Statistical NLP 2017 - Assignment 2

Text Classification Models

Daniel Rivera Ruiz
drr342@nyu.edu

2017-09-27

Part 3. Please submit a short writeup with answers to the following questions:

1. A short description of how you boosted the score in Part 1(d) above the vanilla *performance.*

The *vanilla* performance for the maximum entropy classifier used a feature vector that included only unigram character counts, i.e. the amount of times a given character appeared in the examples.This first approximation yielded a performance of about 64% in the dev set. Nonetheless, by introducing the more complex features listed below, it was possible to obtain much better performances:

a) *Character N-grams.* After performing several experiments, it was decided that including bigram, trigram and 4-gram features delivers the best performance vs. training time ratio. 5-gram (or higher order) features will in some cases increase the performance, but not enough considering the extra time it took to train them. On the other side, the unigram feature was removed because once the higher N-grams were introduced, it was only reducing the performance.

b) *Features related to words.* Additionally to the N-gram characters, two features based on the words conforming the instances were considered in the model: 1) the words themselves (unigram model for words), and 2) the number of words. Spaces and dashes were defined as word separators for these fatures. The number of words proved to be particularly useful to discriminate places, which were oftentimes one-word instances.

c) *Regular expressions features.* Finally, some very specific features were considered after exploring the different data sets and identifying particular patterns that were associated to a certain category. For example, an instance containing the patterns *Co., Inc.* or *Corp.* would most likely belong to the company category. Similar patterns were identified and implemented for the drugs and movies categories.

2. What errors remain in your best classifier and why?

Despite all the efforts and all the features added to the classifier, its performance remains under 90%. After careful observation of the predictions it made, it was found that the most common mistakes were made confussing persons, places and movies. This behavior is not so bad, considering that a lot of movies or places are named after persons, making it almost impossible for the classifier to differentiate them. Other situations similar to this one may be occurring and are most likely the reason why it is so hard to boost the classifier performance even further. Perhaps by implementing a more complex algorithm (like a neural network) or by introducing more complex features the 90% threshold can be crossed.

3. Dev set accuracies for your maximum entropy model and perceptron.

Table 1 shows the highest accuracies achieved in the dev set with the maximum entropy and perceptron models for the proper name classification problem. The maximum entropy model was regularized with $\sigma = 1.0$ and took 77 iterations to converge. For the perceptron model it is worth mentioning that 1) the examples are selected randomly during training and therefore the accuracy may vary from one experiment to another, and 2) the weight vector used by the classifier is the average of the weight vectors calculated after each training instance. This subtle modification improves the accuracy of the perceptron model in about 3% and it also makes it less vulnerable to outputting a weight vector that is highly biased by the last training instance (especially if it happens to be a miss).

Table 1: Dev Set Performance.

Model	Accuracy
Maximum Entropy	89.28%
Perceptron	85.03%