Title - Spam Email Detector

Description - A web interface which allows the user to enter details of an email to check if the email is spam or not spam (ham)

Two Strategies :

1. Classification given labels
2. Anomaly Detection

Both will be implemented user will be given choice as to which he wants to use

User Input:

1. Sender Email : Text
2. Receiver Name: Text
3. Subject : Text
4. Reply To (if present) : Text
5. Forwarded : Yes / No
6. Has Attachment : Yes / No
7. Contains In Line Image : Yes / No
8. Email Text : Text

Possible Features:

* Forwarded : 1 / 0
* Has Attachment : 1 / 0
* Contains inline image : 1 / 0
* Receiver name in text : 1 / 0
* Reply To Present : 1 / 0
* “no-reply” in sender email : 1 / 0
* Subject Empty : 1 / 0
* Email Content Empty : 1 / 0
* "Marketing"/"publicity" in sender email
* Probability of Spamness given Subject after doing NLP
* Probability of Spamness given Email Text after doing NLP

Research:

* <https://pepipost.com/blog/how-to-read-email-headers-to-identify-spam/>

Data Preparation Ideas:

1. Python gmail client
2. Custom Emails Scrapper
3. Public Data Sets

Python Gmail Client

* <https://codehandbook.org/how-to-read-email-from-gmail-using-python/>

Custom Emails Scrapper

Risky as account may be suspended

Public Data Sets

* <https://www.kaggle.com/veleon/ham-and-spam-dataset> (Email Headers)
* <http://www.aueb.gr/users/ion/data/enron-spam/> (Spam Headers)
* <http://www.aueb.gr/users/ion/data/lingspam_public.tar.gz> (Emails subject and text)

Data Gathered:

Cell consists of no. of txt files.

|  |  |  |
| --- | --- | --- |
|  | Header | Body (Subject, Text) |
| Spam | 10502 | 482 |
| Ham | 2552 | 2413 |

Parsing Outline:

Ham and Spam Body : Regex

Ham Header and Spam Header Email : Email Library

Spam Header Text: Regex

CSV’s:

extracted\_body : Ham and Spam Body

extracted\_header\_email : Ham Header and Spam Header Email

extracted\_spam\_header : Spam Header Text

extracted\_header: extracted\_header\_email, extracted\_spam\_header combined

prepped\_body: Cleaned Body

prepped\_header: Features from extracted\_header

NLP Model will be trained on Body and some body part of header.

Validated on remaining part of Header’s body.

Classifier will use features in Header.