Binary Classification

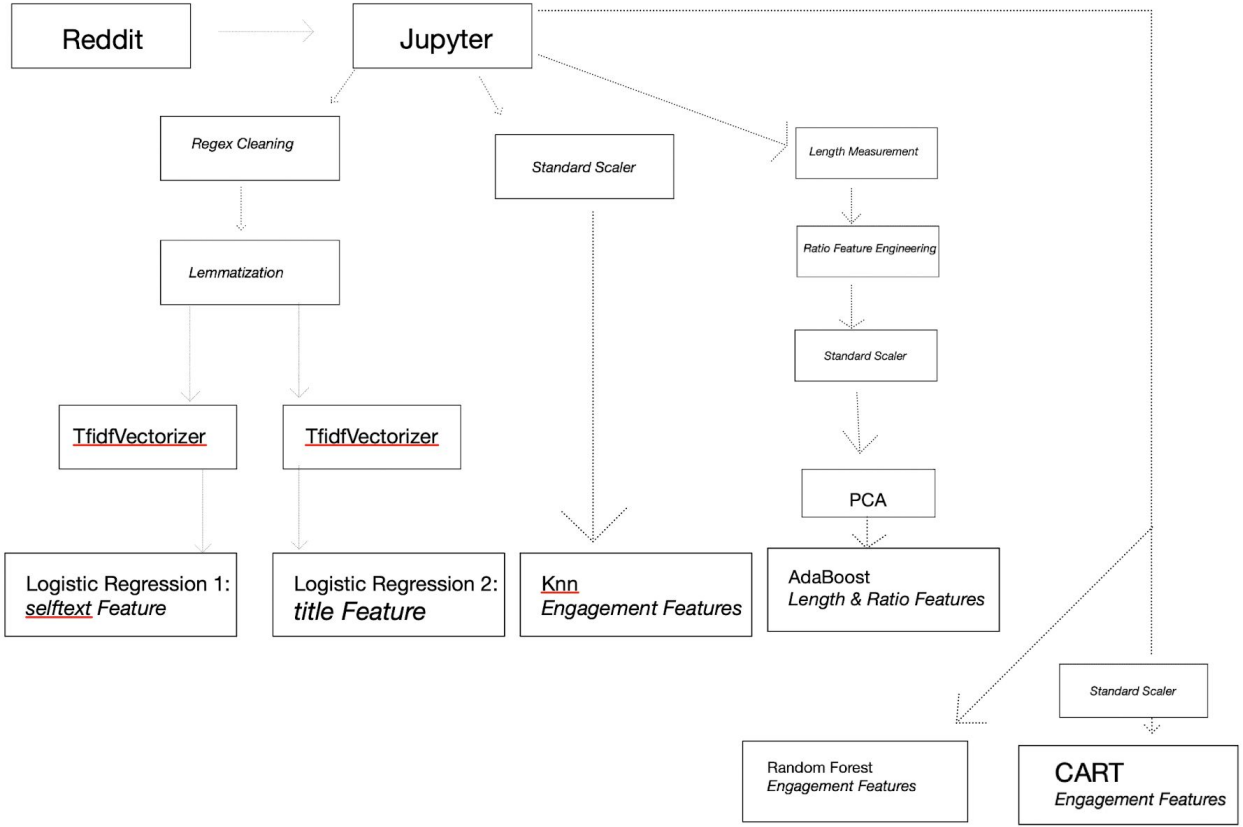
r_investing and r_stocks

Problem Statement

Person “X" works at a fintech startup and wants to create a sentiment indicator of a public company. Person “X" wants to scrape the comments section of a widely-seen submission on the "hot" subsection of either r_invetsing or r_stocks, but no such submission exists. Person “X" must create that submission, but can't figure out which subreddit his submission is a more natural fit for.

Build a model that will tell Person “X" which subreddit he should post to. Assume Person “X" firm has other reddit accounts and can boost their submission's comments, likes, and awards.

Model Flow


Reddit
Jupyter
Regex Cleaning
Standard Scaler
Length Measurement
Lemmatization
Ratio Feature Engineering
Standard Scaler
TfidfVectorizer
TfidfVectorizer
PCA
Logistic Regression 1:
selftext Feature
Logistic Regression 2:
title Feature
Knn
Engagement Features
AdaBoost
Length & Ratio Features
Standard Scaler
Random Forest
Engagement Features
CART
Engagement Features

Features Mapping

- Diversifying the feature space for a stacked model increases model robustness

Text Features	Engagement Features	Length and Ratio Features
title text (title) *submission text (selftext)*	*number of comments (num comments)* *up votes* *total_awards_received*	*characters* *words* *sentences* *characters/sentences* *characters/words* *words/sentences*

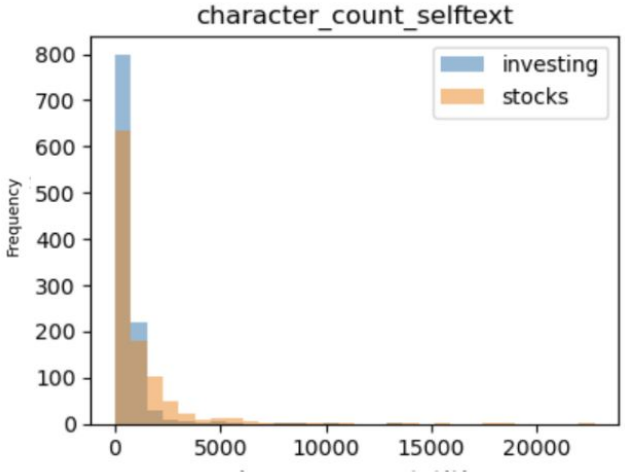
Text Model Results

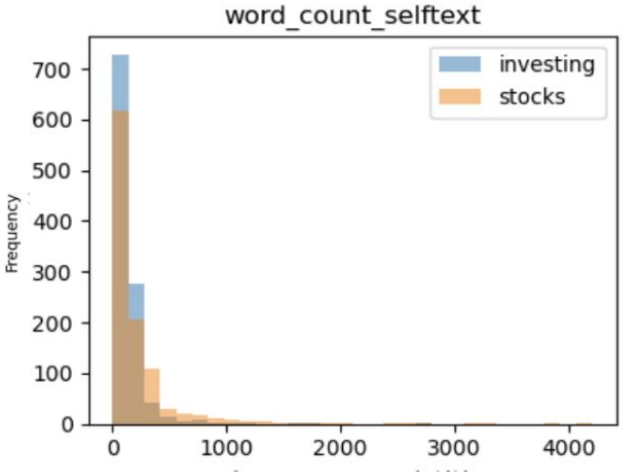
- Title text alone has comparative predictive power to entire submission text.
- Cross val mean and cross val std measures indicate less variance error

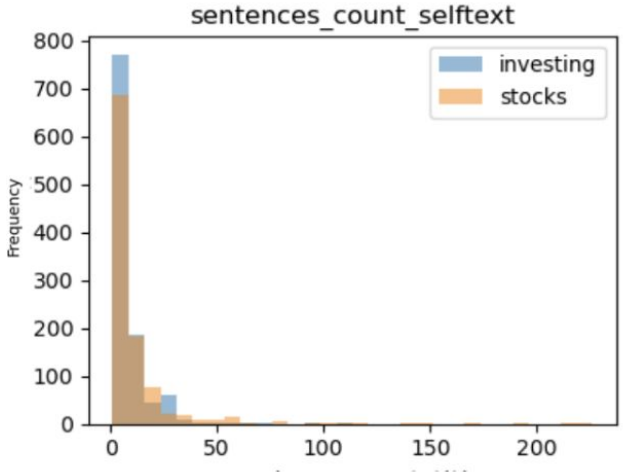
Metrics of Base Models	Logistic selftext	Logistic title
train accuracy	0.900	0.901
validation accuracy	0.777	0.700
cross validation standard deviation	0.0196	0.0157
cross validation standard deviation	0.776	0.716

Length Features

- r_stocks has more sentences, words, and characters per submission than r_investing
- Crossover in distribution leant itself to AdaBoost Decision Trees, Decision Trees, and Random Forest

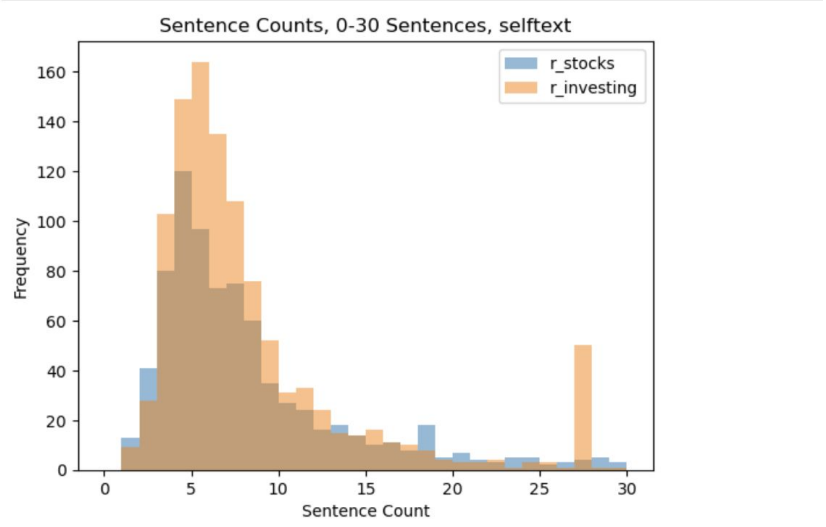


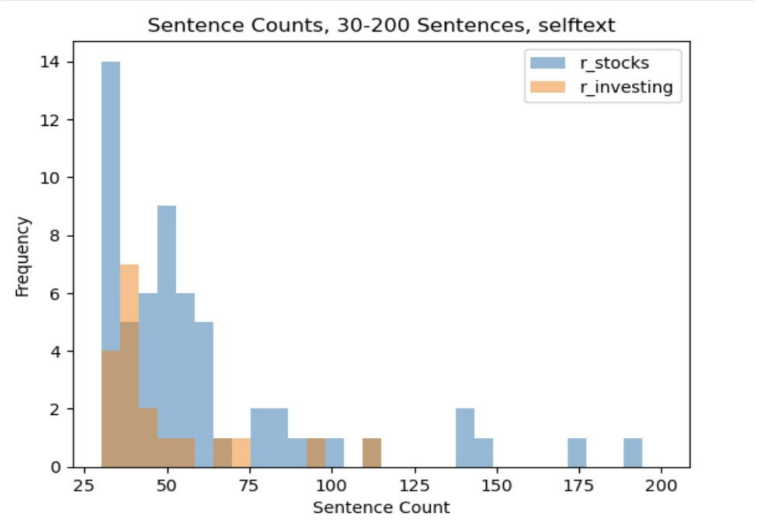




Length Features

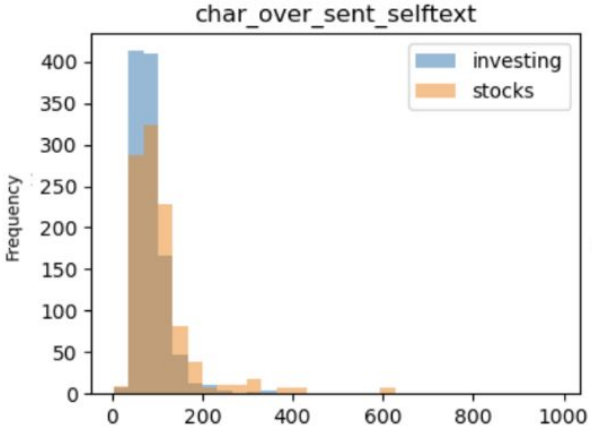
- Magnified crossover of distribution

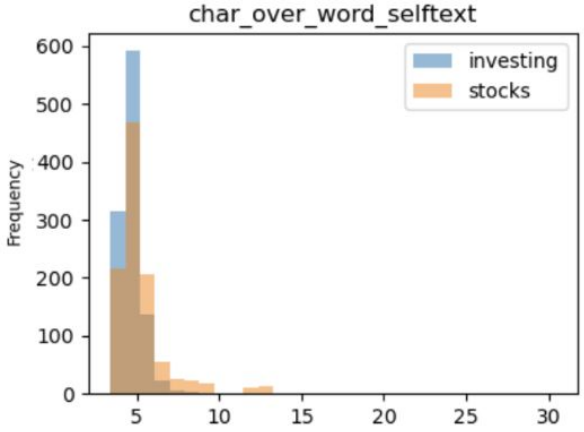


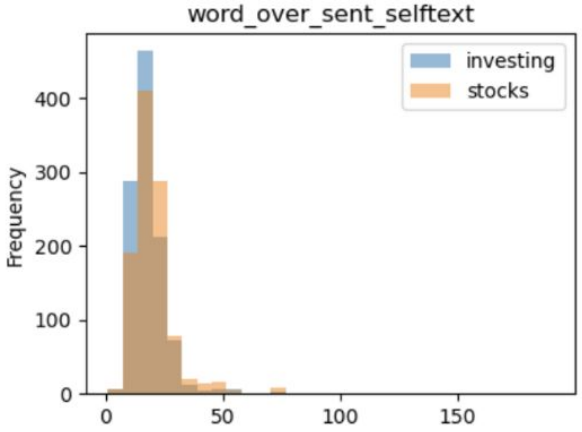


Ratio Features

- High multicollinearity lead to PCA
- Scale differences lead to standard scaler
- R_stocks's submissions have larger words, more words in each sentence and more characters in each sentence

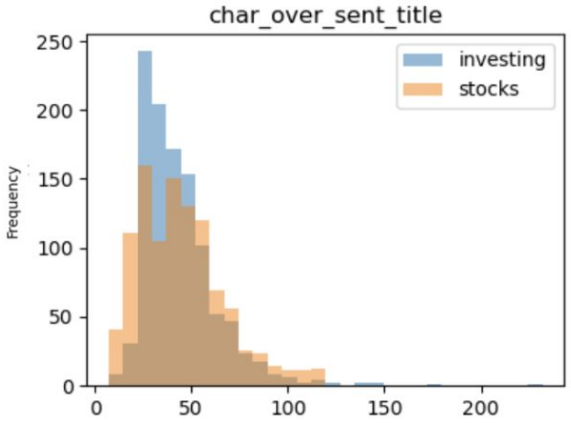


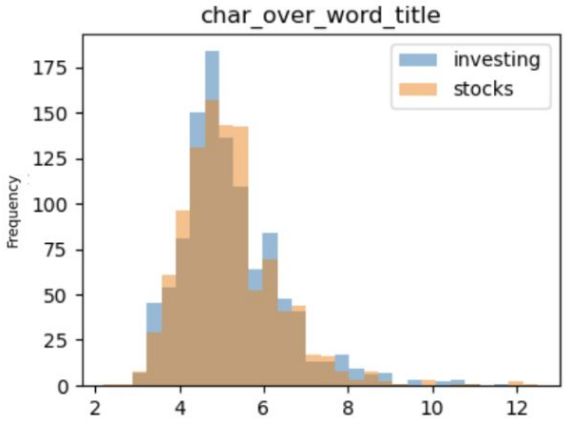


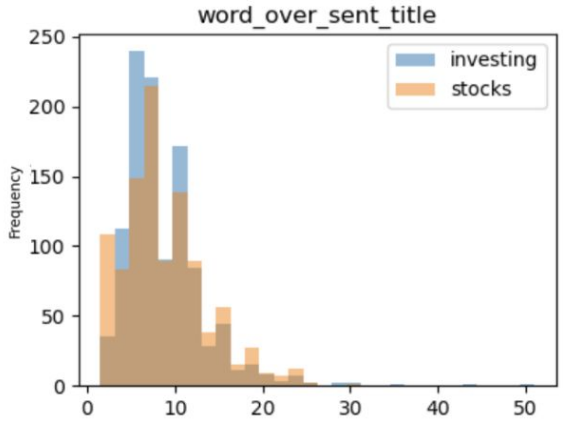


Ratio Features

- Lower kurtosis for r_stock's title ratios for sentence and word structures







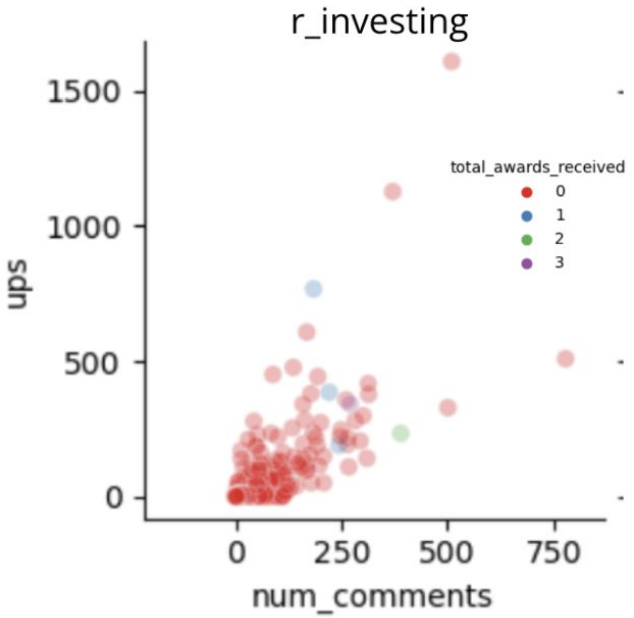
Length and Ratio Features

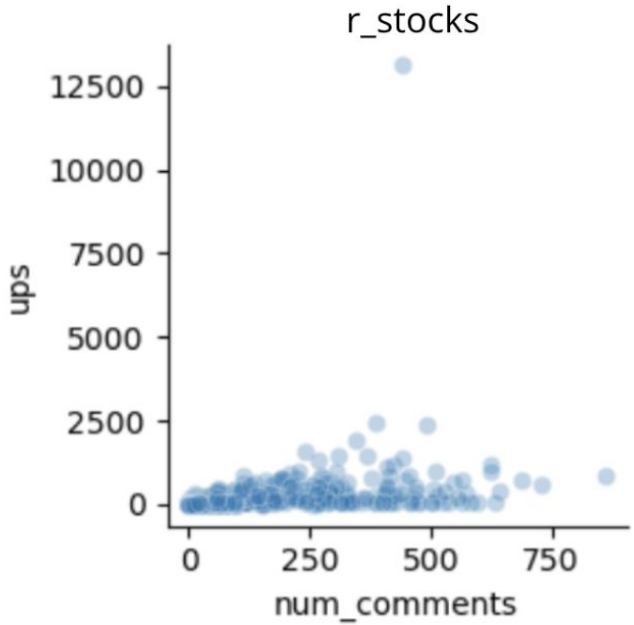
- Moderately overfit from train accuracy to validation accuracy
- Cross val mean and cross val std measures indicate less variance error

Metrics of Base Models	AdaBoost
train accuracy	0.730
validation accuracy	0.643
cross validation standard deviation	0.019
cross validation mean	0.665

Engagement Features- Arriving at Knn

- r_stocks no relationship
- r_investing, positive relationship
- Thought difference in relationship would be captured by RF, CART
- Low observations of 1,2,3 total_awards_received probably made Knn the best





Engagement Models

- Random Forest accuracy was akin to guessing
- Decision tree was extremely overfit
- Mixed results on knn fitness level, with reassuring validation accuracy and cross validation standard deviation, but problematic cross validation mean
- Engagement Features probably have limited predictive power

Metrics of Engagement Models	Knn	RF	Decision Tree
train accuracy	0.68	0.504	0.741
validation accuracy	0.640	0.500	0.579
cross validation standard deviation	0.006	0.004	0.007
cross validation mean	0.601	0.646	0.601

Next Steps

1) Finds counts and distributions of grammar in the text that is reflective of submission sentence complexity such as:

a) Coordinating conjunctions ("for, "and", "nor", "but", "or" "yet"), followed by a comma allows you to join two independent clauses together, creating compound sentences

b) Count the number of commas not associated with a coordinating conjunctions to count the number of complex sentences

c) Count the number of sentences with high word counts and a coordinating conjunction but no comma in order to try to gauge how many submissions might be forgetting to use commas

d) Count the number of sentences with very high comma counts to gauge how many submissions might be misusing commas

e) Combine the submissions with forgotten commas with the submissions with misused commas to see if user grammar (or incorrect grammar) is itself a feature

2) Apply Random Forest to the length and ratio features. RF good with multicollinearity and unscaled data. Also good at incorporating uneven predictive power of different features in the feature set.