What influences a Reddit post's upvotes? Investigating the Influence of Popular Words, Parts of Speech and More

Donald Wolfson
dwolfson@ucsd.edu
University of California San Diego
La Jolla, California, USA

ABSTRACT

When it comes to making a popular Reddit post, there are likely numerous things to consider. Determining if a post is popular can be challenging, but their are factors such as the title, number of comments, awards, and time of day that can play a role in the post's success. How well each of these factors influences the prediction of a post's success is the goal of this paper. In terms of 'success', this will be defined as the number of upvotes, or likes, a Reddit post has. In this paper, we investigate how different pieces of information influence the success of some of the most popular posts of the year from popular subreddits. We do so by focusing primarily on the title of the submission, but also other features such a time of day and week, number of comments, and more. The primary goal of the model produced is to be more optimal than a baseline of predicting the average amount of upvotes for each post. The model produced helps us better understand the importance of a title, and what best influences a post's success.

KEYWORDS

prediction model, linear regression, parts of speech, natural language processing, reddit, python

Reference Format:
Donald Wolfson. 2021. What influences a Reddit post's upvotes? Investigating the Influence of Popular Words, Parts of Speech and More.

1 INTRODUCTION

Before diving into the model, and the different approaches that led to its final form, it's important to first understand where the data comes from. This paper utilize web mined data from this years most popular posts from some of the most popular subreddits. The goal of web mining for data over using the preexisting dataset from the paper that inspired this investigation was the hope to use more recent and relevant data as the latter was dated between 2012 and 2013[4]. The issues with the older dataset is that Reddit has grown a lot since nearly a decade ago. Growth of a social platform often comes with new social norms, expectations, rules, and approaches to posting. The initial goal of web mining data was to make a one-to-one replication of the CSV dataset from the prior paper, but with recent posts. But during the process of web mining, some of the columns were redacted or modified as the approaches and goals of this paper differ from prior research. However, the differences of this paper and prior research will be discussed in a later section.

CSE 158, December 2021, La Jolla, CA, USA

1.1 Web Mining Reddit Posts

Before explaining the process of Web Mining Reddit data, it's important to understand what the data would look like, and how much data was desired. Firstly the goal was to get the top 500 posts of this year from the top 500 subreddits, and store metadata on the each posts in a CSV format as follows: image id, unix time, raw time, title, total votes, reddit id, number of upvotes, subreddit, number of downvotes, local time, score, number of comments, and username. This format is the exact same as the prior research done on this topic [4]. However as will be explained below, a few of these columns did not make it into the final dataset for a variety of reasons.

While Reddit does offer a very powerful API for web mining their platform, a Python Wrapper library was utilized for a more Pythonic and easier time [2]. Using this Python Reddit API Wrapper, PRAW, the first goal was to produce a list of the top 500 most popular public subreddits based on subscriber count. Due to limitation of the API, public subreddits were needed to be found due to quarantined and private subreddits throwing errors if there was any attempt to access their posts. This lead to two of largest subreddits, r/BlackPeopleTwitter and r/ImGoingToHellForThis being excluded from the dataset due to being private. On the same note, r/announcements was excluded from this list due to being an officially run subreddit, and dedicated to rare moments of needed meta-discussions. The last notable feature of this list is that is does include both Safe for Work (SFW), and Not Safe For Work (NSFW) subreddits.

Using the list of the top 500 most popular subreddits, the next script was made to iterate over their submissions sorted by the top of this year. Note that year refers to the last 365 days, not calendar year, and the phrase submission is interchangeable with post. For each submission a list of features was collected in the order of the CSV referenced above. After some tests on a smaller subset of posts, some necessary changes to the format were discovered.

1.2 Format of Mined Data

First, the image id column was determined to be unimportant for the dataset. The content of each posts is not part of this paper's scope, thus labeling images or other content was determined to be unnecessary. Next, the conversion of unix time to both human-readable Strings of UTC and local (Pacific) timestamps was taking a considerably large amount of time per post. The machine this script was running on would not be able to feasibly complete this task for the number of desired posts, so neither timestamp was saved, and instead the unix time was kept. The score value initially represented the difference between the number of upvotes and downvotes, but instead was replaced with the sum of awards given to the post in terms of their cost in Karma. This reduced the amount of redundant

columns, as well as tracked the feature of Karma and awards that Reddit utilizes.

Now that the format of the data has been finalized, it's important to consider some minor issues that are apparent in the actual values of the data. Firstly, Reddit's API has removed direct numbers for upvotes and downvotes in the last few years, and now uses a upvote to downvote ratio [9]. This means that the number of upvotes and downvotes are not accurate, but are good estimates based on the ratio returned by the API and PRAW. Lastly, it is important to note the not all subreddits have had 500 posts in the last 365 days. Thus the total number of data points returns is not 250,000 (500×500), but instead 246, 472 posts. What this can infer is that even though the dataset is sorted by the top posts of the year, the number of upvotes may drastically vary. This may impact the performance of the model based on if and how the data is sorted or shuffled.

The final format of the data was stored in a CSV where each column represented these values: unix time, title, total votes, reddit id, number of upvotes, subreddit, number of downvotes, score, number of comments, username.

2 EXPLORATION OF DATA

Once the data was properly mined, there was a lot of worthwhile questions to be had. As the goal of this model was to primarily understand the influence of the title, most exploration was around this aspect. These explorations can be divided up into trying to answer two questions. First, what are some commonly used words in titles? Next, what parts of speech are present in titles? Since the dataset includes 500 subreddits, looking at the top 5 was a realistic approach to gaining a relative answer without graphing more data than necessary.

2.1 Word Frequencies of Titles

The frequency of words in a title can often be attributed to a specific subreddit. For example, the subreddit r/Music, will likely have a huge amount of posts include words like, "song", "rap", "rock", etc. in their titles. Other subreddits, like r/hmmm almost exclusive have posts with the title being some permutation of the word, "hm", or "hmmm", and so on. Thus, the word frequencies in the title may have a role in the culture of the subreddit, and also which posts are more successful. For example, one can assume a post describing a new album or song won't be very successful in r/hmmm, and a post titled "hmmm" in r/Music will also likely not be very successful there. Each subreddit comes with its own subculture and likely treats titles very differently.

To better understand this, word clouds of the top 200 words are used to help visualize this idea. For these visuals, we'll be looking at the top 5 largest subreddits, r/funny, r/AskReddit, r/gaming, r/aww, and r/Music. Based on the titles, one can likely infer how varying their word clouds may be.

Figure 1: r/AskReddit **Figure 2: r/funny**

Figure 3: r/gaming **Figure 4: r/aww**

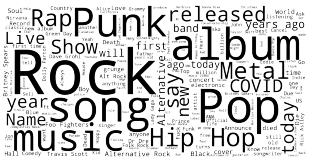

Figure 5: r/Music

As can be seen above, there is very little crossover between subreddits, and their word frequencies. While these visuals may be very insightful into the varying culture and discussions each subreddit has, this does bring up some questions about the model. In a feature vector, will a binary list representing the n most popular words suffice the niches of all subreddits? It's likely that the most popular words across all the subreddits will likely exclude a lot of the niche words of each subreddit that make them successful. These questions, and likely more, will have to be tested on the model.

2.2 Parts of Speech in Titles

As can be seen in the graphs from the previous section, nouns are very prominent in the Word Clouds of the top 5 subreddits. However, it may be worth investigating the Parts of Speech of titles and possibly integrating their frequencies into the model's feature vector. Utilizing Natural Language Processing to get a better understanding of the frequencies of each title's Parts of Speech may help get a better prediction of their success.

The Natural Language Toolkit (NLTK) was used to parse titles for their Parts of Speech [1]. Using this library, titles are first tokenized and then assigned a variety of tags that map to separate parts of speech. While NLTK offers a variety of tags, the script generalizes these tags into more generic bins. For example, NLTK has four tags for nouns to discern combinations of singular, plural, common, and proper nouns. This exploratory scripts bins these tags into one grouping to get a higher level understanding of the frequencies.

To focus on only a few Parts of Speech so as to not complicate the script, the bins include Nouns, Verbs, Adjectives, Pronouns, Adverbs, Determiners, and Others. These bins should give a good idea of how different subreddits may have different frequencies of parts of speech. As with the previous section, this script and visualization will focus on the five largest subreddits: r/funny, r/AskReddit,

r/gaming, r/aww, and r/Music. The graph visualizes each bin, and compares the five subreddits values in terms of their frequency as a percentage of all parts of speech found.

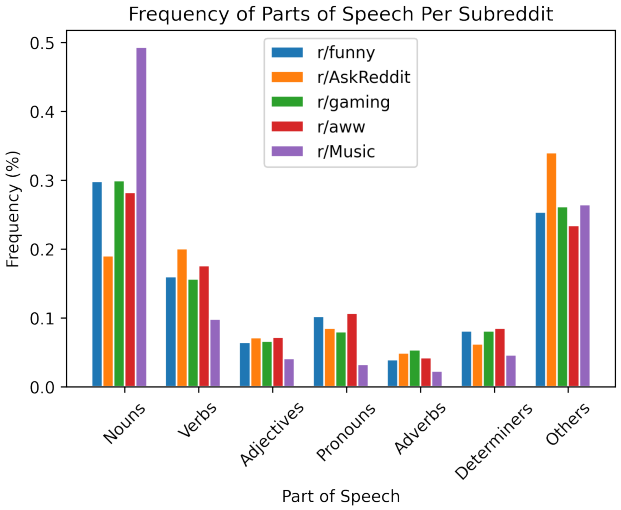

Figure 6: Frequency of Parts of Speech

As can be seen above, Nouns dominate the frequency in titles. Something of note is that r/Music uses a lot more nouns in their title. However, it is interesting to note that each bin is relatively similar to one another. This may be considered a good thing as we face the opposite issue of the Word Clouds in the prior subsection. Instead of having very specialized words per subreddit, the frequencies of parts of speech are relatively similar along the top subreddits. This may result in a more independent feature compared to the word popularity which is dependent upon the subreddit as discussed above. With this information, we can infer a list of either the presence of Parts of Speech, or their frequencies per title may be an optimal addition to the model's feature vector. The final observation is how large the bin for other parts of speech is. This may mean that there could be other worthwhile parts of speech to utilize in a feature vector.

3 PREVIOUS RESEARCH

The research that inspired this investigation was due to the fact the the paper's earliest sections considers this approach Naive [4]. The paper instead focused on the content of the post, and the concept of cross-posting, and re-posting of content such as images in the same or separate communities. Using this approach the paper tried to investigate how similar content with different titles, posted at different times effected the success of the post.

Possibly due to the fact that the paper didn't wish to approach the investigation of how titles influence a post directly, this paper sought to find out what could be done with this approach. However, similar to this previous research, our own model struggles when it comes to word popularity. As discussed in their conclusion, it is hard to discern how much success comes from the content of the post over the title itself posted in the right place at the right time. Notably our model only assess the title, interactions with the post, and the time of submission, while their model also include location such as subreddit.

As stated above, their paper and our exploratory coding of the current dataset both struggle with word popularity of title as it may be too general for all subreddits. As discussed earlier, getting to most popular words appearing in all titles may incorrectly assume words that work well in one community would work well in another. Refer to the example with r/hmmm and r/Music above for clarification.

There is plenty of crossover between the approaches and conclusions of the previous research, but the goal of this paper is less about the content of a post, and more focused on the title. Adding more angles, such as images warrants the more in depth models produced in their paper [4].

4 THE MODEL

As stated in prior sections, the goal of this model is to be able to find what features best influence the prediction of a reddit post's success. Throughout the web mining and exploratory coding process, numerous possible features have been considered and discovered. This section is dedicated to explaining why certain features are included.

4.1 Train, Validation, and Test Sets

Due to the fact that many subreddits that we mined barely had 500 top posts in the past year, it can be assumed the upvotes within this dataset vary widely. As such, our model's distribution of Training to Validation and Test sets should be heavily weighted towards the Training. Thus, this model splits the data 80% for Training, 10% for Validation, and 10% Testing. In terms of raw numbers, this means our model will train on $197,178$ posts, and validate and test on $24,647$ posts respectively.

4.2 What's the Baseline?

To understand what makes one model better than another, we'll need to utilize some type of baseline. For this model, the baseline that all other models will be compared to is a model that simply predicts the average number of upvotes for every post. This model's Mean Standard Error (MSE), will be used as a form of judgment against the other models. The goal is have a better (lower) MSE than the baseline, and get as low as possible with the features present.

In terms of the dataset, the average upvotes per submission is $14,947.35$, the maximum number of upvotes given was $430,539$ and the minimum was 0. Clearly, assuming this average number of upvotes won't be very accurate. When comparing this baseline as a prediction against the actual the upvotes in the validation set the script produces an MSE of $166,498,875.02690643$ when the data is not shuffled, and roughly $500,000,000.0$ when shuffled on average. These are huge MSE's, and likely entail this model may not be very accurate even with the features that will be included. However, while the goal of an MSE is be as close to 0 as possible, an improvement upon the baseline that displays the influence of different features is the primary goal of this model.

4.3 The Features

In terms of the features, 8 were selected and tested on the model. Below are short descriptions of their purpose, format, justification, and how much they influenced the model during ablation.

The first feature added to the feature vector was the score of the post. The score refers to the summed cost in Karma for the awards given to the post. For clarification, Reddit users can pay to award posts with stickers that appear on the top of the post. These stickers vary in price, so the intention of this feature was to hope that the amount of awards in cost given would help predict the success of the post overall. During ablation, this feature displayed a much higher than expected impact on the MSE when removed placing it as the third most influential of the features.

The next feature was the number of comments on the post. The reasoning behind this feature is that interaction likely leads to more upvotes. During ablation, it was shown the removing this feature had the most significant impact on the MSE when removed. As such, this feature was kept.

Two features that were similar in concept were the character length of the title, and the word length of the title. As both are relatively similar in concept, thus justification was that the length of the title may have an influence on the prediction. During ablation, these two were discovered to be the least influential of the 8 features.

As can be seen on the Word Cloud for r/funny, the term "OC" or Original Content is a common attribute of titles on Reddit for users to declare their post as original. This often is used in subreddits for art, or similar content and may attract more attention to the post. However, during ablation this feature was the third least influential on the predictions.

Parts of Speech were surprisingly not as influential in the prediction process as expected. As described in prior sections, the frequencies of a title's parts of speech that were tracked were: Nouns, Verbs, Adjectives, Pronouns, Adverbs, and Determiners. However during ablation, this feature was found to the fifth most influential on the model, being placed lower than the score feature.

Two features that were collected together were the One-Hot Encoding of the hour and weekday. The unix time value from the dataset was converted to local (Pacific) time, and then One-Hot Encoded into a list for the hour of the day, and the weekday. These two features were computed and extended to the feature list together. As such this feature placed as the fourth least influential feature.

The final feature was an n-length binary list representing the n-most popular words found in all title. To find out the best value for n, a script was made that utilized all features listed above, and then iterated over values of n from 0 to 1000 in iterations of 100. For each iteration of n, an binary list of size n that represented the presence any of the n most popular words in the title was added to the feature vector. Then the performance of each model on the validation set and the value of n were stored as pairs to record performance. Thus, the lowest MSE was determined to be the best value for n. From this process, the best value of n found was 600. This feature was the second most influential feature on the model as determined during the ablation tests.

4.4 Linear or Ridge Regression?

While initially the model used Linear Regression, after testing a variety of alphas with Ridge Regression, it was decided that Ridge Regression was the more optimal route. The script that determined the best value for alpha was also the same script as the one used to find the best value for n in the section above. The same script that tested the feature vector with the addition to the n popular words also tested itself against these values for the Ridge's alpha: 0, 0.001, 0.01, 0.1, 1, 10, 100, 1000. This script determined that the best alpha was: 1000.

4.5 To Shuffle or Not

Throughout building the model, the pros and cons of shuffling the data were weighted, and it was decided that shuffling the data would be the fairest approach. Due to the the range of subscribers in the top 500 subreddits being as large as 37, 979, 142 and as small as 10, 47, 267, a sorted dataset would train heavily on very popular subreddits, and test on much smaller subreddits. Another concern is that the more popular subreddits are likely to have all 500 of their possible top posts of the year included, unlike the less active subreddits. As such, shuffling the data ensures that model is more likely to be trained on a wider variety of varying posts. Something to note is that as stated earlier, the baseline MSE difference between the model being shuffled and sorted is nearly triple.

5 RESULTS

Before discussing the official results of the model against the test set, it's worth getting a better understanding of the ablation analysis discussed in the previous section. Below is a table showing the results of the ablation analysis. The three columns are: Features which describes what features are present in each model (row), MSE which is the performance of the model, and %-Change which is the percent change in performance (MSE) from the Baseline. The rows are sorted by the %-Change column.

Features	MSE	%-Change
Baseline	492,722,872.18	0%
All	350,501,965.21	-28.86%
All - Character Length	350,598,726.34	-28.84%
All - Original Content	350,653,732.78	-28.83%
All - OHE. Hour/Week	350.777.364.46	-28.81%
All - Parts of Speech	351,901,479.39	-28.58%
All - Score	370,939,713.74	-24.72%
All - 600 Pop. Words	375.893.943.25	-23.71%
All - Num. Comments	416,686,952.74	-15.43%

As discussed in the prior section on Features, most did not have a significant influence on the MSE. These include the title's character and word length, self declaration of original content, One-Hot Encoding of time of submission, and surprisingly the frequencies of Parts of Speech. As a significant focus of this paper was aimed at finding the importance of the frequencies of parts of speech, it can likely be concluded they don't play as much of a role as expected.

However, some further investigation into parts of speech can still be done. This exploration focus on 6 parts of speech, all of which binned or generalized the tag produces by the NLTK library.

Thus, more expansive research may be done on the the individual tags of parts of speech.

The top 3 features that had an influence on the prediction of posts upvotes were: score, the presence of any of the 600 most popular words, and the number of comments. Based on this list, it's clear that the interaction of awards and comments have clear influence on what the posts upvotes might be. It is notable that the initial predictions and expectations that parts of speech would be a more important feature than popular turned out to be untrue. Given that the top 600 words played a significant role in the prediction model is interesting to know. Further research may be worth looking into what these words are, and then seeing how many are actually important, and not just fluff.

Below are the final results of the model on a newly shuffled dataset. The table follows the prior format, except the rows are now the Baseline against the Test labels, and model's Test performance.

Model	MSE	%-Change
Baseline	501,463,918.97	0%
Test Set	357,822,196.43	-28.64%

While the hope that both the Parts of Speech and Word Popularity would offer a large influence on the model's predictions, it is good to know that at least one played a significant role. Even though the MSE's are very high, a 28.64% performance improvement is certainly a valid and worthwhile improvement.

6 CONCLUSION

This paper aimed to find out what features most influence a Reddit post's success. It is clear that interactions such as awards, and comments play a large role in predicting a post's success. However, the most popular words used overall does offer a large influence as well. Notably, parts of speech offered a much lower influence on prediction than expected. However, both word popularity and parts of speech could both have further investigations going further in-depth into their values. Finally, our total model included 8 features and offered a 28.64% performance improvement compared to the baseline.

ACKNOWLEDGMENTS

We would like to thank frontpagemetrics.com[3] for their data that was parsed to create the underlying dataset of this paper. The data used from their website was the file, 2021-11-19.csv.

We would also like to thank the various third party libraries that were used in this paper and scripts. Without them, none of this work would have been done in the time frame granted. These third party libraries, in no specific order, include: numpy.org[7], scikit-learn.org[8], matplotlib.org[5], Python Reddit API Wrapper (PRAW)[2], wordcloud[6], and nltk.org[1].

REFERENCES

[1] Steven Bird and Liling Tan. [n.d.]. *Natural Language Toolkit.* https://www.nltk.org/install.html
[2] Bryce Boe. [n.d.]. *Python Reddit API Wrapper.* https://praw.readthedocs.io/en/stable/#getting-starteds
[3] FrontPageMetrics. [n.d.]. *Front Page Metrics: List All Subreddits.* https://frontpagemetrics.com/list-all-subreddits
[4] Himabindu Lakkaraju, Julian J. McAuley, and Jure Leskovec. 2013. What's in a Name? Understanding the Interplay between Titles, Content, and Communities in Social Media. In *Proceedings of the Seventh International Conference on Weblogs and Social Media, ICWSM 2013, Cambridge, Massachusetts, USA, July 8-11, 2013*, Emre Kiciman, Nicole B. Ellison, Bernie Hogan, Paul Resnick, and Ian Soboroff (Eds.). The AAAI Press. https://cseweb.ucsd.edu//~jmcauley/pdfs/icwsm13.pdf
[5] matplotlib.org. [n.d.]. *Matplotlib.* https://matplotlib.org
[6] Andreas Mueller. [n.d.]. *wordcloud.* https://pypi.org/project/wordcloud/
[7] NumPy.org. [n.d.]. *NumPy.* https://numpy.org
[8] scikit learn.org. [n.d.]. *scikit-learn.* https://scikit-learn.org/stable/
[9] u/Deimorz. [n.d.]. *reddit changes: individual up/down vote counts no longer visible, % like it" closer to reality, major improvements to "controversial" sorting.* https://www.reddit.com/r/announcements/comments/28hjga/reddit_changes_individual_updown_vote_counts_no/