MAT 345 - PROJECT #2
due Monday, October 12, 2020 at 10:00PM.

OBJECTIVE: In this project, you will implement a spam filter.

GRADING: The assignment is worth 8% of your course grade.

INSTRUCTIONS: Students will work individually on this project, but they may ask questions and clarification from classmates and the instructor. Students must submit their projects on Moodle.

SUBMIT THE FOLLOWING: A copy of your **code** and a **Project Report.** You may use different tools and different programming languages for various parts of this project. You may submit multiple pieces of code to address the steps in the project. Make sure your name is on all files submitted.

PROJECT: Start by downloading the data sets from:

`http://spamassassin.apache.org/old/publiccorpus/`

Read the *readme* file explaining the data. You have 3 options for the data you will work with:

Set A. Use sets 20021010_*easy_ham.tar.bz*2, 20021010_*hard_ham.tar.bz*2, 20021010_*spam.tar.bz*2

Set B. Use sets 20030228_*easy_ham.tar.bz*2, 20030228_*hard_ham.tar.bz*2, 20030228_*spam.tar.bz*2

Set C. Use sets 20030228_*easy_ham.tar.bz*2, 20030228_*easy_ham_2.tar.bz*2, 20030228_*hard_ham.tar.bz*2, 20030228_*spam.tar.bz*2, and 20030228_*spam_2.tar.bz*2

Step 1. Download the set you chose to work with (A, B or C) and save it in folders *easy_ham*, *hard_ham* and *spam*. You now have a few thousand emails that have been labeled as spam or ham.

Step 2. Split the data into *training* and *testing* sets: move every 4^{th} message into *testing* folders. You are left with 75% of data in *training* folders.

Step 3. Build a spam filter by considering the *Subject* line only!

(a) Use the *training* data to make a list of words (made out of letters only) from the *Subject* line. Now you have a list of words to test against: $W = \{w_1, w_2, \ldots, w_N\}$.

(b) For each $w_k \in W$, compute the probabilities $P(w_k \,|\, \text{spam})$ and $P(w_k \,|\, \text{ham})$. Use $\alpha = 1$, $\beta = 2$ for smoothing:

$$P(w_k \,|\, \text{spam}) = \frac{\alpha + (\#\text{ spam containing } w_k)}{\beta + (\#\text{ spam})}\,,\; P(w_k \,|\, \text{ham}) = \frac{\alpha + (\#\text{ ham containing } w_k)}{\beta + (\#\text{ ham})}$$

(c) Output the list of 5 words with highest probabilities $P(\text{spam} \mid w_k)$

(d) Output the list of 5 words with highest probabilities $P(\text{ham} \mid w_k)$

(e) Set up the function

$$P(\text{spam}|\vec{X} = \vec{a}) = \frac{P(\vec{X} = \vec{a} \mid \text{spam})P(\text{spam})}{P(\vec{X} = \vec{a} \mid \text{spam})P(\text{spam}) + P(\vec{X} = \vec{a} \mid \text{ham})P(\text{ham})}.$$

You might want to pre-compute some parameters, as in the Naive Bayes notes.

Step 4. Test the spam filter using the saved *testing* data.

(a) Output the accuracy rate: the proportion of correctly predicted emails to the total number of emails tested.

(b) Output the precision rate: the proportion of spam messages correctly predicted to the number of emails predicted spam.

(c) Output the recall rate: the proportion of spam messages correctly predicted to the number of spam messages.

Step 5. In the **Project Report**, you must include, but are not limited to:

- Your name(s)
- The data set you used (Sets A, B, or C)
- The tools you used in each part of the project.
- Any decisions specific to your spam filter, such as: words that you might have decided to exclude, threshold for the probability of spam (especially if different than 0.5), etc.
- The 5 most "spammiest" and the 5 most "hammiest" words from the training stage: answers (c) and (d) from Step 3.
- The accuracy, precision and recall rates from the testing stage: answers (a), (b), and (c) from Step 4.
- Conclusions on the performance of your spam filter and possible steps you would take to improve it.