Twitter Sentiment Analysis

Shahbaz Masood
Springboard Capstone Project 1

Problem & Motivation

- Sentiment analysis is a multidisciplinary necessity.
  - Online Commerce
  - Voice of the Market (VOM)
  - Voice of the Customer (VOC)
  - Brand Reputation Management
  - Government

- Extract sentiment from 100,000 tweets.
- Dataset acquired online

Data

- Publicly available.
- Data
  - 100,000 tweets
  - Each marked with 0,1 i.e negative positive sentiment

sentiment	tweet
1	@awaketoday awww, thanks
1	@2NiteBoy damn it xD hurry up
1	@celiabb 8 but the next one up would be better
1	@cleeobea drink lots of water- they always help my headaches not seem so intense! and then go to bed!
1	@biggsjm Is it a family tradition?
0	@CrazyBallerina can't. my sal wont be in until next week
0	#followfriday @SusieGennoe - she's only got 35 followers and that makes me so sad
1	@612brisbane try following my public account instead of @rocketpilot
0	@BarackObama don't act like you didn't smoke in your hayday, buster!! Don't take my flavors away
0	I got scolded for not waiting and spending MORE to find my perfect storage solution... saddies Guess I should have *hangs head*

Data Preprocessing

The tweets were processed according to the following and in the same order.

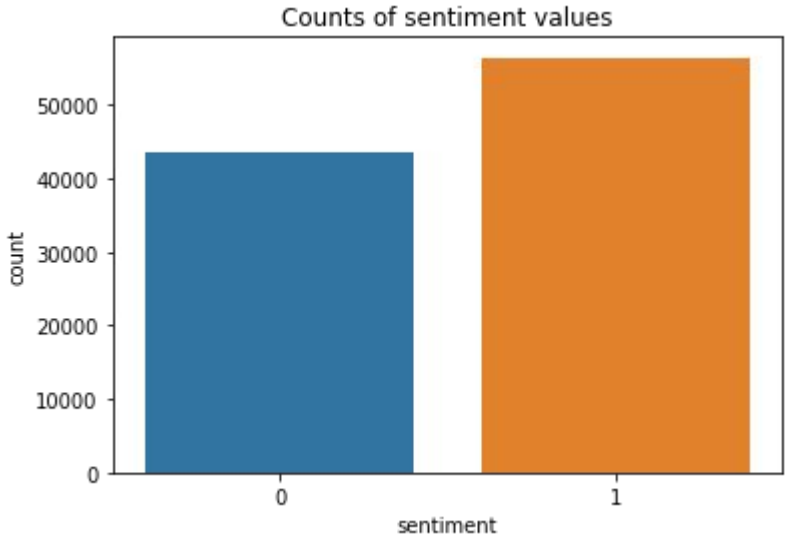
1. Tags removal (i.e '@')
2. Lowercasing
3. Numbers removal
4. HTTP links removal
5. Emojis were processed. Emojis were labelled EMO_POS or EMO_NEG
6. Punctuation removal
7. Removed extra white spaces
8. Words which did not consist of alphabets were removed
9. Stop words were removed (These words add no value to the sentiment of the tweet e.g The, He etc)
10. Character repetitions were removed e.g funnny was changed to funny
11. Words were lemmatized to bring to their basic form e.g adventurous changed to adventure

EDA

Distribution of Sentiment

Roughly 56% tweets are marked positive and 44% as negative

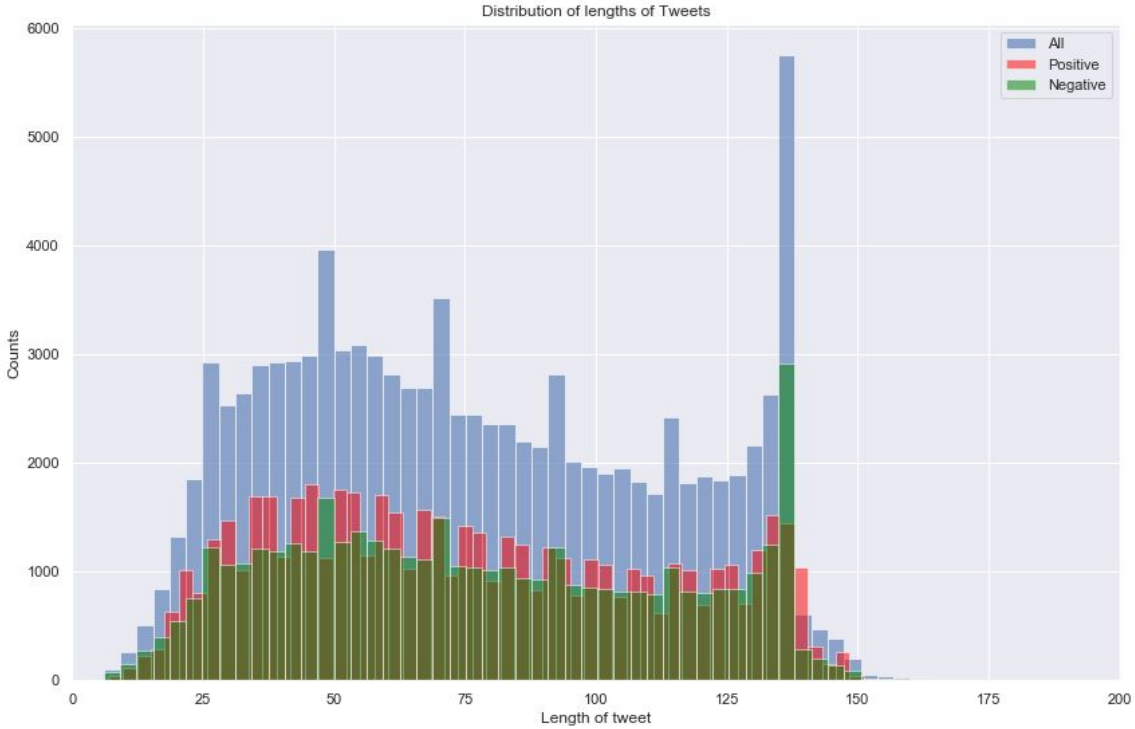
Low class imbalance



EDA

Distribution of length of tweets

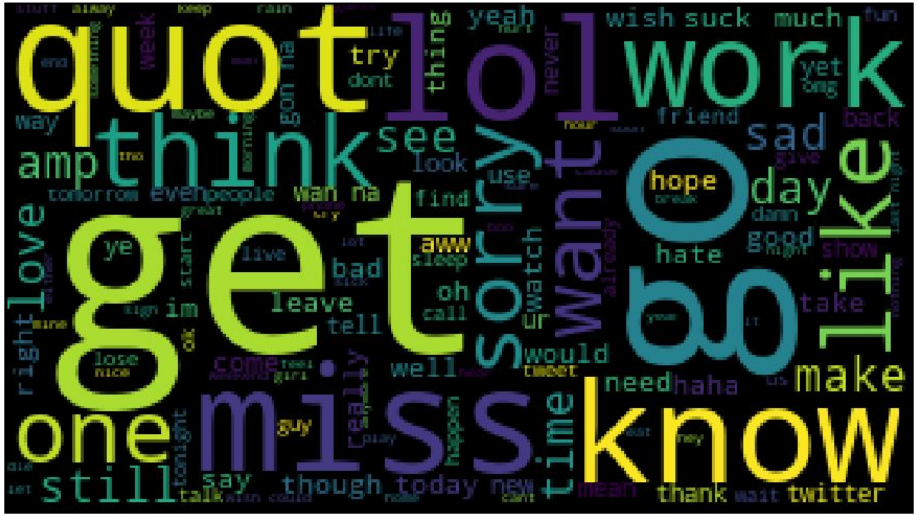
Equal for both sentiments

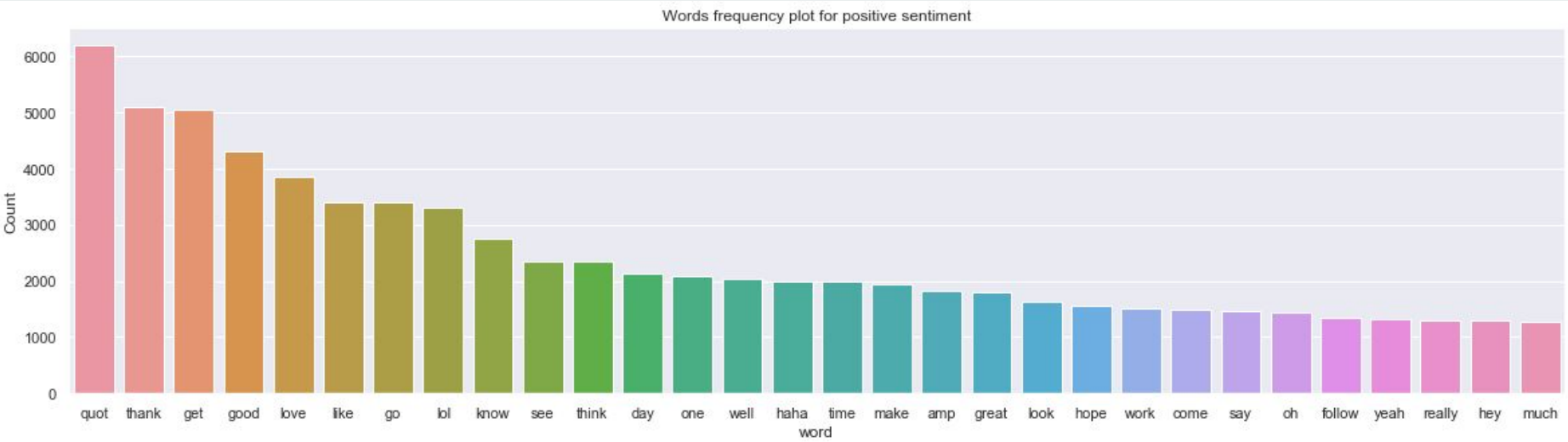


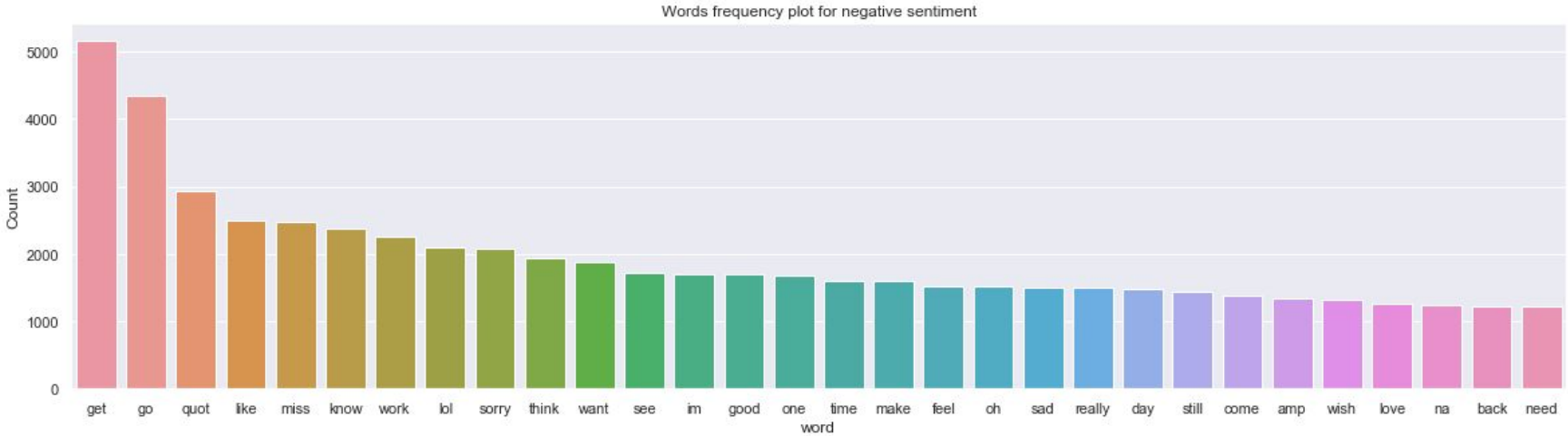
Wordclouds and Countplots

Positive tweets wordcloud

Negative tweets wordcloud




Words frequency plot for positive sentiment
Count
0
1000
2000
3000
4000
5000
6000
quot thank get good love like go lol know see think day one well haha time make amp great look hope work come say oh follow yeah really hey much
word


Words frequency plot for negative sentiment
Count
0
1000
2000
3000
4000
5000
get go quot like miss know work lol sorry think want see im good one time make feel oh sad really day still come amp wish love na back need
word

Initial problems with the data

Upon visual inspection it was noticed that some tweets are:

1. Marked incorrectly e.g. *smoke a little for 420 and chill out!*
2. Had no sentiment but a sentiment is assigned e.g. *called at the Farmers Market in the rain this morning, didn't buy coffee, sorry, I'm a tea drinker really!*
3. Some tweets had multiple sentiments i.e. one sentiment could not be assigned in entirety. E.g. *'i wanted to win but it's allll good lol'*

Modeling Overview

- 3 NLP methods
  - Bag of words
  - TF-IDF
  - Word Embeddings
- 5 ML models
  - Logistic Regression
  - SVM
  - XGBoost
  - NaiveBayes
  - DecisionTree

Data split scheme

	All data	Training data	Cross Validation data	Test data
Percentage	100%	72%	8%	20%
Tweets	100,000	72,000	8,000	20,000

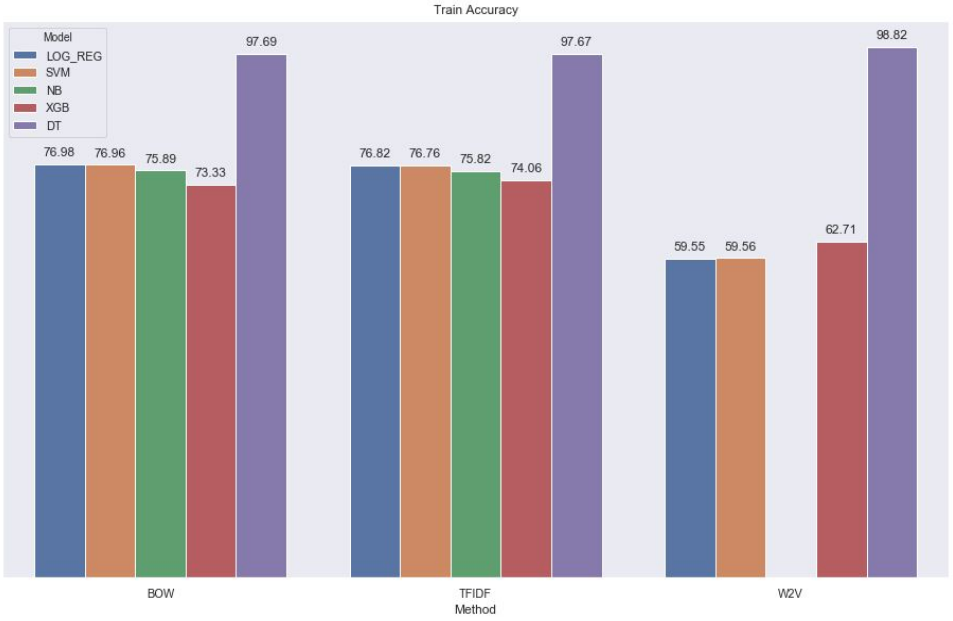
- 3-Fold cross validation
- Hyperparameter tuning
- Homogeneity maintained to ensure comparable results

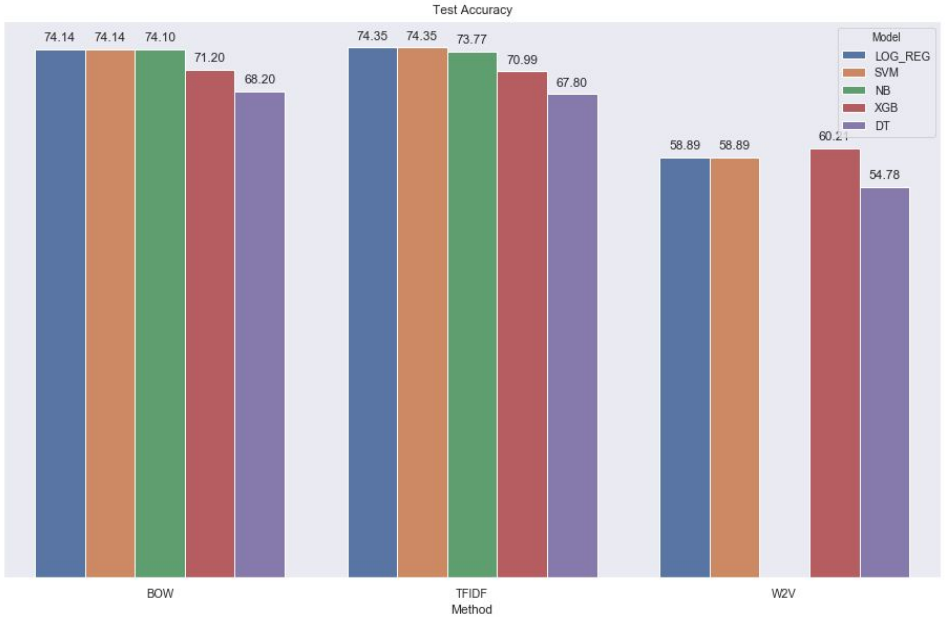
Results

TFIDF - Logistic Regression wins all metrics

Method	Model	TrainAcc	CVAcc	TestAcc	TestF1	TestRecall	TestPrecision	TestROC_AUC
BOW	LOG_REG	0.769830	0.738096	0.741374	0.782981	0.829481	0.741418	0.815214
	SVM	0.769580	0.736983	0.741374	0.782981	0.829481	0.741418	0.815214
	NB	0.758942	0.737095	0.741024	0.776217	0.798542	0.755107	0.812639
	XGB	0.733270	0.715168	0.712021	0.775714	0.885402	0.690207	0.788394
	DT	0.976916	0.668888	0.682018	0.721792	0.733375	0.710569	0.682222
TFIDF	LOG_REG	0.768249	0.739808	0.743524	0.783303	0.824147	0.746317	0.821860
	SVM	0.767643	0.739871	0.743524	0.783303	0.824147	0.746317	0.821860
	NB	0.758217	0.733157	0.737674	0.783277	0.842817	0.731594	0.816258
	XGB	0.740615	0.715230	0.709871	0.773819	0.882379	0.689045	0.787071
	DT	0.976722	0.671601	0.678018	0.719617	0.734619	0.705215	0.671574
W2V	LOG_REG	0.595523	0.593554	0.588909	0.701218	0.857664	0.593041	0.601333
	SVM	0.595561	0.592692	0.588909	0.701218	0.857664	0.593041	0.601333
	XGB	0.627121	0.609594	0.602110	0.702397	0.834815	0.606237	0.632348
	DT	0.988205	0.551437	0.547805	0.598927	0.600284	0.597575	0.535828

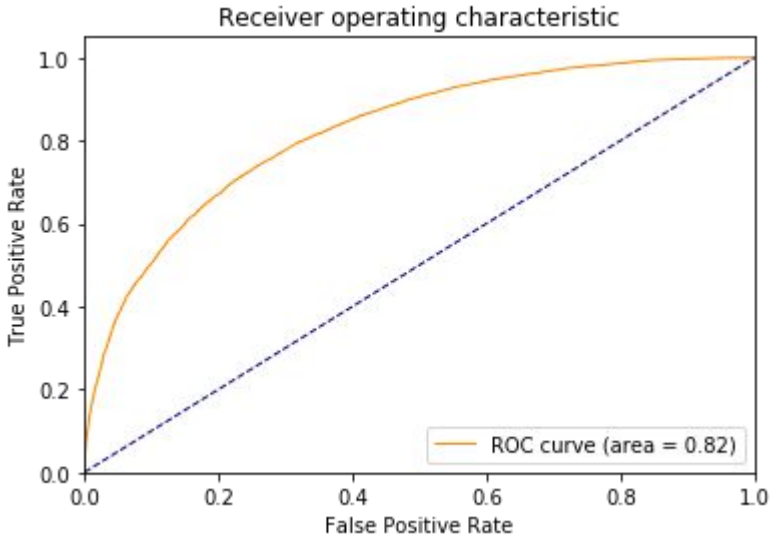
Train/Test Accuracies





Class-wise metrics and ROC

	Negative	Positive	micro avg	macro avg	weighted avg
f1-score	0.705046	0.796944	0.759474	0.750995	0.756934
precision	0.756302	0.761418	0.759474	0.758860	0.759191
recall	0.660296	0.835946	0.759474	0.748121	0.759474



Further Improvements

1. Use more features to train the models for TFIDF.
2. Use a sophisticated model like deep learning where the models are trained with the context of a window of words.
3. BOW and TFIDF can be used with 2-grams or 3-grams
4. Training dataset can be refined to be more accurate.
5. Exclamation marks could be handled specially during the preprocessing of the tweets.