An Investigation of Multiple Language Classifiers COMP 551 - Applied Machine Learning

Thomas Page
Department of Mechanical Engineering
McGill University
Montreal, Canada
thomas.page@mail.mcgill.ca
260771672

David Tamrazov
Computer Science, Arts
McGill University
Montreal, Canada
david.tamrazov@mail.mcgill.ca
260561439

Alexander Wong
Cognitive Science, Arts and Science
McGill University
Montreal, Canada
alexander.wong4@mail.mcgill.ca
260602944

Notorious Language Classifiers
https://github.com/a22wong/notorious-language-classifier

October 23, 2017

I. Introduction

As a result of the first project, corpora in various languages are available to act as training sets for many language related machine learning tasks. The goal of this project is to create classifiers that can determine the language of a sample of text from corpora information developed in the first project. Based on a review of the methods discussed in COMP 551, we construct four classifiers: Naive Bayes, Linear Discriminant Analysis (LDA), Decision Trees, and Extremely Randomized Decision Trees.

II. Related Work

Naive Bayes was chosen as the most promising text classification method for this application based on a previous study in this area. In his masters thesis, Jason Rennie used a Naive Bayes classifier to classify a given news article into 1 of 20 different categories (e.g. Electronics, Religion, Space, etc.). Training on 1000 documents from each category, the Naive Bayes classifier was able to classify the new articles very well [1]. The main difference between that task and this project is that rather than distinguishing between categories, we will be distinguishing between languages, but the same concept applies.

III. Problem Representation

The compiled training dataset contains 276516 labeled language samples from five different languages: Slovak, French, Spanish, German, and Polish. The test set contains 118507 unlabeled samples of zero to ten individual characters. This presented the challenge of classifying seemingly nonsensical data; a common task in practical machine learning. Fig. 1 represents the weight of each language in the training set.

We found that the data required little pre-processing as non-discriminative symbols such as punctuation had already been removed. To process the text, we chose to simply make all characters lowercase as to eliminate the classifier distinguishing between upper- and lower-case symbols. There are many blank lines in the training set, but we found eliminating them to have no effect on our outcome. Nothing is being assigned to a label in these blank lines, and therefore it does not add any weight to the prediction. We also looked into removing numbers from our training set as these have no attachment to language. However, as the corpora are taken from different sources, we decided that some sources may be more likely to include numbers in the dialogue and that we could exploit this distinction.

As mentioned, the test set is a string of characters. Because of this, we chose to train our classifiers on the probability of characters appearing rather than the probability of words or small groups of words appearing. We lost a lot of the natural language data by making our features individual characters, but this seemed to be the only reasonable solution for this unique problem.

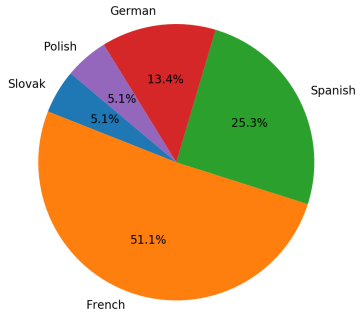

Fig. 1. Training set language distribution.

IV. Testing and Validation

The most effective way to represent the performance metrics of each individual classifier is through their respective confusion matrix. We chose to evaluate our classifiers using a normalized confusion matrix as it highlights important outputs like the true positives and exactly how each class is correctly or incorrectly classified.

A. Naive Bayes

Fig. 2 shows the confusion matrix for the Naive Bayes classifier on the training set. Slovak, French, German, and Polish are all classified 90% correctly. Spanish is the only classifier with a significant difference at 77% correct. It is clear from the matrix that most of the incorrect Spanish classifications are classified at French. This is due in part to both languages sharing many of the same symbols, and because French accounts for over double the training set samples, a Spanish sample is more likely to be classified as French than vice versa. This hypothesis is supported by the confusion matrix, as 16% of Spanish samples are classified as French, but only 7% of French samples are classified as Spanish.

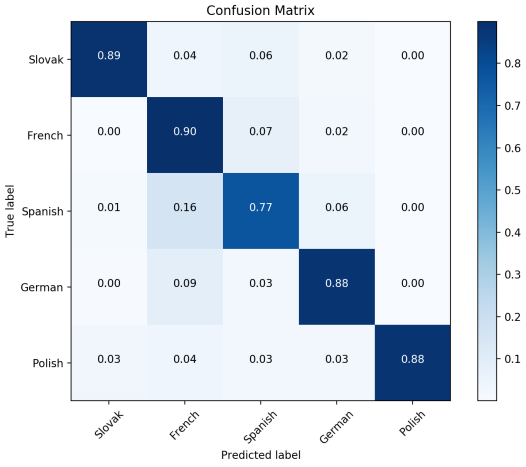

Fig. 2. Naive Bayes confusion matrix.

B. Decision Trees

With a 'nonsensical' test set of only single characters, we decided to try using a Decision Tree to discriminate test data based on what special characters appeared. Naive Bayes simply classified based on

Due to the time complexity of the decision tree implementation, the debugging process was slow and non-optimal

Fig. 3 shows the confusion matrix for the Decision Tree classifier.

There's a total of 333277 instances of special characters in the training set. Only 148 times in the training set did it make a difference

C. Extremely Randomized Trees

V. Discussion

asdf

VI. Statement of Contributions

Thomas was responsible for fully implementing the Naive Bayes classifier. He also wrote the outline of the report and any sections pertaining to Naive Bayes.

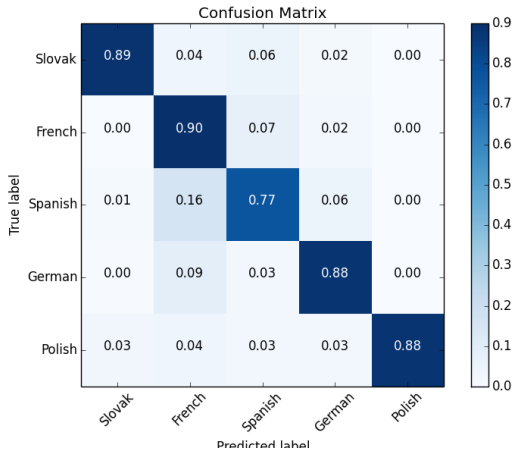

Fig. 3. Naive Bayes confusion matrix.

Alex was responsible for fully implementing the Decision Trees classifier. For the report, he wrote any sections pertaining to Decision Trees.

David was responsible for the LDA and Extremely Randomized Trees classifiers. For the report, he wrote any sections pertaining to LDA or Extremely Randomized Trees.

We hereby state that all the work presented in this report is that of the authors.

References

[1] J. D. M. Rennie, "Improving Multi-class Text Classification with Naive Bayes," M.S. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2001.