Google_colab_dump

June 1, 2018

In [1]:
```
# Install a Drive FUSE wrapper.
# https://github.com/astrada/google-drive-ocamlfuse
!apt-get install -y -qq software-properties-common python-software-properties module-i
!add-apt-repository -y ppa:alessandro-strada/ppa 2>&1 > /dev/null
!apt-get update -qq 2>&1 > /dev/null
!apt-get -y install -qq google-drive-ocamlfuse fuse
```

```
Preconfiguring packages ...
Selecting previously unselected package cron.
(Reading database ... 18298 files and directories currently installed.)
Preparing to unpack .../00-cron_3.0pl1-128ubuntu5_amd64.deb ...
Unpacking cron (3.0pl1-128ubuntu5) ...
Selecting previously unselected package libapparmor1:amd64.
Preparing to unpack .../01-libapparmor1_2.11.0-2ubuntu17.1_amd64.deb ...
Unpacking libapparmor1:amd64 (2.11.0-2ubuntu17.1) ...
Selecting previously unselected package libdbus-1-3:amd64.
Preparing to unpack .../02-libdbus-1-3_1.10.22-1ubuntu1_amd64.deb ...
Unpacking libdbus-1-3:amd64 (1.10.22-1ubuntu1) ...
Selecting previously unselected package dbus.
Preparing to unpack .../03-dbus_1.10.22-1ubuntu1_amd64.deb ...
Unpacking dbus (1.10.22-1ubuntu1) ...
Selecting previously unselected package dirmngr.
Preparing to unpack .../04-dirmngr_2.1.15-1ubuntu8_amd64.deb ...
Unpacking dirmngr (2.1.15-1ubuntu8) ...
Selecting previously unselected package distro-info-data.
Preparing to unpack .../05-distro-info-data_0.36ubuntu0.2_all.deb ...
Unpacking distro-info-data (0.36ubuntu0.2) ...
Selecting previously unselected package libkmod2:amd64.
Preparing to unpack .../06-libkmod2_24-1ubuntu2_amd64.deb ...
Unpacking libkmod2:amd64 (24-1ubuntu2) ...
Selecting previously unselected package kmod.
Preparing to unpack .../07-kmod_24-1ubuntu2_amd64.deb ...
Unpacking kmod (24-1ubuntu2) ...
Selecting previously unselected package lsb-release.
Preparing to unpack .../08-lsb-release_9.20160110ubuntu5_all.deb ...
Unpacking lsb-release (9.20160110ubuntu5) ...
Selecting previously unselected package libgirepository-1.0-1:amd64.
```

```
Preparing to unpack .../09-libgirepository-1.0-1_1.54.1-1_amd64.deb ...
Unpacking libgirepository-1.0-1:amd64 (1.54.1-1) ...
Selecting previously unselected package gir1.2-glib-2.0:amd64.
Preparing to unpack .../10-gir1.2-glib-2.0_1.54.1-1_amd64.deb ...
Unpacking gir1.2-glib-2.0:amd64 (1.54.1-1) ...
Selecting previously unselected package iso-codes.
Preparing to unpack .../11-iso-codes_3.75-1_all.deb ...
Unpacking iso-codes (3.75-1) ...
Selecting previously unselected package libdbus-glib-1-2:amd64.
Preparing to unpack .../12-libdbus-glib-1-2_0.108-2_amd64.deb ...
Unpacking libdbus-glib-1-2:amd64 (0.108-2) ...
Selecting previously unselected package python-apt-common.
Preparing to unpack .../13-python-apt-common_1.4.0~beta3build2_all.deb ...
Unpacking python-apt-common (1.4.0~beta3build2) ...
Selecting previously unselected package python3-apt.
Preparing to unpack .../14-python3-apt_1.4.0~beta3build2_amd64.deb ...
Unpacking python3-apt (1.4.0~beta3build2) ...
Selecting previously unselected package python3-dbus.
Preparing to unpack .../15-python3-dbus_1.2.4-1build3_amd64.deb ...
Unpacking python3-dbus (1.2.4-1build3) ...
Selecting previously unselected package python3-gi.
Preparing to unpack .../16-python3-gi_3.24.1-2build1_amd64.deb ...
Unpacking python3-gi (3.24.1-2build1) ...
Selecting previously unselected package module-init-tools.
Preparing to unpack .../17-module-init-tools_24-1ubuntu2_all.deb ...
Unpacking module-init-tools (24-1ubuntu2) ...
Selecting previously unselected package python-apt.
Preparing to unpack .../18-python-apt_1.4.0~beta3build2_amd64.deb ...
Unpacking python-apt (1.4.0~beta3build2) ...
Selecting previously unselected package python-pycurl.
Preparing to unpack .../19-python-pycurl_7.43.0-2build2_amd64.deb ...
Unpacking python-pycurl (7.43.0-2build2) ...
Selecting previously unselected package python-software-properties.
Preparing to unpack .../20-python-software-properties_0.96.24.17_all.deb ...
Unpacking python-software-properties (0.96.24.17) ...
Selecting previously unselected package python3-software-properties.
Preparing to unpack .../21-python3-software-properties_0.96.24.17_all.deb ...
Unpacking python3-software-properties (0.96.24.17) ...
Selecting previously unselected package software-properties-common.
Preparing to unpack .../22-software-properties-common_0.96.24.17_all.deb ...
Unpacking software-properties-common (0.96.24.17) ...
Selecting previously unselected package unattended-upgrades.
Preparing to unpack .../23-unattended-upgrades_0.98ubuntu1.1_all.deb ...
Unpacking unattended-upgrades (0.98ubuntu1.1) ...
Setting up python-apt-common (1.4.0~beta3build2) ...
Setting up python3-apt (1.4.0~beta3build2) ...
Setting up iso-codes (3.75-1) ...
Setting up distro-info-data (0.36ubuntu0.2) ...
```

```
Setting up python-pycurl (7.43.0-2build2) ...
Setting up lsb-release (9.20160110ubuntu5) ...
Setting up libgirepository-1.0-1:amd64 (1.54.1-1) ...
Setting up libkmod2:amd64 (24-1ubuntu2) ...
Setting up gir1.2-glib-2.0:amd64 (1.54.1-1) ...
Processing triggers for libc-bin (2.26-0ubuntu2.1) ...
Setting up libapparmor1:amd64 (2.11.0-2ubuntu17.1) ...
Setting up unattended-upgrades (0.98ubuntu1.1) ...


Creating config file /etc/apt/apt.conf.d/20auto-upgrades with new version

Creating config file /etc/apt/apt.conf.d/50unattended-upgrades with new version
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Setting up dirmngr (2.1.15-1ubuntu8) ...
Setting up cron (3.0pl1-128ubuntu5) ...
Adding group `crontab' (GID 102) ...
Done.
update-rc.d: warning: start and stop actions are no longer supported; falling back to defaults
update-rc.d: warning: stop runlevel arguments (1) do not match cron Default-Stop values (none)
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Setting up libdbus-1-3:amd64 (1.10.22-1ubuntu1) ...
Setting up kmod (24-1ubuntu2) ...
Setting up libdbus-glib-1-2:amd64 (0.108-2) ...
Setting up python3-gi (3.24.1-2build1) ...
Setting up module-init-tools (24-1ubuntu2) ...
Setting up python3-software-properties (0.96.24.17) ...
Setting up dbus (1.10.22-1ubuntu1) ...
Setting up python-apt (1.4.0~beta3build2) ...
Setting up python3-dbus (1.2.4-1build3) ...
Setting up python-software-properties (0.96.24.17) ...
Setting up software-properties-common (0.96.24.17) ...
Processing triggers for libc-bin (2.26-0ubuntu2.1) ...
Processing triggers for dbus (1.10.22-1ubuntu1) ...
gpg: keybox '/tmp/tmpf8voexvi/pubring.gpg' created
gpg: /tmp/tmpf8voexvi/trustdb.gpg: trustdb created
gpg: key AD5F235DF639B041: public key "Launchpad PPA for Alessandro Strada" imported
gpg: Total number processed: 1
gpg:               imported: 1
Warning: apt-key output should not be parsed (stdout is not a terminal)
Selecting previously unselected package libfuse2:amd64.
(Reading database ... 19706 files and directories currently installed.)
Preparing to unpack .../libfuse2_2.9.7-1ubuntu1_amd64.deb ...
Unpacking libfuse2:amd64 (2.9.7-1ubuntu1) ...
Selecting previously unselected package fuse.
Preparing to unpack .../fuse_2.9.7-1ubuntu1_amd64.deb ...
```

```
Unpacking fuse (2.9.7-1ubuntu1) ...
Selecting previously unselected package google-drive-ocamlfuse.
Preparing to unpack .../google-drive-ocamlfuse_0.6.21-0ubuntu2_amd64.deb ...
Unpacking google-drive-ocamlfuse (0.6.21-0ubuntu2) ...
Setting up libfuse2:amd64 (2.9.7-1ubuntu1) ...
Processing triggers for libc-bin (2.26-0ubuntu2.1) ...
Setting up fuse (2.9.7-1ubuntu1) ...
Setting up google-drive-ocamlfuse (0.6.21-0ubuntu2) ...
```

```
In [0]: # Generate auth tokens for Colab
        from google.colab import auth
        auth.authenticate_user()
```

```
In [3]: # Generate creds for the Drive FUSE library.
        from oauth2client.client import GoogleCredentials
        creds = GoogleCredentials.get_application_default()
        import getpass
        !google-drive-ocamlfuse -headless -id={creds.client_id} -secret={creds.client_secret} <
        vcode = getpass.getpass()
        !echo {vcode} | google-drive-ocamlfuse -headless -id={creds.client_id} -secret={creds.c
```

```
Please, open the following URL in a web browser: https://accounts.google.com/o/oauth2/auth?clie

ůůůůůůůůůů
Please, open the following URL in a web browser: https://accounts.google.com/o/oauth2/auth?clie
Please enter the verification code: Access token retrieved correctly.
```

```
In [0]: # Create a directory and mount Google Drive using that directory.
        !mkdir -p drive
        !google-drive-ocamlfuse drive

        #print('Files in Drive:')
        #!ls drive/

        # Create a file in Drive.
        !echo "This newly created file will appear in your Drive file list." > drive/created.tx
```

```
In [5]: import sqlite3
        import pandas as pd
        import numpy as np
        import nltk
        import string
        import matplotlib.pyplot as plt
        import seaborn as sns
        from sklearn.feature_extraction.text import TfidfTransformer
        from sklearn.feature_extraction.text import TfidfVectorizer
```

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics import confusion_matrix
from sklearn import metrics
from sklearn.metrics import roc_curve, auc
from nltk.stem.porter import PorterStemmer


# using the SQLite Table to read data.
con = sqlite3.connect('drive/database.sqlite')
print(con)

#filtering only positive and negative reviews i.e.
# not taking into consideration those reviews with Score=3
filtered_data = pd.read_sql_query("""
SELECT *
FROM Reviews
WHERE Score != 3
""", con)

# Give reviews with Score>3 a positive rating, and reviews with a score<3 a negative r
def partition(x):
    if x < 3:
        return 'negative'
    return 'positive'

#changing reviews with score less than 3 to be positive and vice-versa
actualScore = filtered_data['Score']
positiveNegative = actualScore.map(partition)
filtered_data['Score'] = positiveNegative
```

```
<sqlite3.Connection object at 0x7effa9e501f0>
```

In [6]:
```
#Sorting data according to ProductId in ascending order
sorted_data=filtered_data.sort_values('ProductId', axis=0, ascending=True, inplace=Fals

#Deduplication of entries
final=sorted_data.drop_duplicates(subset={"UserId","ProfileName","Time","Text"}, keep=
final.shape
```

Out[6]:
```
(364173, 10)
```

In [0]:
```
# value of HelpfulnessNumerator greater than HelpfulnessDenominator is not practically
# possible hence these two rows too are removed from calcualtions
final=final[final.HelpfulnessNumerator<=final.HelpfulnessDenominator]
```

In [8]:
```
import nltk
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /content/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

Out[8]: True

In [0]:
```
import re
import string
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
from nltk.stem.wordnet import WordNetLemmatizer

stop = set(stopwords.words('english')) #set of stopwords
sno = nltk.stem.SnowballStemmer('english') #initialising the snowball stemmer

def cleanhtml(sentence): #function to clean the word of any html-tags
    cleanr = re.compile('<.*?>')
    cleantext = re.sub(cleanr, ' ', sentence)
    return cleantext
def cleanpunc(sentence): #function to clean the word of any punctuation or special cha
    cleaned = re.sub(r'[?|!|\'|"|#]',r'',sentence)
    cleaned = re.sub(r'[.|,|)|(|\|/]',r' ',cleaned)
    return  cleaned
```

In [0]:
```
#Code for implementing step-by-step the checks mentioned in the pre-processing phase
# this code takes a while to run as it needs to run on 500k sentences.
i=0
str1=' '
final_string=[]
all_positive_words=[] # store words from +ve reviews here
all_negative_words=[] # store words from -ve reviews here.
s=''
for sent in final['Text'].values:
    filtered_sentence=[]
    #print(sent);
    sent=cleanhtml(sent) # remove HTML tags
    for w in sent.split():
        for cleaned_words in cleanpunc(w).split():
            if((cleaned_words.isalpha()) & (len(cleaned_words)>2)):
                if(cleaned_words.lower() not in stop):
                    s=(sno.stem(cleaned_words.lower())).encode('utf8')
                    filtered_sentence.append(s)
                    if (final['Score'].values)[i] == 'positive':
                        all_positive_words.append(s) #list of all words used to descri
                    if(final['Score'].values)[i] == 'negative':
                        all_negative_words.append(s) #list of all words used to descri
                else:
                    continue
```

```
                    else:
                        continue
            #print(filtered_sentence)
            str1 = b" ".join(filtered_sentence) #final string of cleaned words
            #print("*****************************************************************")

            final_string.append(str1)
            i+=1
```

```
In [11]: final['CleanedText']=final_string #adding a column of CleanedText which displays the
         print(final['Score'].head(3))
```

```
138706    positive
138688    positive
138689    positive
Name: Score, dtype: object
```

```
In [0]: final.head(3) #below the processed review can be seen in the CleanedText Column

        # store final table into an SQlLite table for future.
        conn = sqlite3.connect('drive/final.sqlite')
        c=conn.cursor()
        conn.text_factory = str
        #final.to_sql('Reviews', conn, flavor=None, schema=None, if_exists='replace', index=Tr
```

```
In [13]: #BoW
         count_vect = CountVectorizer() #in scikit-learn
         final_counts = count_vect.fit_transform(final['Text'].values)
         final_counts.get_shape()
```

```
Out[13]: (364171, 115281)
```

```
In [0]: # TSNE

        from sklearn.manifold import TSNE

        num_points = 5000
        # Picking the top 1000 points as TSNE takes a lot of time for 15K points
        data_1000 = final_counts[0:num_points,:]
        print(type(final_counts))
        #print(final_counts["Score"])
        labels_1000 = final['Score'].head(num_points)

        model = TSNE(n_components=2, random_state=0)
        # configuring the parameteres
        # the number of components = 2
        # default perplexity = 30
        # default learning rate = 200
```

```
        # default Maximum number of iterations for the optimization = 1000

        tsne_data = model.fit_transform(data_1000.toarray())


        # creating a new data frame which help us in ploting the result data
        tsne_data = np.vstack((tsne_data.T, labels_1000)).T
        tsne_df = pd.DataFrame(data=tsne_data, columns=("Dim_1", "Dim_2", "label"))

        # Ploting the result of tsne
        sns.FacetGrid(tsne_df, hue="label", size=6).map(plt.scatter, 'Dim_1', 'Dim_2').add_lege
        plt.show()
```

```
<class 'scipy.sparse.csr.csr_matrix'>
```