Determining User Engagement on a Reddit post by number of comments

Vilo Avila

The problem at hand

Problem Statement

For this particular project we are provided with the following problem statement:

What characteristics of a post on Reddit are most predictive of the overall interaction on a thread (as measured by number of comments)?

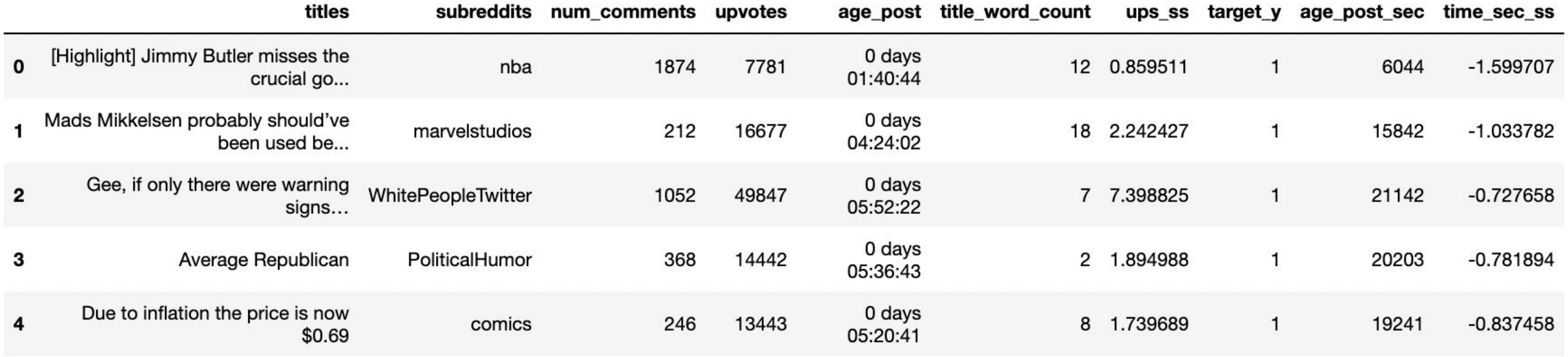
Personal Overall Goal

I will also measure success by choosing the best classification model that will predict classes consistently over the class majority baseline, based on classification metrics such as cross val score, confidence intervals, and confusion matrix metrics.

Python Reddit API Wrapper (PRAW)

- BeautifulSoup and Selenium were first attempted without much success
- Our data points were pulled from reddit over the course of a few days.
- Between 4000-7000 posts were pulled per day as to get a good variety of posts and minimize duplicates
- All data pulled was saved to .csv files for later use
- In total, 31,488 posts along with their corresponding data were pulled from Reddit

Data Snapshot

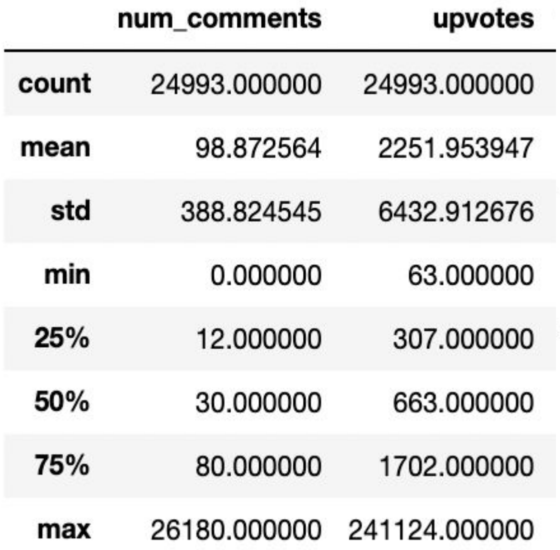


	titles	subreddits	num_comments	upvotes	age_post	title_word_count	ups_ss	target_y	age_post_sec	time_sec_ss
0	[Highlight] Jimmy Butler misses the crucial go...	nba	1874	7781	0 days 01:40:44	12	0.859511	1	6044	-1.599707
1	Mads Mikkelsen probably should've been used be...	marvelstudios	212	16677	0 days 04:24:02	18	2.242427	1	15842	-1.033782
2	Gee, if only there were warning signs…	WhitePeopleTwitter	1052	49847	0 days 05:52:22	7	7.398825	1	21142	-0.727658
3	Average Republican	PoliticalHumor	368	14442	0 days 05:36:43	2	1.894988	1	20203	-0.781894
4	Due to inflation the price is now $0.69	comics	246	13443	0 days 05:20:41	8	1.739689	1	19241	-0.837458

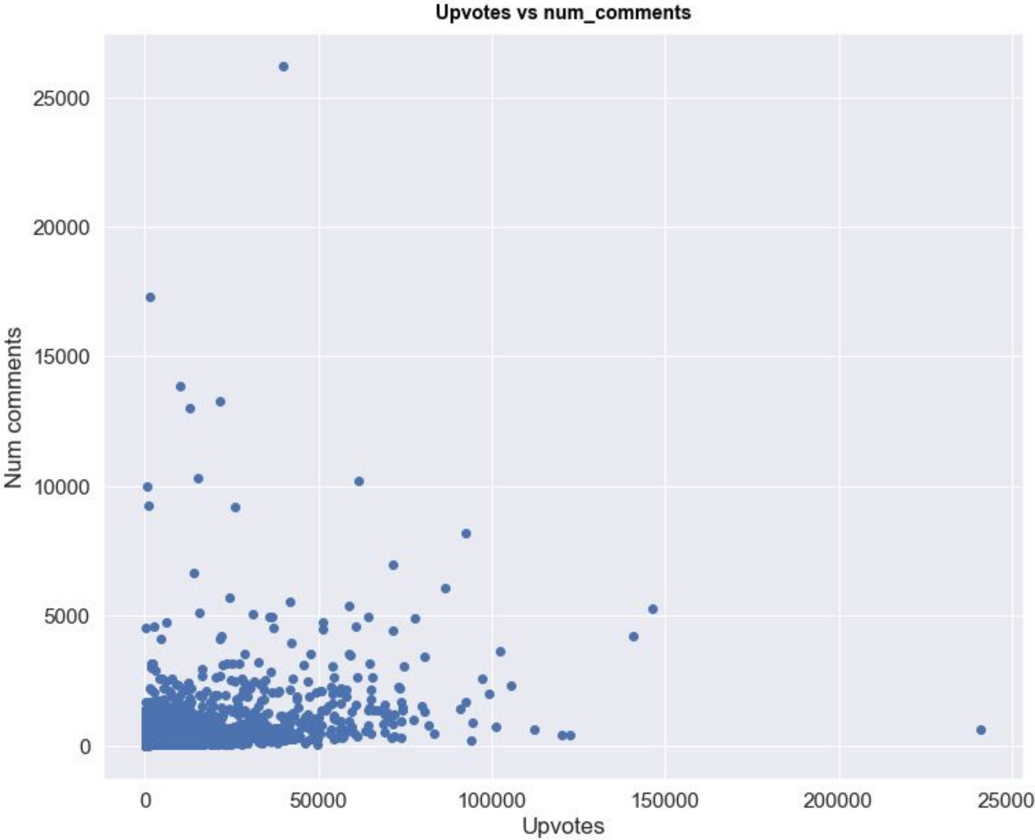
Data Cleaning

Some methods checked for and cleaned.

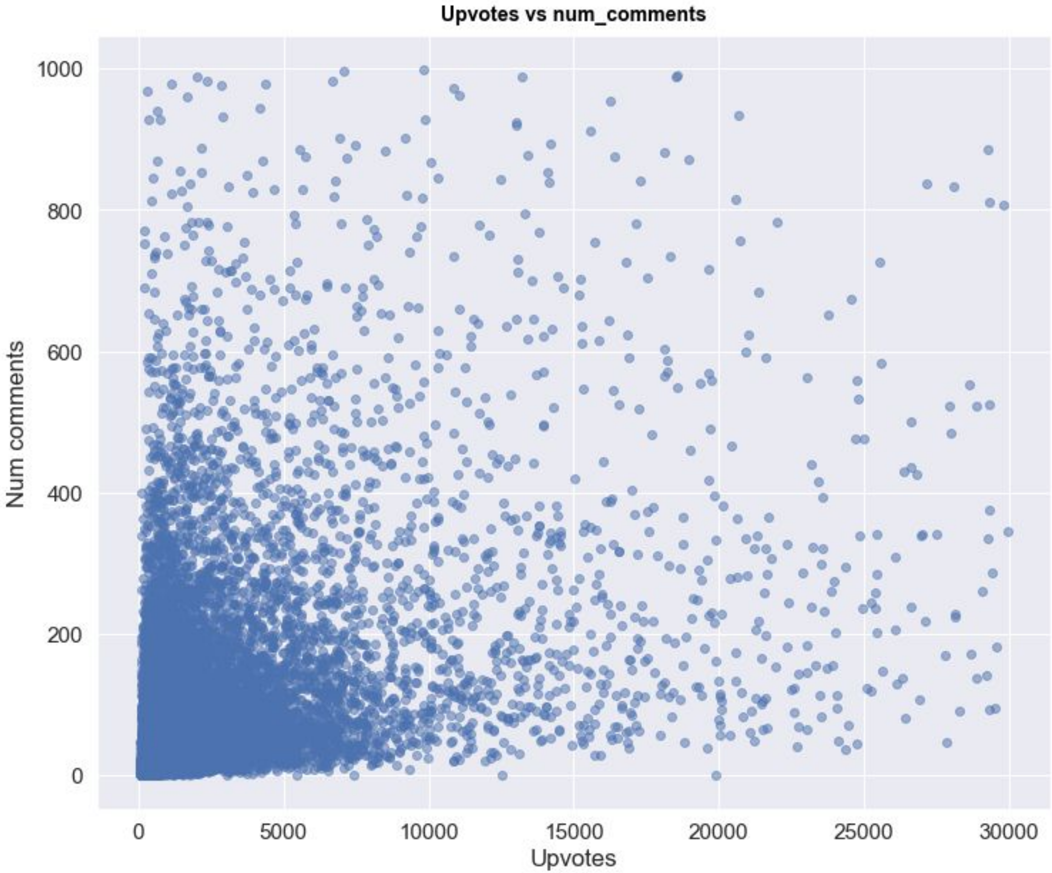
- Null values were checked for in our dataset, none were present.
- Duplicates were checked for and dropped on the basis of Title name (since reddit does not allow title to be changed).
- The last duplicate was kept.
- Titles were cleaned of extra characters and punctuation marks, was lemmatized, and put through TF-IDF vectorizer.
- Subreddits were put through TF-IDF as-is since Reddit controls the subreddit names.



	num_comments	upvotes
count	24993.000000	24993.000000
mean	98.872564	2251.953947
std	388.824545	6432.912676
min	0.000000	63.000000
25%	12.000000	307.000000
50%	30.000000	663.000000
75%	80.000000	1702.000000
max	26180.000000	241124.000000



Some Outliers seen and removed from dataset



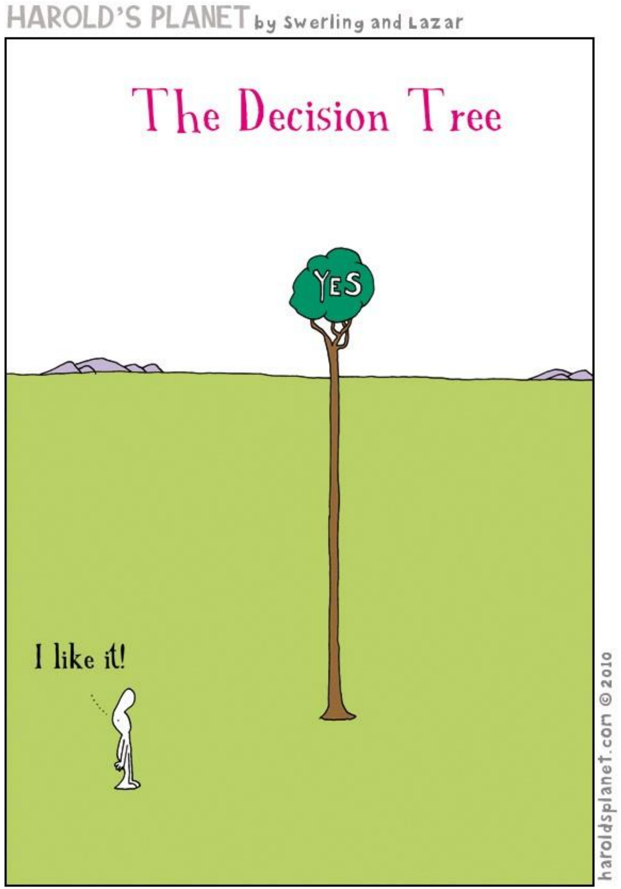
After Removing outliers of over 1000 comments, and 30,000 upvotes

Models Tested For classification

Class majority to beat is 50.52%

- K Nearest Neighbors
- Random Forest
- Extra Trees
- Bagging

Winner: Random Forest

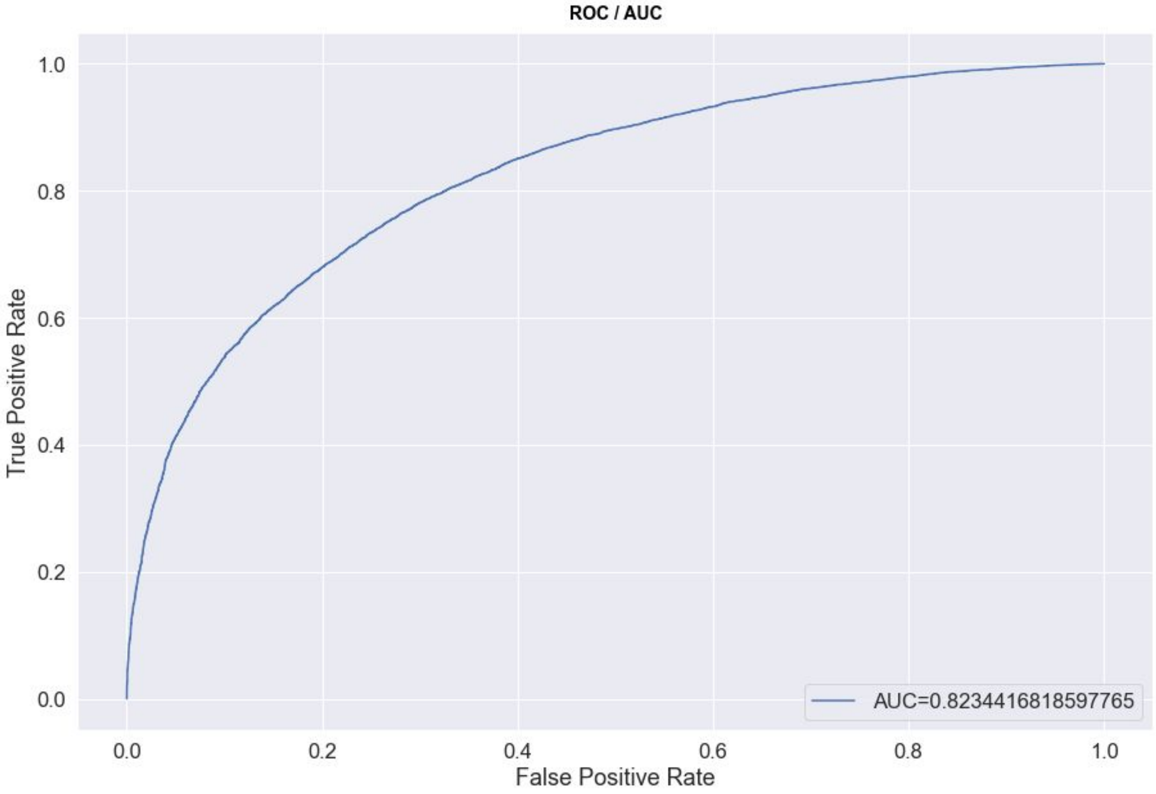


Cross Val Score + Confidence Int.

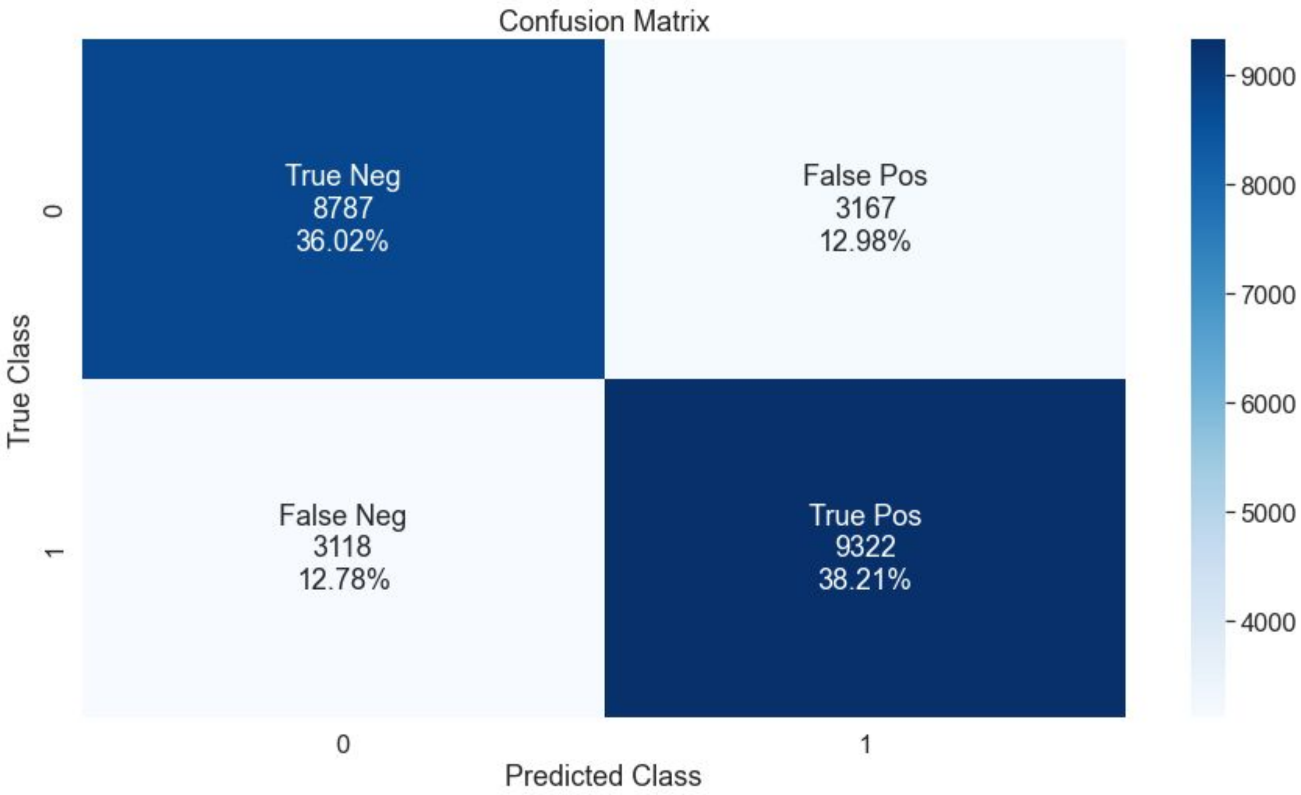
For my Random forest model that scored the best based on cross val scores of X/y_train, X/y_test and overall X/y scores We got the following values:

- X_train/y_train - 74.53%, CI: +- 1.8%
- X_test/y_test - 73.04%, CI: +- 2.4%
- X/y - 73%, CI: +- 1.4%

ROC / AUC curve

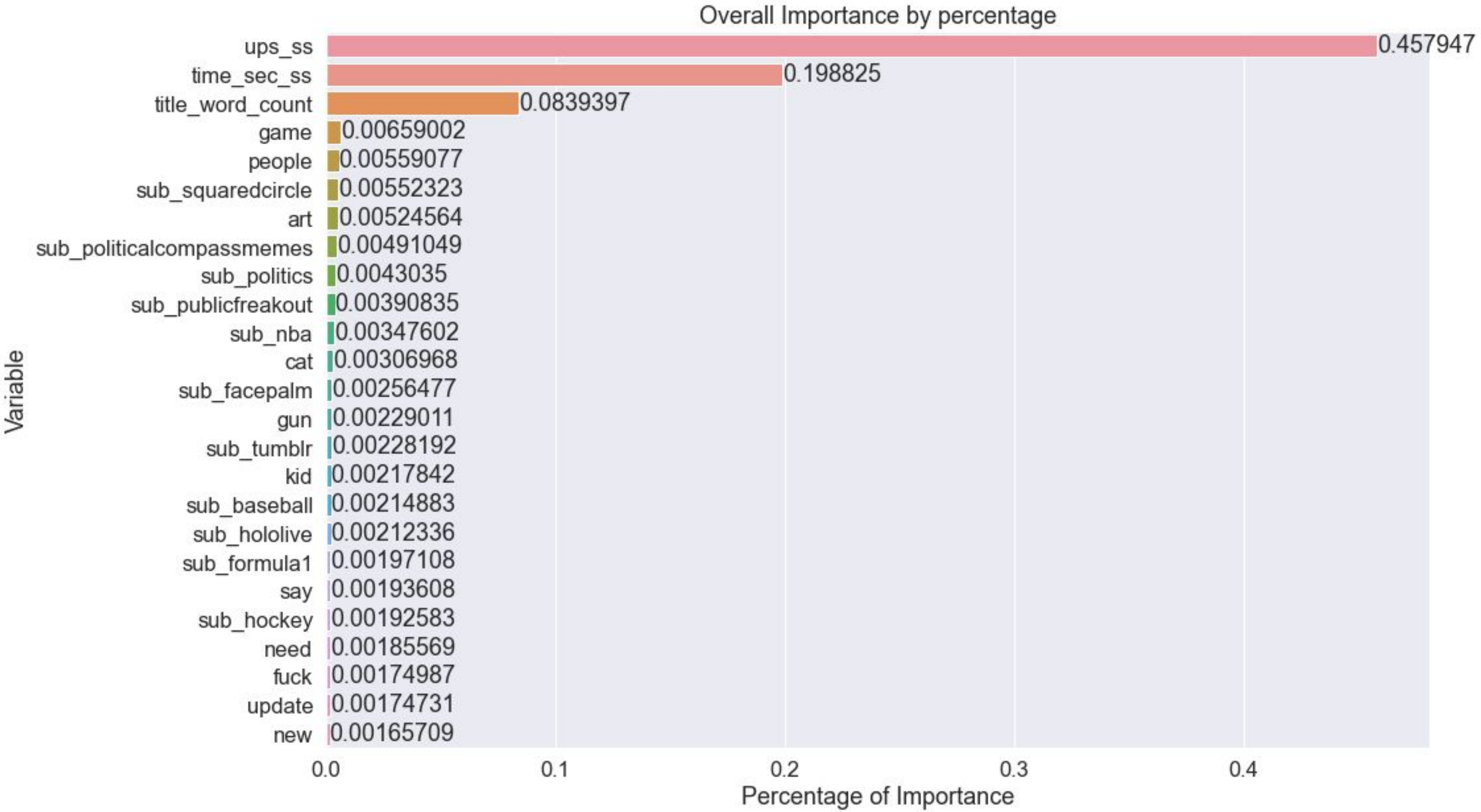


Confusion Matrix

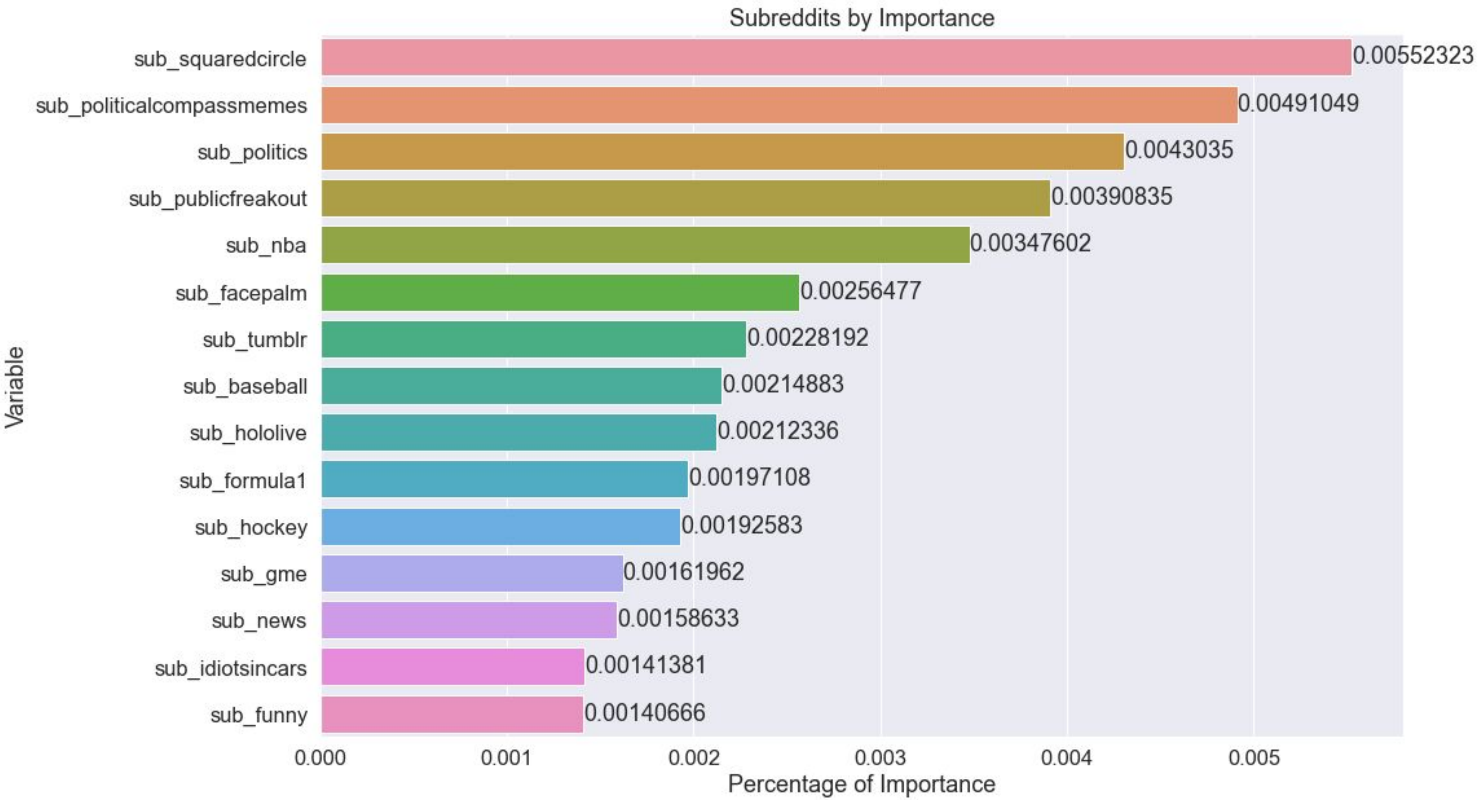


- Accuracy: 74.24%
- Precision: 74.64%
- Sensitivity: 74.94%
- Specificity: 73.51%
- 24,394 cases seen

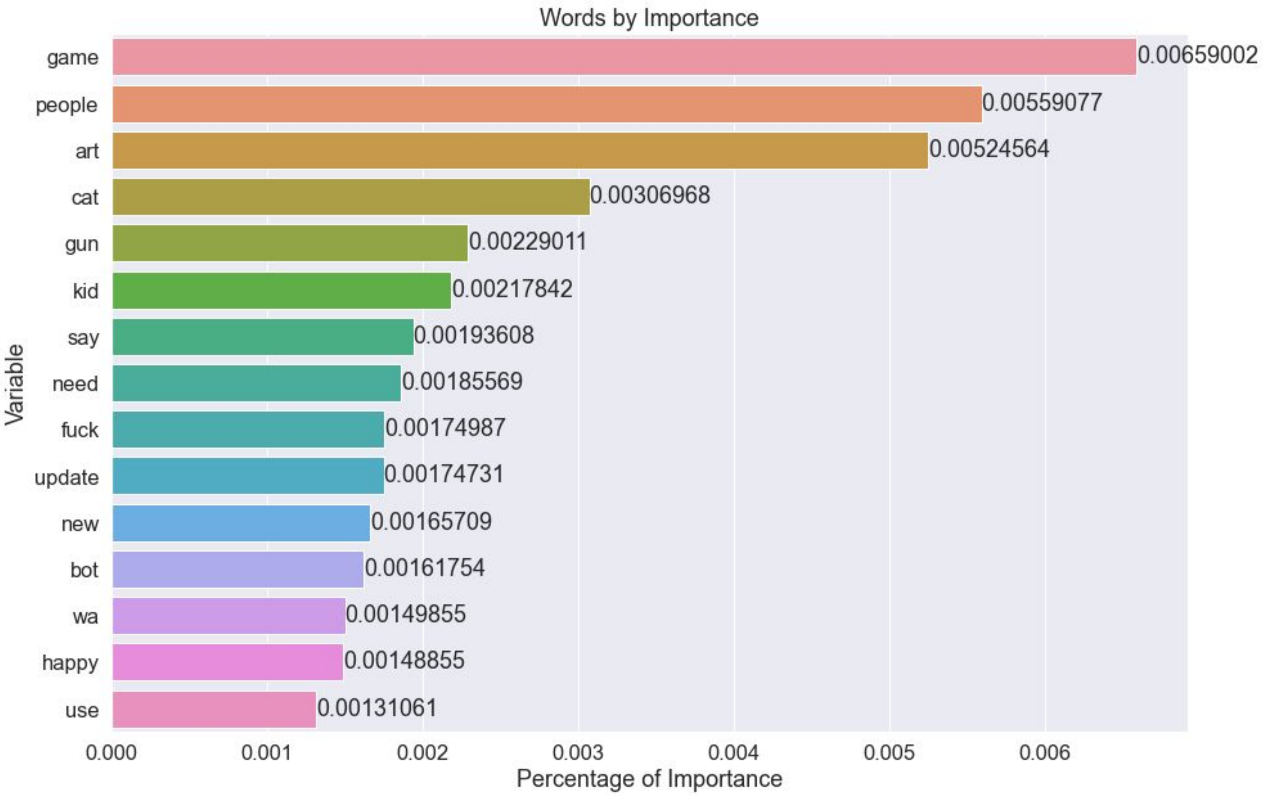
Model Features By Importance



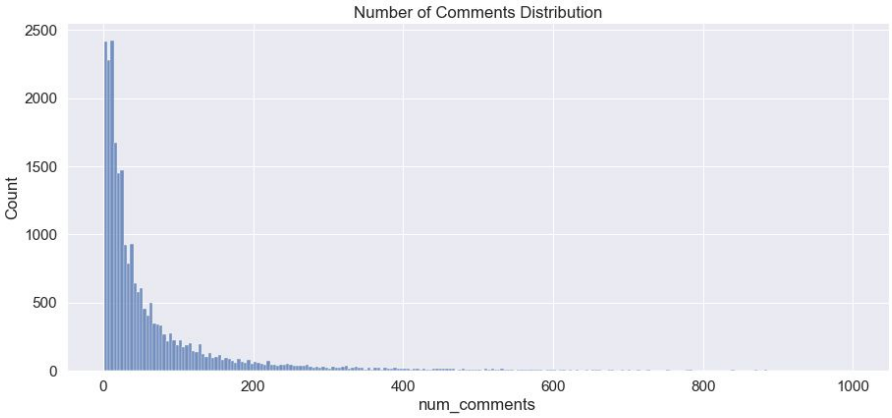
Subreddits by importance

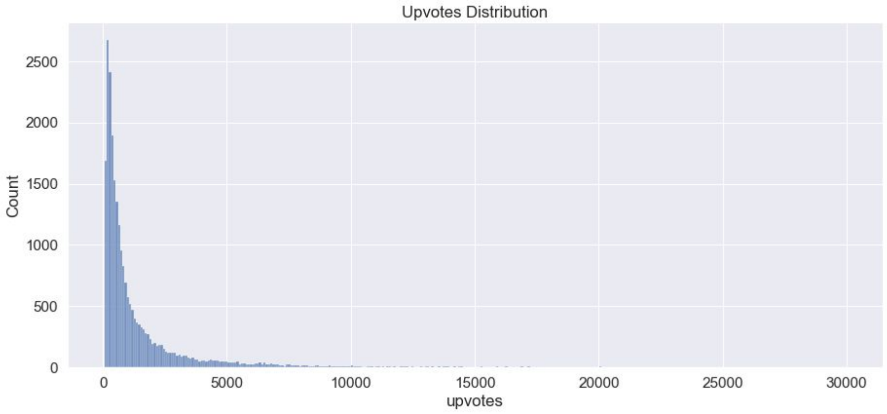


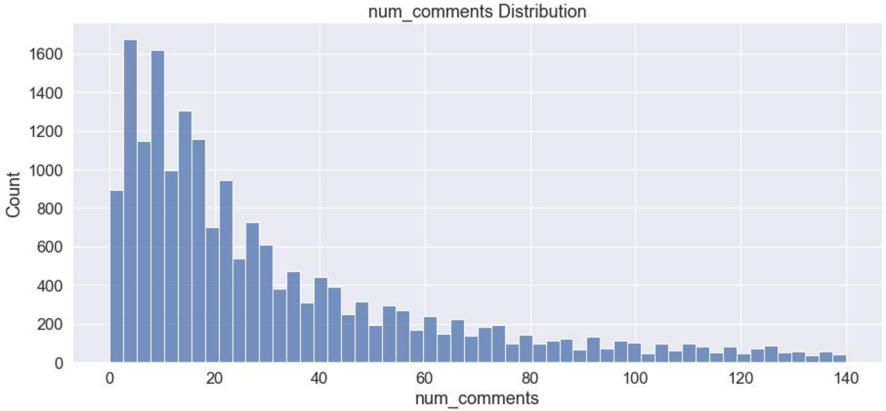
Words in model by Importance

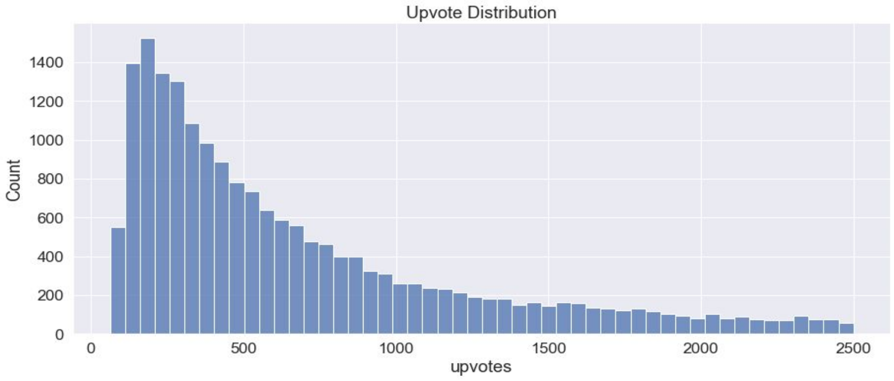


Distributions before and after

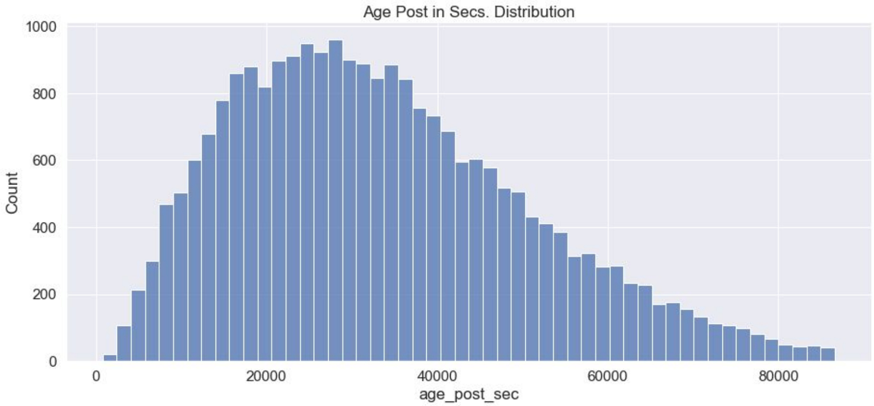


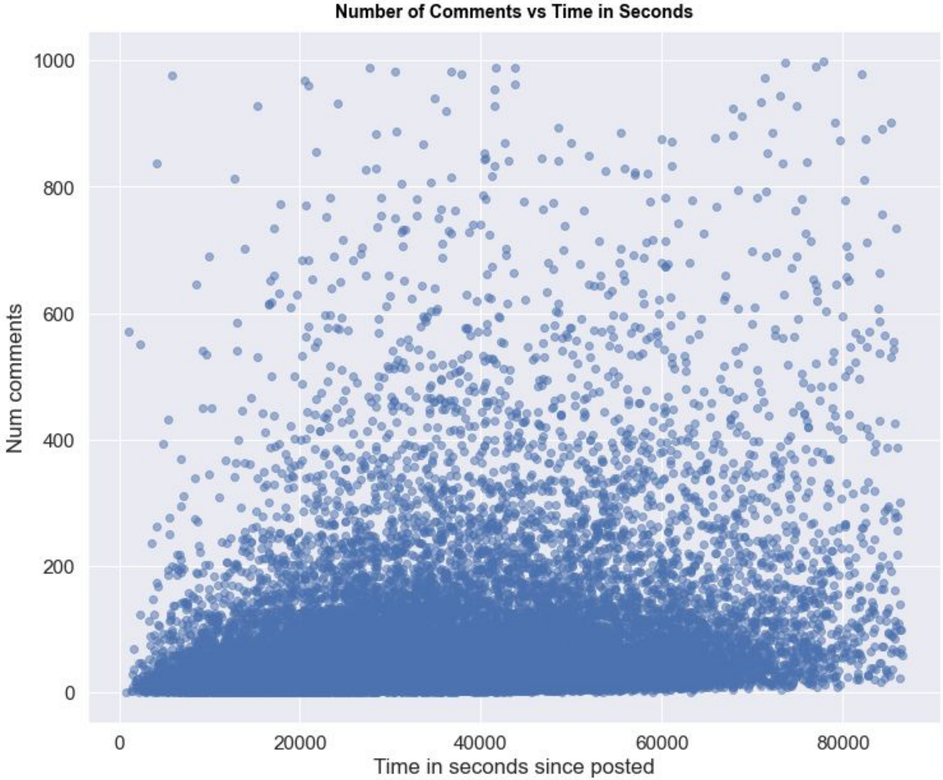






Time in seconds





Conclusions and Findings

After getting the best scores while using the random forest model with our chosen parameters, we extracted which features were of greatest importance to the model itself. It has shown that far and above the amount of upvotes were important in predicting the target feature. Followed by title word count and our standard scaled time in seconds. From this we can deduce that given enough time as a /r/hot post and upvotes start to increase, so too will the number of comments on the post.

Furthermore, we have extracted the top words and subreddits that help our model predict the class with an accuracy of 74.24%. We have demonstrated the top words and subreddits that help our model predict this accuracy.

Recommendations

It would be easy to say that you need to create a post that will get a lot of upvotes and to allow it to sit a length of time, and you will get a fair amount of engagement by way of comments but it doesn’t exactly work that way.

My recommendation would be as follows:

- Try posting in the subreddits I have outlined above as well as using the top words of importance found by the model
- If you see your post getting lots of upvotes early on, it is usually indicative of user engagement that will also come with a fair amount of comments

Room for Improvement

- More datapoints would be gathered per reddit post. Such as the type of post (video, picture, link, article, etc.) to see if this has any influence in user engagement.
- I would get the posting date and time itself, so time of day, day of week, weekend/weekday can be analyzed as influential as well.
- I would really like to also gather posts from /r/new as well. The posts that are in /r/hot are already deemed as "hot" by how many views, upvotes, comments, and shares they are gathering since time of posts. I want to see which posts that are posted to /r/new actually make it to hot, and which ones just fade into obscurity. I think this would really help differentiate the posts that will garnish more user engagement vs the ones that fizzle out and die.
- Hire an assistant!