05.Bayes Classifier - preprocessing

May 2, 2021

0.1 Notebook Imports

[1]:
```
from os import walk
from os.path import join

import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

import nltk
from nltk.stem import PorterStemmer
from nltk.stem import SnowballStemmer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize

from sklearn.model_selection import train_test_split

from bs4 import BeautifulSoup
from wordcloud import WordCloud
from PIL import Image

%matplotlib inline

# for installing a package
# conda install -c channel_name package name
```

0.2 Constants

[2]:
```
EXAMPLE_FILE = 'SpamData/01_Processing/practice_email.txt'

SPAM_1_PATH = 'SpamData/01_Processing/spam_assassin_corpus/spam_1'
SPAM_2_PATH = 'SpamData/01_Processing/spam_assassin_corpus/spam_2'
EASY_NONSPAM_1_PATH = 'SpamData/01_Processing/spam_assassin_corpus/easy_ham_1'
EASY_NONSPAM_2_PATH = 'SpamData/01_Processing/spam_assassin_corpus/easy_ham_2'

SPAM_CAT = 1
HAM_CAT = 0
VOCAB_SIZE = 2500
```

```
WORD_ID_FILE ='SpamData/01_Processing/word-by-id.csv'
DATA_JSON_FILE ='SpamData/01_Processing/email-text-data.json'

TRAINING_DATA_FILE = 'SpamData/02_Training/train-data.txt'
TEST_DATA_FILE = 'SpamData/03_Testing/test-data.txt'

WHALE_FILE = 'SpamData/01_Processing/wordcloud_resources/whale-icon.png'
SKULL_FILE = 'SpamData/01_Processing/wordcloud_resources/skull-icon.png'
THUMBS_UP_FILE = 'SpamData/01_Processing/wordcloud_resources/thumbs-up.png'
THUMBS_DOWN_FILE = 'SpamData/01_Processing/wordcloud_resources/thumbs-down.png'
FONT_BOLD = 'SpamData/01_Processing/wordcloud_resources/OpenSansCondensed-Bold.
 ↪ttf'
FONT_LIGHT = 'SpamData/01_Processing/wordcloud_resources/
 ↪OpenSansCondensed-Light.ttf'
```

0.3 Reading Files

[3]:
```
stream = open(EXAMPLE_FILE,encoding='latin-1')
message = stream.read()
stream.close()

print(type(message))
print(message)
```

```
<class 'str'>
From exmh-workers-admin@redhat.com  Thu Aug 22 12:36:23 2002
Return-Path: <exmh-workers-admin@spamassassin.taint.org>
Delivered-To: zzzz@localhost.netnoteinc.com
Received: from localhost (localhost [127.0.0.1])
        by phobos.labs.netnoteinc.com (Postfix) with ESMTP id D03E543C36
        for <zzzz@localhost>; Thu, 22 Aug 2002 07:36:16 -0400 (EDT)
Received: from phobos [127.0.0.1]
        by localhost with IMAP (fetchmail-5.9.0)
        for zzzz@localhost (single-drop); Thu, 22 Aug 2002 12:36:16 +0100 (IST)
Received: from listman.spamassassin.taint.org (listman.spamassassin.taint.org
[66.187.233.211]) by
    dogma.slashnull.org (8.11.6/8.11.6) with ESMTP id g7MBYrZ04811 for
    <zzzz-exmh@spamassassin.taint.org>; Thu, 22 Aug 2002 12:34:53 +0100
Received: from listman.spamassassin.taint.org (localhost.localdomain
[127.0.0.1]) by
    listman.redhat.com (Postfix) with ESMTP id 8386540858; Thu, 22 Aug 2002
    07:35:02 -0400 (EDT)
Delivered-To: exmh-workers@listman.spamassassin.taint.org
Received: from int-mx1.corp.spamassassin.taint.org (int-
mx1.corp.spamassassin.taint.org
    [172.16.52.254]) by listman.redhat.com (Postfix) with ESMTP id 10CF8406D7
```

```
    for <exmh-workers@listman.redhat.com>; Thu, 22 Aug 2002 07:34:10 -0400
    (EDT)
Received: (from mail@localhost) by int-mx1.corp.spamassassin.taint.org
(8.11.6/8.11.6)
    id g7MBY7g11259 for exmh-workers@listman.redhat.com; Thu, 22 Aug 2002
    07:34:07 -0400
Received: from mx1.spamassassin.taint.org (mx1.spamassassin.taint.org
[172.16.48.31]) by
    int-mx1.corp.redhat.com (8.11.6/8.11.6) with SMTP id g7MBY7Y11255 for
    <exmh-workers@redhat.com>; Thu, 22 Aug 2002 07:34:07 -0400
Received: from ratree.psu.ac.th ([202.28.97.6]) by mx1.spamassassin.taint.org
    (8.11.6/8.11.6) with SMTP id g7MBIhl25223 for <exmh-workers@redhat.com>;
    Thu, 22 Aug 2002 07:18:55 -0400
Received: from delta.cs.mu.OZ.AU (delta.coe.psu.ac.th [172.30.0.98]) by
    ratree.psu.ac.th (8.11.6/8.11.6) with ESMTP id g7MBWel29762;
    Thu, 22 Aug 2002 18:32:40 +0700 (ICT)
Received: from munnari.OZ.AU (localhost [127.0.0.1]) by delta.cs.mu.OZ.AU
    (8.11.6/8.11.6) with ESMTP id g7MBQPW13260; Thu, 22 Aug 2002 18:26:25
    +0700 (ICT)
From: Robert Elz <kre@munnari.OZ.AU>
To: Chris Garrigues <cwg-dated-1030377287.06fa6d@DeepEddy.Com>
Cc: exmh-workers@spamassassin.taint.org
Subject: Re: New Sequences Window
In-Reply-To: <1029945287.4797.TMDA@deepeddy.vircio.com>
References: <1029945287.4797.TMDA@deepeddy.vircio.com>
    <1029882468.3116.TMDA@deepeddy.vircio.com> <9627.1029933001@munnari.OZ.AU>
    <1029943066.26919.TMDA@deepeddy.vircio.com>
    <1029944441.398.TMDA@deepeddy.vircio.com>
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Message-Id: <13258.1030015585@munnari.OZ.AU>
X-Loop: exmh-workers@spamassassin.taint.org
Sender: exmh-workers-admin@spamassassin.taint.org
Errors-To: exmh-workers-admin@spamassassin.taint.org
X-Beenthere: exmh-workers@spamassassin.taint.org
X-Mailman-Version: 2.0.1
Precedence: bulk
List-Help: <mailto:exmh-workers-request@spamassassin.taint.org?subject=help>
List-Post: <mailto:exmh-workers@spamassassin.taint.org>
List-Subscribe: <https://listman.spamassassin.taint.org/mailman/listinfo/exmh-
workers>,
    <mailto:exmh-workers-request@redhat.com?subject=subscribe>
List-Id: Discussion list for EXMH developers <exmh-
workers.spamassassin.taint.org>
List-Unsubscribe: <https://listman.spamassassin.taint.org/mailman/listinfo/exmh-
workers>,
    <mailto:exmh-workers-request@redhat.com?subject=unsubscribe>
List-Archive: <https://listman.spamassassin.taint.org/mailman/private/exmh-
```

```
workers/>
Date: Thu, 22 Aug 2002 18:26:25 +0700


Dear Mr Still

Good tidings to you and all your staff for the festive season ahead (Christmas).
Now to the crux of the matter-in-hand: I am a fully qualified Santa Claus and am
wondering whether you might consider me to run my own "Santa's Grotto" in your
store.
But WAIT! You're probably thinking: "What makes him so special?"
Well, first of all, I have made several changes to the characterisation of
Father Christmas. Rather than greeting the children with shouts of "Ho, ho, ho!"
I prefer to whisper the phrase "Dependence is not unfathomable in this cruel
world we live in". In addition, my gifts are ALL hand-made, ranging from felt
hoops to vanilla-pod holders.
You will note also, from the enclosed sketch, that I have radically redesigned
Santa's outfit and have renamed my character "Lord Buckles". Would you be
interested in employing me? I promise NEVER to let you down.
I look forward to hearing from you.

Best wishes
Robin Cooper
[Excerpt from the book: The Timewaster Letters by Robin Cooper]
```

[4]:
```
import sys
sys.getfilesystemencoding()
```

[4]:
```
'utf-8'
```

[5]:
```
stream = open(EXAMPLE_FILE,encoding='latin-1')

is_body = False
lines = []

for line in stream:
    if is_body:
        lines.append(line)
    elif line == '\n':
        is_body=True


stream.close()

email_body = ' '.join(lines)
print(email_body)
# print(lines)
```

```
 Dear Mr Still

 Good tidings to you and all your staff for the festive season ahead
(Christmas).
 Now to the crux of the matter-in-hand: I am a fully qualified Santa Claus and
am wondering whether you might consider me to run my own "Santa's Grotto" in
your store.
 But WAIT! You're probably thinking: "What makes him so special?"
 Well, first of all, I have made several changes to the characterisation of
Father Christmas. Rather than greeting the children with shouts of "Ho, ho, ho!"
I prefer to whisper the phrase "Dependence is not unfathomable in this cruel
world we live in". In addition, my gifts are ALL hand-made, ranging from felt
hoops to vanilla-pod holders.
 You will note also, from the enclosed sketch, that I have radically redesigned
Santa's outfit and have renamed my character "Lord Buckles". Would you be
interested in employing me? I promise NEVER to let you down.
 I look forward to hearing from you.

 Best wishes
 Robin Cooper
 [Excerpt from the book: The Timewaster Letters by Robin Cooper]
```

0.4 Generator Functions

[6]:
```
def generate_squares(N):
    for my_number in range(N):
        yield my_number**2
```

[7]:
```
for i in generate_squares(5):
    print(i,end=' ->')
```

```
0 ->1 ->4 ->9 ->16 ->
```

0.5 Email Body Extraction

[8]:
```
def email_body_generator(path):

    for root, dirnames, filenames in walk(path):
        for file_name in filenames:

            filepath = join(root,file_name)
            stream = open(filepath,encoding='latin-1')

            is_body = False
            lines = []
```

```
            for line in stream:
                if is_body:
                    lines.append(line)
                elif line == '\n':
                    is_body=True


            stream.close()

            email_body = ' '.join(lines)
            yield file_name,email_body
```

[9]:
```
def df_from_directory(path,classification):
    rows = []
    row_names = []

    for file_name, email_body in email_body_generator(path):
        rows.append({'MESSAGE':email_body,'CATEGORY':classification})
        row_names.append(file_name)
    return pd.DataFrame(rows,index=row_names)
```

[10]:
```
spam_emails = df_from_directory(SPAM_1_PATH,SPAM_CAT)
spam_emails = spam_emails.append(df_from_directory(SPAM_2_PATH,SPAM_CAT))
```

[11]:
```
spam_emails.shape
```

[11]: (1898, 2)

[12]:
```
ham_emails = df_from_directory(EASY_NONSPAM_1_PATH,HAM_CAT)
ham_emails = ham_emails.append(df_from_directory(EASY_NONSPAM_2_PATH,HAM_CAT))
ham_emails.shape
```

[12]: (3901, 2)

[13]:
```
data = pd.concat([spam_emails,ham_emails])
print('Shape of entire dataframe is',data.shape)
data.head()
```

```
Shape of entire dataframe is (5799, 2)
```

[13]:
```
                                                  MESSAGE  \
00001.7848dde101aa985090474a91ec93fcf0  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Tr…
00002.d94f1b97e48ed3b553b3508d116e6a09  1) Fight The Risk of Cancer!\n
http://www.adcl…
00003.2ee33bc6eacdb11f38d052c44819ba6c  1) Fight The Risk of Cancer!\n
http://www.adcl…
```

```
00004.eac8de8d759b7e74154f142194282724
################################################…
00005.57696a39d7d84318ce497886896bf90d  I thought you might like these:\n 1)
Slim Down…

                                        CATEGORY
00001.7848dde101aa985090474a91ec93fcf0         1
00002.d94f1b97e48ed3b553b3508d116e6a09         1
00003.2ee33bc6eacdb11f38d052c44819ba6c         1
00004.eac8de8d759b7e74154f142194282724         1
00005.57696a39d7d84318ce497886896bf90d         1
```

[14]:
```
data.tail()
```

[14]:
```
  MESSAGE  \
01396.61983fbe6ec43f55fd44e30fce24ffa6
http://news.bbc.co.uk/1/hi/england/2515127.stm…
01397.9f9ef4c2a8dc012d80f2ce2d3473d3b7  > >-- be careful when using this one.)
Also, t…
01398.169b51731fe569f42169ae8f948ec676  >>>>> "SM" == Skip Montanaro
<skip@pobox.com> …
01399.ca6b00b7b341bbde9a9ea3dd6a7bf896  So then, "Mark Hammond"
<mhammond@skippinet.co…
01400.f897f0931e461e7b2e964d28e927c35e  Hi there,\n \n Now this is probably of
no use …

                                        CATEGORY
01396.61983fbe6ec43f55fd44e30fce24ffa6         0
01397.9f9ef4c2a8dc012d80f2ce2d3473d3b7         0
01398.169b51731fe569f42169ae8f948ec676         0
01399.ca6b00b7b341bbde9a9ea3dd6a7bf896         0
01400.f897f0931e461e7b2e964d28e927c35e         0
```

0.6 Data Cleaning: Checking for Missing values

[15]:
```
# check if any messages are null
data['MESSAGE'].isnull().values.any()
```

[15]:
```
False
```

[16]:
```
type('')
```

[16]:
```
str
```

[17]:
```
len('')
```

[17]:
```
0
```

[18]:
```
my_var = None # null in py
```

[19]:
```
type(my_var)
```

[19]:
```
NoneType
```

[20]:
```
# check if there are empty strings
(data['MESSAGE'].str.len() == 0).any()
```

[20]:
```
True
```

[21]:
```
(data['MESSAGE'].str.len() == 0).sum()
```

[21]:
```
3
```

[22]:
```
#Challenge number of entries of null
data['MESSAGE'].isnull().sum()
```

[22]:
```
0
```

0.6.1 Locate empty emails

[23]:
```
data[data['MESSAGE'].str.len()==0 ].index
```

[23]:
```
Index(['cmds', 'cmds', 'cmds'], dtype='object')
```

[24]:
```
data.index.get_loc('cmds').any()
```

[24]:
```
True
```

[25]:
```
data.drop(['cmds'],inplace=True)
```

[26]:
```
data.shape
```

[26]:
```
(5796, 2)
```

0.7 Add Document IDs to Track Emails in Dataset

[27]:
```
document_ids = range(0,len(data.index))
data['DOC_ID']=document_ids
```

[28]:
```
data.DOC_ID
```

[28]:
```
00001.7848dde101aa985090474a91ec93fcf0       0
00002.d94f1b97e48ed3b553b3508d116e6a09       1
00003.2ee33bc6eacdb11f38d052c44819ba6c       2
00004.eac8de8d759b7e74154f142194282724       3
00005.57696a39d7d84318ce497886896bf90d       4
```

```
                                                 …
01396.61983fbe6ec43f55fd44e30fce24ffa6    5791
01397.9f9ef4c2a8dc012d80f2ce2d3473d3b7    5792
01398.169b51731fe569f42169ae8f948ec676    5793
01399.ca6b00b7b341bbde9a9ea3dd6a7bf896    5794
01400.f897f0931e461e7b2e964d28e927c35e    5795
Name: DOC_ID, Length: 5796, dtype: int32
```

[29]:
```
data['FILE_NAME']=data.index
data.set_index('DOC_ID',inplace=True)
data.head()
```

[29]:
```
                                                 MESSAGE  CATEGORY  \
DOC_ID
0       <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Tr…         1
1       1) Fight The Risk of Cancer!\n http://www.adcl…         1
2       1) Fight The Risk of Cancer!\n http://www.adcl…         1
3       ##################################################…         1
4       I thought you might like these:\n 1) Slim Down…         1

                                    FILE_NAME
DOC_ID
0       00001.7848dde101aa985090474a91ec93fcf0
1       00002.d94f1b97e48ed3b553b3508d116e6a09
2       00003.2ee33bc6eacdb11f38d052c44819ba6c
3       00004.eac8de8d759b7e74154f142194282724
4       00005.57696a39d7d84318ce497886896bf90d
```

[30]:
```
data.tail()
```

[30]:
```
                                                 MESSAGE  CATEGORY  \
DOC_ID
5791    http://news.bbc.co.uk/1/hi/england/2515127.stm…         0
5792    > >-- be careful when using this one.) Also, t…         0
5793    >>>>> "SM" == Skip Montanaro <skip@pobox.com> …         0
5794    So then, "Mark Hammond" <mhammond@skippinet.co…         0
5795    Hi there,\n \n Now this is probably of no use …         0

                                    FILE_NAME
DOC_ID
5791    01396.61983fbe6ec43f55fd44e30fce24ffa6
5792    01397.9f9ef4c2a8dc012d80f2ce2d3473d3b7
5793    01398.169b51731fe569f42169ae8f948ec676
5794    01399.ca6b00b7b341bbde9a9ea3dd6a7bf896
5795    01400.f897f0931e461e7b2e964d28e927c35e
```

0.8 Save to File Using Pandas

[31]:
```
data.to_json(DATA_JSON_FILE)
```

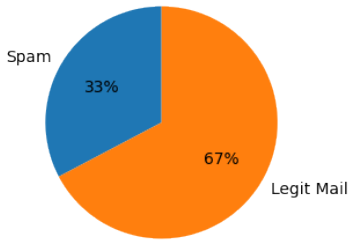
0.9 Number of Spam Messages Visualised (Pie Charts)

[32]:
```
data.CATEGORY.value_counts()
```

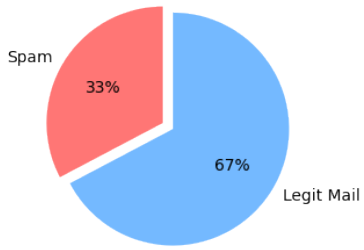
[32]:
```
0    3900
1    1896
Name: CATEGORY, dtype: int64
```

[33]:
```
amount_of_spam = data.CATEGORY.value_counts()[1]
amount_of_ham = data.CATEGORY.value_counts()[0]
```

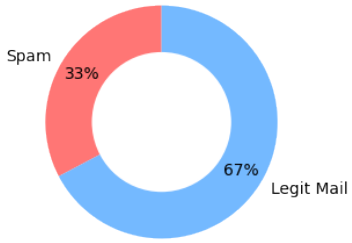
[34]:
```
category_names = ['Spam','Legit Mail']
sizes = [amount_of_spam,amount_of_ham]

plt.figure(figsize=(2,2),dpi=166)
plt.pie(sizes,labels=category_names,textprops={"fontsize":
 ↪6},startangle=90,autopct='%1.0f%%')
plt.show()
```



[35]:
```
category_names = ['Spam','Legit Mail']
sizes = [amount_of_spam,amount_of_ham]
custom_colors =['#ff7675','#74b9ff']

plt.figure(figsize=(2,2),dpi=166)
plt.pie(sizes,labels=category_names,textprops={"fontsize":
 ↪6},startangle=90,autopct='%1.0f%%',colors=custom_colors,explode=[0,0.1])
plt.show()
```

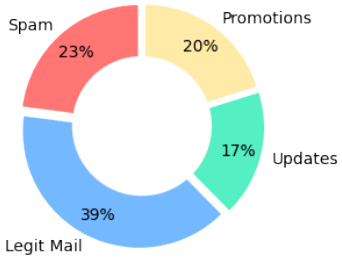


[36]:
```
category_names = ['Spam','Legit Mail']
sizes = [amount_of_spam,amount_of_ham]
custom_colors =['#ff7675','#74b9ff']

plt.figure(figsize=(2,2),dpi=166)
plt.pie(sizes,labels=category_names,textprops={"fontsize":
 ↪6},startangle=90,autopct='%1.0f%%',colors=custom_colors,pctdistance=0.8)

#draw circle
centre_circle = plt.Circle((0,0),radius=0.6,fc='white')
plt.gca().add_artist(centre_circle)

plt.show()
```



[37]:
```
category_names = ['Spam','Legit Mail','Updates','Promotions']
sizes = [25,43,19,22]
custom_colors =['#ff7675','#74b9ff','#55efc4','#ffeaa7']
offset=[0.05,0.05,0.05,0.05]
plt.figure(figsize=(2,2),dpi=166)
plt.pie(sizes,labels=category_names,textprops={"fontsize":
 ↪6},startangle=90,autopct='%1.0f%%',colors=custom_colors,pctdistance=0.
 ↪8,explode=offset)
```

```
#draw circle
centre_circle = plt.Circle((0,0),radius=0.6,fc='white')
plt.gca().add_artist(centre_circle)

plt.show()
```



1 Natural Language Processing

1.1 Text Pre-Processing

[38]:
```
msg = 'All work an no play makes Jack a dull boy.'
msg.lower()
```

[38]:
```
'all work an no play makes jack a dull boy.'
```

1.1.1 Download the NLTK Resources (Tokenizer & StopWords)

[39]:
```
nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\BHANU\AppData\Roaming\nltk_data…
[nltk_data]   Package punkt is already up-to-date!
```

[39]:
```
True
```

[40]:
```
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\BHANU\AppData\Roaming\nltk_data…
[nltk_data]   Package stopwords is already up-to-date!
```

[40]:
```
True
```

[41]:
```
nltk.download('gutenberg')
nltk.download('shakespeare')
```

```
[nltk_data] Downloading package gutenberg to
[nltk_data]     C:\Users\BHANU\AppData\Roaming\nltk_data…
[nltk_data]   Package gutenberg is already up-to-date!
[nltk_data] Downloading package shakespeare to
[nltk_data]     C:\Users\BHANU\AppData\Roaming\nltk_data…
[nltk_data]   Package shakespeare is already up-to-date!
```

[41]: True

1.1.2 Tokenising

[42]:
```
msg = 'All work an no play makes Jack a dull boy.'
word_tokenize(msg.lower())
```

[42]: ['all', 'work', 'an', 'no', 'play', 'makes', 'jack', 'a', 'dull', 'boy', '.']

1.1.3 Removing stop words

[43]:
```
stop_words = set(stopwords.words('english'))
```

[44]:
```
type(stop_words)
```

[44]: set

[45]:
```
if 'this' in stop_words:
    print('found it')
```

```
found it
```

[46]:
```
if 'hello' not in stop_words:
    print('Nope not in here')
```

```
Nope not in here
```

[47]:
```
#Challenge: append non-stop words to filtered_words
msg = 'All work an no play makes Jack a dull boy. To be or not to be.'
words = word_tokenize(msg.lower())
filtered_words = []
for word in words:
    if word not in stop_words:
        filtered_words.append(word)
print(filtered_words)
```

```
['work', 'play', 'makes', 'jack', 'dull', 'boy', '.', '.']
```

1.1.4 Word Stems and Stemming

[48]:
```
msg = 'All work an no play makes Jack a dull boy. To be or not to be. \
    Nobody expects the Spanish Inquisition!'
words = word_tokenize(msg.lower())

# stemmer = PorterStemmer()
stemmer = SnowballStemmer('english')

filtered_words = []
for word in words:
    if word not in stop_words:
        stemmed_word=stemmer.stem(word)
        filtered_words.append(stemmed_word)
print(filtered_words)
```

```
['work', 'play', 'make', 'jack', 'dull', 'boy', '.', '.', 'nobodi', 'expect',
'spanish', 'inquisit', '!']
```

1.1.5 Removing punctuation

[49]:
```
msg = 'All work an no play makes Jack a dull boy. To be or not to be. ??? \
    Nobody expects the Spanish Inquisition!'
words = word_tokenize(msg.lower())


stemmer = SnowballStemmer('english')

filtered_words = []
for word in words:
    if word not in stop_words and word.isalpha() :
        stemmed_word=stemmer.stem(word)
        filtered_words.append(stemmed_word)
print(filtered_words)
```

```
['work', 'play', 'make', 'jack', 'dull', 'boy', 'nobodi', 'expect', 'spanish',
'inquisit']
```

1.1.6 Removing HTML tags from Emails

[50]:
```
soup = BeautifulSoup(data.at[0,'MESSAGE'],'html.parser')
print(soup.prettify())
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
 <head>
  <meta charset="utf-8" content='3D"text/html;' http-equiv="3DContent-T="
ype=""/>
  <meta 5.00.2314.1000"="" content='3D"MSHTML' name="3DGENERATOR"/>
```

```
 </head>
 <body>
  <!-- Inserted by Calypso -->
  <table black;="" border="3D0" cellpadding="3D0" cellspacing="3D2" display:=""
id="3D_CalyPrintHeader_" none"="" r="ules=3Dnone" style='3D"COLOR:'
width='3D"100%"'>
   <tbody>
    <tr>
     <td colspan="3D3">
      <hr color="3Dblack" noshade="" size="3D1"/>
     </td>
    </tr>
   </tbody>
  </table>
 </body>
</html>
<tr>
 <td colspan="3D3">
  <hr color="3Dblack" noshade="" size="3D1"/>
 </td>
</tr>
<!-- End Calypso -->
<!-- Inserted by Calypso=
  -->
<font color="3D#000000" face="3DVERDANA,ARIAL,HELVETICA" size="3D-2">
 <br/>
</font>
&lt;=
 /TR&gt;
<!-- End Calypso -->
<font bold"="" color="3D#ff0000" face='3D"Copperplate' gothic="" ptsize='3D"10"'
size="3D5">
 <center>
  Save up to 70% on Life Insurance.
 </center>
</font>
<font 0000="" bold"="" color="3D#ff=" face='3D"Copperplate' gothic=""
ptsize='3D"10"' size="3D5">
 <center>
  Why Spend More Than You Have To?
  <center>
   <font bold"="" color="3D#ff0000" face='3D"Copperplate' gothic=""
pt='SIZE=3D"10"' size="3D5">
    <center>
     Life Quote Savings
     <center>
      <p align="3Dleft">
      </p>
```

```
      <p align="3Dleft">
      </p>
     </center>
    </center>
   </font>
  </center>
 </center>
</font>
<br/>
<p>
</p>
<center>
 <table border="3D0" bordercolor="3D#111111" cellpadding="3D0" cellspacing="3D0"
wi="dth=3D650">
  <tbody>
  </tbody>
 </table>
 <table border="3D0" bordercolor="3D#111111" cellpadding="3D5" cellspacing="3D0"
wi="dth=3D650">
  <tbody>
   <tr>
    <td colspan="3D2" width='3D"35%"'>
     <b>
      <font face="3DVerdana" size="3D4">
       Ensurin=
 g your
       family's financial security is very important. Life Quote Savings ma=
 kes
       buying life insurance simple and affordable. We Provide FREE Access =
 to The
       Very Best Companies and The Lowest Rates.
      </font>
     </b>
    </td>
   </tr>
   <tr>
    <td align="3Dmiddle" valign="3Dtop" width='3D"18%"'>
     <table bordercolor="3D#111111" width='3D"100%"'>
      <tbody>
       <tr>
        <td %"="" 5px"="" 5px;="" padding-right:="" style='3D"PADDING-LEFT:'
width='3D"100='>
         <font face="3DVerdana" size="3D4">
          <b>
           Life Quote Savings
          </b>
          is FAST, EAS=
 Y and
```

```
            SAVES you money! Let us help you get started with the best val=
 ues in
            the country on new coverage. You can SAVE hundreds or even tho=
 usands
            of dollars by requesting a FREE quote from Lifequote Savings. =
 Our
            service will take you less than 5 minutes to complete. Shop an=
 d
            compare. SAVE up to 70% on all types of Life insurance!
          </font>
         </td>
        </tr>
        <tr>
         <br/>
         <br/>
         <td 5px"="" 5px;="" height="3D50" padding-right:="" style='3D"PADDING-
LEFT:' width='3D"100%"'>
          <p align="3Dcenter">
           <b>
            <font face="3DVerdana" size="3D5">
             <a href='3D"http://website.e365.cc/savequote/"'>
              Click Here For Your=

              Free Quote!
             </a>
            </font>
           </b>
          </p>
         </td>
         <p>
          <font face="3DVerdana" size="3D4">
           <strong>
            <center>
             Protecting your family is the best investment you'll eve=
 r
            make!
             <br/>
            </center>
           </strong>
          </font>
         </p>
        </tr>
       </tbody>
      </table>
     </td>
    </tr>
   </tbody>
  </table>
```

```
</center>
<tr>
 <br/>
 <br/>
</tr>
<tr>
</tr>
<p align="3Dleft">
 <font face='3D"Arial,' helvetica,="" sans-serif"="" size="3D2=">
 </font>
</p>
<p>
</p>
<center>
 <br/>
 <br/>
 <br/>
 <p>
 </p>
 <p align="3Dleft">
  <br/>
 </p>
</center>
<br/>
<br/>
<br/>
<br/>
<p align="3Dcenter">
 <br/>
</p>
<p align="3Dleft">
 <br/>
</p>
<br/>
<br/>
If you are in receipt of this=
  email
       in error and/or wish to be removed from our list,
<a href='3D"mailto:coins@btamail.net.cn"'>
 PLEASE CLICK HERE
</a>
AND TYPE =
 REMOVE. If you
       reside in any state which prohibits e-mail solicitations for insuran=
 ce,
       please disregard this
       email.
<br/>
```

```
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br/>
<br=>
 <br/>
 <br/>
 <br/>
</br=>
<!--=
 CENTER-->
```

[51]:
```
soup.get_text()
```

[51]:
```
"\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n<=\n /TR>\nSave up to 70% on Life Insurance.\nWhy
Spend More Than You Have To?\n \nLife Quote Savings\n
\n\n\n\n\n\n\n\n\n\nEnsurin=\n g your \n       family's financial security is
very important. Life Quote Savings ma=\n kes \n       buying life insurance
simple and affordable. We Provide FREE Access =\n to The \n       Very Best
Companies and The Lowest Rates.\n\n\n\n\n\nLife Quote Savings is FAST, EAS=\n Y
and \n             SAVES you money! Let us help you get started with the best
val=\n ues in \n             the country on new coverage. You can SAVE hundreds
or even tho=\n usands \n             of dollars by requesting a FREE quote from
Lifequote Savings. =\n Our \n             service will take you less than 5
minutes to complete. Shop an=\n d \n             compare. SAVE up to 70% on all
types of Life insurance! \n \n\n\nClick Here For Your=\n  \n             Free
Quote!\n\nProtecting your family is the best investment you'll eve=\n r \n
make!\n\n\n\n\n\n\n\n\nIf you are in receipt of this=\n  email \n       in error
and/or wish to be removed from our list, PLEASE CLICK HERE AND TYPE =\n REMOVE.
If you \n       reside in any state which prohibits e-mail solicitations for
insuran=\n ce, \n       please disregard this \n       email.\n"
```

1.2 Functions for Email Processing

[52]:
```
def clean_message(message,stemmer= PorterStemmer(),stop_words=set(stopwords.
↪words('english'))):

    # Converts to Lowercase and splits up the words
```

```
    words = word_tokenize( message.lower())
    filtered_words = []

    for word in words:
        # Removes stop words and punctuation
        if word not in stop_words and word.isalpha():
            filtered_words.append(stemmer.stem(word))

    return filtered_words
```

[53]:
```
clean_message(email_body)
```

[53]:
```
['dear',
 'mr',
 'still',
 'good',
 'tide',
 'staff',
 'festiv',
 'season',
 'ahead',
 'christma',
 'crux',
 'fulli',
 'qualifi',
 'santa',
 'clau',
 'wonder',
 'whether',
 'might',
 'consid',
 'run',
 'santa',
 'grotto',
 'store',
 'wait',
 'probabl',
 'think',
 'make',
 'special',
 'well',
 'first',
 'made',
 'sever',
 'chang',
 'characteris',
 'father',
```

```
'christma',
'rather',
'greet',
'children',
'shout',
'ho',
'ho',
'ho',
'prefer',
'whisper',
'phrase',
'depend',
'unfathom',
'cruel',
'world',
'live',
'addit',
'gift',
'rang',
'felt',
'hoop',
'holder',
'note',
'also',
'enclos',
'sketch',
'radic',
'redesign',
'santa',
'outfit',
'renam',
'charact',
'lord',
'buckl',
'would',
'interest',
'employ',
'promis',
'never',
'let',
'look',
'forward',
'hear',
'best',
'wish',
'robin',
'cooper',
```

```
 'excerpt',
 'book',
 'timewast',
 'letter',
 'robin',
 'cooper']
```

[54]:
```
def clean_msg_no_html(message,stemmer= PorterStemmer(),stop_words=set(stopwords.
↪words('english'))):

    # Remove html tags
    soup= BeautifulSoup(message,'html.parser')
    strings = soup.get_text()

    # Converts to Lowercase and splits up the words
    words = word_tokenize( strings.lower())
    filtered_words = []

    for word in words:
        # Removes stop words and punctuation
        if word not in stop_words and word.isalpha():
            filtered_words.append(stemmer.stem(word))

    return filtered_words
```

[55]:
```
clean_msg_no_html(data.at[0,'MESSAGE'])
```

[55]:
```
['save',
 'life',
 'insur',
 'spend',
 'life',
 'quot',
 'save',
 'g',
 'famili',
 'financi',
 'secur',
 'import',
 'life',
 'quot',
 'save',
 'ke',
 'buy',
 'life',
 'insur',
 'simpl',
```

```
'afford',
'provid',
'free',
'access',
'best',
'compani',
'lowest',
'rate',
'life',
'quot',
'save',
'fast',
'save',
'money',
'let',
'us',
'help',
'get',
'start',
'best',
'ue',
'countri',
'new',
'coverag',
'save',
'hundr',
'even',
'usand',
'dollar',
'request',
'free',
'quot',
'lifequot',
'save',
'servic',
'take',
'less',
'minut',
'complet',
'shop',
'compar',
'save',
'type',
'life',
'insur',
'click',
'free',
```

```
 'quot',
 'protect',
 'famili',
 'best',
 'invest',
 'r',
 'make',
 'receipt',
 'email',
 'error',
 'wish',
 'remov',
 'list',
 'pleas',
 'click',
 'type',
 'remov',
 'resid',
 'state',
 'prohibit',
 'solicit',
 'ce',
 'pleas',
 'disregard',
 'email']
```

2 Apply Cleaning and Tokenization to all messages

2.0.1 Slicing Dataframes and Series & Creating subsets

[56]:
```
data.iat[3,0]
```

[56]:
```
"##################################################\n #
#\n #                 Adult Club                     #\n #           Offers FREE
Membership               #\n #
#\n ##################################################\n \n >>>>>  INSTANT
ACCESS TO ALL SITES NOW\n >>>>>  Your User Name And Password is.\n >>>>>  User
Name: zzzz@spamassassin.taint.org\n >>>>>  Password: 760382\n \n 5 of the Best
Adult Sites on the Internet for FREE!\n
----------------------------------------\n NEWS 08/18/02\n With just over 2.9
Million Members that signed up for FREE, Last month there were 721,184 New\n
Members. Are you one of them yet???\n ----------------------------------------\n
Our Membership FAQ\n \n Q. Why are you offering free access to 5 adult
membership sites for free?\n A. I have advertisers that pay me for ad space so
you don't have to pay for membership.\n \n Q. Is it true my membership is for
life?\n A. Absolutely you'll never have to pay a cent the advertisers do.\n \n
Q. Can I give my account to my friends and family?\n A. Yes, as long they are
```

```
over the age of 18.\n \n Q. Do I have to sign up for all 5 membership sites?\n
A. No just one to get access to all of them.\n \n Q. How do I get started?\n A.
Click on one of the following links below to become a member.\n \n - These are
multi million dollar operations with policies and rules.\n - Fill in the
required info and they won't charge you for the Free pass!\n - If you don't
believe us, just read their terms and conditions.\n \n
--------------------------\n \n # 5. > Adults Farm\n
http://80.71.66.8/farm/?aid=760382\n Girls and Animals Getting Freaky…FREE
Lifetime Membership!!\n \n # 4. > Sexy Celebes\n
http://80.71.66.8/celebst/?aid=760382\n Thousands Of XXX Celebes doing it…FREE
Lifetime Membership!!\n \n # 3. > Play House Porn\n
http://80.71.66.8/mega/?aid=760382\n Live Feeds From 60 Sites And Web
Cams…FREE Lifetime Membership!!\n \n # 2. > Asian Sex Fantasies\n
http://80.71.66.8/asian/?aid=760382\n Japanese Schoolgirls, Live Sex Shows
…FREE Lifetime Membership!!\n \n # 1. > Lesbian Lace\n
http://80.71.66.8/lesbian/?aid=760382\n Girls and Girls Getting Freaky! …FREE
Lifetime Membership!!\n \n --------------------------\n \n Jennifer Simpson,
Miami, FL\n Your FREE lifetime membership has entertained my boyffriend and I
for\n the last two years!  Your Adult Sites are the best on the net!\n \n Joe
Morgan Manhattan, NY\n Your live sex shows and live sex cams are unbelievable.
The best part\n about your porn sites, is that they're absolutely FREE!\n \n
--------------------------\n \n \n \n \n \n \n \n \n Removal Instructions:\n
\n You have received this advertisement because you have opted in to receive
free adult internet\n offers and specials through our affiliated websites. If
you do not wish to receive further emails\n or have received the email in error
you may opt-out of our database here\n http://80.71.66.8/optout/ . Please allow
24 hours for removal.\n \n vonolmosatkirekpups\n \n"
```

[57]:
```
data.iloc[0:5]
```

[57]:

DOC_ID	MESSAGE	CATEGORY	\
0	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Tr…	1	
1	1) Fight The Risk of Cancer!\n http://www.adcl…	1	
2	1) Fight The Risk of Cancer!\n http://www.adcl…	1	
3	##…	1	
4	I thought you might like these:\n 1) Slim Down…	1	

DOC_ID	FILE_NAME
0	00001.7848dde101aa985090474a91ec93fcf0
1	00002.d94f1b97e48ed3b553b3508d116e6a09
2	00003.2ee33bc6eacdb11f38d052c44819ba6c
3	00004.eac8de8d759b7e74154f142194282724
4	00005.57696a39d7d84318ce497886896bf90d

[58]:
```
data.iloc[5:11]
```

[58]:

DOC_ID	MESSAGE	CATEGORY	FILE_NAME
5	A POWERHOUSE GIFTING PROGRAM You Don't Want To…	1	00006.5ab5620d3d7c6c0db76234556a16f6c1
6	Help wanted. We are a 14 year old fortune 500…	1	00007.d8521faf753ff9ee989122f6816f87d7
7	<html>\n <head>\n <title>ReliaQuote - Save Up …	1	00008.dfd941deb10f5eed78b1594b131c9266
8	TIRED OF THE BULL OUT THERE?\n Want To Stop Lo…	1	00009.027bf6e0b0c4ab34db3ce0ea4bf2edab
9	Dear ricardo1 ,\n \n <html>\n <body>\n <center…	1	00010.445affef4c70feec58f9198cfbc22997
10	Cellular Phone Accessories All At Below Wholes…	1	00011.61816b9ad167657773a427d890d0468e

[59]:
```
first_emails = data.MESSAGE.iloc[0:3]
nested_list = first_emails.apply(clean_msg_no_html)
```

[60]:
```
# flat_list=[]
# for sub_list in nested_list:
#     for item in sub_list:
#         flat_list.append(item)

flat_list=[item for sub_list in nested_list for item in sub_list]
len(flat_list)
```

[60]: 192

[61]:
```
%%time

# use apply() on all messages in the dataframe
nested_list = data.MESSAGE.apply(clean_msg_no_html)
```

```
C:\Users\BHANU\anaconda3\lib\site-packages\bs4\__init__.py:414:
MarkupResemblesLocatorWarning: "http://www.post-
gazette.com/columnists/20020905brian5
" looks like a URL. Beautiful Soup is not an HTTP client. You should probably
use an HTTP client like requests to get the document behind the URL, and feed
that document to Beautiful Soup.
  warnings.warn(
Wall time: 2min 8s
```

[62]:
```
nested_list.head()
```

[62]:
```
DOC_ID
0    [save, life, insur, spend, life, quot, save, g…
1    [fight, risk, cancer, http, slim, guarante, lo…
2    [fight, risk, cancer, http, slim, guarante, lo…
3    [adult, club, offer, free, membership, instant…
4    [thought, might, like, slim, guarante, lose, l…
Name: MESSAGE, dtype: object
```

[63]:
```
nested_list.tail()
```

[63]:
```
DOC_ID
5791    [http, bizarr, collect, stuf, anim, could, fet…
5792    [care, use, one, also, realli, cute, thing, ja…
5793    [sm, skip, montanaro, write, jeremi, put, anot…
5794    [mark, hammond, like, given, zodb, sound, attr…
5795    [hi, probabl, use, whatsoev, also, problem, re…
Name: MESSAGE, dtype: object
```

2.0.2 Using logic to slice dataframes

[64]:
```
data[data.CATEGORY == 1].shape
```

[64]:
```
(1896, 3)
```

[65]:
```
data[data.CATEGORY == 1].tail()
```

[65]:
```
                                                 MESSAGE  CATEGORY  \
DOC_ID
1891    <html>\n <head>\n <meta http-equiv="content-ty…         1
1892    This is a multi-part message in MIME format.\n…         1
1893    Dear Subscriber,\n \n If I could show you a wa…         1
1894    ****Mid-Summer Customer Appreciation SALE!****…         1
1895    ATTN:SIR/MADAN      \n \n                       …         1

                                    FILE_NAME
DOC_ID
1891    01396.e80a10644810bc2ae3c1b58c5fd38dfa
1892    01397.f75f0dd0dd923faefa3e9cc5ecb8c906
1893    01398.8ca7045aae4184d56e8509dc5ad6d979
1894    01399.2319643317e2c5193d574e40a71809c2
1895    01400.b444b69845db2fa0a4693ca04e6ac5c5
```

[66]:
```
# Challenge: indies of spam and ham mails
doc_ids_spam = data[data.CATEGORY == 1].index
doc_ids_ham = data[data.CATEGORY == 0].index
```

2.0.3 Subsetting a Series with an Index

[67]:
```
type(doc_ids_ham)
```

[67]:
```
pandas.core.indexes.numeric.Int64Index
```

[68]:
```
type(nested_list)
```

[68]:
```
pandas.core.series.Series
```

[69]:
```
nested_list_ham = nested_list.loc[doc_ids_ham]
nested_list_spam = nested_list.loc[doc_ids_spam]
```

[70]:
```
flat_list_ham = [item for sub_list in nested_list_ham for item in sub_list]
normal_words = pd.Series(data=flat_list_ham).value_counts()

normal_words.shape[0] # total number of unique words
```

[70]:
```
20815
```

[71]:
```
normal_words[:10]
```

[71]:
```
http      7563
use       3633
list      2880
one       2373
get       2286
mail      2255
would     2003
like      1931
messag    1849
work      1800
dtype: int64
```

[72]:
```
flat_list_spam = [item for sub_list in nested_list_spam for item in sub_list]
spammy_words = pd.Series(data=flat_list_spam).value_counts()

spammy_words.shape[0]
```

[72]:
```
13242
```

[73]:
```
spammy_words[:10]
```

[73]:
```
http      3097
email     3090
free      2585
click     2058
receiv    1989
```

```
list      1971
get       1914
pleas     1852
busi      1792
order     1746
dtype: int64
```

2.1 Creating a Word Cloud

[74]:
```
word_cloud = WordCloud().generate(email_body)
plt.imshow(word_cloud,interpolation='bilinear')
plt.axis('off')
plt.show()
```

[75]:
```
example_corpus = nltk.corpus.gutenberg.words('melville-moby_dick.txt')
len(example_corpus)
```

[75]: 260819

[76]:
```
type(example_corpus)
```

[76]: nltk.corpus.reader.util.StreamBackedCorpusView

[77]:
```
example_corpus
```

[77]: ['[', 'Moby', 'Dick', 'by', 'Herman', 'Melville', …]

[78]:
```
word_list = [''.join(word) for word in example_corpus]
novel_as_string = ' '.join(word_list)
```

[79]:
```
icon = Image.open(WHALE_FILE)
image_mask = Image.new('RGB',size=icon.size,color=(255,255,255))
image_mask.paste(icon,box = icon)
rgb_array = np.array(image_mask) # converts image object to array

word_cloud =␣
 ↪WordCloud(mask=rgb_array,background_color='white',max_words=400,colormap='ocean')
word_cloud.generate(novel_as_string)

plt.figure(figsize=[16,8])
plt.imshow(word_cloud,interpolation='bilinear')
plt.axis('off')
plt.show()
```

[80]:
```
rgb_array.shape
```

[80]:
```
(1024, 2048, 3)
```

[81]:
```
rgb_array[1023,2047]
```

[81]:
```
array([255, 255, 255], dtype=uint8)
```

[82]:
```
rgb_array[500,1000]
```

[82]:
```
array([0, 0, 0], dtype=uint8)
```

[83]:
```
#Challenge: Use skull image for wordcloud of shakespeare's play hamlet
hamlet = nltk.corpus.shakespeare.words('hamlet.xml')
```

```
hamlet_list = [''.join(word) for word in hamlet]
hamlet_as_string = ' '.join(hamlet_list)
```

[84]:
```
skull_icon = Image.open(SKULL_FILE)
simage_mask = Image.new('RGB',size=skull_icon.size,color=(255,255,255))
simage_mask.paste(skull_icon,box = skull_icon)
srgb_array = np.array(simage_mask) # converts image object to array

sword_cloud =␣
 ↪WordCloud(mask=srgb_array,background_color='white',max_words=600,colormap='bone')
sword_cloud.generate(hamlet_as_string)

plt.figure(figsize=[16,8])
plt.imshow(sword_cloud,interpolation='bilinear')
plt.axis('off')
plt.show()
```

3 Word Cloud of Ham and spam messages

[85]:
```
icon = Image.open(THUMBS_UP_FILE)
image_mask = Image.new('RGB',size=icon.size,color=(255,255,255))
image_mask.paste(icon,box = icon)
rgb_array = np.array(image_mask) # converts image object to array

#Generate string
ham_str = ' '.join(flat_list_ham)
```

```
#output due to stemmed word
word_cloud =␣
 ↪WordCloud(mask=rgb_array,background_color='white',max_words=500,colormap='winter',font_path
word_cloud.generate(ham_str)


plt.figure(figsize=[16,8])
plt.imshow(word_cloud,interpolation='bilinear')
plt.axis('off')
plt.show()
```

[86]:
```
icon = Image.open(THUMBS_DOWN_FILE)
image_mask = Image.new('RGB',size=icon.size,color=(255,255,255))
image_mask.paste(icon,box = icon)
rgb_array = np.array(image_mask) # converts image object to array

#Generate string
spam_str = ' '.join(flat_list_spam)

word_cloud =␣
 ↪WordCloud(mask=rgb_array,background_color='white',max_words=2000,colormap='gist_heat',font_
word_cloud.generate(spam_str)


plt.figure(figsize=[16,8])
plt.imshow(word_cloud,interpolation='bilinear')
plt.axis('off')
plt.show()
```

4 Generate Vocabulary and Dictionary

[87]:
```
stemmed_nested_list = data.MESSAGE.apply(clean_msg_no_html)
flat_stemmed_list = [item for sublist in stemmed_nested_list for item in␣
 ↪sublist]
```

```
C:\Users\BHANU\anaconda3\lib\site-packages\bs4\__init__.py:414:
MarkupResemblesLocatorWarning: "http://www.post-
gazette.com/columnists/20020905brian5
" looks like a URL. Beautiful Soup is not an HTTP client. You should probably
```

```
use an HTTP client like requests to get the document behind the URL, and feed
that document to Beautiful Soup.
  warnings.warn(
```

[88]:
```
unique_words = pd.Series(flat_stemmed_list).value_counts()
print('No. of unique words',unique_words.shape[0])
unique_words.head()
```

```
No. of unique words 27334
```

[88]:
```
http     10660
use       5019
list      4851
email     4367
get       4200
dtype: int64
```

[89]:
```
frequent_words = unique_words[:VOCAB_SIZE]
print('Most Common Words: \n',frequent_words[:10])
```

```
Most Common Words:
 http     10660
use       5019
list      4851
email     4367
get       4200
mail      3983
one       3907
free      3202
time      3042
work      2883
dtype: int64
```

4.1 Create a Vocabulary DataFrame eith a WORD_ID

[90]:
```
word_ids = list(range(0,VOCAB_SIZE))
vocab = pd.DataFrame({"VOCAB_WORD":frequent_words.index.values},index= word_ids)
vocab.index.name = 'WORD_ID'
vocab.head()
```

[90]:
```
         VOCAB_WORD
WORD_ID
0              http
1               use
2              list
3             email
4               get
```

4.2 Save the vocabulary as csv file

[91]:
```
vocab.to_csv(WORD_ID_FILE,index_label=vocab.index.name,header=vocab.VOCAB_WORD.
↪name)
```

4.3 Exercise: Checking if the word is Part of the vocabulary

[92]:
```
any(vocab.VOCAB_WORD == 'machine') #inefficient
```

[92]:
```
False
```

[93]:
```
# 'machine' in set(vocab.VOCAB_WORD)
# 'learn' in set(vocab.VOCAB_WORD)
# 'fun' in set(vocab.VOCAB_WORD)
# 'data' in set(vocab.VOCAB_WORD)
# 'science' in set(vocab.VOCAB_WORD)
# 'app' in set(vocab.VOCAB_WORD)
'brewery' in set(vocab.VOCAB_WORD)
```

[93]:
```
False
```

4.4 Exercise: Find the email with most number of words

[94]:
```
# Python list comprehension
clean_email_length = [len(sublist) for sublist in stemmed_nested_list]
print('No. of stemmed words in the longest mail',max(clean_email_length))
```

```
No. of stemmed words in the longest mail 7671
```

[95]:
```
print('Email position in the list (and the dataframe)',np.
↪argmax(clean_email_length))
```

```
Email position in the list (and the dataframe) 5775
```

[96]:
```
stemmed_nested_list[np.argmax(clean_email_length)][:5]
```

[96]:
```
['yahoo', 'group', 'sponsor', 'dvd', 'free']
```

[97]:
```
data.at[np.argmax(clean_email_length),'MESSAGE']
```

[97]:
```
: Archive format has changed. Make sure you\n        synchronize your replicas
before upgrading, to avoid spurious\n        conflicts. The first sync after
upgrading will be slow.\n      * New/improved functionality:\n           + A new
preference -sortbysize controls the order in which\n             changes are
displayed to the user: when it is set to true,\n             the smallest
changed files are displayed first. (The default\n              setting is
false.)\n           + A new preference -sortnewfirst causes newly created files
to\n             be listed before other updates in the user interface.\n
+ We now allow the ssh protocol to specify a port.\n           + Incompatible
```

```
change: The unison: protocol is deprecated, and\n             we added file: and
socket:. You may have to modify your\n             profiles in the .unison
directory. If a replica is specified\n             without an explicit protocol,
we now assume it refers to a\n             file. (Previously "//saul/foo" meant
to use SSH to connect to\n             saul, then access the foo directory. Now
it means to access\n             saul via a remote file mechanism such as samba;
the old\n             effect is now achieved by writing ssh://saul/foo.)\n
+ Changed the startup sequence for the case where roots are\n             given
but no profile is given on the command line. The new\n             behavior is
to use the default profile (creating it if it\n             does not exist), and
temporarily override its roots. The\n             manual claimed that this case
would work by reading no\n             profile at all, but AFAIK this was never
true.\n           + In all user interfaces, files with conflicts are always\n
listed first\n           + A new preference \'sshversion\' can be used to
control which\n             version of ssh should be used to connect to the
server. Legal\n             values are 1 and 2. (Default is empty, which will
make unison\n             use whatever version of ssh is installed as the
default \'ssh\'\n             command.)\n           + The situation when the
permissions of a file was updated the\n             same on both side is now
handled correctly (we used to report\n             a spurious conflict)\n      *
Improvements for the Windows version:\n           + The fact that filenames are
treated case-insensitively under\n             Windows should now be handled
correctly. The exact behavior\n             is described in the cross-platform
section of the manual.\n           + It should be possible to synchronize with
Windows shares,\n             e.g., //host/drive/path.\n           + Workarounds
to the bug in syncing root directories in\n             Windows. The most
difficult thing to fix is an ocaml bug:\n             Unix.opendir fails on c:
in some versions of Windows.\n      * Improvements to the GTK user interface
(the Tk interface is no\n        longer being maintained):\n           + The UI
now displays actions differently (in blue) when they\n             have been
explicitly changed by the user from Unison\'s\n             default
recommendation.\n           + More colorful appearance.\n           + The
initial profile selection window works better.\n           + If any transfers
failed, a message to this effect is\n             displayed along with
\'Synchronization complete\' at the end of\n             the transfer phase (in
case they may have scrolled off the\n             top).\n           + Added a
global progress meter, displaying the percentage of\n             total bytes
that have been transferred so far.\n      * Improvements to the text user
interface:\n           + The file details will be displayed automatically when
a\n             conflict is been detected.\n           + when a warning is
generated (e.g. for a temporary file left\n             over from a previous run
of unison) Unison will no longer\n             wait for a response if it is
running in -batch mode.\n           + The UI now displays a short list of
possible inputs each time\n             it waits for user interaction.\n
+ The UI now quits immediately (rather than looping back and\n
starting the interaction again) if the user presses \'q\' when\n
asked whether to propagate changes.\n           + Pressing \'g\' in the text
```

```
user interface will proceed\n             immediately with propagating updates,
without asking any more\n             questions.\n      * Documentation and
installation changes:\n           + The manual now includes a FAQ, plus sections
on common\n             problems and on tricks contributed by users.\n
+ Both the download page and the download directory explicitly\n             say
what are the current stable and beta-test version\n             numbers.\n
+ The OCaml sources for the up-to-the-minute developers\'\n             version
(not guaranteed to be stable, or even to compile, at\n             any given
time!) are now available from the download page.\n           + Added a
subsection to the manual describing cross-platform\n             issues (case
conflicts, illegal filenames)\n      * Many small bug fixes and random
improvements.\n        \n    Changes since 2.3.1:\n      * Several bug fixes.
The most important is a bug in the rsync module\n        that would occasionally
cause change propagation to fail with a\n        \'rename\' error.\n        \n
Changes since 2.2:\n      * The multi-threaded transport system is now disabled
by default.\n        (It is not stable enough yet.)\n      * Various bug
fixes.\n      * A new experimental feature:\n        The final component of a
-path argument may now be the wildcard\n        specifier *. When Unison sees
such a path, it expands this path on\n        the client into into the
corresponding list of paths by listing\n        the contents of that
directory.\n        Note that if you use wildcard paths from the command line,
you\n        will probably need to use quotes or a backslash to prevent the *\n
from being interpreted by your shell.\n        If both roots are local, the
contents of the first one will be\n        used for expanding wildcard paths.
(Nb: this is the first one\n        after the canonization step -- i.e., the one
that is listed first\n        in the user interface -- not the one listed first
on the command\n        line or in the preferences file.)\n        \n    Changes
since 2.1:\n      * The transport subsystem now includes an implementation by
Sylvain\n        Gommier and Norman Ramsey of Tridgell and Mackerras\'s rsync\n
protocol. This protocol achieves much faster transfers when only a\n
small part of a large file has been changed by sending just diffs.\n        This
feature is mainly helpful for transfers over slow links---on\n        fast local
area networks it can actually degrade performance---so\n        we have left it
off by default. Start unison with the -rsync\n        option (or put rsync=true
in your preferences file) to turn it on.\n      * ``Progress bars\'\' are now
diplayed during remote file transfers,\n        showing what percentage of each
file has been transferred so far.\n      * The version numbering scheme has
changed. New releases will now be\n        have numbers like 2.2.30, where the
second component is\n        incremented on every significant public release and
the third\n        component is the ``patch level.\'\'\n      * Miscellaneous
improvements to the GTK-based user interface.\n      * The manual is now
available in PDF format.\n      * We are experimenting with using a multi-
threaded transport\n        subsystem to transfer several files at the same
time, making much\n        more effective use of available network bandwidth.
This feature is\n        not completely stable yet, so by default it is disabled
in the\n        release version of Unison.\n        If you want to play with the
multi-threaded version, you\'ll need\n        to recompile Unison from sources
```

(as described in the\n documentation), setting the THREADS flag in Makefile.OCaml to\n true. Make sure that your OCaml compiler has been installed with\n the -with-pthreads configuration option. (You can verify this by\n checking whether the file threads/threads.cma in the OCaml\n standard library directory contains the string -lpthread near the\n end.)\n \n Changes since 1.292:\n * Reduced memory footprint (this is especially important during the\n first run of unison, where it has to gather information about all\n the files in both repositories).\n * Fixed a bug that would cause the socket server under NT to fail\n after the client exits.\n * Added a SHIFT modifier to the Ignore menu shortcut keys in GTK\n interface (to avoid hitting them accidentally).\n \n Changes since 1.231:\n * Tunneling over ssh is now supported in the Windows version. See\n the installation section of the manual for detailed instructions.\n * The transport subsystem now includes an implementation of the\n rsync protocol, built by Sylvain Gommier and Norman Ramsey. This\n protocol achieves much faster transfers when only a small part of\n a large file has been changed by sending just diffs. The rsync\n feature is off by default in the current version. Use the -rsync\n switch to turn it on. (Nb. We still have a lot of tuning to do:\n you may not notice much speedup yet.)\n * We\'re experimenting with a multi-threaded transport subsystem,\n written by Jerome Vouillon. The downloadable binaries are still\n single-threaded: if you want to try the multi-threaded version,\n you\'ll need to recompile from sources. (Say make THREADS=true.)\n Native thread support from the compiler is required. Use the\n option -threads N to select the maximal number of concurrent\n threads (default is 5). Multi-threaded and single-threaded\n clients/servers can interoperate.\n * A new GTK-based user interface is now available, thanks to Jacques\n Garrigue. The Tk user interface still works, but we\'ll be shifting\n development effort to the GTK interface from now on.\n * OCaml 3.00 is now required for compiling Unison from sources. The\n modules uitk and myfileselect have been changed to use labltk\n instead of camltk. To compile the Tk interface in Windows, you\n must have ocaml-3.00 and tk8.3. When installing tk8.3, put it in\n c:\\Tcl rather than the suggested c:\\Program Files\\Tcl, and be sure\n to install the headers and libraries (which are not installed by\n default).\n * Added a new -addversionno switch, which causes unison to use\n unison-<currentversionnumber> instead of just unison as the remote\n server command. This allows multiple versions of unison to coexist\n conveniently on the same server: whichever version is run on the\n client, the same version will be selected on the server.\n \n Changes since 1.219:\n * INCOMPATIBLE CHANGE: Archive format has changed. Make sure you\n synchronize your replicas before upgrading, to avoid spurious\n conflicts. The first sync after upgrading will be slow.\n * This version fixes several annoying bugs, including:\n + Some cases where propagation of file permissions was not\n working.\n + umask is now ignored when creating directories\n + directories are create writable, so that a read-only\n directory and its contents

can be propagated.\n + Handling of warnings generated by the server.\n + Synchronizing a path whose parent is not a directory on both\n sides is now flagged as erroneous.\n + Fixed some bugs related to symnbolic links and nonexistant\n roots.\n o When a change (deletion or new contents) is propagated\n onto a \'follow\'ed symlink, the file pointed to by the\n link is now changed. (We used to change the link itself,\n which doesn\'t fit our assertion that \'follow\' means the\n link is completely invisible)\n o When one root did not exist, propagating the other root\n on top of it used to fail, becuase unison could not\n calculate the working directory into which to write\n changes. This should be fixed.\n * A human-readable timestamp has been added to Unison\'s archive\n files.\n * The semantics of Path and Name regular expressions now correspond\n better.\n * Some minor improvements to the text UI (e.g. a command for going\n back to previous items)\n * The organization of the export directory has changed --- should be\n easier to find / download things now.\n \n Changes since 1.200:\n * INCOMPATIBLE CHANGE: Archive format has changed. Make sure you\n synchronize your replicas before upgrading, to avoid spurious\n conflicts. The first sync after upgrading will be slow.\n * This version has not been tested extensively on Windows.\n * Major internal changes designed to make unison safer to run at the\n same time as the replicas are being changed by the user.\n * Internal performance improvements.\n \n Changes since 1.190:\n * INCOMPATIBLE CHANGE: Archive format has changed. Make sure you\n synchronize your replicas before upgrading, to avoid spurious\n conflicts. The first sync after upgrading will be slow.\n * A number of internal functions have been changed to reduce the\n amount of memory allocation, especially during the first\n synchronization. This should help power users with very big\n replicas.\n * Reimplementation of low-level remote procedure call stuff, in\n preparation for adding rsync-like smart file transfer in a later\n release.\n * Miscellaneous bug fixes.\n \n Changes since 1.180:\n * INCOMPATIBLE CHANGE: Archive format has changed. Make sure you\n synchronize your replicas before upgrading, to avoid spurious\n conflicts. The first sync after upgrading will be slow.\n * Fixed some small bugs in the interpretation of ignore patterns.\n * Fixed some problems that were preventing the Windows version from\n working correctly when click-started.\n * Fixes to treatment of file permissions under Windows, which were\n causing spurious reports of different permissions when\n synchronizing between windows and unix systems.\n * Fixed one more non-tail-recursive list processing function, which\n was causing stack overflows when synchronizing very large\n replicas.\n \n Changes since 1.169:\n * The text user interface now provides commands for ignoring files.\n * We found and fixed some more non-tail-recursive list processing\n functions. Some power users have reported success with very large\n replicas.\n * INCOMPATIBLE CHANGE: Files ending in .tmp are no longer ignored\n automatically. If

you want to ignore such files, put an\n appropriate ignore pattern in your profile.\n * INCOMPATIBLE CHANGE: The syntax of ignore and follow patterns has\n changed. Instead of putting a line of the form\n \n ignore = <regexp>\n in your profile (.unison/default.prf), you should put:\n \n ignore = Regexp <regexp>\n Moreover, two other styles of pattern are also recognized:\n \n ignore = Name <name>\n matches any path in which one component matches <name>, while\n \n ignore = Path <path>\n matches exactly the path <path>.\n Standard ``globbing\'\' conventions can be used in <name> and\n <path>:\n + a ? matches any single character except /\n + a * matches any sequence of characters not including /\n + [xyz] matches any character from the set {x, y, z }\n + {a,bb,ccc} matches any one of a, bb, or ccc.\n See the user manual for some examples.\n \n Changes since 1.146:\n * Some users were reporting stack overflows when synchronizing huge\n directories. We found and fixed some non-tail-recursive list\n processing functions, which we hope will solve the problem. Please\n give it a try and let us know.\n * Major additions to the documentation.\n \n Changes since 1.142:\n * Major internal tidying and many small bugfixes.\n * Major additions to the user manual.\n * Unison can now be started with no arguments -- it will prompt\n automatically for the name of a profile file containing the roots\n to be synchronized. This makes it possible to start the graphical\n UI from a desktop icon.\n * Fixed a small bug where the text UI on NT was raising a \'no such\n signal\' exception.\n \n Changes since 1.139:\n * The precompiled windows binary in the last release was compiled\n with an old OCaml compiler, causing propagation of permissions not\n to work (and perhaps leading to some other strange behaviors we\'ve\n heard reports about). This has been corrected. If you\'re using\n precompiled binaries on Windows, please upgrade.\n * Added a -debug command line flag, which controls debugging of\n various modules. Say -debug XXX to enable debug tracing for module\n XXX, or -debug all to turn on absolutely everything.\n * Fixed a small bug where the text UI on NT was raising a \'no such\n signal\' exception.\n \n Changes since 1.111:\n * INCOMPATIBLE CHANGE: The names and formats of the preference files\n in the .unison directory have changed. In particular:\n + the file ``prefs\'\' should be renamed to default.prf\n + the contents of the file ``ignore\'\' should be merged into\n default.prf. Each line of the form REGEXP in ignore should\n become a line of the form ignore = REGEXP in default.prf.\n * Unison now handles permission bits and symbolic links. See the\n manual for details.\n * You can now have different preference files in your .unison\n directory. If you start unison like this\n \n unison profilename\n (i.e. with just one ``anonymous\'\' command-line argument), then the\n file ~/.unison/profilename.prf will be loaded instead of\n default.prf.\n * Some improvements to terminal handling in the text user interface\n * Added a switch -killServer that terminates the remote server\n process when the unison client is shutting down, even when using\n sockets for

```
communication. (By default, a remote server created\n        using ssh/rsh is
terminated automatically, while a socket server\n        is left running.)\n
* When started in \'socket server\' mode, unison prints \'server\n
started\' on stderr when it is ready to accept connections. (This\n        may
be useful for scripts that want to tell when a socket-mode\n        server has
finished initalization.)\n      * We now make a nightly mirror of our current
internal development\n        tree, in case anyone wants an up-to-the-minute
version to hack\n        around with.\n      * Added a file CONTRIB with some
suggestions for how to help us make\n        Unison better.\n        \n \n \n To
unsubscribe from this group, send an email to:\n unison-announce-
unsubscribe@egroups.com\n \n  \n \n Your use of Yahoo! Groups is subject to
http://docs.yahoo.com/info/terms/ \n \n \n \n'
```

5 Generate Features and Sparse Matrix

5.1 Creating a dataframe with one word per column

[98]:
```
type(stemmed_nested_list)
```

[98]:
```
pandas.core.series.Series
```

[99]:
```
word_columns_df = pd.DataFrame.from_records(stemmed_nested_list.to_list())
```

[100]:
```
word_columns_df.shape
```

[100]:
```
(5796, 7671)
```

5.1.1 Splitting the data into a taraining and Testing Dataset

[101]:
```
X_train,X_test,y_train,y_test = train_test_split(word_columns_df,data.
↪CATEGORY,test_size=0.3,random_state=42)
```

[102]:
```
print('No of training samples ',X_train.shape[0])
print('Fraction of training set ',X_train.shape[0]/word_columns_df.shape[0])
```

```
No of training samples  4057
Fraction of training set  0.6999654934437544
```

[103]:
```
X_train.index.name = X_test.index.name = 'DOC_ID'
X_test.head()
```

[103]:

DOC_ID	0	1	2	3	4	5	6
4675	interest	alway	wonder	thing	bad	exampl	goto
4220	url	http	date	final	gdc	europ	review
2484	stephen	william	mailto	swilliam	weaken	food	transact
2418	el	mon	sep	bitbitch	wrote	eugen	mani
5110	music	school	joke	american	conductor	european	conductor

```
             7         8             9     …  7661  7662  7663  7664  7665  \
DOC_ID                                       …
4675      languag   support          goto  …  None  None  None  None  None
4220    confernec   session           ect  …  None  None  None  None  None
2484     argument      note  neighborhood  …  None  None  None  None  None
2418         homo    friend         lover  …  None  None  None  None  None
5110         talk  european     conductor  …  None  None  None  None  None

        7666  7667  7668  7669  7670
DOC_ID
4675    None  None  None  None  None
4220    None  None  None  None  None
2484    None  None  None  None  None
2418    None  None  None  None  None
5110    None  None  None  None  None

[5 rows x 7671 columns]
```

[104]:
```
y_test.head()
```

[104]:
```
DOC_ID
4675    0
4220    0
2484    0
2418    0
5110    0
Name: CATEGORY, dtype: int64
```

5.1.2 Create a sparse matrix to training data

[105]:
```
word_index = pd.Index(vocab.VOCAB_WORD)
type(word_index[3])
```

[105]:
```
str
```

[106]:
```
word_index.get_loc('thu')
```

[106]:
```
392
```

[107]:
```
def make_sparse_matrix(df,indexed_words,labels):
    """
    Returns sparse matrix as DataFrames

    df: A Dataframe with words in columns with a document id as an␣
 ↪index(X_train or X_test)
    indexed_words: index of words ordered by order id
```

```
    labels: Category as a series(y_train or y_test)
    """

    nr_rows = df.shape[0]
    nr_cols = df.shape[1]
    word_set = set(indexed_words)
    dict_list = []

    for i in range(nr_rows):
        for j in range(nr_cols):

            word = df.iat[i,j]
            if word in word_set:
                doc_id = df.index[i]
                word_id = indexed_words.get_loc(word)
                category = labels.at[doc_id]

                item = {'LABEL':category,'DOC_ID':doc_id,'OCCURENCE':
↪1,'WORD_ID':word_id}

                dict_list.append(item)

    return pd.DataFrame(dict_list)
```

[108]:
```
%%time
sparse_train_df = make_sparse_matrix(X_train,word_index,y_train)
```

```
Wall time: 8min 52s
```

[109]:
```
sparse_train_df.shape
```

[109]:
```
(429235, 4)
```

[110]:
```
sparse_train_df[-5:]
```

[110]:

	LABEL	DOC_ID	OCCURENCE	WORD_ID
429230	1	860	1	47
429231	1	860	1	1430
429232	1	860	1	26
429233	1	860	1	21
429234	1	860	1	126

5.1.3 Combine Occurences with the pandas groupby() Method

[111]:
```
train_grouped = sparse_train_df.groupby(['DOC_ID','WORD_ID','LABEL']).sum()
train_grouped.head()
```

[111]:
```
                     OCCURENCE
DOC_ID WORD_ID LABEL
0      2       1              1
       3       1              2
       4       1              1
       7       1              3
       11      1              1
```

[112]:
```
vocab.at[3,'VOCAB_WORD']
```

[112]:
```
'email'
```

[113]:
```
train_grouped = train_grouped.reset_index()
train_grouped.head()
```

[113]:
```
   DOC_ID  WORD_ID  LABEL  OCCURENCE
0       0        2      1          1
1       0        3      1          2
2       0        4      1          1
3       0        7      1          3
4       0       11      1          1
```

[114]:
```
train_grouped.tail()
```

[114]:
```
        DOC_ID  WORD_ID  LABEL  OCCURENCE
258359    5795     2040      0          2
258360    5795     2041      0          1
258361    5795     2277      0          1
258362    5795     2325      0          1
258363    5795     2381      0          1
```

[115]:
```
vocab.at[2038,'VOCAB_WORD']
```

[115]:
```
'experienc'
```

[116]:
```
data.MESSAGE[5795]
```

[116]:
```
"Hi there,\n \n Now this is probably of no use to you whatsoever, but…\n \n
Not a deb-head, but I also have a problem regards NVidia. I've two machines,\n
pretty much the same hardware. \n athlon 1700XP's, both have NVidia cards, one
Gforce3 Ti and a Gforce2MX,\n both use the same driver module. \n \n Both have
same kernel version albeit it compiled for their particular\n hardware.\n \n One
works perfectly, the other fails to load. When I check with lsmod, I can\n see
the NVdriver is loaded, but not used ;--(\n Thus when I startx, it bombs out. \n
\n IU still have X with the default nv driver and crappy accelleration -->\n
650fps with glxgears [should be over 2000fps]\n Its not a hardware issue with
the cards as I swapped them over and get the\n same symptoms. \n \n I reckon my
mobo is cack, I've tried swapping slots around, reserving\n resources etc all to
```

```
no avail. \n Should there be an interrupt for the nv card? I haven't checked the
other\n box yet.\n \n Regards,\n CW\n \n ------------------\n On Tue, 03 Dec
2002, Mark Page wrote:\n \n > Was running Debian Woody perfectly well until I
installed the NVIDIA 3D \n > tar packages.\n \n You probably stil are ;)\n \n >
Both the kernel and GLX packages installed with no problem and I then \n >
amended the XFree config file appropriately (according to all \n >
instructions). Neither GDM nor KDM will now fire up returning me to the \n >
console screen. \n \n ie X is broken. \n \n > Running startx I isolated the
problem to a failure to load the nvidia\n > kernel module - By cd'ing to the
kernel module file and re-running 'make\n > install' I can get a workable
xserver \n \n so you fixed X.\n \n > minus the preferences you would get from
either gdm or kdm and to be\n > honest the GUI is horrible.\n \n If X is working
and you get a brief Nvidia splash screen then this is\n likely nothing to do
with the driver itself.  You should be able to use kdm\n or gdm with the nvidia
Xserver.  What exactly did you change in your\n XF86Config?  Will gdm/kdm not
start?\n \n > I followed the nvidia suggestion of 'make install \n >
SYSINCLUDE=/path/to/kernel/headers' and whilst this appears to install \n > ok
upon reboot I am back to the same console login and having to go \n > through
the same reinstall of the kernel module. I have tried rewriting \n > the XF
config file to it's original state with no success.\n \n Well, this sounds like
the NVIDIA kernel module isn't loading on boot.\n When it *is* working do a
quick \n \n /sbin/lsmod\n \n and look for the module name (something like
'NVDriver').  Then put this on\n a new line in \n \n /etc/modules\n \n in order
that linux will load it on boot.  You can just do\n \n modprobe nvdriver\n \n to
load it by hand.  Your make install did this as well as reinstalling the\n
driver in /lib etc.  You should also get a message saying loading the\n nvidia
driver taints your kernel (which it does, in the sense that it's not\n open
source).\n \n > Is this a problem anyone else has encountered and what is the
best \n > solution? Can I rid myself of the tar file installations and find some
\n > specific .deb packages?\n \n The nvidia driver source is in the Woody
tree.\n \n http://packages.debian.org/stable/x11/nvidia-kernel-src.html\n \n so
you can do\n \n apt-get install nvidia-kernel-src\n \n and then compile,
although I've always gone the route you're doing now.\n \n Ideally the nvidia
driver would be a part of the kernel from day 1.\n However, nvidia have not open
sourced it (most of the above source is\n pre-compiled object code), but in
order for it to work with your specific\n kernel you must compile against your
kernel headers (as you did).  As\n debian has lots of potential kernels they
can't have one nvdriver package\n and have chosen instead to provide the source.
Actually I don't know if\n they're permitted to distribute binaries by nvidia's
license anyway.\n \n Gavin\n \n -- \n Irish Linux Users' Group: ilug@linux.ie\n
http://www.linux.ie/mailman/listinfo/ilug for (un)subscription information.\n
List maintainer: listmaster@linux.ie\n \n -- \n Irish Linux Users' Group:
ilug@linux.ie\n http://www.linux.ie/mailman/listinfo/ilug for (un)subscription
information.\n List maintainer: listmaster@linux.ie\n \n \n"
```

[117]:
```
train_grouped.shape
```

[117]: (258364, 4)

5.1.4 Save training Data as .txt file

[118]:
```
np.savetxt(TRAINING_DATA_FILE,train_grouped,fmt="%d")
```

[119]:
```
train_grouped.columns
```

[119]: Index(['DOC_ID', 'WORD_ID', 'LABEL', 'OCCURENCE'], dtype='object')

[120]:
```
# Challenge do same for test data
sparse_test_df = make_sparse_matrix(X_test,word_index,y_test)
```

[121]:
```
test_grouped = sparse_test_df.groupby(['DOC_ID','WORD_ID','LABEL']).sum()
test_grouped = test_grouped.reset_index()
np.savetxt(TEST_DATA_FILE,test_grouped,fmt="%d")
```

5.2 Pre-Processing Subtleties and Checking the Understanding

Challenge: We started with 5796 emails. We split it into 4057 emails for training and 1739 emails for testing. How many individual emails were included in testing.txt file? Count the number in the test_grouped DataFrame. After splitting and shifting our data, how many emails were included in the X_test_dataframe?Is the number same? If not which emails were excluded and why? Compare DOC_ID values to findout

[122]:
```
train_doc_ids= set(train_grouped.DOC_ID)
test_doc_ids= set(test_grouped.DOC_ID)
```

[123]:
```
len(test_doc_ids)
```

[123]: 1724

[124]:
```
len(X_test)
```

[124]: 1739

[125]:
```
set(X_test.index.values) - test_doc_ids
```

[125]: {134, 179, 240, 274, 298, 339, 439, 471, 670, 734, 765, 945, 1544, 1670, 1700}

[126]:
```
data.MESSAGE[134]
```

[126]:
```
'------=_NextPart_000_00E8_85C13B1D.B7243B86\n Content-Type: text/html;
charset="iso-8859-1"\n Content-Transfer-Encoding: base64\n \n \n
PGh0bWw+DQoNCjxib2R5IGJnY29sb3I9IiNGRkZGRkYiIHRleHQ9IiMwMDAw\n
MDAiPiANCjxwIGFsaWduPSJjZW50ZXIiPjxhIGhyZWY9Imh0dHA6Ly93d3cu\n
ZGlyZWN0d2Vic3RvcmUuY29tL3RveXMvaW5kZXguaHRtbCI+PGltZyBzcmM9\n
Imh0dHA6Ly93d3cuZGlyZWN0d2Vic3RvcmUuY29tL21waWMuanBnIiB3aWR0\n
```

```
aD0iNTAwIiBoZWlnaHQ9IjMzOSIgYm9yZGVyPSIwIj48L2E+PGJyPiANCjxm\n
b250IHNpemU9IjMiIGZhY2U9IkFyaWFsLCBIZWx2ZXRpY2EsIHNhbnMtc2Vy\n
aWYiPjxhIGhyZWY9Imh0dHA6Ly93d3cuZGlyZWN0d2Vic3RvcmUuY29tL3Rv\n
eXMvaW5kZXguaHRtbCI+PGI+RU5URVIgDQpOT1cgaWYgeW91IGFyZSAxOCBh\n
bmQgb3Zlcjwvb48L2E+PC9mb250PjwvcD4gDQo8cCBhbGlnbj0iY2VudGVy\n
Ij48Zm9udCBmYWNlPSJBcmlhbCwgSGVsdmV0aWNhLCBzYW5zLXNlcmlmIiBz\n
aXplPSI0IiBjb2xvcj0iI0ZGMDAwMCI+PGI+U1BFQ0lBTCANCk9GRkVSPGJy\n
PiANCjxmb250IHNpemU9IiBUiPjMwIERheXMgPGk+RlJFRSBhY2Nlc3M8L2k+\n
IDwvZm9udD48YnI+IA0KdG8gdGhlIGxhcmdlc3QgQWR1bHRzaXRlIG9uIHRo\n
ZSB3ZWIuPGJyPiANCjwvYj4gPGZvbnQgc2l6ZT0iMiI+Zm9yIG9yZGVycyBv\n
dmVyICQxMDA8L2ZvbnQ+PC9mb250PjwvcD4gDQo8cCBhbGlnbj0iY2VudGVy\n
Ij48Zm9udCBzaXplPSIyIiBmYWNlPSJBcmlhbCwgSGVsdmV0aWNhLCBzYW5z\n
LXNlcmlmIj48Yj48Zm9udCBzaXplPSI0IiBjb2xvcj0iIzAwMDBGRiI+UmVh\n
ZHkgDQp0byBnbyBTaG9wcGluZz8gPC9mb250PjwvYj48YnI+IA0KaW4gYWW91IGNh\n
biBmZWVsIHNhZmUgc2hvcHBpbmcgb3VyIHNlY3VyZSBvbmxpbmUgc3RvcmUu\n
IDxicj4gDQpZb3VyIHByaXZhY3kgYW5kIGNvbmZpZGVudGlhbGl0eSBpcyBv\n
dXIgcHJpb3JpdHkuIDxicj4gDQpBbGwgb3JkZXJzIGFyZSBwYWNrZWQgaW4g\n
YSBkaXNjcmVldCBwcmlvcml0eSBtYWlsIGJveC4gPGJyPiANCldlIG5ldmVy\n
IHNoYXJlIHlvdXIgaW5mb3JtYXRpb24gd2l0aCBhbnlvbmUuIE9yZGVycyBz\n
aGlwcGVkIHdpdGhpbiAyNCBob3Vycy48L2ZvbnQ+PC9wPiANCg0KPHA gYWxp\n
Z249ImNlbnRlciI+PGZvbnQgc2l6ZT0iMyIgZmFjZT0iQXJpYWwsIEhlbHZl\n
dGljYSwgc2Fucy1zZXJpZiI+PGEgaHJlZj0iaHR0cDovL3d3dy5kaXJlY3R3\n
ZWJzdG9yZS5jb20vdG95cy9pbmRleC5odG1sIj48Yj48Zm9udCBzaXplPSI1\n
Ij5FTlRFUiANCk5PVzwvZm9udD48L2I+PC9hPjwvZm9udD48L3A+IA0KPC9h\n
YWxpZ249ImNlbnRlciI+Jm5ic3A7PC9wPiANCjxwIGFsaWduPSJjZW50ZXIi\n
PiZuYnNwOzwvcD4gDQo8cCBhbGlnbj0iY2VudGVyIj48Zm9udCBzaXplPSIx\n
IiBmYWNlPSJBcmlhbCwgSGVsdmV0aWNhLCBzYW5zLXNlcmlmIj5UaGlzIEUt\n
bWFpbGluZyANCmhhcyBiZWVuIHNlbnQgdG8geW91IGFzIGEgcGVyc29uIGlu\n
dGVyZXN0ZWQgaW4gdGhlIGluZm9ybWF0aW9uIGVuY2xvc2VkLiA8YnI+IA0K\n
SWYgdGhpcyByZWFjaGVkIHlvdSBieSBlcnJvciwgb3IgeW91IGRvIG5vdCB3\n
aXNoIHRvIHJlY2VpdmUgdGhpcyBpbmZvcm1hdGlvbiANCm9yIHR5cGUgb2Yg\n
aW5mb3JtYXRpb24gaW4gdGhlIGZ1dHVyZSwgPGJyPiANCnBsZWFzZSBjbGlj\n
ayBvbiB0aGUgd29yZDxhIGhyZWY9Im1haWx0bzptaW5pcmVtQGZpcmVtYWls\n
LmRlIj4gUkVNT1ZFIDwvYT5hbmQgDQpyZXBseSB0aGlzIGVtYWlsIHRvIHVz\n
LCB5b3Ugd2lsbCBiZSB0YWtlbiBvZmYgb3VyIGxpc3QgaW1tZWRpYXRlbHkg\n
YW5kIE5FVkVSIA0KcmVjZWl2ZSBhbnkgZW1haWxzIGZyb20gdXMuPC9mb250\n
PjwvcD4gDQo8cCBhbGlnbj0iY2VudGVyIj4mbmJzcDs8L3A+IA0KPHAgYWxp\n
Z249ImNlbnRlciI+Jm5ic3A7PC9wPiANCjwvYT4gDQo8cCBhbGlnbj0iY2Vu\n
YnNwOzwvcD4gDQo8cCBhbGlnbj0iY2VudGVyIj4mbmJzcDs8L3A+IA0KPHAg\n
YWxpZ249ImNlbnRlciI+Jm5ic3A7PC9wPiANCjwvYm9keT4gDQo8L2h0bWw+\n
DQoxMzkzUHdxYjAtMjYxU0JjSDAzNTdtaEdxMS0xMDN1VHVQOTA3NkFyd3gw\n
LTI1Nm5ad2M0NjA2aWtDSzUtNDUyS2VzdzM1NjhFT29JNC00NjFQbDc3\n \n'
```

[127]:
```
data.loc[765]
```

[127]:
```
MESSAGE      <html><body><IMG SRC='http://master2.free4all…
CATEGORY                                                 1
```

```
FILE_NAME                00266.12e00174bc1346952a8ba2c430e48bf6
Name: 765, dtype: object
```

[128]:
```
clean_msg_no_html(data.at[765,'MESSAGE'])
```

[128]:
```
[]
```

[129]:
```
clean_message(data.at[765,'MESSAGE'])
```

[129]:
```
['html', 'bodi', 'img', 'center', 'img']
```

[ ]:
```

```