Master's Thesis

Analyze and predict news popularity on NY Times' Twitter - focus on the content of tweets

Luqi Chen

Columbia University

Table of Contents

ABSTRACT

With the rapid development of social network platforms such as Twitter, Facebook, YouTube, Weibo, online social media is transforming the traditional news industry as they act as crucial sources of information. Moreover, the influence is exponentially expanded by functions like "retweet" by which individuals can pass along information to one another. Facing this challenge, how can big news agencies like New York Times react in order to produce popular news and attract the largest amount of audience? In the paper, we focus on the content of the news itself and suggest that a better performance of news agencies on online social media can be achieved partly by adjusting the news content. We utilize the scraped information of New York Times' tweets and try to analyze the relationship between content features (length, hashtag, simplicity, sentiment, terms, named entities) and popularity of the tweets (which can be indicated by number of retweets/likes). This study then applies machine learning methods to build the prediction model.

1 INTRODUCTION

1.1 Statement of the Research Question

Due to the "word-of-mouth" diffusion mechanism which has been studied by many scholars (Bakshy, Hofman, Mason, & Watts, 2011), Twitter, a previously communication and microblogging platform, is serving increasingly as a new medium in real-time happenings (Kwak, Lee, Park, & Moon, 2010). On this platform, everyone who has an account can subscribe to a news channel by "following", express his or her attitude by "like", and disseminate the news by retweeting. Affected by these features, many traditional news agencies became super nodes on Twitter, like BBC and CNN. Though previously these large news agencies had dominated the competition because of the high cost of news reporting and disseminating, the competition on social media service like Twitter has been more or less a competition of content.

Predicting news popularity on Twitter, especially by using the news content, is of great theoretical and practical importance. Theoretically, taking the advantage of Twitter's features, this serves as a natural experiment to test whether or not the content itself can affect information "cascades" --- the number of users in each influence tree (Bakshy et al., 2011), or in other words, to which extent will the receiver of the news decide to pass along it to others just because of the content? Practically, as large news agencies produce tons of news every day, predicting news popularity with the news content can help them optimize the recommendation system, help them choose the most popular news to post on their Twitter account, so that their news can reach the largest population. What's more, as every tweet posted is only a news summary with a URL link navigating to the original article, this study can help large new agencies draft their summaries in a more-likely-to-be-popular way so that they can gain more page views in their original website. Also, the study can benefit small news agencies which are growing and utilizing Twitter as one of the main platforms, they can concentrate on improving the content of their news to reach increasing amount of audience and expand their institutional influence and reputation in the news industry.

Taken together, researching news popularity on Twitter basing on news content is both necessary and important in the current social media society. This study chooses New York Times as our target news agency and study the predictive value of different content features for news popularity by building and evaluating different prediction models.

1.2 Rationale for the Proposed Study

Since Twitter came into use in 2006, the rapid growth of Twitter user population and emerging influence of tweets have caught much attention from scholars. However, most of the prior works analyzed "popularity of tweets", not specifically "popularity of news", this study is going to narrow down to focus on NY times' tweets which are all essentially news. In addition,

not much work has been done focusing particularly on the content of tweets, and those which focused on a combination of content-based and author-based features led to contradictory results about predicting power of content-based features, this paper is going to use tweet data released from only one twitter account: "nytimes", so that we can control for the author-based features at the first step. At last, previously the measure of tweet popularity had been limited to looking at retweets only, this study contributes by look at the number of "likes" too, which has never been used before.

2 LITERATURE REVIEW

2.1 Predicting Posts Popularity on Online Social Media

Online social media is gaining increasingly attention and prior research has analyzed popularity of posts on different social network platforms. In early work, Szabo et al. (Szabo & Huberman, 2010) investigated **stories** and **videos** on **Digg** and **YouTube** and found strong linear correlation between popularities of content at later and earlier times, thus proposed three models to predict future time popularity by measuring the popularity of the content in a given time. Recently, Trzcinski et al. (Trzcinski & Rokita, 2017) looked at videos on **YouTube** and **Facebook**, his Support Vector Regression models indicate that **"visual features computed before the publication of the video"** alone can be helpful to predict popularity in the future, while **"social features"** and **early view counts** can act as a supplement. Meanwhile Chinese scholars have paid more attention to **Sina Microblog platform** which is a main network platform in China. Kuang et al. (Kuang, Tang, & Guo, 2014) claimed that "importance of the content" and "response time" are nice predictors of times of retweeting if tied with the main social events in China; Lin et al. (Lin, Li, Chang, &Kinshuk, 2017) incorporated even more features (39 features regarding author, text, recipient and relationship) when predicting retweeting behaviors.

2.2 Predicting Tweets Popularity on Twitter

Twitter, in particular, has attracted much research interest in recent years. Research concerning prediction of popularity of tweets has studied the explanatory power of a variety of features, these features can be divided into two parts, (1) Author-based features: features related to the twitter account who posted the tweet originally, for example, number of followers, number of tweets, page-rank, etc. (2) Content-based features: features related to the content of the tweet itself, like length, URLs, hashtag, username, named entities, sentiment, category of content, subjectivity, topics, terms, etc. As I am mainly focusing my attention on content-based features, we will briefly review the author part and discuss more on the content part.

2.2.1 Author-based Features

Some of the studies observe no evidence in proving the relationship between author-based features and number of retweets, Cha et al. (Cha & Gummadi, 2010) for example, argued that high number of followers does not necessarily lead to increase in number of retweets, another study carried out by Kwak et al. illustrated the difference between ranking by the number of followers and ranking by retweets (Kwak, Lee, Park, & Moon, 2010).

However, these findings are contradicted by other analysis where influence of author-based features are examined together with content-based features. Several features of the author were proved important in predicting retweetability: (1) number of followers (Bakshy et al., 2011; Jenders, Kasneci, & Naumann, 2013; Martin, Hofman, Sharma, Anderson, & Watts, 2016; Suh, Hong, Pirolli, & Chi, 2010) (2) number of followees and the age of the account (Martin et al., 2016; Suh et al., 2010) (3) total count of tweets (Martin et al., 2016) (4) degree distribution (Hong, Dan, & Davison, 2011) (5) source of the news (Bandari, Asur, & Huberman, 2012; Wu & Shen, 2015) (6) “social features” especially number of followers and lists (Petrovic, Osborne, & Lavrenko,

2011). In addition, a study focusing on predicting the popularity of newly emerging hashtags (Ma, Sun, & Cong, 2013) tested a range of content-based and author-based features and concluded that author-based features out-performs content-based features.

2.2.2 Content-based Features

There is a debate about which, if any content features lead to higher popularity of news articles on Twitter and my research is going to test previously suggested features and also propose an additional measure of popularity.

Obvious content features: length, URLs, hashtag, mentions, named entities

Several studies have focused on or involved obvious features and most of them led to positive findings. For example, A logistic regression was performed by Naveed et al. (Naveed, Gottron, Kunegis, & Alhadi, 2011) to prove that containing hashtags, usernames and URLs play roles in increasing the likelihood of being retweeted. Suh et al. pointed out the significant effects of hashtags and URLs on retweet probabilities (Suh et al., 2010), Petrovic et al. (Petrovic et al., 2011) claimed that "it is possible to predict retweets just by looking at the text of the tweet", these are confirmed by Jenders et al. (Jenders et al., 2013) who used generalized linear model to prove that number of URLs and mentions have large influence on the prediction, followed by tweet length and number of hashtags.

The negative record come from Bandari et al. (Bandari, Asur, & Huberman, 2012) who pointed out that named entities do not improve prediction models much.

This study is going to include all the obvious content features mentioned except URLs, because every tweet from New York Times contains a URL which links to the original article. For "named entities", I just use a feature named "president" to represent if the tweet contains "Trump" or not.

Latent content features: sentiment, category of content, subjectivity, topics, terms

Most research which analyzed sentiment of tweet content showed significant evidence of its predicting power. Naveed et al. features (Naveed et al., 2011) argued negative sentiments play roles in increasing the likelihood of being retweeted, this is confirmed by Jenders et al. (Jenders et al., 2013) who used generalized linear model to prove that tweets with negative sentiments are much more likely to be retweeted, and again by Wu & Shen (Wu & Shen, 2015) who reported the correlation between negative sentiment of news and retweet popularity.

A slightly different finding carried out by Bakshy et al. (Bakshy et al., 2011) revealed that although content that elicits more positive generate larger cascades on average, which means that sentiment of tweets is descriptively important, it does not improve the model fit and was not as informative as author-based features.

When looking at category of content and subjectivity, we see more negative findings. A common finding by Bandari et al. (Bandari, Asur, & Huberman, 2012) and Martin et al. (Martin et al., 2016) is that content features cannot improve predictive power. Among which Bandari et al. focused on subjectivity and categories, Martin et al. investigated domain, tweet time, spam score and category.

Terms and topics are more advanced features which usually require TF-IDF and topic modeling methods. A typical research in this area was done by Hong & Davison before (Hong & Davison, 2010), in which they used several different topic modeling schemes together with TF-IDF weighting scores as features and compared their performance on retweet prediction. They concluded that topic mixture can be supplement features in classifying “retweet or not”, but most of the results are worse compared by the baseline, TF-IDF.

Among these features, this study is going to select sentiment, subjectivity, terms, and add readability to investigate. I am not touching category of content because there is only one category here that is news. Topic analysis will not be included either considering the way that news pop out on Twitter: they just come one by one and the user cannot filter by category.

Overall content features performance

Other studies I did not include in the above review did not specify the predictive value of specific content feature, but viewed content-based features as a whole. They agree on that content-based features did not act as important factor. By running multiple linear regression on one baseline and seven compared models, Maleewong (Maleewong 2016) reports that the baseline model which contains only content-based and author-based features has a significantly lower accuracy than other models and only reached an adjusted R square of 0.453, adding retweeter-based features can improve the model to a large extent; Zaman et al. (Zaman, Herbrich, Van Gael, & Stern, 2010) got a slightly different finding: "the most of the predictive power comes from the tweeter and retweeter". But their results agree on the conclusion that content-based features are not important in predicting tweet popularity.

3 CONCEPTAULIZE NEWS TWEET POPULARITY

In Twitter, retweeting has been considered as the most important information propagation as it has viral nature and can directly affect cascade size (Petrovic et al., 2011; Wu & Shen, 2015), till now to the best of my knowledge all the articles use retweet, either retweet probability or retweet number, as indicator and measurement of tweet popularity, this is agreed and continued by this paper which will use the number of retweets as the first measure of news tweet popularity.

In addition, I noticed that each tweet has a number of "favorites", which shows how many people favors this tweet. I suggest that count of "favorites" is an indicator of how much

acknowledgements it gained from the audience, so it can act as a supplement measure of news tweet popularity and will be discussed in this paper too.

4 HYPOTHESES

Content-based features (length, hashtag, mentions, named entities, simplicity, sentiment, and TF-IDF of terms) have predictive value for number of **retweets** and **favorites** for New York Times's news tweets.

1) Linear prediction model which incorporates the seven features (length, hashtag, mentions, named entity, subjectivity, readability and sentiment) plus TF-IDF features can have high explanatory power (with an adjusted R square more than 0.5) and some of the features can significantly affect number of retweets and likes.

2) Classification models using the seven features (length, hashtag, mentions, named entity, subjectivity, readability and sentiment) can outperform a naïve approach.

3) Classification models using the seven features (length, hashtag, mentions, named entity, subjectivity, readability and sentiment) plus TF-IDF features can outperform a naïve approach to a large extent.

5 DATA

5.1 Dataset Collection

Data is comprised of a set of news tweets posted by New York Times' official Twitter account (https://twitter.com/nytimes) on 30 random days from September to October. The sample contains 2309 observations without any missing values and there are 3 variables in the raw dataset: text of the tweet, retweet number of the tweet and likes number of the tweet. The content-based features are generated from the text of the tweet by feature engineering steps.

This data was collected by data scraping in Python. Raw data was accessed by Twitter API's "GET statuses/user_timeline" application which can return a collection of the most recent Tweets posted by the user indicated by the screen_name or user_id parameters (https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-user_timeline). I got the most recent tweets of New York Times by specifying "screen_name = "nytimes".

As Twitter only allows me to scrape the information of most recent 200 tweets every day, it is impossible to get the whole 2309 tweets at once at a certain time point and divide the retweet count and likes count by number of days since it posted. According to the finding that approximately 75% of retweet actions occur within the first day (Kwak, Lee, Park, & Moon, 2010), this study approximately assume that the retweet count after 24 hours equals the final counts. Thus, I developed a "Plus One Day" scraping collection mechanism, namely, scrape tweets data of day (T) at the end of day (T+1). At last, I then appended all the data frames together and got my final dataset for analysis.

Figure 1: *Excerpt of the raw dataset collected by web scraping*

	created_at	retweet_count	favorite_count	text
0	Mon Sep 09 03:37:47 +0000 2019	36	152	b'The good, the bad, and the ugly side of havi...
1	Mon Sep 09 03:10:13 +0000 2019	43	163	b'In the Albanian capital of Tirana, the count...
2	Mon Sep 09 02:50:02 +0000 2019	22	97	b'The 17 Democratic candidates who are lagging...
3	Mon Sep 09 02:49:40 +0000 2019	4898	0	b"RT @npfandos: NEW: I got ahold of a draft co...
4	Mon Sep 09 02:32:03 +0000 2019	247	531	b"President Trump's plan to secretly meet with...
5	Mon Sep 09 02:03:50 +0000 2019	86	0	b'RT @TheWeekly: Estonia, a small European dem...
6	Mon Sep 09 02:01:14 +0000 2019	21	0	b'RT @TheWeekly: Were live! Watch #TheWeeklyNY...
7	Mon Sep 09 01:43:04 +0000 2019	686	0	b'RT @mattbpurdy: How Trumps high-risk gambit ...
8	Mon Sep 09 01:24:51 +0000 2019	116	621	b"For Paul Eng, trying to recreate his family'...
9	Mon Sep 09 01:11:58 +0000 2019	601	2934	b'Breaking News: Rafael Nadal has defeated Dan...

5.2 Data Cleaning and Preprocessing

Figure 2: *Excerpt of the raw dataset-identify retweets*

	created_at	retweet_count	favorite_count	text	is_retweet
0	Mon Sep 09 03:37:47 +0000 2019	36	152	b'The good, the bad, and the ugly side of havi...	No
1	Mon Sep 09 03:10:13 +0000 2019	43	163	b'In the Albanian capital of Tirana, the count...	No
2	Mon Sep 09 02:50:02 +0000 2019	22	97	b'The 17 Democratic candidates who are lagging...	No
3	Mon Sep 09 02:49:40 +0000 2019	4898	0	b"RT @npfandos: NEW: I got ahold of a draft co...	Yes
4	Mon Sep 09 02:32:03 +0000 2019	247	531	b"President Trump's plan to secretly meet with...	No
5	Mon Sep 09 02:03:50 +0000 2019	86	0	b'RT @TheWeekly: Estonia, a small European dem...	Yes
6	Mon Sep 09 02:01:14 +0000 2019	21	0	b'RT @TheWeekly: Were live! Watch #TheWeeklyNY...	Yes
7	Mon Sep 09 01:43:04 +0000 2019	686	0	b'RT @mattbpurdy: How Trumps high-risk gambit ...	Yes
8	Mon Sep 09 01:24:51 +0000 2019	116	621	b"For Paul Eng, trying to recreate his family'...	No
9	Mon Sep 09 01:11:58 +0000 2019	601	2934	b'Breaking News: Rafael Nadal has defeated Dan...	No

Firstly, “New York Times” retweets others’ tweets at times, these are not news tweets created by “New York Times” and should be left out of this analysis. For example, in Figure 1 above, there are 4 tweet texts that contains “RT”, which shows they are retweets themselves. I utilized Python’s Pandas and NumPy packages to get rid of these retweets. After deleting retweets, there remain 2200 records.

Figure 3: *Excerpt of the dataset without retweets (2200 rows and 4 columns)*

	created_at	retweet_count	favorite_count	text
0	Mon Sep 09 03:37:47 +0000 2019	36	152	'The good, the bad, and the ugly side of havin...
1	Mon Sep 09 03:10:13 +0000 2019	43	163	'In the Albanian capital of Tirana, the countr...
2	Mon Sep 09 02:50:02 +0000 2019	22	97	'The 17 Democratic candidates who are lagging ...
4	Mon Sep 09 02:32:03 +0000 2019	247	531	"President Trump's plan to secretly meet with ...
8	Mon Sep 09 01:24:51 +0000 2019	116	621	"For Paul Eng, trying to recreate his family's...
9	Mon Sep 09 01:11:58 +0000 2019	601	2934	'Breaking News: Rafael Nadal has defeated Dani...
10	Mon Sep 09 01:01:23 +0000 2019	251	475	'In Fashionopolis, Dana Thomas, a veteran styl...
11	Mon Sep 09 00:30:04 +0000 2019	36	233	'This flexible soba noodle dish also works wel...
12	Mon Sep 09 00:00:03 +0000 2019	311	1005	'A dramatic revolt by British Conservative Par...
13	Sun Sep 08 23:15:20 +0000 2019	44	185	'The bag is to Berlin what a New Yorker tote i...

Secondly, to prepare for the generation of latent content features such as sentiment score and TF-IDF scores, the text data is supposed to be cleaned and should contain no URLs, hashtags and mentions. Python's *preprocessor* package can do all of these for us at the same time.

Figure 4: *Excerpt of the dataset with "clean_text" (2200 rows and 5 columns)*

	created_at	retweet_count	favorite_count	text	clean_text
0	Mon Sep 09 03:37:47 +0000 2019	36	152	'The good, the bad, and the ugly side of havin...	'The good, the bad, and the ugly side of havin...
1	Mon Sep 09 03:10:13 +0000 2019	43	163	'In the Albanian capital of Tirana, the countr...	'In the Albanian capital of Tirana, the countr...
2	Mon Sep 09 02:50:02 +0000 2019	22	97	'The 17 Democratic candidates who are lagging ...	'The Democratic candidates who are lagging beh...
4	Mon Sep 09 02:32:03 +0000 2019	247	531	"President Trump's plan to secretly meet with ...	"President Trump's plan to secretly meet with ...
8	Mon Sep 09 01:24:51 +0000 2019	116	621	"For Paul Eng, trying to recreate his family's...	"For Paul Eng, trying to recreate his family's...
9	Mon Sep 09 01:11:58 +0000 2019	601	2934	'Breaking News: Rafael Nadal has defeated Dani...	'Breaking News: Rafael Nadal has defeated Dani...
10	Mon Sep 09 01:01:23 +0000 2019	251	475	'In Fashionopolis, Dana Thomas, a veteran styl...	'In Fashionopolis, Dana Thomas, a veteran styl...
11	Mon Sep 09 00:30:04 +0000 2019	36	233	'This flexible soba noodle dish also works wel...	'This flexible soba noodle dish also works wel...
12	Mon Sep 09 00:00:03 +0000 2019	311	1005	'A dramatic revolt by British Conservative Par...	'A dramatic revolt by British Conservative Par...
13	Sun Sep 08 23:15:20 +0000 2019	44	185	'The bag is to Berlin what a New Yorker tote i...	'The bag is to Berlin what a New Yorker tote i...

As the data frame above cannot illustrate the difference between "text" and "clean_text", some examples from the dataset are selected here for reference:

Table 1: Comparison of 'text' and 'clean_text'

text	clean_text
'The good, the bad, and the ugly side of having a movie crew take over your home https://t.co/b5o4vry1Nf'	'The good, the bad, and the ugly side of having a movie crew take over your home '
'In his first extensive comments since the Harvey Weinstein story broke, Bob Weinstein explained to our reporters @JodiKantor and @mega2e that he mistakenly saw his brothers problem as sex addiction and eventually "got worn out" trying to intervene\nhttps://t.co/UmYkSqeJE8'	'In his first extensive comments since the Harvey Weinstein story broke, Bob Weinstein explained to our reporters and that he mistakenly saw his brothers problem as sex addiction and eventually "got worn out" trying to intervene

'Kamala Harris says people in El Paso asked her, "Do you think Trump is responsible for what happened?" At the #DemDebate, Harris says she replied, "Obviously he didn\'t pull the trigger, but he\'s certainly been tweeting out the ammunition." https://t.co/T1SPFJHLMv https://t.co/JrPRyMcBTh'	'Kamala Harris says people in El Paso asked her, "Do you think Trump is responsible for what happened?" At the , Harris says she replied, "Obviously he didn\'t pull the trigger, but he\'s certainly been tweeting out the ammunition." '

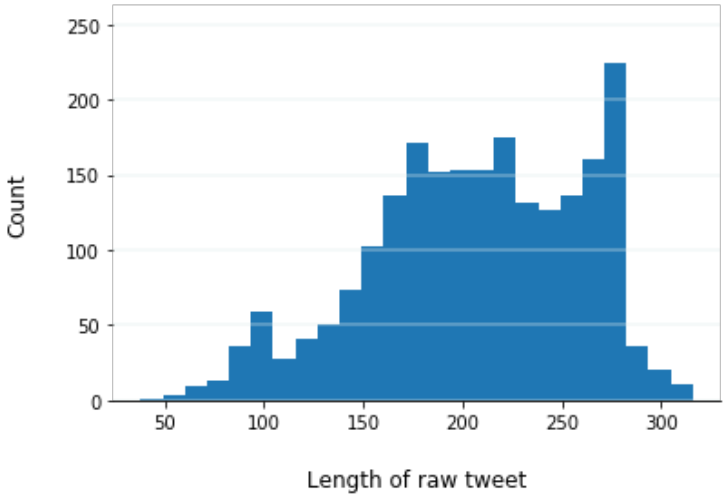
5.3 Feature Engineering

5.3.1 Obvious Content Features

Length of tweet

This analysis was performed using un-preprocessed tweets, which is the "text" column in my dataset. The reason of doing so is to capture the length of the raw tweet seen by the audience. Python's *len* method is used to calculate the length of un-preprocessed tweets, which is the total count of letters in the sentence, including white spaces and punctuation. The new variable is called "length".

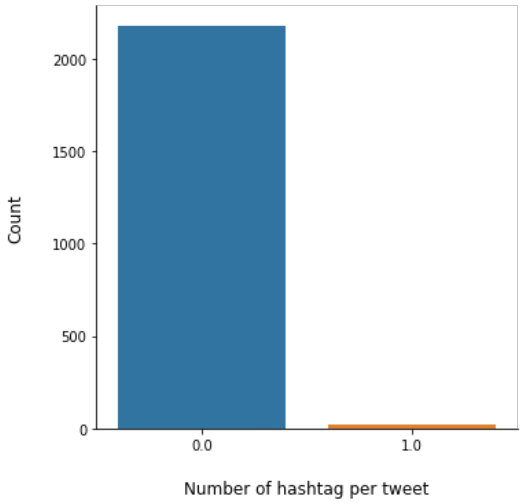
Figure 5: *Distribution of "length"*



Count of hashtags

A new variable called “hashtag” shows the count of hashtags (#) in every un-preprocessed tweet.

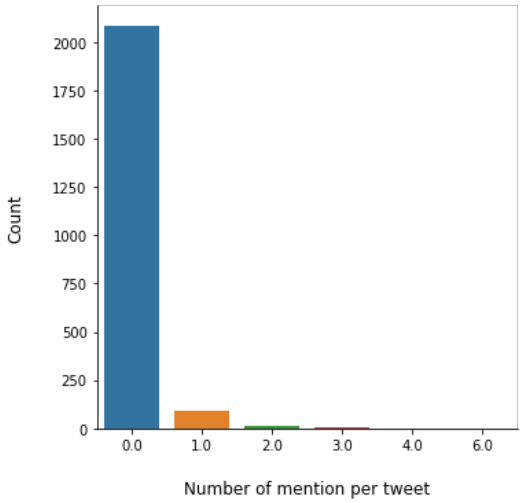
Figure 6: *Distribution of “hashtag”*



Count of mentions

A new variable called “mention” shows the count of mentions (@) in every un-preprocessed tweet.

Figure 7: *Distribution of “mention”*



Named entity (Trump)

A new binary variable called “president” shows if the un-preprocessed tweet contains “Trump” or not.

Figure 8: *Distribution of “*president*”*

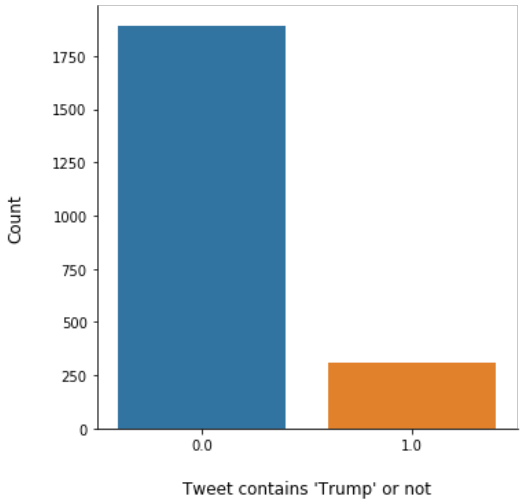


Figure 9: *Excerpt of the dataset with obvious content features (2200 rows and 9 columns)*

	created_at	retweet_count	favorite_count	text	clean_text	length	hashtag	mention	president
0	Mon Sep 09 03:37:47 +0000 2019	36	152	'The good, the bad, and the ugly side of havin...	'The good, the bad, and the ugly side of havin...	105.0	0.0	0.0	0.0
1	Mon Sep 09 03:10:13 +0000 2019	43	163	'In the Albanian capital of Tirana, the countr...	'In the Albanian capital of Tirana, the countr...	115.0	0.0	0.0	0.0
2	Mon Sep 09 02:50:02 +0000 2019	22	97	'The 17 Democratic candidates who are lagging ...	'The Democratic candidates who are lagging beh...	180.0	0.0	0.0	0.0
4	Mon Sep 09 02:32:03 +0000 2019	247	531	"President Trump's plan to secretly meet with ...	"President Trump's plan to secretly meet with ...	154.0	0.0	0.0	1.0
8	Mon Sep 09 01:24:51 +0000 2019	116	621	"For Paul Eng, trying to recreate his family's...	"For Paul Eng, trying to recreate his family's...	277.0	0.0	0.0	0.0
9	Mon Sep 09 01:11:58 +0000 2019	601	2934	'Breaking News: Rafael Nadal has defeated Dani...	'Breaking News: Rafael Nadal has defeated Dani...	143.0	0.0	0.0	0.0
10	Mon Sep 09 01:01:23 +0000 2019	251	475	'In Fashionopolis, Dana Thomas, a veteran styl...	'In Fashionopolis, Dana Thomas, a veteran styl...	272.0	0.0	0.0	0.0
11	Mon Sep 09 00:30:04 +0000 2019	36	233	'This flexible soba noodle dish also works wel...	'This flexible soba noodle dish also works wel...	141.0	0.0	0.0	0.0
12	Mon Sep 09 00:00:03 +0000 2019	311	1005	'A dramatic revolt by British Conservative Par...	'A dramatic revolt by British Conservative Par...	281.0	0.0	0.0	1.0
13	Sun Sep 08 23:15:20 +0000 2019	44	185	'The bag is to Berlin what a New Yorker tote i...	'The bag is to Berlin what a New Yorker tote i...	230.0	0.0	0.0	0.0

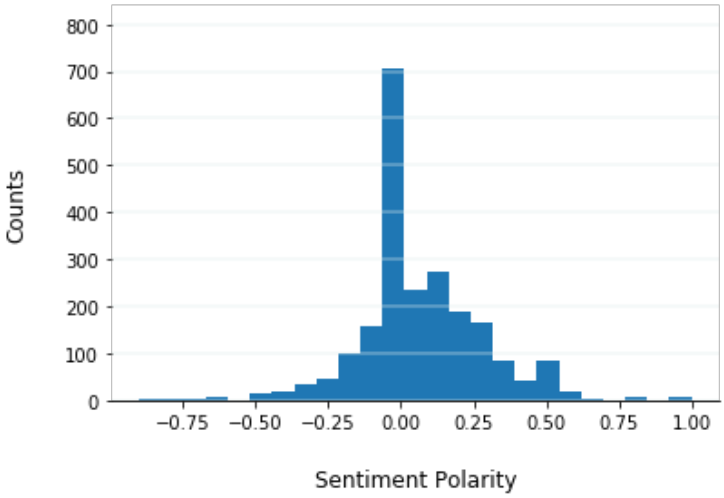
5.3.2 Latent Content Features

Sentiment score

The sentiment score here means sentiment polarity score of a sentence. The sentiment property of Python’s *TextBlob* package can return both the polarity score and subjectivity score of

a sentence. The polarity score is a float within the range [-1.0, 1.0] where 1.0 is very positive and -1.0 is very negative. A new variable called "sentiment" shows sentiment polarity score of "clean_text".

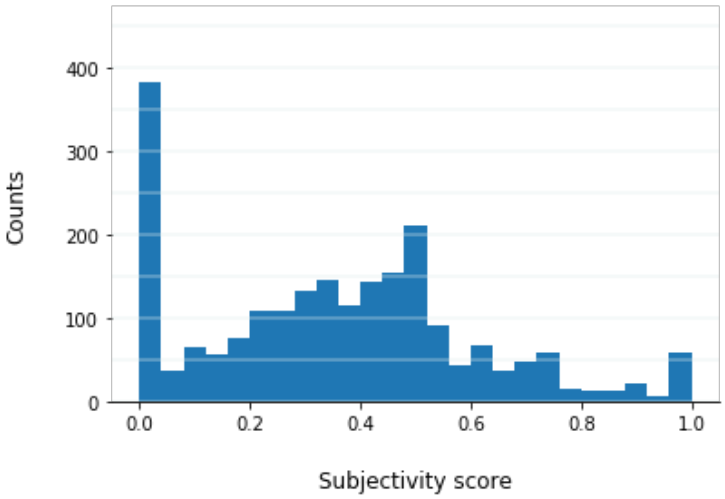
Figure 10: *Distribution of "sentiment"*



Subjectivity score

As mentioned above, the sentiment property of Python's *TextBlob* package can return both the polarity score and subjectivity score of a sentence. The subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective. A new variable called "subjectivity" shows subjectivity score of "clean_text".

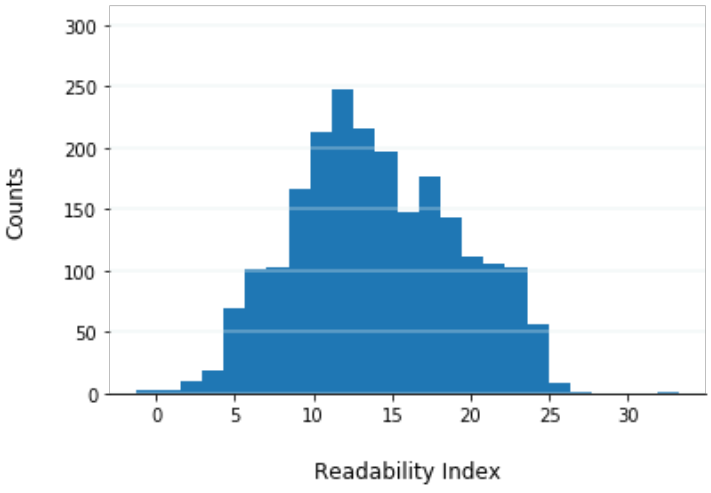
Figure 11: *Distribution of "subjectivity"*



Readability Index

To identify the readability of a tweet text, Python's *textstat* package is used to give a "automated_readability_index" (ARI) for each "clean_text". The ARI index approximates the grade level needed to comprehend the text. For example, if the ARI is 6.5, then the grade level to comprehend the text is 6th to 7th grade. A new variable called "readability" shows readability index of "clean_text".

Figure 12: *Distribution of "readability"*



TF-IDF score of terms

TF-IDF (term frequency–inverse document frequency) is the product of TF (term frequency) and IDF (inverse document frequency). Every term in a document can be given a TF-IDF score (also called TF-IDF weights in some other articles) to represent its frequency in the document relatively to its frequency in the corpus. For example, if a term appears frequently in one document and rarely appear in other documents in the corpus, its TF-IDF score in that document will be very high. In other words, it is a statistic measure which determines the importance of a term in a document relatively to its importance in the whole corpus. Examining the TF-IDF scores of every term in every cleaned tweet text is essential in the analysis because terms are the smallest unit of document contents, utilizing terms can help this study make the most of the cleaned tweet text and also reveal some important content information of popular tweets.

$$TF_{t,d} = \frac{Number\ of\ times\ term\ t\ appears\ in\ document\ d}{Total\ number\ of\ terms\ in\ document\ d}$$

$$IDF_t = \log \frac{Number\ of\ documents\ in\ corpus}{Number\ of\ documents\ with\ term\ t}$$

$$TF - IDF = TF_{t,d} \times IDF_t$$

To ensure that this analysis will get the TF-IDF scores of meaningful terms, the calculation of TF-IDF scores of terms should be based on fully cleaned text. Extra text splitting and cleaning is performed here: Firstly, every tweet text is splitted into a list of words; Secondly, punctuation, stop words and non-alphabetic words are removed. Thirdly, every word in the list is lowercased and stemmed. Lastly, the list of words are combined into a sentence again. This time it is a fully-cleaned sentence.

Looping over the "clean_text" column of the data frame gives us a list of fully-cleaned sentences which is the corpus in our analysis. Then Python's *TfidfVectorizer* function in the *sklearn* package is utilized to get the TF-IDF score of every term in every document.

Figure 13: *Excerpt of the dataset with all the content features (2200 rows and 6830 columns)*

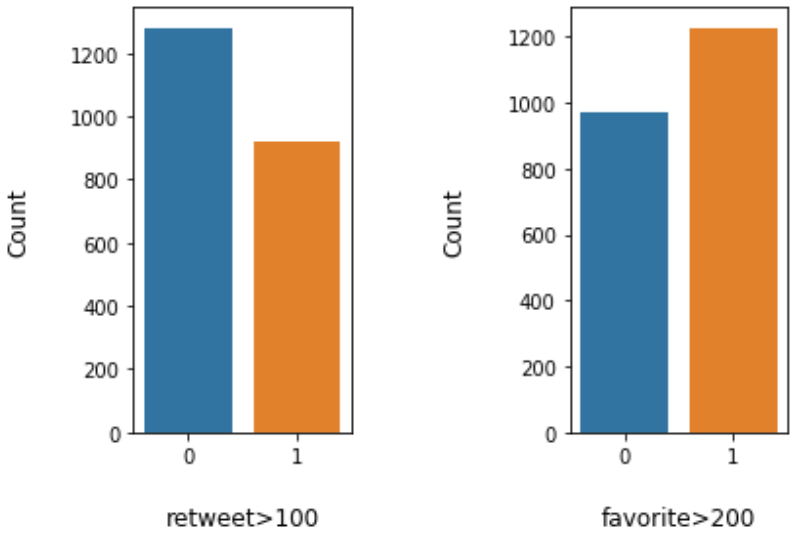
	created_at	retweet_count	favorite_count	text	clean_text	length	hashtag	mention	president	sentiment	...	zest	zhang	zimbabw
0	Mon Sep 09 03:37:47 +0000 2019	36	152	'The good, the bad, and the ugly side of havin...	'The good, the bad, and the ugly side of havin...	105.0	0.0	0.0	0.0	-0.233333	...	0.0	0.0	0.0
1	Mon Sep 09 03:10:13 +0000 2019	43	163	'In the Albanian capital of Tirana, the countr...	'In the Albanian capital of Tirana, the countr...	115.0	0.0	0.0	0.0	0.216667	...	0.0	0.0	0.0
2	Mon Sep 09 02:50:02 +0000 2019	22	97	'The 17 Democratic candidates who are lagging ...	'The Democratic candidates who are lagging beh...	180.0	0.0	0.0	0.0	-0.050000	...	0.0	0.0	0.0
3	Mon Sep 09 02:32:03 +0000 2019	247	531	"President Trump's plan to secretly meet with ...	"President Trump's plan to secretly meet with ...	154.0	0.0	0.0	1.0	-0.100000	...	0.0	0.0	0.0
4	Mon Sep 09 01:24:51 +0000 2019	116	621	"For Paul Eng, trying to recreate his family's...	"For Paul Eng, trying to recreate his family's...	277.0	0.0	0.0	0.0	0.166667	...	0.0	0.0	0.0
5	Mon Sep 09 01:11:58 +0000 2019	601	2934	'Breaking News: Rafael Nadal has defeated Dani...	'Breaking News: Rafael Nadal has defeated Dani...	143.0	0.0	0.0	0.0	0.325000	...	0.0	0.0	0.0
6	Mon Sep 09 01:01:23 +0000 2019	251	475	'In Fashionopolis, Dana Thomas, a veteran styl...	'In Fashionopolis, Dana Thomas, a veteran styl...	272.0	0.0	0.0	0.0	0.140000	...	0.0	0.0	0.0
7	Mon Sep 09 00:30:04 +0000 2019	36	233	'This flexible soba noodle dish also works wel...	'This flexible soba noodle dish also works wel...	141.0	0.0	0.0	0.0	0.200000	...	0.0	0.0	0.0
8	Mon Sep 09 00:00:03 +0000 2019	311	1005	'A dramatic revolt by British Conservative Par...	'A dramatic revolt by British Conservative Par...	281.0	0.0	0.0	1.0	-0.083810	...	0.0	0.0	0.0
9	Sun Sep 08 23:15:20 +0000 2019	44	185	'The bag is to Berlin what a New Yorker tote i...	'The bag is to Berlin what a New Yorker tote i...	230.0	0.0	0.0	0.0	0.151515	...	0.0	0.0	0.0

5.4 Dependent Variable Transformation

As stated in part 3 above, this paper will use both the count of retweets and the count of favorites as indicators of news tweet popularity, so in the regression models, "retweet_count" and "favorite_count" will both serve as dependent variables.

In addition, continuous dependent variables must be transformed into categorical variables to serve as class labels in classification prediction models. It is crucial to set two proper thresholds of count of retweets and count of favorites to identify a new tweet is "popular" or not. My goal is to set thresholds that are not only reasonable, but also result in an approximately balanced dataset. As the median of "retweet_count" and "favorite_count" are 81.5 and 228.5 respectively, the thresholds are set to be 100 and 200 respectively. Dependent variable "retweet>100" is coded "1" when retweet_count >100, otherwise "retweet>100" is coded "0". Similarly, "favorite>200" is coded "1" when retweet_count >200, otherwise "favorite>200" is coded "0".

Figure 14: *Distribution of "retweet>100" and "favorite>200"*



5.5 Descriptive Statistics

My final dataset ready for modeling is a data frame with 2200 rows and 6832 columns, there are 4 dependent variables ("retweet_count", "favorite_count", "retweet>100" and favorite>200"), 6825 independent variables. Of the 6825 independent variables, 7 are named features: length, hashtag, mention, president, sentiment, subjectively and readability, the others are TF-IDF scores of terms.

Table 2: Summary Statistics

Variable	Number of Observations	Mean	Standard Deviation	Min	Max
length	2200	206.78	53.85	38	316
hashtag	2200	0.01	0.10	0	1
mention	2200	0.07	0.32	0	6
president	2200	0.14	0.35	0	1
sentiment	2200	0.07	0.21	-0.9	1.0
subjectivity	2200	0.35	0.25	0.0	1.0
readability	2200	14.04	5.20	-1.2	33.2
retweet_count	2200	263.98	961.97	1	24406
favorite_count	2200	765.41	3326.01	0	87095
retweet>100	2200	0.42	0.49	0	1
favorite>200	2200	0.56	0.50	0	1

5.6 Feature Selection

It is widely known that having too many predictor features compared with the number of training samples can deteriorate prediction models performance, both for regression and classification problems (Hsieh 1998; Allen 1974). Various adjustments or measures can be applied to mitigate the influence.

For linear regression models, lasso regression can be used to obtain the subset of predictor variables. Lasso does this by penalizing on the model parameters and that causes some parameters to shrink to zero.

For classification models, an automatic feature selection technique called "Recursive Feature Elimination" is utilized in this analysis. The algorithm starts with full model and runs series of models that evaluate prediction error on ytrain after dropping a feature at each time. The least important features are pruned from current set of features. That procedure is recursively repeated on the pruned set until the desired number of features to select is eventually reached. The dimension of the dataset is then reduced to 2200 X 1000 in this analysis.

5.7 Imbalanced Data and Cross Validation

Although reaching a balancing sample is taken into consideration when choosing the classification barrier of dependent variables, "retweet_count =100" and "favorite_count=200" is still slightly different from the median of "retweet_count" and "favorite_count", which resulted in a slightly imbalanced sample. To maximize accuracy and reduce error, oversampling method is used to balance the dataset and cross validation is used to measure the accuracy.

Synthetic minority oversampling technique (SMOTE) is chosen as the imbalanced technique in this analysis. Different from the ordinary oversampling methods that increase the size of the minority class by randomly repeating minority class samples, SMOTE takes into account the correlations of the features between data points and the result accuracy for classification models can be reliable (El-Sayed 2016).

k-fold cross validation can be divided into several steps: (1) Split the original sample into k equal sized sub samples (2) Choose one of the subsamples as a test set, and the remaining (k-1) subsamples as training set, the model is fit to the training set and predictive accuracy is assessed

using the test set, thus get the cross validation score(or other validation result) (3) Choose another subsample as the test set and repeat this process, until every subsample serve as the test set, average the scores and get the mean cross validation score (or other validation results). k-fold cross validation is used for providing a less-biased estimate of model prediction performance. The reason is that it makes sure that every observation in the original dataset has appeared in both the training and test set and every observation appeared in the test set only once.

Slightly different from k-fold cross validation, Stratified k-fold cross validation is mostly used in classification problems and ensures that relative class frequencies in each fold reflect relative class frequencies on the whole dataset. This analysis used stratified 10-fold cross validation.

The challenges raised by the joint use of sampling techniques and cross validation has been addressed (Blagus 2015). Blagus emphasized that sampling methods should not be performed on the entire dataset before cross validation, instead, it should be performed only on the training set of each subsampling process during cross validation. Incorrect implementation of cross validation with sampling techniques performed before it will result in overoptimistic model accuracy estimates. SMOTE is performed on each of the 9-fold training set in 10 times of cross validation to generate estimate of prediction accuracy.

6 REGRESSION MODELS

6.1 Linear Regression (OLS)

Linear regression is a simple approach to supervised learning when the dependent variable is continuous, it assumes that the dependence of Y on X_1, X_2, …X_p is linear. The following model is applied to training data to predict count of retweet and count of favorite respectively:

$$\text{retweet\_count / favorite\_count} = \beta_0 + \beta_1 \text{length} + \beta_2 \text{hashtag} + \beta_3 \text{mention} + \beta_4 \text{president} + \beta_5 \text{sentiment} + \beta_6 \text{subjectivity} + \beta_7 \text{readability} + \varepsilon \quad (1)$$

OLS model then uses the least squares fitting procedure where β_0, β_1, β_2,… β_7 are estimated using the values that minimize

$$\text{RSS} = \sum_{i=1}^{n} (\gamma_i - \beta_0 - \sum_{j=1}^{7} \beta_j x_{ij})^2$$

The result after fitting the models is reported in table 3 below.

Table 3: OLS regression results of training set

OLS Regression Models	**Model 1** **Predict retweet_count**	**Model 2** **Predict favorite_count**
length	-0.1305 (0.541)	-0.7034 (1.848)
hashtag	-119.4082 (242.249)	-345.1399 (826.687)
mention	-73.9959 (74.848)	-202.1338 (255.523)
president	479.5068*** (74.443)	1193.6604*** (254.040)
sentiment	89.3043 (124.216)	489.9320 (423.891)
subjectivity	1.2293 (105.068)	11.7890 (358.547)
readability	1.3400 (5.585)	-7.3155 (19.059)
_cons	215.2877** (107.016)	841.2837** (365.197)
adj_R squared	0.027	0.015
N	1650	1650

Result of the OLS regressions show that when predicting popularity with the 7 content features, the adjusted R square values are very low, both for predicting count of retweet and count of favorite, indicating that these 7 features don't have much predicting power in this regression model. What's more, "president" is the only feature that has a statistically significant relationship with "retweet_count" and "favorite_count", controlling for length, number of hashtags, number of mentions, sentiment polarity, subjectivity and readability score, a tweet that contains "Trump" will receive about 480 more retweets and 1994 more favorites compared with a tweet that does not contain "Trump".

6.2 Lasso Regression

Lasso is a technique that constrains the coefficient estimates and shrinks some of them towards zero. To test the linear model performance on all the content features (7 named content features + 6818 TF-IDF score features), this analysis fit a model containing all 6825 features and use Lasso for feature selection. The original model is as below:

$$\text{retweet\_count / favorite\_count} = \beta_0 + \beta_1 \text{length} + \beta_2 \text{hashtag} + \beta_3 \text{mention} + \beta_4 \text{president} + \beta_5 \text{sentiment} + \beta_6 \text{subjectivity} + \beta_7 \text{readability} + \beta_8 \text{TFIDF\_term1} + \beta_9 \text{TFIDF\_term2} + \ldots \beta_{6825} \text{TFIDF\_term6818} + \varepsilon \quad (2)$$

Different from simple OLS regression, the lasso coefficients minimize:

$$\sum_{i=1}^{n} (\gamma_i - \beta_0 - \sum_{j=1}^{6825} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{6825} |\beta_j| = \text{RSS} + \lambda \sum_{j=1}^{6825} |\beta_j|$$

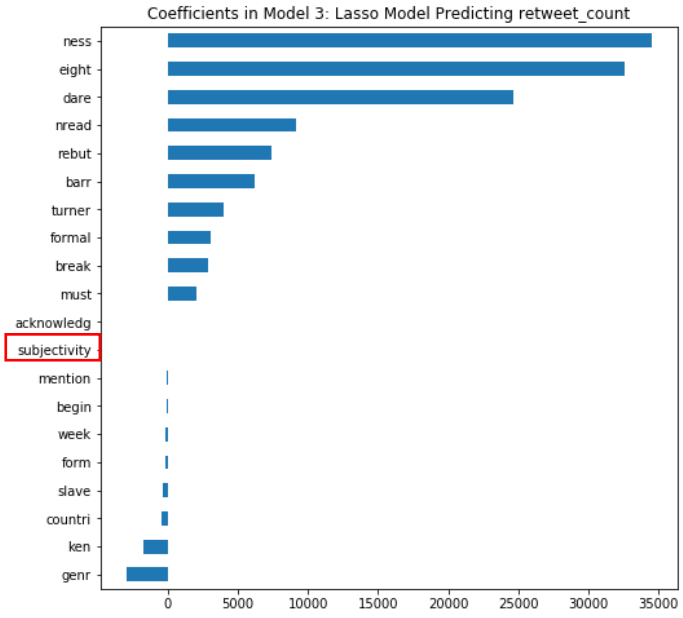
The following table 4 reports prediction result of the 2 lasso models.

Table 4: Lasso regression results

Lasso Regression Models	Model 3 Predict retweet_count	Model 4 Predict favorite_count
Training set score	0.66	0.88
Test set score (R^2)	0.21	-0.01
Number of coefficients used	47	198

As model 4 is performing extremely bad, even worse than random guessing, further coefficient analysis is only applied for model 3. Figure 15 visualizes top coefficients in model 3.

Figure 15: *Coefficients in model 3*



Content features that are not TF-IDF scores is marked out to differentiate. In figure 15, top 10 positive coefficients and top 10 negative coefficients are plotted out. Most of the top features are TF-IDF scores, only "subjectivity" stands out as an exception, it has a negative coefficient in predicting count of retweets.

7 CLASSIFICATION MODELS

7.1 Baseline

Considering that the data is imbalanced, the baseline accuracy is the probability of correctly predicting the class label by guessing all the data points as the majority class:

Table 5: Baseline prediction accuracy (10-fold cross validation)

Baseline Models	Class	Percentage	Baseline Accuracy
Predict retweet>100	0	59%	0.59
	1	41%	
Predict favorite>200	0	44%	0.56
	1	56%	

7.2 Logistic Regression

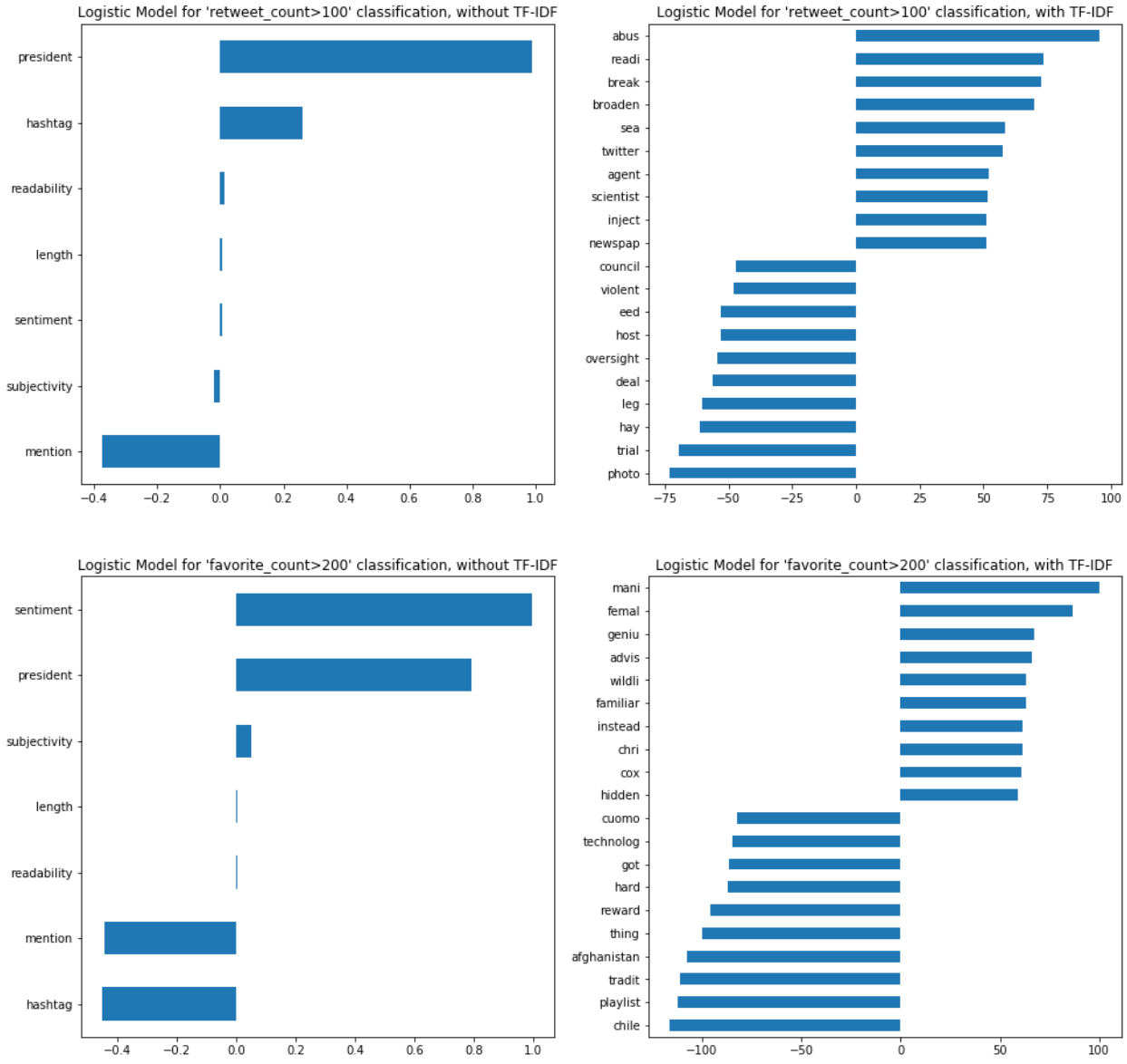
The first predictive model used to perform binary classification is logistic regression. It predicts output class labels by first generating the probability that output is 1, which is P(x) in the following formula:

$$P(x) = \frac{1}{1+e^{-(\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p)}}$$

When P(x) >= 0.5, the predicted output is 1, when P(x) < 0.5, the predicted output is 0. The parameters are then estimated by maximum likelihood.

Logistic model containing all the content features and using "Recursive Feature Elimination" for feature selection reached a 10-fold cross validation accuracy score of 0.90 for "retweet>100" and 0.85 for "favorite>200", both the accuracy estimates are much higher than the baseline model. The following figure plots the important features according to coefficients.

Figure 16: *Coefficients of top features in logistic regression*



7.3 KNN Classification

K nearest neighbors is a simple algorithm for classification problems, it makes prediction with new test observations by calculating observation's similarity to all observations in training data using distance measurement. After finding k nearest neighbors in training data, it takes majority vote from these neighbors and assign the category to observation. The formula for the distance is:

$$d(x, x') = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2 \ldots + (x_n - x_n')^2}$$

To define the "k" parameter, figure 17 is plotted to illustrate how the model accuracy changes along with k.

Figure 17: *Accuracy of KNN models*

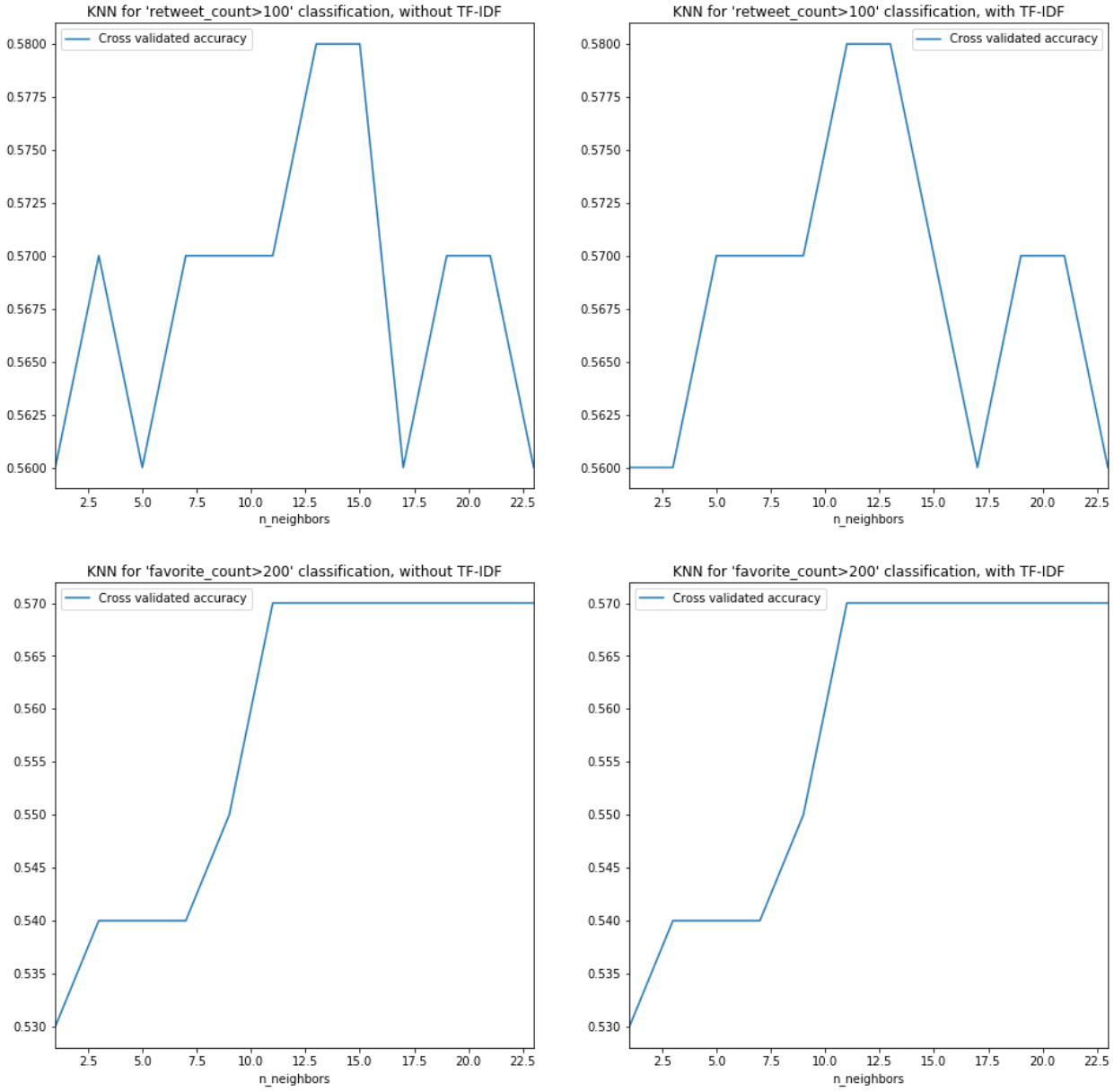


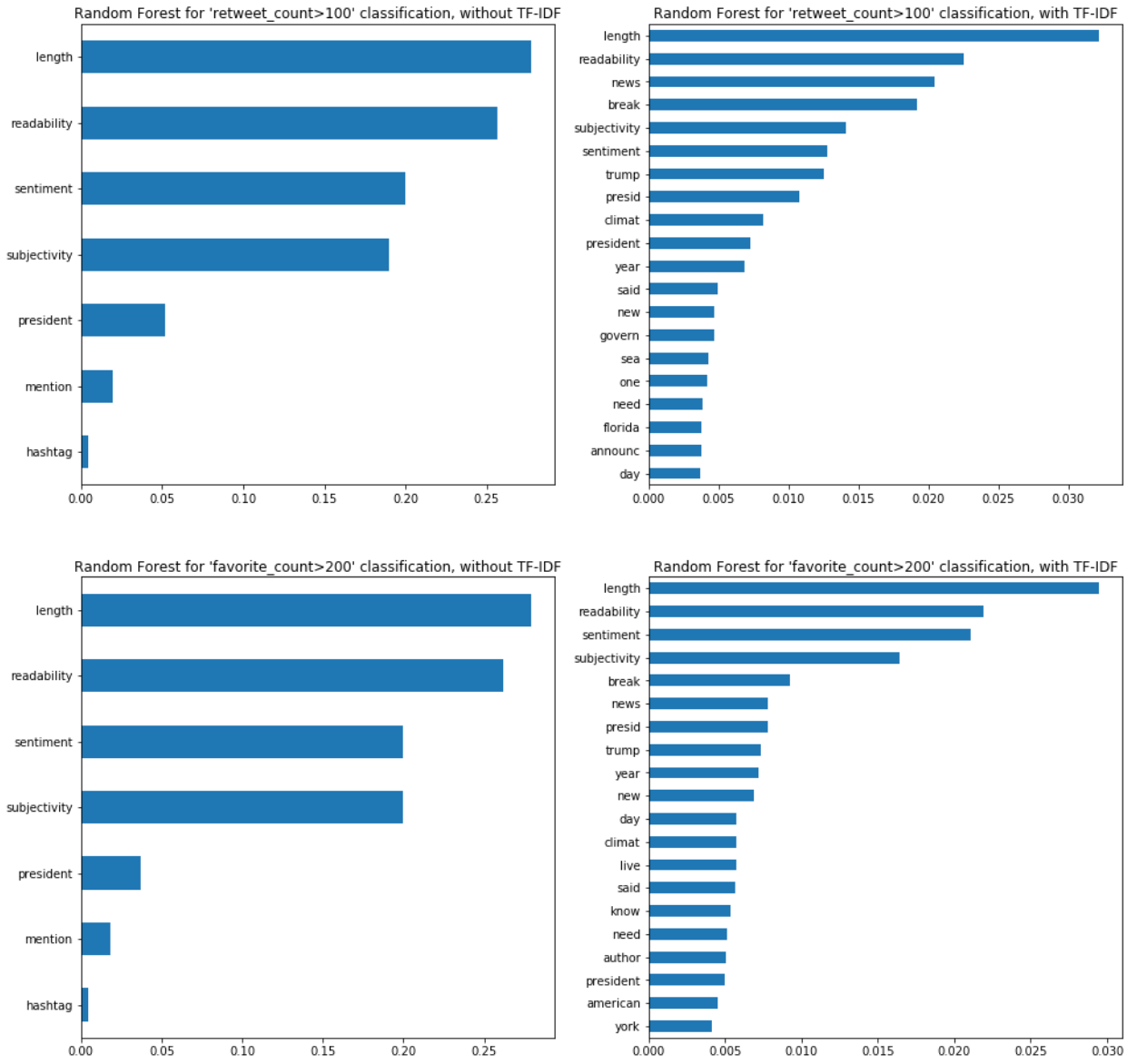
Figure 17 shows that the highest accuracy score for predicting “retweet>100” is 0.58, and for predicting “favorite>200” is 0.57, they are similar to the baseline model accuracy scores.

7.4 Random Forest Classification

Random forest is a popular classification algorithm that has proved high-accuracy performance and at the same time returns measures of feature importance. It uses an ensemble of classification trees and allow them to vote for the most popular class. Bagging is used to create a bootstrap sample of the data for tree building and random variable selection is performed during each tree building process. An important parameter to be chosen is “n_estimators”, which is number of trees to generate for the model. This analysis used n_estimators=20,50,100,150,200, and picked the highest accuracy rate for each of the 4 models and produced plots that show feature importance of top features.

Random forest model containing all the content features and using “Recursive Feature Elimination” for feature selection reached a 10-fold cross validation accuracy score of 0.69 for “retweet>100” and 0.64 for “favorite>200”, both the accuracy estimates are higher than the baseline model.

Figure 18: *Feature importance of top features in random forest models*



8 DISCUSSION

8.1 Results

Three hypotheses were raised at the beginning of this research:

1) Linear prediction model which incorporates the seven features (length, hashtag, mentions, named entity, subjectivity, readability and sentiment) plus TF-IDF features can have

high explanatory power (with an adjusted R square more than 0.5) and some of the features can significantly affect number of retweets and likes.

2) Classification models using the seven features (length, hashtag, mentions, named entity, subjectivity, readability and sentiment) can outperform a naïve approach.

3) Classification models using the seven features (length, hashtag, mentions, named entity, subjectivity, readability and sentiment) plus TF-IDF features can outperform a naïve approach to a large extent.

Table 6: Model results

Regression Models						
	Features		Dependent Variable		Performance	
	Length, hashtag, mention, president, sentiment, subjectivity, readability	TF-IDF scores	retweet _count	favorite _count	R squared	
OLS regression	√		√		0.03	
OLS regression	√			√	0.02	
Lasso regression	√	√	√		0.21	
Lasso regression	√	√		√	-0.01	
Classification Models						
	Features		Dependent Variable		Performance	
	Length, hashtag, mention, president, sentiment, subjectivity, readability	TF-IDF scores	retweet> 100	favorite> 200	Cross validated Accuracy	Baseline Accuracy
Logistic regression	√		√		0.61	0.59
Logistic regression	√	√	√		0.90	0.59
Logistic regression	√			√	0.58	0.56
Logistic regression	√	√		√	0.85	0.56
KNN	√		√		0.58	0.59
KNN	√	√	√		0.58	0.59
KNN	√			√	0.57	0.56
KNN	√	√		√	0.57	0.56
Random forest	√		√		0.59	0.59
Random forest	√	√	√		0.69	0.59
Random forest	√			√	0.56	0.56
Random forest	√	√		√	0.64	0.56

According to the "Regression Models" part in table 6, the first hypothesis is not supported. The best linear model only reached a R squared of 0.21 which shows that the features don't have much explanatory power for count of retweets or count of favorites. In addition, the only feature

that has a significant effect on count of rewets and count of favorites is “president”, the effect is strong and positive.

With regard to the second hypothesis, it is proved to be right by logistic regression model for both “**retweet>100**” classification and “**favorite>200**” classification.

Adding TF-IDF scores as features adds to the accuracy of the classification models, the third hypothesis is proved to be right by both logistic regression model and random forest classification model, for both “**retweet>100**” classification and “**favorite>200**” classification.

On top of model performance and examination of the hypotheses, one crucial implication of this study is to identify important features that can predict tweet news popularity. As logistic regression model is the top-performing classification model and the coefficients have positive and negative directions, this analysis relies mostly on figure 16, *Coefficients of top features in logistic regressions*, to get the top features in the prediction model. and summarized 3 main findings and possible explanations:

Finding 1: high frequency of “abuse”, “break”, “scientist” and “newspaper” can lead to higher probability of being retweeted > 100 times, this may because “abuse” is a topic that attracts much public attention, “break” is related to “breaking news” that makes the news stand out in its title and “scientist” and “newspaper” make the news tweet more reliable, so people tend to retweet these news more often.

Finding 2: “president”, which indicates if the news contains “Trump” or not, keeps to be a quite important feature either at predicting “retweet>100” or “favorite>200”, this is consistent with the finding from linear regression.

Finding 3: “sentiment” plays a much larger positive role in affecting the probability of being favorited > 200 times than in affecting the probability of being retweeted > 100 times, this

is easy to interpret because people may be more willing to favorite those tweets that express positive feelings.

8.2 Conclusion and Future Work

Twitter plays an essential role in today's social network and also serves as an important social media platform for news to spread. This paper aims to research news popularity on Twitter basing on news content, the analysis first scraped tweet news text together with their count of retweets and favorites from New York Times' twitter account and then examined content-based features of news by dividing it into two categories: obvious content features (length, hashtag, mentions and named entities) and latent content features (sentiment, simplicity, readability and TF-IDF of terms). At last, regression models and classification models are performed to predict retweet/favorites using these content-based features.

To conclude, **content-based features** (length, hashtag, mentions, named entities, sentiment, simplicity, readability and TF-IDF of terms) have essential predictive value for classifying popular tweets and unpopular tweets if the classification barrier is set to be **retweet count = 100** and **favorite count = 200**. Results showed that this analysis achieved an overall accuracy of 0.90 using Logistic Regression as the classifier. However, these features have little explanatory power if directly predicting the exact count of retweets and count of favorites, indicated by the inadequate regression results.

It is also noted that overall "TF-IDF of terms" performs better than other content-based features, which reveals that specific words may catch audience attention and influence the popularity of the news in a larger extent than the other index calculated in this paper (like sentiment, simplicity, readability).

The result of this research can be utilized by news agencies which have accounts on Twitter to help promote news content. For supernodes on Twitter like New York Times, their number of followers, lists and sources are almost constant, this study tells that there is still much to do to increase their online news popularity by adjusting the content of news. As we know "New York Times" has its newspaper and online news website, the twitter platform is a supplement to interact with their audience and do marketing. When they choose what to put in their twitter account and how to summarize it to a tweet, there is an "art of repackaging". According to the findings listed above, it may be a good idea to try add "breaking news" at the beginning to attract more retweets; Also, news tweets that covers popular politic and social topics should be prioritized in the selection process; Positive sentiment in the content of new tweets will help increase number of "favorites".

Due to the time restrict of this project, the data collection process cannot last more than one month so the sample size is relatively small. Moreover, the "Plus One Day" scraping collection mechanism used here can only get an estimate of final counts of retweet and likes, which may result in bias in the prediction analysis. In addition, future study may consider the content category of the news to add more power to the prediction model.

Bibliography

Allen, D.M. (1974). The relationship between variable selection and data agumentation and a method for prediction, *Technometrics*, *16*(1), 125-127. http://doi.org/10.1080/00401706.1974.10489157

Bakshy, E., Hofman, J. M., Mason, W. A., & Watts, D. J. (2011). *Everyone's an influencer*. 65. https://doi.org/10.1145/1935826.1935845

Blagus, R., Lusa, L. (2015) Joint use of over- and under-sampling techniques and cross-validation for the development and assessment of prediction models. *BMC Bioinformatics 16*(1), 1-10. http://doi:10.1186/s12859-015-0784-9

El-Sayed, A. A., M. Mahmood, M. A., Meguid, N. A., & Hefny, H. A. (2015). Handling autism

imbalanced data using synthetic minority over-sampling technique (SMOTE). *2015 Third World Conference on Complex Systems (WCCS)*, 1-5. http://doi.org/10.1109/ICoCS.2015.7483267

Hong, L., Dan, O., & Davison, B. D. (2011). Predicting popular messages in Twitter. *Proceedings of the 20th International Conference Companion on World Wide Web, WWW 2011*, 57–58. https://doi.org/10.1145/1963192.1963222

Hong, L., & Davison, B. D. (2010). Empirical study of topic modeling in Twitter. *SOMA 2010 - Proceedings of the 1st Workshop on Social Media Analytics*, 80–88. https://doi.org/10.1145/1964858.1964870

Hsieh, P., & Landgrebe D. (1998). Classification of high dimensional data. *ECE Technical Reports*.52. http://internal-pdf//CLASSIFICATION OF HIGHDIMENSIONAL DATA-3415241872/CLASSIFICATION OF HIGHDIMENSIONAL DATA.pdf

Jenders, M., Kasneci, G., & Naumann, F. (2013). Analyzing and predicting viral tweets. *WWW 2013 Companion - Proceedings of the 22nd International Conference on World Wide Web*, 657–664.

Kuang, L., Tang, X., & Guo, K. (2014). Predicting the times of retweeting in microblogs. *Mathematical Problems in Engineering*, *2014*. https://doi.org/10.1155/2014/604294

Martin, T., Hofman, J. M., Sharma, A., Anderson, A., & Watts, D. J. (2016). Exploring limits to prediction in complex social systems. *25th International World Wide Web Conference, WWW 2016*, 683–694. https://doi.org/10.1145/2872427.2883001

Naveed, N., Gottron, T., Kunegis, J., & Alhadi, A. C. (2011). Bad news travel fast: A content-based analysis of interestingness on twitter. *Proceedings of the 3rd International Web Science Conference, WebSci 2011*. https://doi.org/10.1145/2527031.2527052

Petrovic, S., Osborne, M., & Lavrenko, V. (2011). Rt to win! predicting message propagation in twitter. *Proceedings of the Fifth International Conference on Weblogs and Social Media - ICWSM '11*, 586–589. Retrieved from http://homepages.inf.ed.ac.uk/miles/papers/icwsm11.pdf%5Cnhttp://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/viewPDFInterstitial/2754/3209

Suh, B., Hong, L., Pirolli, P., & Chi, E. H. (2010). Want to be retweeted? Large scale analytics on factors impacting retweet in twitter network. *Proceedings - SocialCom 2010: 2nd IEEE International Conference on Social Computing, PASSAT 2010: 2nd IEEE International*

Conference on Privacy, Security, Risk and Trust, 177–184. https://doi.org/10.1109/SocialCom.2010.33

Szabo, G., & Huberman, B. A. (2010). Predicting the popularity of online content. *Communications of the ACM*, *53*(8), 80–88. https://doi.org/10.1145/1787234.1787254

Trzcinski, T., & Rokita, P. (2017). Predicting Popularity of Online Videos Using Support Vector Regression. *IEEE Transactions on Multimedia*, *19*(11), 2561–2570. https://doi.org/10.1109/TMM.2017.2695439

Wu, B., & Shen, H. (2015). Analyzing and predicting news popularity on Twitter. *International Journal of Information Management*, *35*(6), 702–711. https://doi.org/10.1016/j.ijinfomgt.2015.07.003

Zaman, T. R., Herbrich, R., Van Gael, J., & Stern, D. (2010). Predicting Information Spreading in Twitter. *Proceedings of Computational Social Science and the Wisdom of Crowds Workshop*, 1–4. Retrieved from http://www.herbrich.me/papers/nips10_twitter.pdf