CA02: Spam eMail Detection using Naive Bayes Classification Algorithm

Assignment Description

In this exercise we shall train the model with set of emails labelled as either from Spam or Not Spam. There are 702 emails equally divided into spam and non spam category. Next, we shall test the model on 260 emails. We shall ask model to predict the category of this emails and compare the accuracy with correct classification that we already know.

Instructions

The data folders are in a Zip file at BrightSpace CA02 assignment folder. The assignment instruction document and the provided .ipynb file is also at BrightSpace.

What you are supposed to do:

1. Read the code and understand the logic of every line of the code
2. Complete the code (there are missing areas of the code and it is mentioned in the commented cells), such that the code runs properly and produces the exact same result as displayed at the end of the notebook.
3. Comment the code thoroughly using "markdown" texts and comment lines so that the logic of the code is easily understandable. It's your job to explain clearly your understanding of the process at every step.

IMPORTANT:

- Your data folders "test-mails" and "train-mails" must be under your current folder where you will create your notebook.
- In your code you must use the **relative** path names of the data folder locations ("./test-mails", "./train-mails").
- **DO NOT CHANGE THE DATA FOLDER and DATA FILE NAMES** or tamper with the ORIGINAL DATA, as your program will be evaluated by running against the original data.

Some Code Explanation

Cleaning and Preparing the data

We have two folders ***test-mails*** and ***train-mails***. We will use train-mails to train the model. The sample email data set looks like:

```
Subject: re : 2 . 882 s – > np np> deat : sun , 15 dec 91 2 : 25 : 2 est > : michael <
mmorse @ vm1 . yorku . ca > > subject : re : 2 . 864 query
> > wlodek zadrozny ask " anything interest " > construction " s > np np " . . . second ,
> much relate : consider construction form > discuss list late reduplication ? > logical
sense " john mcnamara name " tautologous thus , > level , indistinguishable " , , here ? "
. ' john mcnamara name ' tautologous support those logic–base semantics irrelevant natural
language . sense tautologous ? supplies value attribute follow attribute value . fact
value name–attribute relevant entity ' chaim shmendrik ' , ' john mcnamara name ' false .
tautology , . ( reduplication , either . )
```

First line is subject and the content starts from the third line.
If you navigate to any of the train-mails or test-mails, you shall see file names in two patterns:

number–numbermsg[number].txt : example **3–1msg1.txt** (this are non spam emails)ORspmsg[Number].txt : example **spmsga162.txt (**these files are of spam emails)**.**

The very first step in text data mining task is to clean and prepare the data for a model. In **cleaning** we remove the non required words, expressions and symbols from text.

Consider a text: *"Hi, this is Alice. Hope you are doing well and enjoying your vacation."*

Here the words like is, this, are, and etc. don't really contribute to the analysis. Such words are also called **stop words**. Hence in this exercise, we consider only most frequent 3000 words of dictionary from email.

After cleaning what we need from every email document, we should have some matrix representation of the word frequency.

Submitting Your Work

1. Complete your work in the Colab / Jupyter NB environment
2. Upload your code in the CA Assignment Submission Folder by the deadline
3. Create a GitHub Folder for CA02 in your GitHub repository
4. Upload the Notebook to this GitHub repository and Commit / Push. Do not forget to add a Readme.md file in your GitHub repository folder.

5. Upload your Notebook file (.ipynb) **and** Readme.md file at the BrightSpace Assignment Folder. **Completing and uploading a Readme.md file is a must** for EVERY computer assignment in this course. Google and study the syntax of creating .md files so that you can use various fonts and header hierarchy. Make them beautiful! Most text editors allow creating .md files and it's syntax.

Why should you create a Readme.md file in GitHub? (.md stands for "mark down")

In the Open Source community, it is an unspoken culture to provide some information about your code so that others can understand and re-use your code. This information is traditionally documented in the form of a Readme.md file accompanying every code of library of code. The kind of information you should include in your Readme.md file:

- A brief description of the purpose of your program and what it is doing
- What libraries are needed in the environment to be able to run the code
- What versions of various software / libraries are you using
- What dataset are you using and their source
- Acknowledgement and link to source code of anyone else's code, if you are reusing them
- How to install and run your code along with datasets to be able to run your program successfully

Additionally, your peer reviewer will also evaluate the readability of your Readme.md file and will comment on that.