Assignment 3
CSE 517: Natural Language Processing

University of Washington

Winter 2016
Due: February 17, 2016, 1:30 pm

1 Semisupervised Text Categorization

Often, we have a lot of unlabeled data and very few labeled examples. Methods that try to combine labeled and unlabeled data are called "semisupervised." This problem asks you to implement a semisupervised learner and report on its performance.

Download and untar `http://courses.cs.washington.edu/courses/cse517/16wi/assignments/a3.tar.gz`. You'll see:

- `train.tsv` and `dev.tsv`: tab-separated files, each line containing a filename and one of 19 labels
- `labeled`: a directory containing the files listed in `train.tsv` and `dev.tsv`; each contains text
- `unlabeled`: a directory containing about ten times as many files as `labeled`; each contains text

The data are derived from speeches by U.S. presidential candidates in 2008 and 2012. The labels encode—in human-readable form—the candidate's name and the year of the race. To make the task more interesting, the speeches have been chopped up into segments of 140 characters or less. Some segments are probably very easy to guess, and others are not (try it). The assignment of segments to training, development, and unlabeled pools was random (probabilities $\frac{10}{121}$, $\frac{1}{121}$, $\frac{10}{11}$, respectively), so it's reasonable to think of all of the data as having been drawn from the same distribution.

The files in `unlabeled` are not, strictly speaking, a test set, because you're going to use them to build your classifier. We are, however, going to measure the accuracy of your classifications of these files. This setup is sometimes called "transductive" learning.

The files listed in `dev.tsv` are not to be used in training; they are your development data.

Here is a simple approach to semi-supervised learning; let $\mathcal{D}_l = \{(\boldsymbol{x}_i, \ell_i)\}_{i=1}^n$ be the labeled data and $\mathcal{D}_u = \{\bar{\boldsymbol{x}}_i\}_{i=1}^m$ be the unlabeled data.

1. $t \leftarrow 0$
2. $\mathcal{D}_l^{(0)} \leftarrow \mathcal{D}_l$
3. Train a supervised classifier on $\mathcal{D}_l^{(t)}$.
4. Label each instance in $\mathcal{D}_u$, letting $\hat{\mathcal{D}}_u$ denote $\{(\bar{\boldsymbol{x}}_i, \text{classify}(\bar{\boldsymbol{x}}_i)\}_{i=1}^m$.

5. Let $\mathcal{D}_l^{(t+1)} \leftarrow \mathcal{D}_l^{(0)} \cup \hat{\mathcal{D}}_u$.
6. $t \leftarrow t + 1$
7. If not done, go to step 3.

In English, this is an approach in which a model is iteratively trained on its own output (sometimes called "self training"). There are many variations on this approach! One is only add some instances to $\mathcal{D}_l^{(t+1)}$ in step 5, perhaps the ones the classifier is most confident about. Another is to add "softly" labeled instances; this is a variant of the EM algorithm. Note also that the stopping criterion is left unspecified; you might continue until very few unlabeled examples' hypothesized labels change, or until accuracy stops improving on a development set, or something else.

You do not need to use the approach above. You may use existing tools, but you may not use any additional data. If you are unsure whether something you are considering using is a "tool" or "data," ask us.

Your deliverables for this problem are:

Question 1: Submit a file named `unlabeled.tsv` formatted exactly like `train.tsv`, but giving labels for the filenames in the `unlabeled` directory.

Question 2: Succinctly write up a description of your approach. You're welcome to use existing tools; this is the place to mention that. Features matter a lot, so be sure to clearly explain yours.

Question 3: Include a plot in which each point is a training run; the x-axis is the fraction of the unlabeled data used in your semisupervised method, and the y-axis is the development-set accuracy. Show accuracies for five settings: 0% (supervised learning without using the unlabeled data at all), 10%, 20%, 50%, and 100% of the unlabeled data (at least). Note: semisupervised learning is challenging; often these curves don't show big benefits over the supervised version. Do what you can.

Question 4: Include the same data in a table, as well.

Question 5: Include a paragraph summarizing the results. Did it work? If not, why do you think it didn't work?

Like the first assignment and the project, this problem lets you choose what to focus on. You could build a simple classifier and focus on preprocessing (maybe using methods taught in class) and/or feature engineering. You could stick with tried-and-true features and focus on optimizing your classifier. You could explore different ways to use the unlabeled data. Any or all of these are fine.

2 Naïve Bayes

In class on 1/27, it was mentioned—but not explained—that the naïve Bayes classifier is an example of a linear classifier. In this exercise, you will work through this claim.

First, consider the naïve Bayes decision rule, written in terms of the observed feature values $\boldsymbol{f} = \langle f_1, \ldots, f_d \rangle$.

$$\text{classify}(\boldsymbol{f}) = \underset{\ell \in \mathcal{L}}{\text{argmax}}\, \pi_\ell \prod_{j=1}^{d} \theta_{f_j|j,\ell} \tag{1}$$

For every label $\ell^* \in \mathcal{L}$, every feature random variable indexed by $j \in \{1, \ldots, d\}$, and every value $f^* \in \mathcal{F}_j$ in that random variable's range, we can define a binary feature:

$$\phi_{\ell^*,j,f^*}(\boldsymbol{f}, \ell) = \mathbf{1}\{\ell = \ell^* \wedge f_j = f^*\} \tag{2}$$

Further, for every $\ell^* \in \mathcal{L}$, define a feature:

$$\phi_{\ell^*}(\boldsymbol{f}, \ell) = \mathbf{1}\{\ell = \ell^*\} \tag{3}$$

The form of these "$\boldsymbol{\phi}$" features more closely matches those of the discriminative classifiers we discussed on 1/27 and 2/1: they are functions of input/output pairs to real values (in this case, binary values).

Question 6: To show that naïve Bayes is a linear classifier, define the values of $\mathbf{w}$ that make equation 1 equivalent to

$$\text{classify}(\boldsymbol{f}) = \underset{\ell \in \mathcal{L}}{\text{argmax}}\, \mathbf{w} \cdot \boldsymbol{\phi}(\boldsymbol{f}, \ell) \tag{4}$$

where $\boldsymbol{\phi}$ includes all features of the form ϕ_{ℓ^*,j,f^*} and ϕ_{ℓ^*} defined in equations 2–3.