*Climate* *Change* *Belief* *Analysis*

Introduction

Many companies are built around lessening their environmental impact and thus they offer products that are environmentally friendly and sustainable. With that, they would like to determine how people perceive climate change.

This notebook aims to build Machine Learning Classification model(s) that is able to classify accurately whether or not a person believes in climate change based on their novel tweet.

This will be done by importing necesarry libraries as well as the training and test datasets. Data cleaning follows together with exploratory data analysis.We then wrap up the notebook by diving into different classification techniques under the Modelling section which will be followed by insights and a conclusion.

The following are some of the few classification techniques explored in this notebook:

1.Support Vector Machines

2.Naive Bayes Classfier

3.K-Nearest Neighbors

4.Random Forest

5.Tree based models

Because this is **sentiment analysis**,the interpretation and classification of emotions within text data, we will use some natural language preprocessing techniques libraries like nltk,spacy together with scikit learn.

*COMET*

In [1]:

!pip install comet\_ml

Collecting comet\_ml

Downloading comet\_ml-3.1.12-py2.py3-none-any.whl (214 kB)

|████████████████████████████████| 214 kB 8.0 MB/s

Requirement already satisfied: wurlitzer>=1.0.2 in /opt/conda/lib/python3.7/site-packages (from comet\_ml) (2.0.0)

Collecting comet-git-pure>=0.19.11

Downloading comet\_git\_pure-0.19.16-py3-none-any.whl (409 kB)

|████████████████████████████████| 409 kB 18.7 MB/s

Collecting everett[ini]>=1.0.1; python\_version >= "3.0"

Downloading everett-1.0.2-py2.py3-none-any.whl (31 kB)

Requirement already satisfied: jsonschema!=3.1.0,>=2.6.0 in /opt/conda/lib/python3.7/site-packages (from comet\_ml) (3.2.0)

Requirement already satisfied: six in /opt/conda/lib/python3.7/site-packages (from comet\_ml) (1.14.0)

Requirement already satisfied: nvidia-ml-py3>=7.352.0 in /opt/conda/lib/python3.7/site-packages (from comet\_ml) (7.352.0)

Collecting netifaces>=0.10.7

Downloading netifaces-0.10.9.tar.gz (28 kB)

Requirement already satisfied: requests>=2.18.4 in /opt/conda/lib/python3.7/site-packages (from comet\_ml) (2.23.0)

Requirement already satisfied: websocket-client>=0.55.0 in /opt/conda/lib/python3.7/site-packages (from comet\_ml) (0.57.0)

Requirement already satisfied: urllib3>=1.24.1 in /opt/conda/lib/python3.7/site-packages (from comet-git-pure>=0.19.11->comet\_ml) (1.24.3)

Requirement already satisfied: certifi in /opt/conda/lib/python3.7/site-packages (from comet-git-pure>=0.19.11->comet\_ml) (2020.4.5.2)

Collecting configobj; extra == "ini"

Downloading configobj-5.0.6.tar.gz (33 kB)

Requirement already satisfied: importlib-metadata; python\_version < "3.8" in /opt/conda/lib/python3.7/site-packages (from jsonschema!=3.1.0,>=2.6.0->comet\_ml) (1.6.0)

Requirement already satisfied: setuptools in /opt/conda/lib/python3.7/site-packages (from jsonschema!=3.1.0,>=2.6.0->comet\_ml) (46.1.3.post20200325)

Requirement already satisfied: pyrsistent>=0.14.0 in /opt/conda/lib/python3.7/site-packages (from jsonschema!=3.1.0,>=2.6.0->comet\_ml) (0.16.0)

Requirement already satisfied: attrs>=17.4.0 in /opt/conda/lib/python3.7/site-packages (from jsonschema!=3.1.0,>=2.6.0->comet\_ml) (19.3.0)

Requirement already satisfied: chardet<4,>=3.0.2 in /opt/conda/lib/python3.7/site-packages (from requests>=2.18.4->comet\_ml) (3.0.4)

Requirement already satisfied: idna<3,>=2.5 in /opt/conda/lib/python3.7/site-packages (from requests>=2.18.4->comet\_ml) (2.9)

Requirement already satisfied: zipp>=0.5 in /opt/conda/lib/python3.7/site-packages (from importlib-metadata; python\_version < "3.8"->jsonschema!=3.1.0,>=2.6.0->comet\_ml) (3.1.0)

Building wheels for collected packages: netifaces, configobj

Building wheel for netifaces (setup.py) ... - \ | / - \ done

Created wheel for netifaces: filename=netifaces-0.10.9-cp37-cp37m-linux\_x86\_64.whl size=37295 sha256=90b8d09a34393b0468797e3b9221d8170952df2d7c73fbf0d8bbeda06827c4bb

Stored in directory: /root/.cache/pip/wheels/5f/09/cf/2b1aa8371c071fa89518ac0bbda1b8cca4e65b6e2538af4192

Building wheel for configobj (setup.py) ... - \ done

Created wheel for configobj: filename=configobj-5.0.6-py3-none-any.whl size=34546 sha256=874e8af48e45be50c4fc707f14af1f1d3af9f2402b5da95436dfe77eba5cec6b

Stored in directory: /root/.cache/pip/wheels/0d/c4/19/13d74440f2a571841db6b6e0a273694327498884dafb9cf978

Successfully built netifaces configobj

Installing collected packages: comet-git-pure, configobj, everett, netifaces, comet-ml

Successfully installed comet-git-pure-0.19.16 comet-ml-3.1.12 configobj-5.0.6 everett-1.0.2 netifaces-0.10.9

In [2]:

import comet\_ml

Imports

Import Libraries

In [3]:

!pip install spacy

!pip install NLTK

Requirement already satisfied: spacy in /opt/conda/lib/python3.7/site-packages (2.2.4)

Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /opt/conda/lib/python3.7/site-packages (from spacy) (2.0.3)

Requirement already satisfied: wasabi<1.1.0,>=0.4.0 in /opt/conda/lib/python3.7/site-packages (from spacy) (0.6.0)

Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /opt/conda/lib/python3.7/site-packages (from spacy) (3.0.2)

Requirement already satisfied: catalogue<1.1.0,>=0.0.7 in /opt/conda/lib/python3.7/site-packages (from spacy) (1.0.0)

Requirement already satisfied: blis<0.5.0,>=0.4.0 in /opt/conda/lib/python3.7/site-packages (from spacy) (0.4.1)

Requirement already satisfied: setuptools in /opt/conda/lib/python3.7/site-packages (from spacy) (46.1.3.post20200325)

Requirement already satisfied: requests<3.0.0,>=2.13.0 in /opt/conda/lib/python3.7/site-packages (from spacy) (2.23.0)

Requirement already satisfied: thinc==7.4.0 in /opt/conda/lib/python3.7/site-packages (from spacy) (7.4.0)

Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /opt/conda/lib/python3.7/site-packages (from spacy) (1.0.2)

Requirement already satisfied: plac<1.2.0,>=0.9.6 in /opt/conda/lib/python3.7/site-packages (from spacy) (1.1.3)

Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /opt/conda/lib/python3.7/site-packages (from spacy) (4.45.0)

Requirement already satisfied: srsly<1.1.0,>=1.0.2 in /opt/conda/lib/python3.7/site-packages (from spacy) (1.0.2)

Requirement already satisfied: numpy>=1.15.0 in /opt/conda/lib/python3.7/site-packages (from spacy) (1.18.1)

Requirement already satisfied: importlib-metadata>=0.20; python\_version < "3.8" in /opt/conda/lib/python3.7/site-packages (from catalogue<1.1.0,>=0.0.7->spacy) (1.6.0)

Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /opt/conda/lib/python3.7/site-packages (from requests<3.0.0,>=2.13.0->spacy) (1.24.3)

Requirement already satisfied: chardet<4,>=3.0.2 in /opt/conda/lib/python3.7/site-packages (from requests<3.0.0,>=2.13.0->spacy) (3.0.4)

Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/lib/python3.7/site-packages (from requests<3.0.0,>=2.13.0->spacy) (2020.4.5.2)

Requirement already satisfied: idna<3,>=2.5 in /opt/conda/lib/python3.7/site-packages (from requests<3.0.0,>=2.13.0->spacy) (2.9)

Requirement already satisfied: zipp>=0.5 in /opt/conda/lib/python3.7/site-packages (from importlib-metadata>=0.20; python\_version < "3.8"->catalogue<1.1.0,>=0.0.7->spacy) (3.1.0)

Requirement already satisfied: NLTK in /opt/conda/lib/python3.7/site-packages (3.2.4)

Requirement already satisfied: six in /opt/conda/lib/python3.7/site-packages (from NLTK) (1.14.0)

In [4]:

import numpy as np

import pandas as pd

import sklearn

import matplotlib.pyplot as plt

from numpy import arange

import seaborn as sns

*#Natural Language Processing*

import re

import spacy.cli

import nltk

from nltk.corpus import stopwords

from nltk.tokenize import word\_tokenize

from sklearn.feature\_extraction.text import TfidfVectorizer

*#Matrix measurement*

from sklearn.metrics import accuracy\_score, f1\_score

from sklearn.metrics import classification\_report

from sklearn.metrics import confusion\_matrix

from sklearn.metrics import f1\_score, precision\_score, recall\_score

*#Resampling techniques*

from sklearn.utils import resample

from imblearn.over\_sampling import SMOTE

*#Machine Learning Models*

from sklearn.naive\_bayes import MultinomialNB, BernoulliNB

from sklearn.svm import LinearSVC, SVC

from sklearn.linear\_model import SGDClassifier, LinearRegression

from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier, ExtraTreesClassifier, BaggingClassifier

from sklearn.tree import DecisionTreeClassifier

from sklearn.neighbors import KNeighborsClassifier

from sklearn.gaussian\_process import GaussianProcessClassifier

from sklearn.model\_selection import train\_test\_split

from sklearn.model\_selection import cross\_val\_score

from sklearn.model\_selection import GridSearchCV

from sklearn.model\_selection import KFold

from wordcloud import WordCloud

*# from google.colab import drive*

In [5]:

spacy.cli.download('en\_core\_web\_sm')

nltk.download('punkt')

nltk.download('wordnet')

nltk.download('stopwords')

stop = nltk.corpus.stopwords.words('english')

✔ Download and installation successful

You can now load the model via spacy.load('en\_core\_web\_sm')

[nltk\_data] Downloading package punkt to /usr/share/nltk\_data...

[nltk\_data] Package punkt is already up-to-date!

[nltk\_data] Downloading package wordnet to /usr/share/nltk\_data...

[nltk\_data] Package wordnet is already up-to-date!

[nltk\_data] Downloading package stopwords to /usr/share/nltk\_data...

[nltk\_data] Package stopwords is already up-to-date!

In [6]:

nlp = spacy.load('en\_core\_web\_sm')

Read in the datasets

In [7]:

train = pd.read\_csv('https://raw.githubusercontent.com/Stephane-Masamba/Team\_4\_CPT\_ML-Classification/Mikael/train%20(1).csv')

print(train.head())

sentiment message tweetid

0 1 PolySciMajor EPA chief doesn't think carbon di... 625221

1 1 It's not like we lack evidence of anthropogeni... 126103

2 2 RT @RawStory: Researchers say we have three ye... 698562

3 1 #TodayinMaker# WIRED : 2016 was a pivotal year... 573736

4 1 RT @SoyNovioDeTodas: It's 2016, and a racist, ... 466954

In [8]:

test = pd.read\_csv('https://raw.githubusercontent.com/Stephane-Masamba/Team\_4\_CPT\_ML-Classification/Mikael/test%20(1).csv')

print(test.head())

message tweetid

0 Europe will now be looking to China to make su... 169760

1 Combine this with the polling of staffers re c... 35326

2 The scary, unimpeachable evidence that climate... 224985

3 @Karoli @morgfair @OsborneInk @dailykos \nPuti... 476263

4 RT @FakeWillMoore: 'Female orgasms cause globa... 872928

Data description

The data available aggregates tweets pertaining to climate change collected between Apr 27, 2015 and Feb 21, 2018.

In total, 43943 tweets were collected.

Each tweet is labelled as one of four classes.

Classes and description

Class 2 : News-the tweet links to factual news about climate change

Class 1 : Pro-the tweet supports the belief of man-made climate change

Class 0 : Neutral-the tweet neither supports nor refutes the belief of man-made climate change

Class -1 : Anti-the tweet does not believe in man-made climate change

Variables definitions

Sentiment: Sentiment of tweet

Message: Tweet body

Tweetid: Twitter unique id

Data Cleaning

Knowing that we are dealing with text data, we decided to first clean the data by making all tweets lower-case, removing punctuation marks and removing white spaces before doing anything else. Also, replacing all links with the word 'LINK' and all user handles with 'USER\_REF'

In [9]:

def clean\_text(df):

i = 0

for tweet **in** df['message']:

tweet = tweet.lower()

tweet = re.sub(r'http\S+', 'LINK', tweet)

tweet = re.sub(r'@\S+', 'USER\_REF', tweet)

tweet = re.sub(r'[^\w\s]', '', tweet)

tweet = tweet.lstrip()

tweet = tweet.rstrip()

tweet = tweet.replace(' ', ' ')

df.loc[i, 'message'] = tweet

i += 1

In [10]:

clean\_text(train)

train

Out[10]:

|  | sentiment | message | tweetid |
| --- | --- | --- | --- |
| 0 | 1 | polyscimajor epa chief doesnt think carbon dio... | 625221 |
| 1 | 1 | its not like we lack evidence of anthropogenic... | 126103 |
| 2 | 2 | rt USER\_REF researchers say we have three year... | 698562 |
| 3 | 1 | todayinmaker wired 2016 was a pivotal year in ... | 573736 |
| 4 | 1 | rt USER\_REF its 2016 and a racist sexist clima... | 466954 |
| ... | ... | ... | ... |
| 15814 | 1 | rt USER\_REF they took down the material on glo... | 22001 |
| 15815 | 2 | rt USER\_REF how climate change could be breaki... | 17856 |
| 15816 | 0 | notiven rt nytimesworld what does trump actual... | 384248 |
| 15817 | -1 | rt USER\_REF hey liberals the climate change cr... | 819732 |
| 15818 | 0 | rt USER\_REF USER\_REF climate change equation i... | 806319 |

15819 rows × 3 columns

In [11]:

clean\_text(test)

test

Out[11]:

|  | message | tweetid |
| --- | --- | --- |
| 0 | europe will now be looking to china to make su... | 169760 |
| 1 | combine this with the polling of staffers re c... | 35326 |
| 2 | the scary unimpeachable evidence that climate ... | 224985 |
| 3 | USER\_REF USER\_REF USER\_REF USER\_REF \nputin go... | 476263 |
| 4 | rt USER\_REF female orgasms cause global warmin... | 872928 |
| ... | ... | ... |
| 10541 | rt USER\_REF brb writing a poem about climate c... | 895714 |
| 10542 | 2016 the year climate change came home during ... | 875167 |
| 10543 | rt USER\_REF pacific countries positive about f... | 78329 |
| 10544 | rt USER\_REF youre so hot you must be the cause... | 867455 |
| 10545 | rt USER\_REF climate change is a global issue t... | 470892 |

10546 rows × 2 columns

Here we add 'LINK' and 'USER\_REF' to our list of stop words and remove all stop words from each tweet

In [12]:

def remove\_stopwords(df):

my\_stop\_words = stopwords.words('english')

my\_stop\_words.append('LINK')

my\_stop\_words.append('USER\_REF')

df\_index = 0

for tweet **in** df['message']:

tweet = word\_tokenize(tweet)

tweet = [word for word **in** tweet if **not** word **in** my\_stop\_words]

tweet = ' '.join(tweet)

df.loc[df\_index, 'message'] = tweet

df\_index += 1

return df

In [13]:

remove\_stopwords(train)

Out[13]:

|  | sentiment | message | tweetid |
| --- | --- | --- | --- |
| 0 | 1 | polyscimajor epa chief doesnt think carbon dio... | 625221 |
| 1 | 1 | like lack evidence anthropogenic global warming | 126103 |
| 2 | 2 | rt researchers say three years act climate cha... | 698562 |
| 3 | 1 | todayinmaker wired 2016 pivotal year war clima... | 573736 |
| 4 | 1 | rt 2016 racist sexist climate change denying b... | 466954 |
| ... | ... | ... | ... |
| 15814 | 1 | rt took material global warming lgbt rights he... | 22001 |
| 15815 | 2 | rt climate change could breaking 200millionyea... | 17856 |
| 15816 | 0 | notiven rt nytimesworld trump actually believe... | 384248 |
| 15817 | -1 | rt hey liberals climate change crap hoax ties ... | 819732 |
| 15818 | 0 | rt climate change equation 4 screenshots | 806319 |

15819 rows × 3 columns

In [14]:

remove\_stopwords(test)

Out[14]:

|  | message | tweetid |
| --- | --- | --- |
| 0 | europe looking china make sure alone fighting ... | 169760 |
| 1 | combine polling staffers climate change womens... | 35326 |
| 2 | scary unimpeachable evidence climate change al... | 224985 |
| 3 | putin got jill trump doesnt believe climate ch... | 476263 |
| 4 | rt female orgasms cause global warming sarcast... | 872928 |
| ... | ... | ... |
| 10541 | rt brb writing poem climate change climatechan... | 895714 |
| 10542 | 2016 year climate change came home hottest yea... | 875167 |
| 10543 | rt pacific countries positive fiji leading glo... | 78329 |
| 10544 | rt youre hot must cause global warming aldubla... | 867455 |
| 10545 | rt climate change global issue thats getting w... | 470892 |

10546 rows × 2 columns

Find all named entities for each tweet

In [15]:

def entities(df):

df\_index = 0

for tweet **in** df['message']:

tweet = nlp(tweet)

for entity **in** tweet.ents:

df.loc[df\_index, 'message'] = df.loc[df\_index, 'message'].replace(str(entity.text), str(entity.label\_))

df\_index += 1

return df

In [16]:

entities(train)

Out[16]:

|  | sentiment | message | tweetid |
| --- | --- | --- | --- |
| 0 | 1 | ORG chief doesnt think carbon dioxide main cau... | 625221 |
| 1 | 1 | like lack evidence anthropogenic global warming | 126103 |
| 2 | 2 | rt researchers say three years act climate cha... | 698562 |
| 3 | 1 | todayinmaker wired 2016 pivotal year war clima... | 573736 |
| 4 | 1 | rt 2016 racist sexist climate change denying b... | 466954 |
| ... | ... | ... | ... |
| 15814 | 1 | rt took material global warming lgbt rights he... | 22001 |
| 15815 | 2 | rt climate change could breaking 200millionyea... | 17856 |
| 15816 | 0 | notiven rt nytimesworld trump actually believe... | 384248 |
| 15817 | -1 | rt hey liberals climate change crap hoax ties ... | 819732 |
| 15818 | 0 | rt climate change equation 4 screenshots | 806319 |

15819 rows × 3 columns

In [17]:

entities(test)

Out[17]:

|  | message | tweetid |
| --- | --- | --- |
| 0 | LOC looking GPE make sure alone fighting clima... | 169760 |
| 1 | combine polling staffers climate change womens... | 35326 |
| 2 | scary unimpeachable evidence climate change al... | 224985 |
| 3 | putin got jill trump doesnt believe climate ch... | 476263 |
| 4 | rt female orgasms cause global warming sarcast... | 872928 |
| ... | ... | ... |
| 10541 | rt brb writing poem climate change climatechan... | 895714 |
| 10542 | 2016 year climate change came home hottest yea... | 875167 |
| 10543 | rt pacific countries positive fiji leading glo... | 78329 |
| 10544 | rt youre hot must cause global warming aldubla... | 867455 |
| 10545 | rt climate change global issue thats getting w... | 470892 |

10546 rows × 2 columns

Lemmatize every word in each tweet

In [18]:

def lem\_text(df):

df\_index = 0

for tweet **in** df['message']:

tweet = nlp(tweet)

for token **in** tweet:

df.loc[df\_index, 'message'] = df.loc[df\_index, 'message'].replace(str(token.text), str(token.lemma\_))

df\_index += 1

return df

In [19]:

lem\_text(train)

Out[19]:

|  | sentiment | message | tweetid |
| --- | --- | --- | --- |
| 0 | 1 | ORG chief donot think carbon dioxide main caus... | 625221 |
| 1 | 1 | like lack evidence anthropogenic global warming | 126103 |
| 2 | 2 | rt researchers say three years act climate cha... | 698562 |
| 3 | 1 | todayinmaker wired 2016 pivotal year war clima... | 573736 |
| 4 | 1 | rt 2016 racist sexist climate change denying b... | 466954 |
| ... | ... | ... | ... |
| 15814 | 1 | rt took material global warming lgbt rights he... | 22001 |
| 15815 | 2 | rt climate change could breaking 200millionyea... | 17856 |
| 15816 | 0 | notiven rt nytimesworld trump actually believe... | 384248 |
| 15817 | -1 | rt hey liberals climate change crap hoax ties ... | 819732 |
| 15818 | 0 | rt climate change equation 4 screenshots | 806319 |

15819 rows × 3 columns

In [20]:

lem\_text(test)

Out[20]:

|  | message | tweetid |
| --- | --- | --- |
| 0 | LOC look GPE make sure alone fight climate change | 169760 |
| 1 | combine polling staffers climate change womens... | 35326 |
| 2 | scary unimpeachable evidence climate change al... | 224985 |
| 3 | putin got jill trump doesnt believe climate ch... | 476263 |
| 4 | rt female orgasms cause global warming sarcast... | 872928 |
| ... | ... | ... |
| 10541 | rt brb writing poem climate change climatechan... | 895714 |
| 10542 | 2016 year climate change came home hottest yea... | 875167 |
| 10543 | rt pacific countries positive fiji leading glo... | 78329 |
| 10544 | rt youre hot must cause global warming aldubla... | 867455 |
| 10545 | rt climate change global issue thats getting w... | 470892 |

10546 rows × 2 columns

Exploratory Data Analysis

Now it is time to analyse the data and how it is structured.

Missing values

In [21]:

train.isnull().sum()

Out[21]:

sentiment 0

message 0

tweetid 0

dtype: int64

In [22]:

test.isnull().sum()

Out[22]:

message 0

tweetid 0

dtype: int64

There are no missing values in both the train and test datasets.

Sentiment Classes

In [23]:

train.sentiment.value\_counts()

Out[23]:

1 8530

2 3640

0 2353

-1 1296

Name: sentiment, dtype: int64

In [24]:

counts = train["sentiment"].value\_counts()

plt.bar(range(len(counts)), counts)

plt.xticks([0, 1, 2, 3], ['Pro', 'News', 'Neutral', 'Anti'])

plt.ylabel("Total per class")

plt.xlabel("Sentiment Classes")

plt.show()

In [25]:

*#Percentage of the major class*

len(train[train.sentiment==1])/len(train.sentiment)

Out[25]:

0.5392249826158417

The 'Pro' climate change class accounts for 54% of the data set and the remaining 46% is shared amongst the three other classes. This imbalance indicates how biased our model might me thus exploring resampling techniques might be required in order to improve the model accuracy.

Word Cloud

Below we show some of the words used in the different tweets under each class.

In [26]:

*#word clouds*

news = train[train['sentiment'] == 2]['message']

pro = train[train['sentiment'] == 1]['message']

neutral =train[train['sentiment'] == 0]['message']

Anti = train[train['sentiment'] ==-1]['message']

news = [word for line **in** news for word **in** line.split()]

pro = [word for line **in** pro for word **in** line.split()]

neutral = [word for line **in** neutral for word **in** line.split()]

Anti= [word for line **in** Anti for word **in** line.split()]

news = WordCloud(

background\_color='white',

max\_words=20,

max\_font\_size=40,

scale=5,

random\_state=1,

collocations=False,

normalize\_plurals=False

).generate(' '.join(news))

pro = WordCloud(

background\_color='white',

max\_words=20,

max\_font\_size=40,

scale=5,

random\_state=1,

collocations=False,

normalize\_plurals=False

).generate(' '.join(pro))

neutral = WordCloud(

background\_color='white',

max\_words=20,

max\_font\_size=40,

scale=5,

random\_state=1,

collocations=False,

normalize\_plurals=False

).generate(' '.join(neutral))

Anti = WordCloud(

background\_color='white',

max\_words=20,

max\_font\_size=40,

scale=5,

random\_state=1,

collocations=False,

normalize\_plurals=False

).generate(' '.join(Anti))

fig, axs = plt.subplots(2, 2, figsize = (20, 10))

fig.tight\_layout(pad = 0)

axs[0, 0].imshow(news)

axs[0, 0].set\_title('Words from news tweets', fontsize = 20)

axs[0, 0].axis('off')

axs[0, 1].imshow(pro)

axs[0, 1].set\_title('Words from pro tweets', fontsize = 20)

axs[0, 1].axis('off')

axs[1, 0].imshow(Anti)

axs[1, 0].set\_title('Words from anti tweets', fontsize = 20)

axs[1, 0].axis('off')

axs[1, 1].imshow(neutral)

axs[1, 1].set\_title('Words from neutral tweets', fontsize = 20)

axs[1, 1].axis('off')

plt.savefig('joint\_cloud.png')

Define our feature and target variable to use when modelling

In [27]:

X = train['message']

X

Out[27]:

0 ORG chief donot think carbon dioxide main caus...

1 like lack evidence anthropogenic global warming

2 rt researchers say three years act climate cha...

3 todayinmaker wired 2016 pivotal year war clima...

4 rt 2016 racist sexist climate change denying b...

...

15814 rt took material global warming lgbt rights he...

15815 rt climate change could breaking 200millionyea...

15816 notiven rt nytimesworld trump actually believe...

15817 rt hey liberals climate change crap hoax ties ...

15818 rt climate change equation 4 screenshots

Name: message, Length: 15819, dtype: object

In [28]:

y = train['sentiment']

y

Out[28]:

0 1

1 1

2 2

3 1

4 1

..

15814 1

15815 2

15816 0

15817 -1

15818 0

Name: sentiment, Length: 15819, dtype: int64

Convert features to machine language

In [29]:

tf\_vecto = TfidfVectorizer(lowercase = True,stop\_words = 'english',ngram\_range=(1, 2))

X = tf\_vecto.fit\_transform(X)

Splitting the data

In [30]:

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.2, random\_state = 42)

In [31]:

*#X\_test = test['message']*

In [32]:

*#X\_test = tf\_vect.transform(X\_test)*

Modelling

Modelling with maximum data cleaning

The function below takes in a model and splits our training data into a train and test set. Once this is done, the function calculates the accuracy score for both the train and test set and returns the avergace accuracy score for both.

In [33]:

def accuracy(model):

features = train['message']

target = train['sentiment']

train\_scores = []

test\_scores = []

*#tf\_vect = TfidfVectorizer(ngram\_range=(1, 2))*

tf\_vecto = TfidfVectorizer(lowercase = True,stop\_words = 'english',ngram\_range=(1, 2))

folds = KFold(n\_splits=5, shuffle=True)

for train\_index, test\_index **in** folds.split(features):

x\_train, x\_test = features.iloc[train\_index], features.iloc[test\_index]

y\_train, y\_test = target.iloc[train\_index], target.iloc[test\_index]

x\_train = tf\_vecto.fit\_transform(x\_train)

x\_test = tf\_vecto.transform(x\_test)

model.fit(x\_train, y\_train)

train\_predictions = model.predict(x\_train)

test\_predictions = model.predict(x\_test)

train\_score = accuracy\_score(y\_train, train\_predictions)

train\_scores.append(train\_score)

test\_score = accuracy\_score(y\_test, test\_predictions)

test\_scores.append(test\_score)

avg\_train\_accuracy = np.mean(train\_scores)

avg\_test\_accuracy = np.mean(test\_scores)

return [avg\_train\_accuracy, avg\_test\_accuracy]

Support Vector Machines(SVM)

SVM or Support Vector Machine is a linear model for classification and regression problems. It can solve linear and non-linear problems and work well for many practical problems. The idea of SVM is simple: The algorithm creates a line or a hyperplane which separates the data into classes

In [34]:

sv = SVC()

sv\_accuracy = accuracy(sv)

sv\_accuracy

Out[34]:

[0.9736392974898864, 0.7089582534784105]

Bernoulli Naive Bayes

The Bernoulli naive Bayes classifier assumes that all our features are binary such that they take only two values (e.g. a nominal categorical feature that has been one-hot encoded)

In [35]:

bernoulli = BernoulliNB()

bernoulli\_accuracy = accuracy(bernoulli)

bernoulli\_accuracy

Out[35]:

[0.7307194017944867, 0.6198876828436253]

Multi Nomial Naive Bayes

In summary, Naive Bayes classifier is a general term which refers to conditional independence of each of the features in the model, while Multinomial Naive Bayes classifier is a specific instance of a Naive Bayes classifier which uses a multinomial distribution for each of the features

In [36]:

mnb = MultinomialNB()

mnb\_accuracy = accuracy(mnb)

mnb\_accuracy

Out[36]:

[0.7225014385463489, 0.6372087501943497]

Stochastic Gradient Descent (SGD)

SGD is a simple,efficient approach to fitting linear classifiers and regressors under convex loss functions such as (linear) Support Vector Machines and Logistic Regression.It has received a considerable amount of attention just recently in the context of large-scale learning.

In [37]:

sgd = SGDClassifier()

sgd\_accuracy = accuracy(sgd)

sgd\_accuracy

Out[37]:

[0.9513085626973943, 0.7277954285746261]

Random Forest

The random forest is a classification algorithm consisting of many decisions trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

In [38]:

rand\_forest = RandomForestClassifier()

rand\_forest\_accuracy = accuracy(rand\_forest)

rand\_forest\_accuracy

Out[38]:

[0.9971711223308846, 0.6729251143016219]

K-Nearest Neighbors (KNN)

In pattern recognition, the k-nearest neighbors algorithm (k-NN) is a non-parametric method proposed by Thomas Cover used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. ... In k-NN classification, the output is a class membership.

In [39]:

knn = KNeighborsClassifier()

knn\_accuracy = accuracy(knn)

knn\_accuracy

Out[39]:

[0.7347651210951334, 0.6292425496606024]

Gradient Boosting Classifier

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.

In [40]:

grad\_booster = GradientBoostingClassifier()

grad\_booster\_accuracy = accuracy(grad\_booster)

grad\_booster\_accuracy

Out[40]:

[0.6832132330280252, 0.6407486131722951]

Extra Trees Classifier

An extra-trees classifier is a class that implements a meta estimator that fits a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The number of trees in the forest.

In [41]:

extra\_trees = ExtraTreesClassifier()

extra\_trees\_accuracy = accuracy(extra\_trees)

extra\_trees\_accuracy

Out[41]:

[0.9972027353858925, 0.7001082163271359]

Bagging

A Bagging classifier is an ensemble meta-estimator that fits base classifiers each on random subsets of the original dataset and then aggregate their individual predictions (either by voting or by averaging) to form a final prediction. ... If None, then the base estimator is a decision tree.

In [42]:

bagging = BaggingClassifier()

bagging\_accuracy = accuracy(bagging)

bagging\_accuracy

Out[42]:

[0.9705259510264878, 0.6159042428394365]

Decision Tree Classifier

The decision tree classifier (Pang-Ning et al., 2006) creates the classification model by building a decision tree. Each node in the tree specifies a test on an attribute, each branch descending from that node corresponds to one of the possible values for that attribute.

In [43]:

dec\_tree = DecisionTreeClassifier()

dec\_tree\_accuracy = accuracy(dec\_tree)

dec\_tree\_accuracy

Out[43]:

[0.9971711223308846, 0.5830325192561112]

Linear SVC

The objective of a Linear SVC (Support Vector Classifier) is to fit to the data provided, returning a "best fit" hyperplane that divides, or categorizes, data. After getting the hyperplane,the features are fed to the classifier to see what the "predicted" class is.

In [44]:

linear\_sv = LinearSVC()

linear\_sv\_accuracy = accuracy(linear\_sv)

linear\_sv\_accuracy

Out[44]:

[0.9940419880710543, 0.7271642965658953]

Plot all models and respective accuracy scores

Below is a graph of all models which allows for simplified comparison

In [45]:

models = ['SVC', 'Bernoulli', 'Multinomial Naive Bayes', 'SGDClassifier', 'Random Forest', 'KNearestNeighbours', 'Gradient Booster', 'Extra Trees', 'Bagging', 'Decision Tree', 'Linear SV']

bar\_widths = [sv\_accuracy[1], bernoulli\_accuracy[1], mnb\_accuracy[1], sgd\_accuracy[1], rand\_forest\_accuracy[1], knn\_accuracy[1], grad\_booster\_accuracy[1], extra\_trees\_accuracy[1], bagging\_accuracy[1], dec\_tree\_accuracy[1], linear\_sv\_accuracy[1]]

bar\_positions = arange(11) + 0.75

tick\_positions = range(1,12)

fig, ax = plt.subplots()

ax.barh(bar\_positions, bar\_widths, 0.5)

ax.set\_yticks(tick\_positions)

ax.set\_yticklabels(models)

ax.set\_ylabel('Model')

ax.set\_xlabel('Accuracy')

ax.set\_title('Accuracy For Each Model Trained')

plt.show()

Fit best models and use the metrics module

In [46]:

linear\_sv.fit(X\_train, y\_train)

Out[46]:

LinearSVC()

In [47]:

*#confusion matrix and classification\_report*

y\_pred = linear\_sv.predict(X\_test)

print(confusion\_matrix(y\_test,y\_pred))

print('**\n\n**Accuracy score: ' + str(accuracy\_score(y\_test, y\_pred)))

print("**\n\n**Classification Report:**\n\n**", classification\_report(y\_test,y\_pred,target\_names=['Anti', 'Neutral','Pro','News']))

[[ 99 30 142 7]

[ 10 147 250 18]

[ 9 46 1585 115]

[ 3 10 199 494]]

Accuracy score: 0.7348293299620733

Classification Report:

precision recall f1-score support

Anti 0.82 0.36 0.50 278

Neutral 0.63 0.35 0.45 425

Pro 0.73 0.90 0.81 1755

News 0.78 0.70 0.74 706

accuracy 0.73 3164

macro avg 0.74 0.58 0.62 3164

weighted avg 0.73 0.73 0.72 3164

Below is a visual of the confusion matrix

In [48]:

sentiment\_code = {1:'Pro', 2:'News', 0:'Neutral', -1:'Anti'}

In [49]:

train['sentiment\_code'] = train['sentiment'].map(sentiment\_code)

In [50]:

aux\_train = train[['sentiment', 'sentiment\_code']].drop\_duplicates().sort\_values('sentiment\_code')

conf\_matrix = confusion\_matrix(y\_test, y\_pred)

plt.figure(figsize=(12.8,6))

sns.heatmap(conf\_matrix,

annot=True,

cbar=False,

fmt='g',

xticklabels=aux\_train['sentiment'].values,

yticklabels=aux\_train['sentiment'].values,

cmap="Blues")

plt.ylabel('Predicted')

plt.xlabel('Actual')

plt.title('Confusion matrix')

plt.show()

In [51]:

sgd.fit(X\_train,y\_train)

Out[51]:

SGDClassifier()

In [52]:

y\_predict = sgd.predict(X\_test)

print(confusion\_matrix(y\_test,y\_predict))

print('**\n\n**Accuracy score: ' + str(accuracy\_score(y\_test, y\_pred)))

print("**\n\n**Classification Report:**\n\n**", classification\_report(y\_test,y\_predict,target\_names=['Anti', 'Neutral','Pro','News']))

[[ 90 25 154 9]

[ 6 124 278 17]

[ 3 34 1605 113]

[ 4 6 219 477]]

Accuracy score: 0.7348293299620733

Classification Report:

precision recall f1-score support

Anti 0.87 0.32 0.47 278

Neutral 0.66 0.29 0.40 425

Pro 0.71 0.91 0.80 1755

News 0.77 0.68 0.72 706

accuracy 0.73 3164

macro avg 0.75 0.55 0.60 3164

weighted avg 0.73 0.73 0.70 3164

In [53]:

sentiment\_code = {1:'Pro', 2:'News', 0:'Neutral', -1:'Anti'}

In [54]:

train['sentiment\_code'] = train['sentiment'].map(sentiment\_code)

In [55]:

aux\_train = train[['sentiment', 'sentiment\_code']].drop\_duplicates().sort\_values('sentiment\_code')

conf\_matrix = confusion\_matrix(y\_test, y\_predict)

plt.figure(figsize=(12.8,6))

sns.heatmap(conf\_matrix,

annot=True,

cbar=False,

fmt='g',

xticklabels=aux\_train['sentiment'].values,

yticklabels=aux\_train['sentiment'].values,

cmap="Blues")

plt.ylabel('Predicted')

plt.xlabel('Actual')

plt.title('Confusion matrix')

plt.show()

Modelling with resampled data

Since there has already been an observation that the data is imbalanced, one must attempt improving the accuracy of the above modelling.

This will be done by introducing resampling techniques, Downsampling and SMOTE.

Downsampling

In [56]:

train = pd.read\_csv('https://raw.githubusercontent.com/Stephane-Masamba/Team\_4\_CPT\_ML-Classification/Mikael/train%20(1).csv')

print(train.head())

sentiment message tweetid

0 1 PolySciMajor EPA chief doesn't think carbon di... 625221

1 1 It's not like we lack evidence of anthropogeni... 126103

2 2 RT @RawStory: Researchers say we have three ye... 698562

3 1 #TodayinMaker# WIRED : 2016 was a pivotal year... 573736

4 1 RT @SoyNovioDeTodas: It's 2016, and a racist, ... 466954

In [57]:

test = pd.read\_csv('https://raw.githubusercontent.com/Stephane-Masamba/Team\_4\_CPT\_ML-Classification/Mikael/test%20(1).csv')

print(test.head())

message tweetid

0 Europe will now be looking to China to make su... 169760

1 Combine this with the polling of staffers re c... 35326

2 The scary, unimpeachable evidence that climate... 224985

3 @Karoli @morgfair @OsborneInk @dailykos \nPuti... 476263

4 RT @FakeWillMoore: 'Female orgasms cause globa... 872928

Clean the data by calling the cleaning functions

In [58]:

clean\_text(train)

remove\_stopwords(train)

entities(train)

lem\_text(train)

Out[58]:

|  | sentiment | message | tweetid |
| --- | --- | --- | --- |
| 0 | 1 | ORG chief donot think carbon dioxide main caus... | 625221 |
| 1 | 1 | like lack evidence anthropogenic global warming | 126103 |
| 2 | 2 | rt researchers say three years act climate cha... | 698562 |
| 3 | 1 | todayinmaker wired 2016 pivotal year war clima... | 573736 |
| 4 | 1 | rt 2016 racist sexist climate change denying b... | 466954 |
| ... | ... | ... | ... |
| 15814 | 1 | rt took material global warming lgbt rights he... | 22001 |
| 15815 | 2 | rt climate change could breaking 200millionyea... | 17856 |
| 15816 | 0 | notiven rt nytimesworld trump actually believe... | 384248 |
| 15817 | -1 | rt hey liberals climate change crap hoax ties ... | 819732 |
| 15818 | 0 | rt climate change equation 4 screenshots | 806319 |

15819 rows × 3 columns

In [59]:

train\_majority = train[train.sentiment== 1]

train\_0 = train[train.sentiment== 0]

train\_2 = train[train.sentiment== 2]

train\_minority = train[train.sentiment==-1]

In [60]:

*# Downsample majority classes*

train\_majority\_downsampled = resample(train\_majority,

replace=False, *# sample without replacement*

n\_samples=1296, *# to match minority class*

random\_state=123) *# reproducible results*

train\_0\_downsampled = resample(train\_0,

replace=False,

n\_samples=1296,

random\_state=123)

train\_2\_downsampled = resample(train\_2,

replace=False,

n\_samples=1296,

random\_state=123)

*# Combine minority class with downsampled majority class*

train\_downsampled1 = pd.concat([train\_0\_downsampled,train\_2\_downsampled])

train\_downsampled2 = pd.concat([train\_majority\_downsampled, train\_minority])

train\_downsampled = pd.concat([train\_downsampled1, train\_downsampled2])

In [61]:

train\_downsampled

Out[61]:

|  | sentiment | message | tweetid |
| --- | --- | --- | --- |
| 13470 | 0 | rt really god global warming exist movie pulp ... | 574574 |
| 8148 | 0 | also storm global warming maybe probably | 761606 |
| 11544 | 0 | rt savor ur cold ones u climate change gon na ... | 639609 |
| 7282 | 0 | rt ive begun miss days endless papers global w... | 121132 |
| 12634 | 0 | omg celebrities moral actions climate change s... | 744560 |
| ... | ... | ... | ... |
| 15704 | -1 | rt point wthe delusional left literally every ... | 890582 |
| 15705 | -1 | rt al gore admits paris accord wont solve issu... | 380307 |
| 15711 | -1 | rt funny people believe climate change bc scie... | 390339 |
| 15806 | -1 | rt al gore idiot thinks climate change cause p... | 213863 |
| 15817 | -1 | rt hey liberals climate change crap hoax ties ... | 819732 |

5184 rows × 3 columns

In [62]:

train\_downsampled['sentiment'].value\_counts()

Out[62]:

-1 1296

2 1296

1 1296

0 1296

Name: sentiment, dtype: int64

In [63]:

counts = train["sentiment"].value\_counts()

counti = train\_downsampled['sentiment'].value\_counts()

plt.bar(range(len(counts)), counts)

plt.bar(range(len(counts)),counti,color='red')

plt.xticks([0, 1, 2, 3], ['Pro', 'News', 'Neutral', 'Anti'])

plt.ylabel("Total per class")

plt.xlabel("Sentiment Classes")

plt.legend(['original','resampled'])

plt.show()

Now that the data is balanced, we go ahead and build a model. This time we check the performance with onlyy the two best performing models as per the above graph.

In [64]:

X\_down = train['message']

X\_down

Out[64]:

0 ORG chief donot think carbon dioxide main caus...

1 like lack evidence anthropogenic global warming

2 rt researchers say three years act climate cha...

3 todayinmaker wired 2016 pivotal year war clima...

4 rt 2016 racist sexist climate change denying b...

...

15814 rt took material global warming lgbt rights he...

15815 rt climate change could breaking 200millionyea...

15816 notiven rt nytimesworld trump actually believe...

15817 rt hey liberals climate change crap hoax ties ...

15818 rt climate change equation 4 screenshots

Name: message, Length: 15819, dtype: object

In [65]:

y\_down = train['sentiment']

y\_down

Out[65]:

0 1

1 1

2 2

3 1

4 1

..

15814 1

15815 2

15816 0

15817 -1

15818 0

Name: sentiment, Length: 15819, dtype: int64

In [66]:

X\_down = tf\_vecto.fit\_transform(X\_down)

X\_down

Out[66]:

<15819x88009 sparse matrix of type '<class 'numpy.float64'>'

with 300338 stored elements in Compressed Sparse Row format>

In [67]:

*#train\_test\_split*

X\_train1,X\_test1,y\_train1,y\_test1 = train\_test\_split(X\_down,y\_down,test\_size=0.2,random\_state=0)

In [68]:

lsvm = LinearSVC()

lsvm.fit(X\_train1, y\_train1)

Out[68]:

LinearSVC()

In [69]:

*#confusion matrix and classification\_report*

y\_pred1 = lsvm.predict(X\_test1)

print(confusion\_matrix(y\_test1,y\_pred1))

print('**\n\n**Accuracy score: ' + str(accuracy\_score(y\_test1, y\_pred1)))

print("**\n\n**Classification Report:**\n\n**",classification\_report(y\_test1,y\_pred1,target\_names=['Anti', 'Neutral','Pro','News']))

[[ 77 23 129 20]

[ 12 149 278 34]

[ 12 50 1530 109]

[ 4 8 199 530]]

Accuracy score: 0.7225031605562579

Classification Report:

precision recall f1-score support

Anti 0.73 0.31 0.44 249

Neutral 0.65 0.32 0.42 473

Pro 0.72 0.90 0.80 1701

News 0.76 0.72 0.74 741

accuracy 0.72 3164

macro avg 0.72 0.56 0.60 3164

weighted avg 0.72 0.72 0.70 3164

In [70]:

sentiment\_code = {1:'Pro', 2:'News', 0:'Neutral', -1:'Anti'}

In [71]:

train['sentiment\_code'] = train['sentiment'].map(sentiment\_code)

In [72]:

aux\_train = train[['sentiment', 'sentiment\_code']].drop\_duplicates().sort\_values('sentiment\_code')

conf\_matrix = confusion\_matrix(y\_test1, y\_pred1)

plt.figure(figsize=(12.8,6))

sns.heatmap(conf\_matrix,

annot=True,

cbar=False,

fmt='g',

xticklabels=aux\_train['sentiment'].values,

yticklabels=aux\_train['sentiment'].values,

cmap="Blues")

plt.ylabel('Predicted')

plt.xlabel('Actual')

plt.title('Confusion matrix')

plt.show()

SMOTE

Using data from the first data split

In [73]:

print(X\_train.shape,y\_train.shape)

(12655, 88009) (12655,)

In [74]:

smote = SMOTE("minority")

X\_sm , y\_sm = smote.fit\_resample(X\_train,y\_train)

/opt/conda/lib/python3.7/site-packages/imblearn/utils/\_validation.py:638: FutureWarning: Pass sampling\_strategy=minority as keyword args. From version 0.9 passing these as positional arguments will result in an error

FutureWarning)

In [75]:

print(X\_sm.shape,y\_sm.shape)

(18412, 88009) (18412,)

In [76]:

ls= LinearSVC()

ls.fit(X\_sm, y\_sm)

*#confusion matrix and classification\_report*

y\_predsm = ls.predict(X\_test)

print(confusion\_matrix(y\_test,y\_predsm))

print('**\n\n**Accuracy score: ' + str(accuracy\_score(y\_test, y\_predsm)))

print("**\n\n**Classification Report:**\n\n**",classification\_report(y\_test,y\_predsm,target\_names=['Anti', 'Neutral','Pro','News']))

[[ 134 18 119 7]

[ 42 118 248 17]

[ 35 46 1561 113]

[ 8 12 196 490]]

Accuracy score: 0.7278761061946902

Classification Report:

precision recall f1-score support

Anti 0.61 0.48 0.54 278

Neutral 0.61 0.28 0.38 425

Pro 0.73 0.89 0.80 1755

News 0.78 0.69 0.74 706

accuracy 0.73 3164

macro avg 0.68 0.59 0.62 3164

weighted avg 0.72 0.73 0.71 3164

In [77]:

sentiment\_code = {1:'Pro', 2:'News', 0:'Neutral', -1:'Anti'}

In [78]:

train['sentiment\_code'] = train['sentiment'].map(sentiment\_code)

In [79]:

aux\_train = train[['sentiment', 'sentiment\_code']].drop\_duplicates().sort\_values('sentiment\_code')

conf\_matrix = confusion\_matrix(y\_test, y\_predsm)

plt.figure(figsize=(12.8,6))

sns.heatmap(conf\_matrix,

annot=True,

cbar=False,

fmt='g',

xticklabels=aux\_train['sentiment'].values,

yticklabels=aux\_train['sentiment'].values,

cmap="Blues")

plt.ylabel('Predicted')

plt.xlabel('Actual')

plt.title('Confusion matrix')

plt.show()

Modelling with minimal data cleaning

To conclude the modelling section, we will explore what one could have done by intuition. This is just to check how accurate our model would be if we did not introduce the 'fancy' cleaning functions.

In [80]:

train = pd.read\_csv('https://raw.githubusercontent.com/Stephane-Masamba/Team\_4\_CPT\_ML-Classification/Mikael/train%20(1).csv')

print(train.head())

sentiment message tweetid

0 1 PolySciMajor EPA chief doesn't think carbon di... 625221

1 1 It's not like we lack evidence of anthropogeni... 126103

2 2 RT @RawStory: Researchers say we have three ye... 698562

3 1 #TodayinMaker# WIRED : 2016 was a pivotal year... 573736

4 1 RT @SoyNovioDeTodas: It's 2016, and a racist, ... 466954

In [81]:

test = pd.read\_csv('https://raw.githubusercontent.com/Stephane-Masamba/Team\_4\_CPT\_ML-Classification/Mikael/test%20(1).csv')

print(test.head())

message tweetid

0 Europe will now be looking to China to make su... 169760

1 Combine this with the polling of staffers re c... 35326

2 The scary, unimpeachable evidence that climate... 224985

3 @Karoli @morgfair @OsborneInk @dailykos \nPuti... 476263

4 RT @FakeWillMoore: 'Female orgasms cause globa... 872928

In [82]:

sample = pd.read\_csv('https://raw.githubusercontent.com/Stephane-Masamba/Team\_4\_CPT\_ML-Classification/Mikael/sample\_submission.csv')

print(sample.head())

tweetid sentiment

0 169760 1

1 35326 1

2 224985 1

3 476263 1

4 872928 1

This time we only call one cleaning function which is the clean\_text() function. This does not include tokenization as well as lemmatization

In [83]:

clean\_text(train)

remove\_stopwords(train)

Out[83]:

|  | sentiment | message | tweetid |
| --- | --- | --- | --- |
| 0 | 1 | polyscimajor epa chief doesnt think carbon dio... | 625221 |
| 1 | 1 | like lack evidence anthropogenic global warming | 126103 |
| 2 | 2 | rt researchers say three years act climate cha... | 698562 |
| 3 | 1 | todayinmaker wired 2016 pivotal year war clima... | 573736 |
| 4 | 1 | rt 2016 racist sexist climate change denying b... | 466954 |
| ... | ... | ... | ... |
| 15814 | 1 | rt took material global warming lgbt rights he... | 22001 |
| 15815 | 2 | rt climate change could breaking 200millionyea... | 17856 |
| 15816 | 0 | notiven rt nytimesworld trump actually believe... | 384248 |
| 15817 | -1 | rt hey liberals climate change crap hoax ties ... | 819732 |
| 15818 | 0 | rt climate change equation 4 screenshots | 806319 |

15819 rows × 3 columns

In [84]:

clean\_text(test)

remove\_stopwords(test)

Out[84]:

|  | message | tweetid |
| --- | --- | --- |
| 0 | europe looking china make sure alone fighting ... | 169760 |
| 1 | combine polling staffers climate change womens... | 35326 |
| 2 | scary unimpeachable evidence climate change al... | 224985 |
| 3 | putin got jill trump doesnt believe climate ch... | 476263 |
| 4 | rt female orgasms cause global warming sarcast... | 872928 |
| ... | ... | ... |
| 10541 | rt brb writing poem climate change climatechan... | 895714 |
| 10542 | 2016 year climate change came home hottest yea... | 875167 |
| 10543 | rt pacific countries positive fiji leading glo... | 78329 |
| 10544 | rt youre hot must cause global warming aldubla... | 867455 |
| 10545 | rt climate change global issue thats getting w... | 470892 |

10546 rows × 2 columns

Features and Modelling

In [85]:

X\_min = train['message']

X\_min

Out[85]:

0 polyscimajor epa chief doesnt think carbon dio...

1 like lack evidence anthropogenic global warming

2 rt researchers say three years act climate cha...

3 todayinmaker wired 2016 pivotal year war clima...

4 rt 2016 racist sexist climate change denying b...

...

15814 rt took material global warming lgbt rights he...

15815 rt climate change could breaking 200millionyea...

15816 notiven rt nytimesworld trump actually believe...

15817 rt hey liberals climate change crap hoax ties ...

15818 rt climate change equation 4 screenshots

Name: message, Length: 15819, dtype: object

In [86]:

y\_min = train['sentiment']

y\_min

Out[86]:

0 1

1 1

2 2

3 1

4 1

..

15814 1

15815 2

15816 0

15817 -1

15818 0

Name: sentiment, Length: 15819, dtype: int64

Convert to machine language

In [87]:

X\_min = tf\_vecto.fit\_transform(X\_min)

In [88]:

*#train\_test\_split*

X\_train2,X\_test2,y\_train2,y\_test2 = train\_test\_split(X\_min,y\_min,test\_size=0.2,random\_state=0)

Stochastic Gradient Descent

In [89]:

sgd = SGDClassifier()

sgd\_accuracy = accuracy(sgd)

sgd\_accuracy

Out[89]:

[0.951261165592169, 0.7281121436904986]

Linear SVC

In [90]:

linear\_sv = LinearSVC()

linear\_sv\_accuracy = accuracy(linear\_sv)

linear\_sv\_accuracy

Out[90]:

[0.9939629616772251, 0.729186333127226]

In [91]:

linear\_sv.fit(X\_train2, y\_train2)

Out[91]:

LinearSVC()

In [92]:

*#confusion matrix and classification\_report*

y\_pred2 = linear\_sv.predict(X\_test2)

print(confusion\_matrix(y\_test1,y\_pred1))

print('**\n\n**Accuracy score: ' + str(accuracy\_score(y\_test2, y\_pred1)))

print("**\n\n**Classification Report:**\n\n**",classification\_report(y\_test2,y\_pred2,target\_names=['Anti', 'Neutral','Pro','News']))

[[ 77 23 129 20]

[ 12 149 278 34]

[ 12 50 1530 109]

[ 4 8 199 530]]

Accuracy score: 0.7225031605562579

Classification Report:

precision recall f1-score support

Anti 0.73 0.31 0.44 249

Neutral 0.65 0.32 0.42 473

Pro 0.72 0.90 0.80 1701

News 0.76 0.72 0.74 741

accuracy 0.72 3164

macro avg 0.72 0.56 0.60 3164

weighted avg 0.72 0.72 0.70 3164

In [93]:

sentiment\_code = {1:'Pro', 2:'News', 0:'Neutral', -1:'Anti'}

In [94]:

train['sentiment\_code'] = train['sentiment'].map(sentiment\_code)

In [95]:

aux\_train = train[['sentiment', 'sentiment\_code']].drop\_duplicates().sort\_values('sentiment\_code')

conf\_matrix = confusion\_matrix(y\_test1, y\_pred1)

plt.figure(figsize=(12.8,6))

sns.heatmap(conf\_matrix,

annot=True,

cbar=False,

fmt='g',

xticklabels=aux\_train['sentiment'].values,

yticklabels=aux\_train['sentiment'].values,

cmap="Blues")

plt.ylabel('Predicted')

plt.xlabel('Actual')

plt.title('Confusion matrix')

plt.show()

Results and ending comet experiment

In [96]:

*# import comet\_ml in the top of your file*

from comet\_ml import Experiment

*# Add the following code anywhere in your machine learning file*

experiment = Experiment(api\_key="kyaDe1YHDUV60KbpzF3dVpIuk",

project\_name="general", workspace="rachel-ramonyai")

---------------------------------------------------------------------------

ImportError Traceback (most recent call last)

<ipython-input-96-0be51143aeb7> in <module>

**4** # Add the following code anywhere in your machine learning file

**5** experiment = Experiment(api\_key="kyaDe1YHDUV60KbpzF3dVpIuk",

----> 6 project\_name="general", workspace="rachel-ramonyai")

/opt/conda/lib/python3.7/site-packages/comet\_ml/\_\_init\_\_.py in \_\_init\_\_(self, api\_key, project\_name, workspace, log\_code, log\_graph, auto\_param\_logging, auto\_metric\_logging, parse\_args, auto\_output\_logging, log\_env\_details, log\_git\_metadata, log\_git\_patch, disabled, log\_env\_gpu, log\_env\_host, display\_summary, log\_env\_cpu, display\_summary\_level, optimizer\_data)

**216** display\_summary\_level=display\_summary\_level,

**217** log\_env\_cpu=log\_env\_cpu,

--> 218 optimizer\_data=optimizer\_data,

**219** )

**220**

/opt/conda/lib/python3.7/site-packages/comet\_ml/experiment.py in \_\_init\_\_(self, project\_name, workspace, log\_code, log\_graph, auto\_param\_logging, auto\_metric\_logging, parse\_args, auto\_output\_logging, log\_env\_details, log\_git\_metadata, log\_git\_patch, disabled, log\_env\_gpu, log\_env\_host, display\_summary, log\_env\_cpu, display\_summary\_level, optimizer\_data)

**380** ALREADY\_IMPORTED\_MODULES

**381** )

--> 382 raise ImportError(msg)

**383**

**384** # Generate a unique identifier for this experiment.

ImportError: You must import Comet before these modules: tensorflow, tensorboard

In [97]:

f1 = f1\_score(y\_test, y\_predsm,average='macro')

precision = precision\_score(y\_test, y\_pred,average='macro')

recall = recall\_score(y\_test, y\_pred,average='macro')

In [98]:

params = {"kernel": 'linear',

"model\_type": "SVC",

"stratify": True

}

In [99]:

params = {

"model\_type": "Best LinearSVC",

"stratify": True

}

metrics = {"f1": f1,

"recall": recall,

"precision": precision

}

In [100]:

*# Log our parameters and results*

experiment.log\_parameters(params)

experiment.log\_metrics(metrics)

---------------------------------------------------------------------------

NameError Traceback (most recent call last)

<ipython-input-100-26114776ea3c> in <module>

**1** # Log our parameters and results

----> 2 experiment.log\_parameters(params)

**3** experiment.log\_metrics(metrics)

NameError: name 'experiment' is not defined

In [101]:

experiment.end()

---------------------------------------------------------------------------

NameError Traceback (most recent call last)

<ipython-input-101-86c369aed6fc> in <module>

----> 1 experiment.end()

NameError: name 'experiment' is not defined

Model Analysis and Insights

We worked with various models, and at first the linear SV model and the SGD classifier performed the best, with around 70% accuracy.

We then decided to fit these two models and increase their performance. Both of the models achieved different precision,recall & F1 scores.

With the SV model performing better with precision, but acheiving an accuarcy of 72%. While the SGD model achieved better scores in the other categories, however both models achieved the same accuracy at first.

The last model we made after extensive cleaning achieved a lower accuracy by 1%, but the predictions between anti,pro,neutral and news were closer together. Without a clear leader. This better reflects the real world findings.

The 'Pro' climate change class accounts for 54% of the data set and the remaining 46% is shared amongst the three other classes.

This 54% count for 'Pro' class means more people agree that climate change is a real threat to our eco-system. Companies that offer products and services which are environmentally friendly and sustainable are more likely to receive support from people based on the shared sentiments and beliefs about climate change.

The market reasearch done by these companies has positive results which is an indication for prospects of goood profit margins for eco-friendly products and services.

Accuracy on holdout data set is always lower and this shows that our model tends to be slightly overfitting the training data...best performing model is linear svc.

Conclusion

Our model predicted the news category best, while the other categories did improve, with our model being able to predict anti sentiment the second best. Although only more accurate by a couple of percent. So overall the model performed as anticipated.

Through the classification techniques, companies will be able to access a broad base of consumer sentiments, spanning multiple demographic and geographic categories, this will increase their insights and inform strategies.

This includes adding to their market research efforts in gauging how their product or service may be received.

Based on the findings it can be concluded that more customers will be receptive to eco-friendly products and services provided by companies.

The results prove that more people believe that climate change is a threat to our lives and that it is up to us to reduce our carbon footprint.