Project:

```
!pip install xgboost
!pip install lightgbm
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('fivethirtyeight')
import warnings
warnings.filterwarnings('ignore')
import nltk
import re
from nltk.stem import PorterStemmer # for stemming
from nltk.stem import WordNetLemmatizer # for lemmatization
from nltk.corpus import stopwords
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
from sklearn.preprocessing import LabelEncoder
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from xgboost import XGBClassifier
from lightgbm import LGBMClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: xgboost in /usr/local/lib/python3.8/dist-packages (0.90)
Requirement already satisfied: scipy in /usr/local/lib/python3.8/dist-packages (from xgboost) (1.7.3)
Requirement already satisfied: numpy in /usr/local/lib/python3.8/dist-packages (from xgboost) (1.21.6)
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: lightgbm in /usr/local/lib/python3.8/dist-packages (2.2.3)
Requirement already satisfied: numpy in /usr/local/lib/python3.8/dist-packages (from lightgbm) (1.21.6)
Requirement already satisfied: scipy in /usr/local/lib/python3.8/dist-packages (from lightgbm) (1.7.3)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.8/dist-packages (from lightgbm) (1.0.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.8/dist-packages (from scikit-learn->lightgbm) (3.1.0)
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.8/dist-packages (from scikit-learn->lightgbm) (1.2.0)
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
[nltk_data] Downloading package wordnet to /root/nltk_data...
```

```
%matplotlib inline
```

```
data = pd.read_csv('/content/gender-classifier.csv', encoding = 'latin1')
data.head()
```

	_unit_id	_golden	_unit_state	_trusted_judgments	_last_judgment_at	gender
0	815719226	False	finalized	3	10/26/15 23:24	male
1	815719227	False	finalized	3	10/26/15 23:30	male

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20050 entries, 0 to 20049
Data columns (total 26 columns):
 #   Column                 Non-Null Count  Dtype  
---  ------                 --------------  -----  
 0   _unit_id               20050 non-null  int64  
 1   _golden                20050 non-null  bool   
 2   _unit_state            20050 non-null  object 
 3   _trusted_judgments     20050 non-null  int64  
 4   _last_judgment_at      20000 non-null  object 
 5   gender                 19953 non-null  object 
 6   gender:confidence      20024 non-null  float64
 7   profile_yn             20050 non-null  object 
 8   profile_yn:confidence  20050 non-null  float64
 9   created                20050 non-null  object 
 10  description            16306 non-null  object 
 11  fav_number             20050 non-null  int64  
 12  gender_gold            50 non-null     object 
 13  link_color             20050 non-null  object 
 14  name                   20050 non-null  object 
 15  profile_yn_gold        50 non-null     object 
 16  profileimage           20050 non-null  object 
 17  retweet_count          20050 non-null  int64  
 18  sidebar_color          20050 non-null  object 
 19  text                   20050 non-null  object 
 20  tweet_coord            159 non-null    object 
 21  tweet_count            20050 non-null  int64  
 22  tweet_created          20050 non-null  object 
 23  tweet_id               20050 non-null  float64
 24  tweet_location         12566 non-null  object 
 25  user_timezone          12252 non-null  object 
dtypes: bool(1), float64(3), int64(5), object(17)
memory usage: 3.8+ MB
```

```
# Drop unique attribute columns and redundant
df = data[['gender', 'description', 'text', "name"]]
df.head()
```

	gender	description	text	name
0	male	i sing my own rhythm.	Robbie E Responds To Critics After Win Against...	sheezy0
1	male	I'm the author of novels filled with family dr...	ÛÏIt felt like they were my friends and I was...	DavdBurnett
2	male	louis whining and squealing and all	i absolutely adore when louis starts the songs...	lwtprettylaugh
3	male	Mobile guy. 49ers, Shazam, Google, Klein... Re	Hi @JordanSpieth - Looking at the ... do you	douggarland

```
# Check for Null Values
print(df.isna().sum())
df.dropna(axis=0, inplace=True)
```

```
gender           97
description    3744
text              0
name              0
dtype: int64
```

```
# Explore gender counts and strata
df['gender'].value_counts()
```

```
female     5725
male       5469
```

```
brand      4328
unknown     702
Name: gender, dtype: int64
```

```
# Parse only Male and Female
df = df[(df['gender'] == "male") | (df['gender'] == "female")]
df.head()
```

	gender	description	text	name
0	male	i sing my own rhythm.	Robbie E Responds To Critics After Win Against...	sheezy0
1	male	I'm the author of novels filled with family dr...	ÛÏIt felt like they were my friends and I was...	DavdBurnett
2	male	louis whining and squealing and all	i absolutely adore when louis starts the songs...	lwtprettylaugh
3	male	Mobile guy. 49ers, Shazam, Google, Kleiner Pe...	Hi @JordanSpieth - Looking at the url - do you...	douggarland

```
df['gender'].value_counts()
```

```
female    5725
male      5469
Name: gender, dtype: int64
```

```
print("Number of instances: ", len(df))
```

```
Number of instances:  11194
```

```
# Encoding Gender Labels
label_map = {"female":1, "male":0}
df["label"] = df["gender"].map(label_map)
df = df.drop(["gender"], axis=1)
df.head()
```

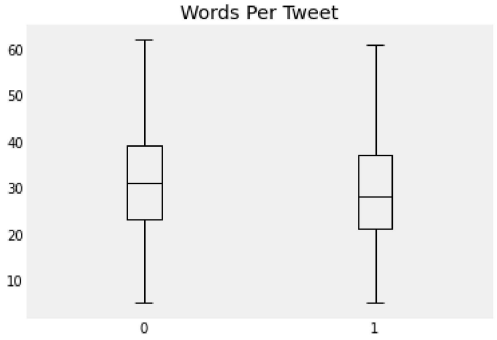
	description	text	name	label
0	i sing my own rhythm.	Robbie E Responds To Critics After Win Against...	sheezy0	0
1	I'm the author of novels filled with family dr...	ÛÏIt felt like they were my friends and I was...	DavdBurnett	0
2	louis whining and squealing and all	i absolutely adore when louis starts the songs...	lwtprettylaugh	0
3	Mobile guy. 49ers, Shazam, Google, Kleiner Pe...	Hi @JordanSpieth - Looking at the url - do you...	douggarland	0

```
#df["text"] = df["description"] + ", " + df["text"]
#df = df.drop(["description"], axis=1)
df["text"] = df["name"] + ", " + df["description"] + ", " + df["text"]
df = df.drop(["description", "name"], axis=1)
df.head()
```

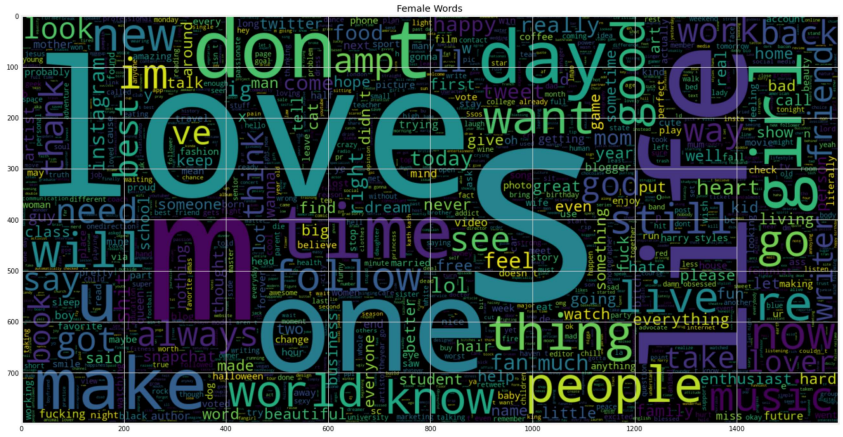
	text	label
0	sheezy0, i sing my own rhythm., Robbie E Respo...	0
1	DavdBurnett, I'm the author of novels filled w...	0
2	lwtprettylaugh, louis whining and squealing an...	0
3	douggarland, Mobile guy. 49ers, Shazam, Googl...	0
4	WilfordGemma, Ricky Wilson The Best FRONTMAN/K...	1

```
# Data Preprocessing
text_cleaning_re = "@\S+|https?:\S+|http?:\S|[^A-Za-z0-9]+"

def preprocessing(regex, text):
  text = re.sub(regex, ' ', str(text).lower()).strip()
  return text

df.text = df.text.apply(lambda x: preprocessing(text_cleaning_re, x))
```

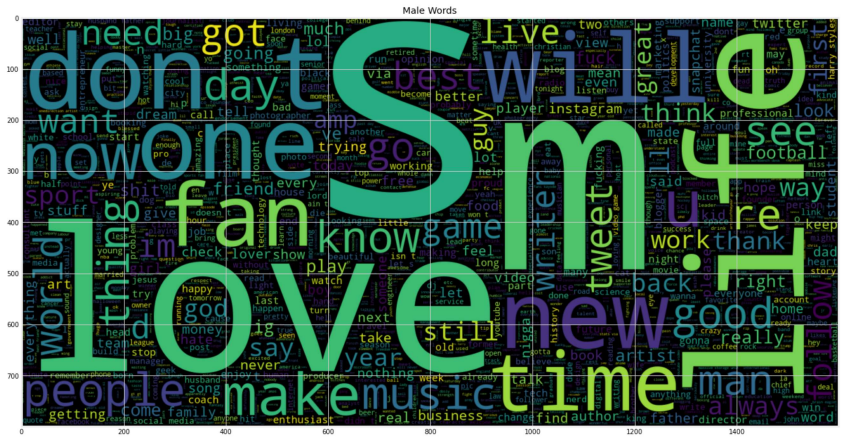
```
df.loc[:, "Words Per Tweet"] = df["text"].str.split().str.len()
df.boxplot("Words Per Tweet", by="label", grid=False, showfliers=False,
           color="black")
plt.suptitle("")
plt.xlabel("")
plt.show()
df = df.drop(["Words Per Tweet"], axis=1)
```




```
from wordcloud import WordCloud

plt.figure(figsize = (20,20))
wc = WordCloud(max_words = 2000 , width = 1600 , height = 800).generate(" ".join(df[df.label == 1].text))
plt.imshow(wc , interpolation = 'bilinear')
plt.title("Female Words")
plt.show()
```




```
plt.figure(figsize = (20,20))
wc = WordCloud(max_words = 2000 , width = 1600 , height = 800).generate(" ".join(df[df.label == 0].text))
plt.imshow(wc , interpolation = 'bilinear')
plt.title("Male Words")
plt.show()
```




```
df_xg = df.copy()
df_xg = df_xg[["label", "text"]]
```

```
train_data, test_data = train_test_split(df_xg, test_size = 0.1, random_state=7)
train_data, val_data = train_test_split(train_data, test_size=0.20, random_state=7)

print("Train Data size:", len(train_data))
print("Val Data size", len(val_data))
print("Test Data size", len(test_data))
```

```
Train Data size: 8059
Val Data size 2015
Test Data size 1120
```

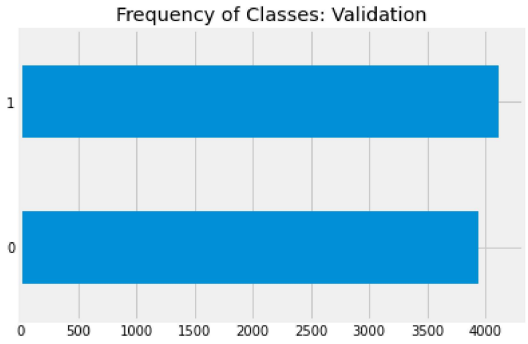
```
# Print Sample
train_data.head(10)
```

	label	text
10822	1	bestdateever relationships couples dating can ...
24	0	jhurkett bsc economics graduate coys james bon...
4066	1	tamrynseale 5sos magcon 1d the vamps the tide ...
6465	0	ity17 god family football if i aint the best j...
11470	0	poeboy412 you came here for a reason just foll...
18634	0	warriorbob9 the 9 is silent my most and last f...
3852	0	alexclegg93 northumbria university 22 and to m...
16737	1	tayedris the world isn t as cruel as you take ...
7444	0	zelakto broadcaster on twitch linux server adm...
14575	1	goddardtara urban studies phd student research...

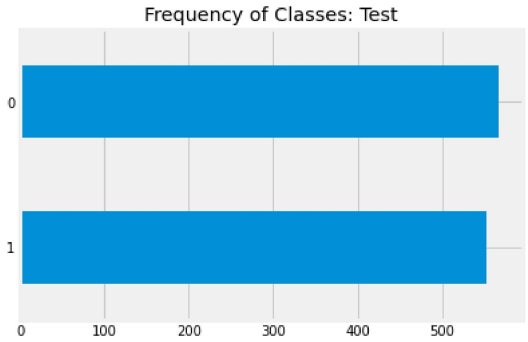
```
# Split the data into features and labels
y_train = train_data["label"]
y_val = val_data["label"]
y_test = test_data["label"]
```

```
# Features
x_train = train_data.drop(["label"], axis=1)
x_val = val_data.drop(["label"], axis=1)
x_test = test_data.drop(["label"], axis=1)
```

```
y_train.value_counts(ascending=True).plot.barh()

plt.title("Frequency of Classes: Validation")
plt.show()
```




```
y_test.value_counts(ascending=True).plot.barh()
plt.title("Frequency of Classes: Test")
plt.show()
```



Machine Learning Approaches

[ ] ↳ *12 cells hidden*

Transformer Model Approach with Bert

[ ] ↳ *17 cells hidden*

Attempting to classify using only the profile description

[ ] ↳ *9 cells hidden*

Augmenting data

[ ] ↳ *14 cells hidden*