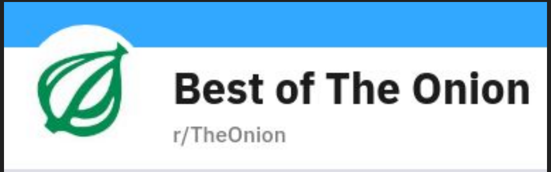
Reddit Fake News Classification

Project 3 -Web APIs & NLP

Background



Best of The Onion

r/TheOnion

- Fake news, misinformation, sensationalism, yellow journalism etc. has been around as long as print media has been in existence
- Common uses range from telling lies to sell more than competitors all the way to using it to justify atrocities against other people.
- Since the advent of the internet it has become easier to generate and distribute misinformation
- Reddit is a popular site where users share content including news articles in some communities or subreddits.

Problem Statement

To combat misinformation Reddit would like to label submissions of news articles as either fake news or real new based on the text in the post. This would increase user trust on the site and ultimately profits. We will use r/theonion and r/nottheonion subreddits as proxies to test if it is possible to get accurate classifications before employing the system site wide.

The Data

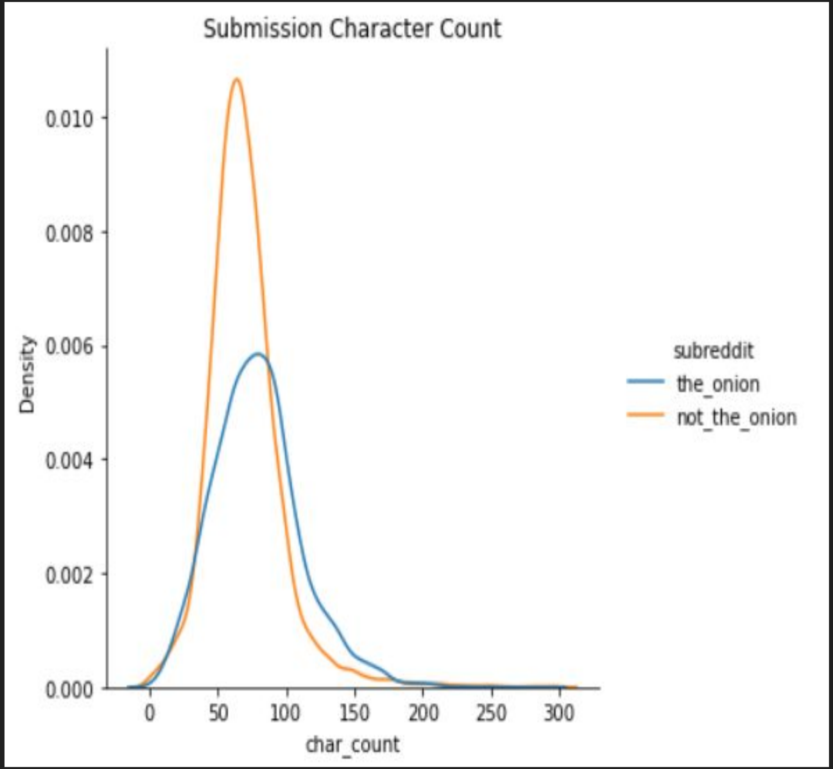
- Gathered using PushShift Reddit API
- 18451 posts from August 2,2021 - May 21, 2013
  - 8486 submissions from r/theonion - 55%
  - 9965 from r/nottheonion - 45%
- About 30 posts per month
- Focusing on title of the post

(Minimal) Data Cleaning

- Drop Duplicates
- Check for non-news submissions by moderators
- Remove URLs to avoid modeling based on urls
  - .com/ http/https/www
- Remove Reddit Markdown
  - Bold and Italic with “*” ->> Censored words
  - Bold and Italic with “_”
- Minimal cleaning to simulate real world submissions
  - Other subreddits may/maynot have rules about what is allowed to be posted

Other Notable Features (not modeled)

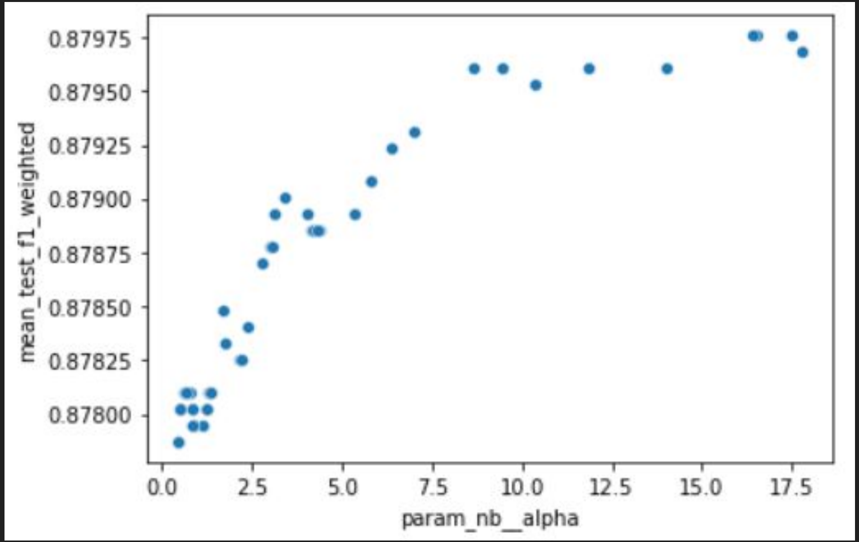
- Number of non-alphanumeric characters
  - No difference
- Number of characters
  - r/thonion submissions on average 8 characters longer
- Number of words
  - r/onion on average 1.5 words longer
- r/nottheonion slightly more negative sentiment but that could be by design

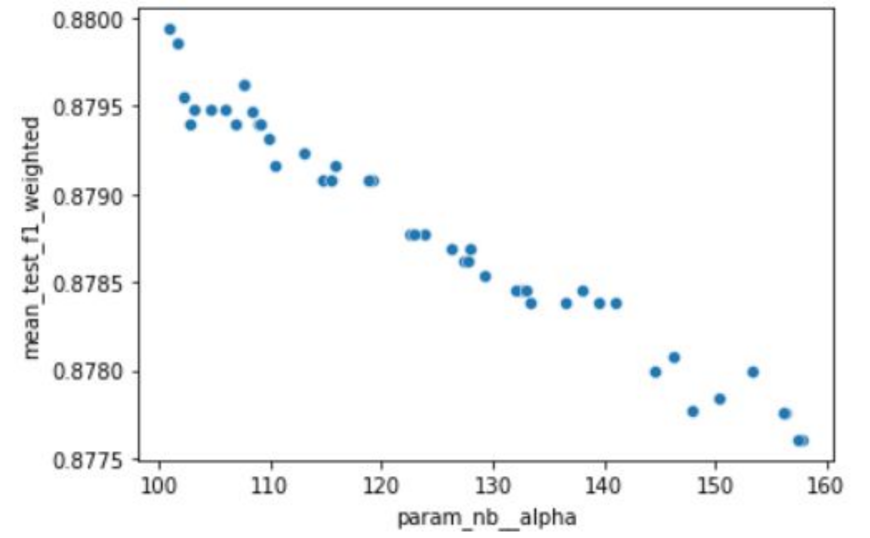


Modeling Methodology

- Make baseline model for comparison using Dummy Classifier
- Use default hyperparameters for CountVectorizer and Multinomial Naive Bayes model
- Tune relevant hyperparameters for best score on metrics using RandomizedSearchCV and GridSearchCV
- Metrics:
  - Accuracy score
  - Weighted F1

Hyperparameter Tuning





Final Hyperparameters

Transformer/Estimator	Hyperparameters	Value
CountVectorizer	Lowercase	False
CountVectorizer	stop_words	None
CountVectorizer	ngram_range(1,2)	(1,2)
CountVectorizer	max_df	0.6
CountVectorizer	min_df	0.003
Multinomial Naive Bayes	alpha	103

Model Metrics

Baseline Model	Accuracy Score	Weighted F1 Score
Training Set	0.4995	0.5008
Test Set	0.5121	0.5134

Final Model	Accuracy Score	Weighted F1 Score
Training Set	0.8800	0.8802
Test Set	0.8911	0.8913

Top 10 Features

r/nottheonion	r/theonion
ad_1	To
arrested after	Of
charged	The
court	In
game	For
jail	And
kill	On
make	With
UK	You
accused of	This

Decision Tree

Baseline Model	Accuracy Score	Weighted F1 Score
Training Set	0.9998	0.9998
Test Set	0.6935	0.6937

Very overfit consider bagging or random forest for better results

Recommendations

- The significant improvement from the baseline model suggests that it is possible to label submissions as “fake news” or “real news” based on the title.
- Recommend training model on more subreddits
- Recommend testing other classification models
- Limitations of current model:
  - Not much data cleaning
  - Basic transformer to vectorize text (TFIDF)
  - Only two subreddits used
  - Only used text and not other information about posts