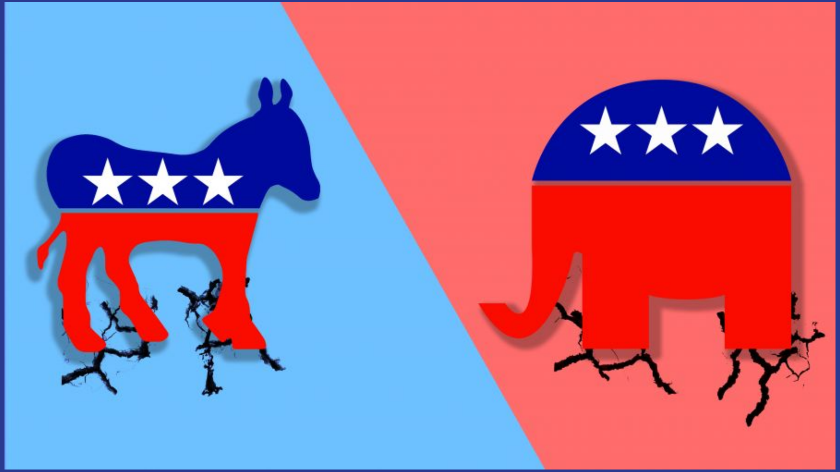
Predicting Political Affiliation using Natural Language Processing



Problem

Potential Stakeholders

- YouTube
- Facebook
- Instagram

Context

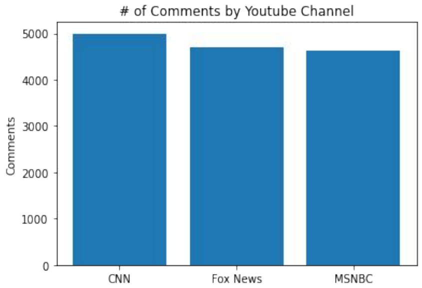
- Free services rely on ad revenue through user engagement.
- These services want to maximize user engagement
- Past user engagements can be used to improve user's experience

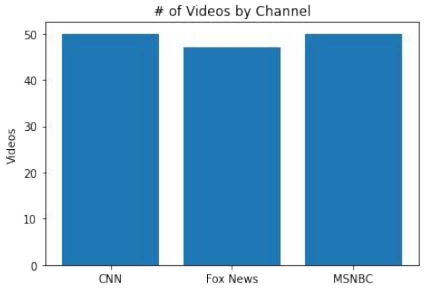
Problem statement

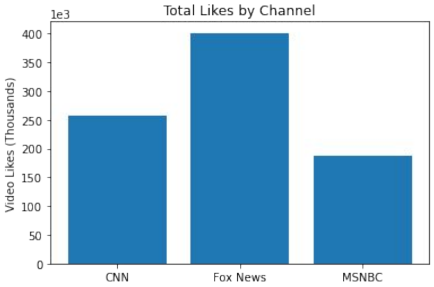
- Can comments be used to predict qualities about users in order to improve user experience and maximize user engagement?

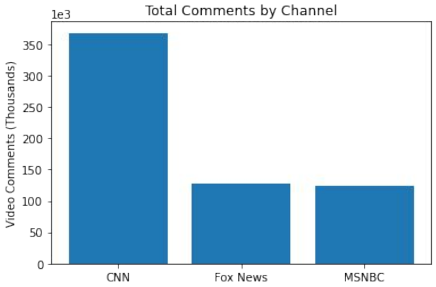
Data Wrangling

- Source: Google API
- Channels: CNN, Fox, MSNBC
- 150 Videos regarding Covid-19
- 100 top comments per video
- Observation Comment
  - Comment Likes
  - Video Views
  - Video Likes
  - Video Comment Count



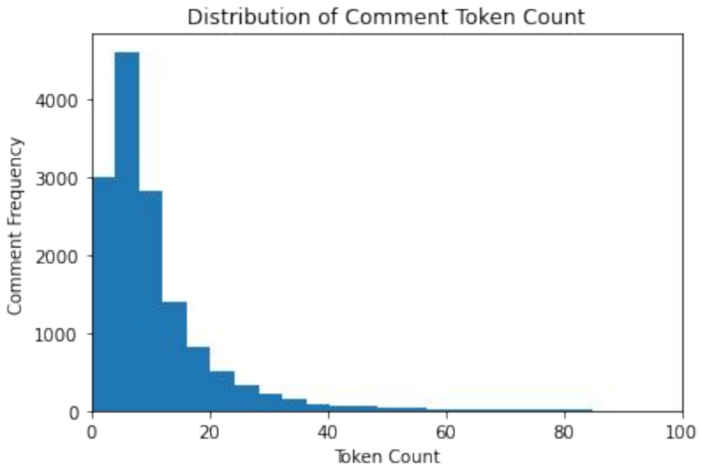


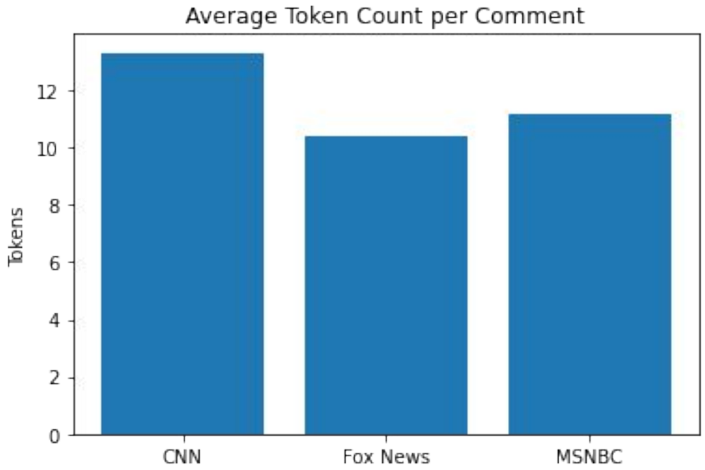




- Final Data: 14,329 comments
  - CNN - 4,998
  - Fox - 4,699
  - MSNBC - 4,632
- 147 Videos
  - CNN - 50
  - Fox - 47
  - MSNBC - 50
- User Engagement
  - Fox News leads video likes
  - CNN leads comment counts

Exploratory Data Analysis

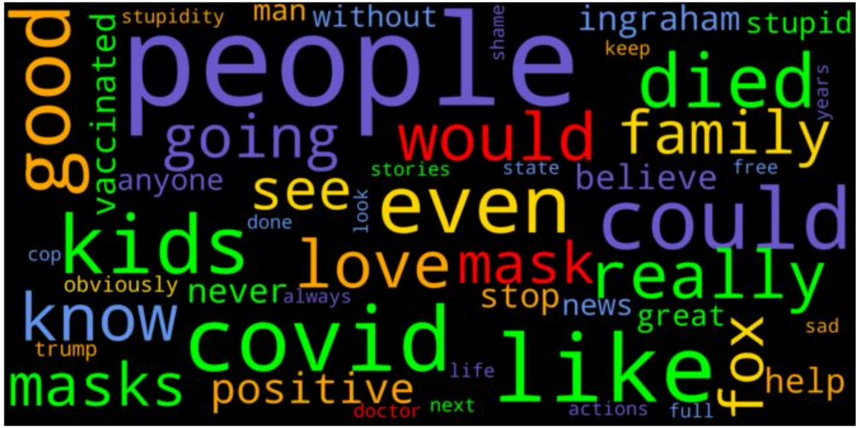




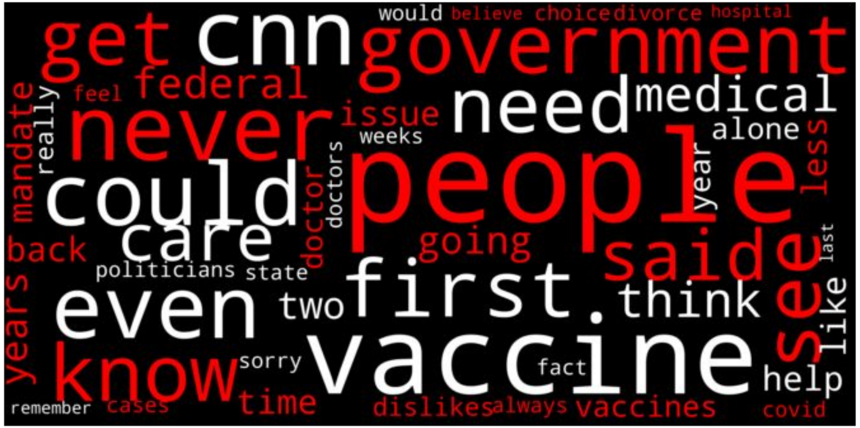
- EDA Preprocessing
  - Lowercase
  - Tokenized
  - Non-alpha characters stripped
  - Stopword removal
- Tokens, Bigrams, Trigrams
- Grouping
  - By News Network
  - By Comment Likes
- Ranked by counts and TF-IDF weights

Counts

Most Frequent MSNBC Tokens (Most Liked Comments)



Most Frequent CNN Tokens (Most Liked Comments)

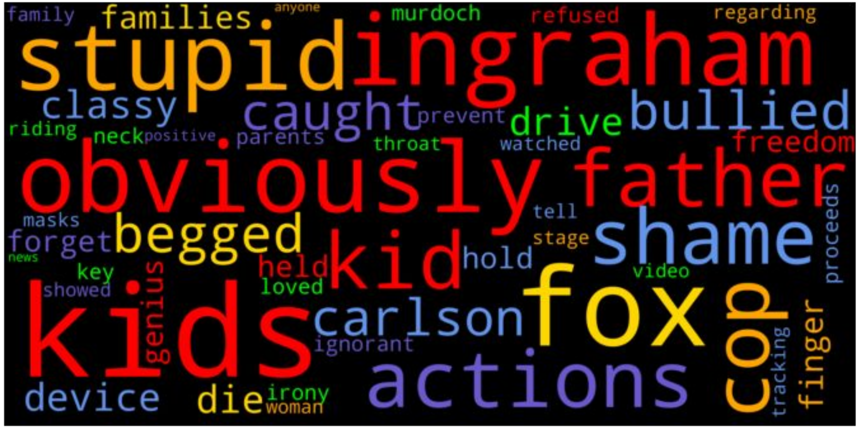


Most Frequent FOX Tokens (Most Liked Comments)

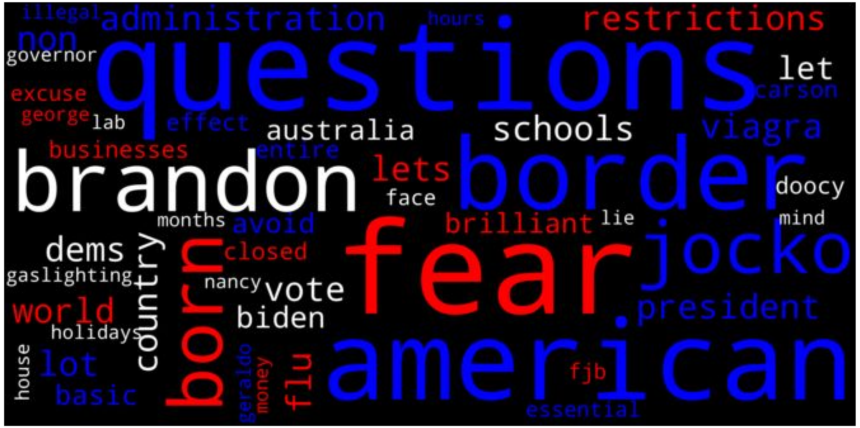
TF-IDF Weights

Top CNN Tokens by TFIDF (Most liked Comments)

Top MSNBC Tokens by TFIDF (Most liked Comments)



Top FOX Tokens by TFIDF (Most liked Comments)



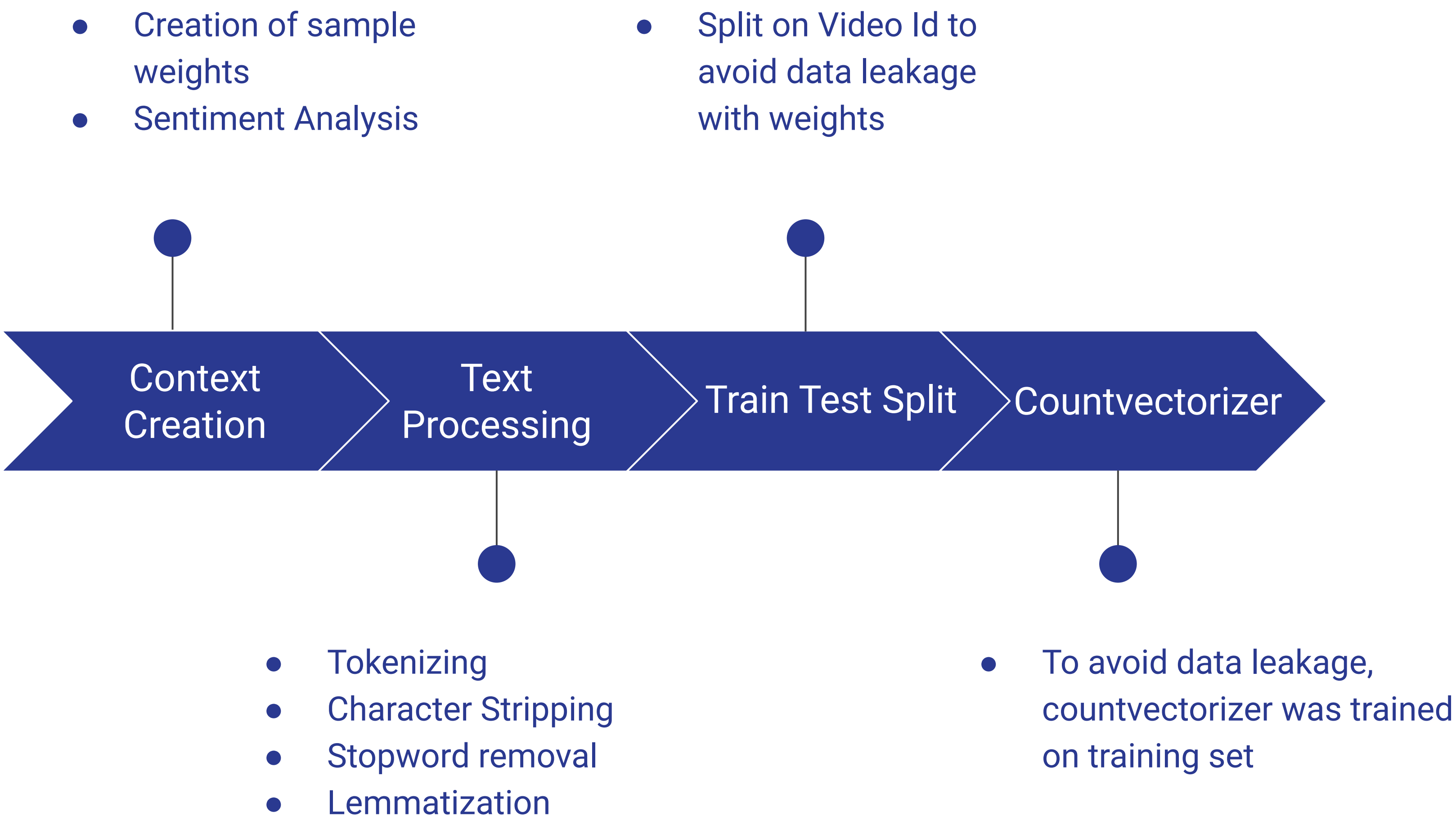
Bi-gram Counts

CNN	FOX	MSNBC
'god bless', 56	'go brandon', 166	'let go', 67
'fully vaccinated', 54	'let go', 163	'go brandon', 59
'many people', 53	'god bless', 78	'get vaccinated', 39
'let go', 53	'gon na', 36	'gon na', 37
'go brandon', 48	'fox news', 29	'fully vaccinated', 32
'gon na', 40	'southern border', 27	'wear mask', 27
'get vaccinated', 39	'peter doocy', 26	'public health', 24
'got covid', 39	'thank god', 25	'health care', 23
'natural immunity', 35	'got covid', 22	'south africa', 23
'two years', 32	'side effects', 21	'fox news', 22

Bi - gram TF-IIDF

CNN	FOX	MSNBC
'lying know', 0.07067932899858004	'ben carson', 0.0887200120593771	'blah blah', 0.05706452815024745
'bari weiss', 0.04573368346966943	'dr oz', 0.07393334338281425	'cbd oil', 0.052309150804393495
'community schools', 0.04157607588151767	'thank tucker', 0.05914667470625141	'diet plan', 0.04755377345853954
'sorry loss', 0.04157607588151767	'dr carson', 0.05421778514739712	'brian williams', 0.042798396112685586
'know know', 0.03836123142780878	'hard evidence', 0.04928889558854284	'hahaha hahaha', 0.042798396112685586

Preprocessing


Creation of sample weights
Sentiment Analysis
Split on Video Id to avoid data leakage with weights
Context Creation
Text Processing
Train Test Split
Countvectorizer
Tokenizing
Character Stripping
Stopword removal
Lemmatization
To avoid data leakage, countvectorizer was trained on training set

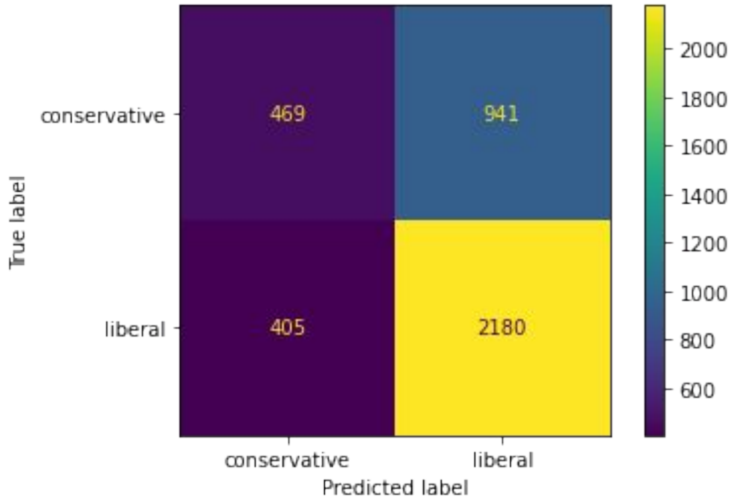
Modeling

Overview

- Logistic Regression
- Random Forest
- Multinomial Naive Bayes
- Support Vector Machines

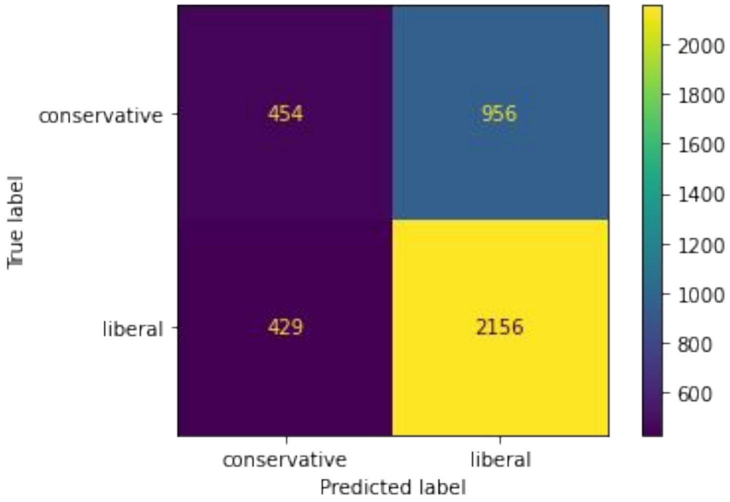
- Each model was trained with 7 different feature sets.
    - Count vectors
    - 2 weighted Count vectors
    - 2 Sentiment polarity features for each weighted set

Logistic Regression + Count Vector



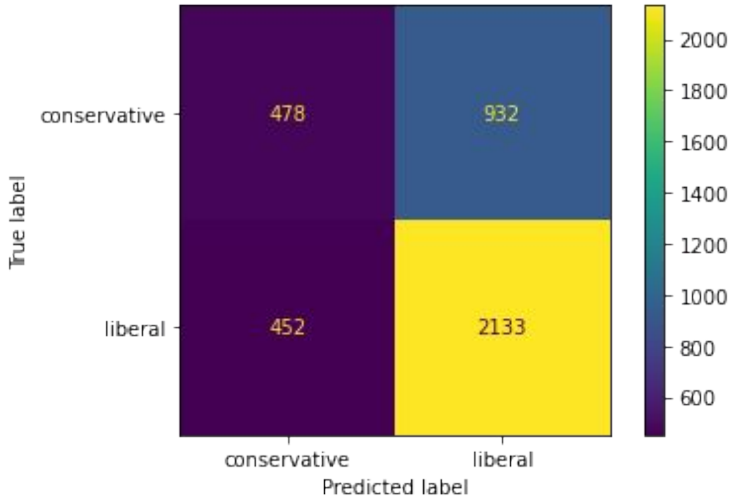
	precision	recall	f1-score	support
0	0.54	0.33	0.41	1410
1	0.70	0.84	0.76	2585
accuracy			0.66	3995
macro avg	0.62	0.59	0.59	3995
weighted avg	0.64	0.66	0.64	3995

Random Forest + Weighted(1) Count Vector



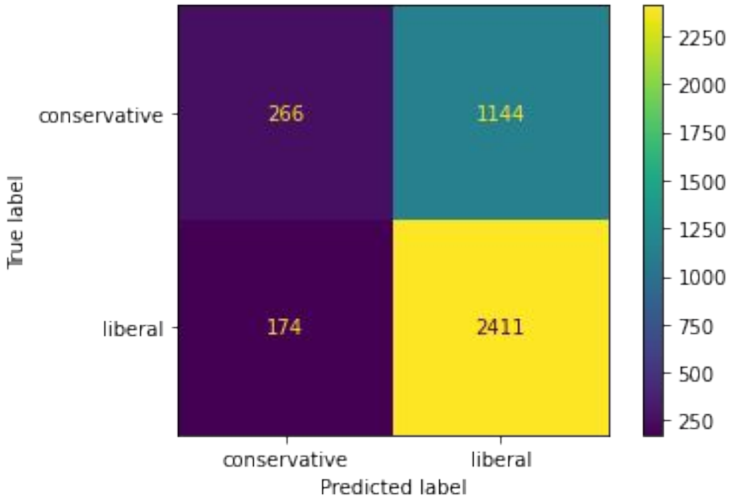
	precision	recall	f1-score	support
0	0.51	0.32	0.40	1410
1	0.69	0.83	0.76	2585
accuracy			0.65	3995
macro avg	0.60	0.58	0.58	3995
weighted avg	0.63	0.65	0.63	3995

Multinomial Naive Bayes + Weighted(1) Count Vector with Sentiment Polarity



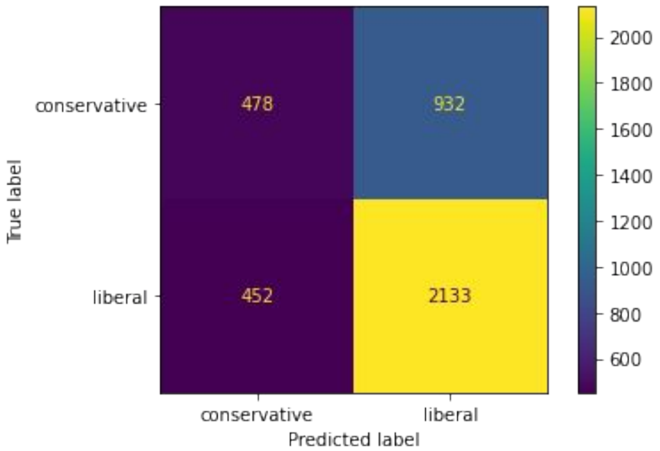
	precision	recall	f1-score	support
0	0.51	0.34	0.41	1410
1	0.70	0.83	0.76	2585
accuracy			0.65	3995
macro avg	0.60	0.58	0.58	3995
weighted avg	0.63	0.65	0.63	3995

Support Vector Machines - Weighted(2) Count Vector



	precision	recall	f1-score	support
0	0.60	0.19	0.29	1410
1	0.68	0.93	0.79	2585
accuracy			0.67	3995
macro avg	0.64	0.56	0.54	3995
weighted avg	0.65	0.67	0.61	3995

Final Model Choice



	precision	recall	f1-score	support
0	0.51	0.34	0.41	1410
1	0.70	0.83	0.76	2585
accuracy			0.65	3995
macro avg	0.60	0.58	0.58	3995
weighted avg	0.63	0.65	0.63	3995

- **Multinomial Naive Bayes**
  - **Weighted Comment likes : Views**
  - **Textblob sentiment feature**
- All the models had similar performance, with accuracies between 65 - 67%. The chosen model, however, had the higher F1 scores despite having a lower accuracy. Overall this model was better at identifying the minority class.

Considerations for Future

- Gather larger data set
  - Expand Video topic query
- Gather richer data set
  - Include additional news networks