Assignment: Language Identification

11-411/11-611 NLP Teaching Staff

Due: February 16, 2023

1 Introduction

In this assignment, you will create a ngram (upto bi-gram for this assignment) character based Naive Bayes text classifier. Your classifier will distinguish between 6 different languages

1.1 Data

The training dataset includes 53,856 different sentences distributed among 6 different languages which include Hausa, Indonesisan, Manobo, Tagalog, Swahili and Nahuatl. The dev set (for testing locally) includes 17,791 different sentences from the same 6 languages. The train and dev files have one sentence per line. The language of the sentence is given first, followed by a tab character, followed by the sentence (each word is separated by whitespace).

2 Learning Objectives

1. Understanding the basics of text classification and language identification.
2. Implementation of Naive Bayes classifier for language identification.
3. Understanding of the Bayes theorem and its application in text classification.
4. Understanding of the performance evaluation metrics for text classification.

3 Task: Programming

In your file naivebayes.py, create a Naive Bayes classifier using the provided template.

1. **Bayes' Rule:** For each sentence, the Naive Bayes classifier returns the class $\hat{c}$ which has the maximum posterior probability. By applying Bayes' rule we can get:

$$\hat{c} = \arg\max_{c \in C} P(c \mid sentence) = \arg\max_{c \in C} \frac{P(sentence \mid c)P(c)}{P(sentence)} \tag{1}$$

   Note that $P(sentence)$ can be dropped in the calculation as it is a normalizing constant and does not affect the argmax calculation.

2. **Prior and Likelihood:** The Naive Bayes model assumes that all n-grams in a sentence are independent of one another given the class. The likelihood of a sentence can be written as:

$$P(f_1, \ldots, f_n \mid c) = \prod_{i=1}^{n} P(f_i \mid c) \tag{2}$$

   where f_i is the i^{th} term in the sentence and c is the class of the sentence. You will need to write a program to estimate the multinomial distributions $P(term - class)$ and the prior $P(c)$ from the training dataset.

3. **Add-one Smoothing:** Use add-one smoothing when estimating probability. Make sure to add the size of the vocabulary (number of unique character n-grams in all classes) to the denominator when normalizing. (Reference - Lecture 6, Slide 40)

4. **Calculations in Log Space:** Probability calculations are done in log space to avoid underflow and increase speed.

5. **Implementation:** All of the languages use the roman alphabet without diacritics, so identification of the languages is based on frequency of ngram (e.g., bigrams). To implement the Train and Test for Naive Bayes classifier - (Refer Lecture 6, Slides 44-45)

3.1 Small Dataset for Debugging

We have provided you with a small dataset and the reference output at various steps. Make use of that to debug your code before running it on the language dataset.

3.2 Evaluation

The points have been divided depending on the performance of your model on the test set (on gradescope). All metrics have been already implemented in the handout.

The rubrik is as follows:

1. 10 points — macro-averaged F1 > 0.90
2. 10 points — micro-averaged F1 > 0.90
3. 15 points — macro-averaged F1 > 0.98

4. 15 points — micro-averaged F1 > 0.98

To evaluate your implementation, train a Naive Bayes classifier using the training data train.txt. Use the dev file dev.txt to evaluate your classifier locally.

The code can be run using the command

python3 naivebayes.py train.txt dev.txt

4 Deliverables

We will be using Gradescope for submission of this homework.
Please submit a zip archive named handin.zip containing the following items.
Please do not put them in a folder inside the archive.

1. A file called naivebayes.py which implements your classifier and reports MP, MR, MF, MAP, MAR, MAF.

2. A file called README. Whether or not you collaborated with other individuals or employed outside resources, you must include a section in your README documenting your consultation (or non-consultation) of other persons and resources. If you have any additional information to give us, you should also put it there.

All deliverables are due by 11:59pm EST on 16th Februrary, 2023.