UNIVERSITY OF CALIFORNIA -
LOS ANGELES

SEPARATING TRUTH FROM FICTION

STATS 141

Twitter and the Arab Spring

Authors:
Joshua DERENSKI
Chris DONG
Rita HSU
Amy KE
Yeseul KIM
Sharon XU

Supervisor:
Vivian LEW
Professor:
Maryam ESFANDIARI
Teaching Assistant:
Elliot KANG

March 17, 2016

Contents

Twitter and the Arab Spring

Joshua Derenski, Chris Dong, Rita Hsu, Amy Ke, Yeseul Kim, Sharon Xu

***Abstract*—The prominence of social media as a communication platform and news source is highlighted by its significant role in the dissemination of information during the Arab Spring. This series of protests culminated into violently opposed riots, and were highly publicized and discussed throughout the world via Twitter. In this paper, we analyze a sample of these tweets spanning a twelve day time period, highlighting important implications in the characteristics of users and their tweets. We begin by modeling Twitter data with time series analysis, tracing credibility through the investigation of a prominent rumor. In order to understand the implications of tweet content, we then utilize machine learning based approaches to quantify sentiment, semantic themes, and user credibility. The result of this analysis suggests that Twitter activity during this time period is highly dynamic, and that particularly sensational rumors can spread despite credible, substantiated sources. Significant factors that determine the popularity of a particular tweet lies on a large spectrum, ranging from audiences associated with users to the content of an individual's tweet.**

I. Introduction

THE Arab Spring refers to a series of anti-government protests that erupted into civil wars across Eastern Africa and the Middle East at the end of 2010. This was a time of great turmoil for this area and the rest of the world, and several leaders were forcibly removed from power. One of the most significant of these individuals was former Libyan dictator Muammar Gaddafi. Gaddafi responded to the protests that took place in Libya with brutal force, where mercenaries and loyalists attacked and killed many of those who opposed his regime. Eventually, this leader was removed from power and later killed.

During the course of the revolution, social networking sites such as Twitter played an essential role in organizing these protests and disseminating up-to-date information about the events that transpired. With history as a resource, a sample of tweets during this time period can be analyzed in order to gain insight into reception of these events on social media and the implications behind the public sentiments of users around the world.

Our sample of tweets spans the time period between February 15, 2011 to February 26th, 2011. During this time frame, a series of protests in Africa and the Middle East in areas such as Tripoli and Egypt as well a major speech made by Gaddafi himself sparked rapid discussion across the world through this online medium. Through the unique content and patterns of dispersion of information in our dataset, we aim to analyze not only the content and sentiments behind peoples responses to the events that transpired, but also the underlying implications of credibility that accompany these characteristics across the world.

II. The Data

National Public Radio broadcaster Andy Carvin rose to prominence during the Arab Spring uprising through his documentation of the revolution in the form of live tweets. Using Carvins carefully sourced insights as a reference, the data set was generated on Twitter analytics website TweetReach from queries based on keywords extracted from his tweets.

The dataset is comprised of 2,423,916 Twitter posts over the course of twelve days between February 15 and February 26, 2011. These posts are in the form of original tweets, in which users original content is posted; retweets, where other users share other users original posts; and replies, in which users post original responses to users original posts.

We assume a significant degree of sampling bias in the data, meaning the results from any analysis performed on this data set may not be generalized in any way. Nevertheless, the data presents a unique opportunity to understand the international reception to and the lasting consequences of the Arab Spring, which may stimulate ideas for future studies.

Time series analysis is performed at several points in this report. However, this analysis is based on sample measurements of the variable of interest rather than actual population measurements. Thus, forecasting actual future values is impossible. Nevertheless, because of the magnitude of this particular dataset and the kinds of patterns we have observed, we are confident that the time series of sample measurements we have are qualitatively very similar to what the actual population time series may look like for the chosen phenomena. Thus, the time series analyses in this report will be focused on questions of general forecastability of the phenomena we observe, and on the distinct features present in the data.

In order to prepare for content driven analysis, several steps are taken to clean the data. For sentiment analysis, we utilize M.F. Porter's Snowball stemming algorithm, for instance converting the words “stemming”, ”stemmed”, and “stems” to “stem” in order to reduce repetition. We then convert the words to lowercase for proper matching and consistency. Words like “not” are preserved because they may affect the outcoming sentiment. Arabic code is removed because it can pose problems due to unicode containing Roman letters and numbers. For topic modeling analysis, a similar algorithm is applied to clean for stems. In addition, punctuation and commonly used words in grammar construct are removed, and foreign language content that is ubiquitous in the dataset in this case will be kept.

III. Time Series Modeling

We begin with an overview of the tweet activity during the twelve day time frame, modeling the sample time series for the total hourly number of tweets. Because the date and time corresponding to each tweet is included in the dataset, we aggregate tweets by date and hour. The resulting time series is a sample series due to the total frequencies derived from this dataset originating from a sample total rather than population. However, because this sample is large enough, the time series exhibits traits that are most likely present in that which we would like to observe – the time series of population totals. Thus, it is worthwhile to model this sample time series. We proceed by assessing both the major features of tweet activity in this dataset, attempting to model the sample time series and performing forecasting for this time series.

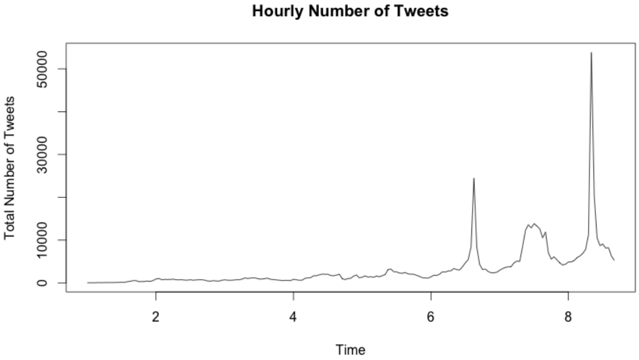
The forecasting is not for predictive purposes, but to assess the forecastability of the phenomenon we are observing. This is a reasonable thing to do with this series because, as mentioned before, the sample series has traits that are undoubtedly present in the population time series. Thus, assessing forecastability of the sample series will give us an idea of how possible it is to predict Twitter activity based on historical activity, and the problems that can arise when trying to forecast a phenomenon as dynamic as such. In particular, since this Twitter data comes from a time in history when significant world events were happening, we will examine forecasting in the face of "shocks", or events which change the behavior of the time series we are observing.

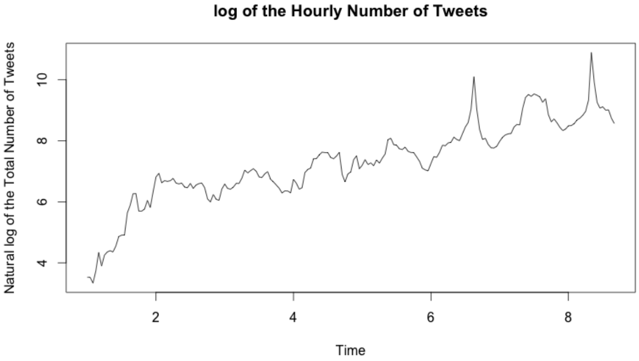
A. The Twitter Data

We examine the sample time series for the total hourly number of tweets present in this dataset. This may give us insight into the events of Arab Spring that generated a lot of social network activity, and how the behavior of the network as a whole evolved during this twelve day period. The first thing that will be noticed in the hourly activity is that there are several substantial peaks of activity in the series, and these peaks correlated with several significant events that happened during this period. The sample series for the daily number of tweets is also presented here, but no modeling will be attempted for this series. Even at the daily level, we see distinct trends in Tweet activity, and at the hourly level we see seasonality.

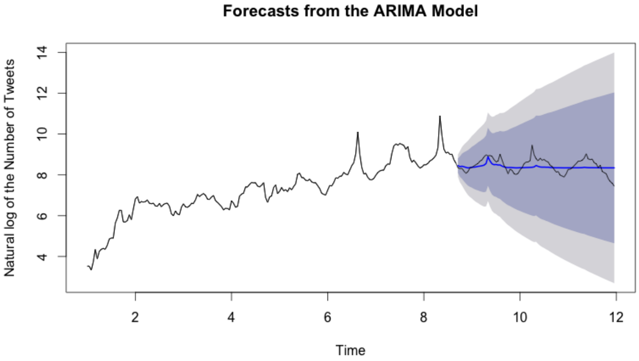
B. Our First Attempt

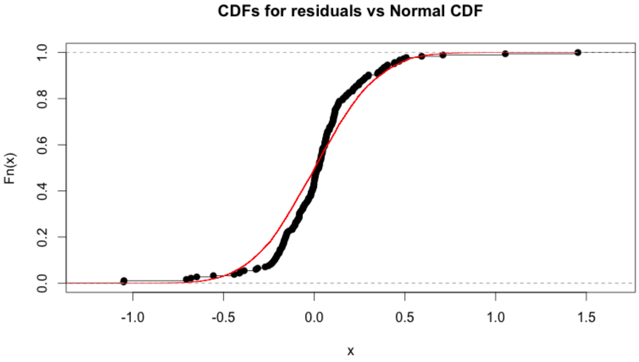
Our first attempt to model this series will be an attempt to predict the last 3 cycles in the series. We will attempt this with an ARIMA model, an exponential model and a naive model. Below are graphs of the sample we have.





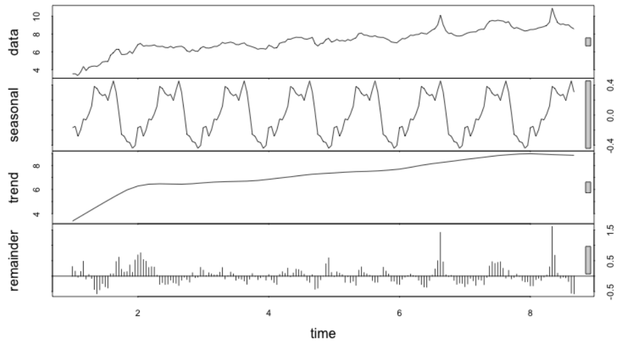
First, we will consider an ARIMA model generated by R Here are the forecasts generated from this model.





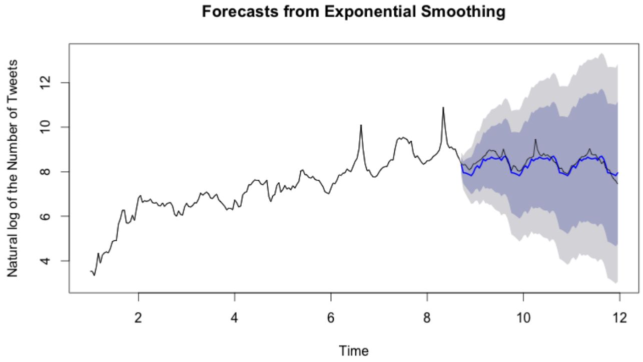
The mean absolute percentage error for the ARIMA model was 0.271. We should also consider the distribution of the residuals for this model. We are comparing the CDF of the residuals to the CDF of a normal distribution with mean 0, and a standard deviation equal to the standard deviation of the residuals.

A second method we will use is called exponential smoothing, which attempts to predict future observations by weighting past observations, and these weights decrease exponentially as we look at observations further and further back in time. To effectively model this time series, we will first decompose it into its trend, seasonal and random components.



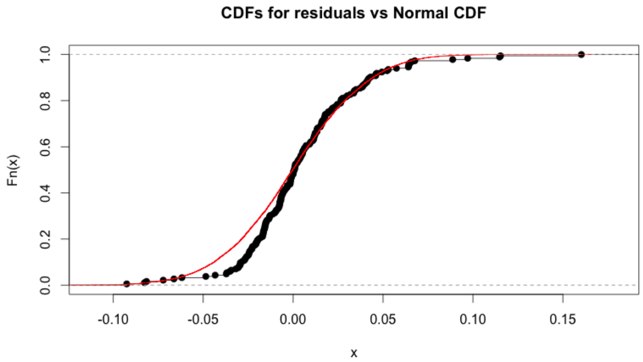
From this decomposition, we assume that there is no trend (this is because it appears that the trend is starting to plateau), we assume an additive seasonality component (because the amplitude is constant), and a multiplicative error term (because the noise is more prevalent in some areas than others).

Here is the forecast made by this model.

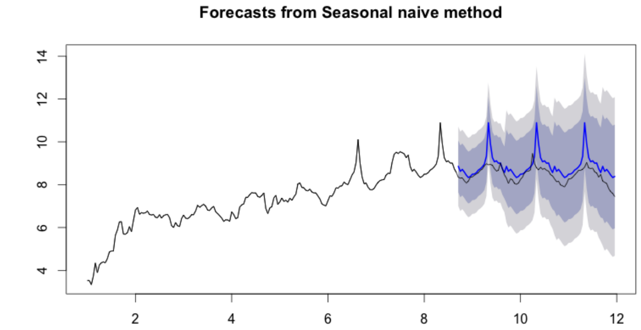


The mean absolute percentage error for the exponential model was 0.18.

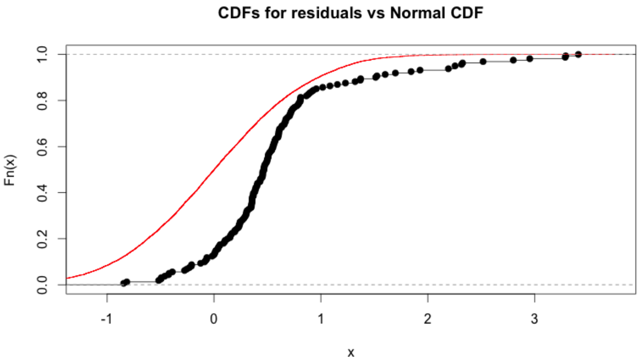
We will again consider the distribution of the residuals for this model, relative to a normal distribution.



Here is our naive model and the forecasts generated. The the mean absolute percentage error for the model was 0.568.



Here is the CDF of the residuals for this model.

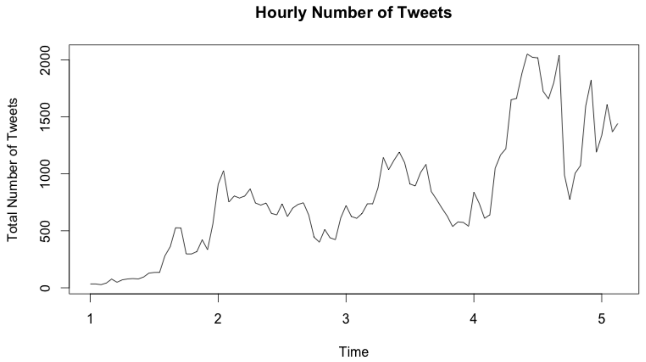


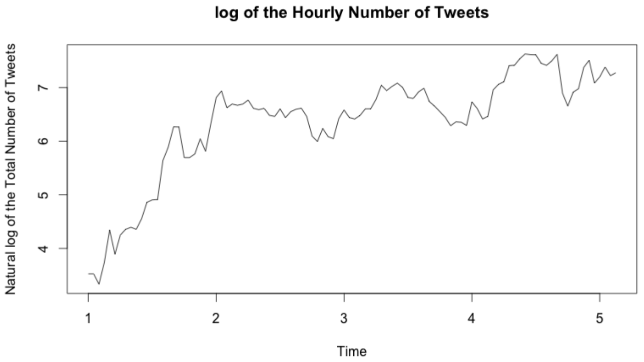
C. Discussion of First Attempt

We can see that exponential smoothing has outperformed our ARIMA model in this case; the reason for this is because exponential smoothing better models the seasonality in the data in this case than the ARIMA model does. Both models outperform the naive model. The reason the naive model performed so poorly is that, in the last cycle available in the same time series, there is a major outlier. It is also worth noting that the residuals for both the ARIMA and exponential model are approximately normally distributed with mean zero, while the residuals for the naive model is not normally distributed. This means that not only are the ARIMA and exponential models more accurate than the naive model, but the prediction intervals from these two models are also more valid than those produced by the naive model.

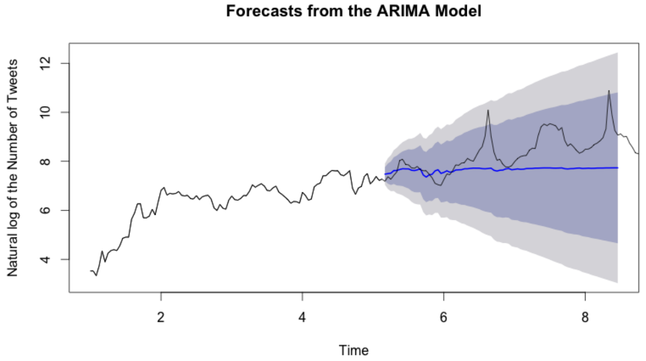
D. Our Second Attempt

In the original time series, we noticed that Twitter activity was highly dynamic during this time period. We might wonder, if we only have the time series of observations before activity really increases, if we can "predict" this increase of activity, and if so, how well. We will attempt this with both an ARIMA model and exponential smoothing. Below is our sample.



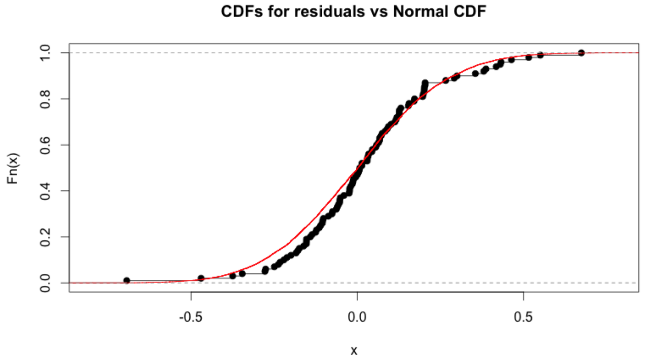


First, we consider an ARIMA model; here are the forecasts generated from this model.

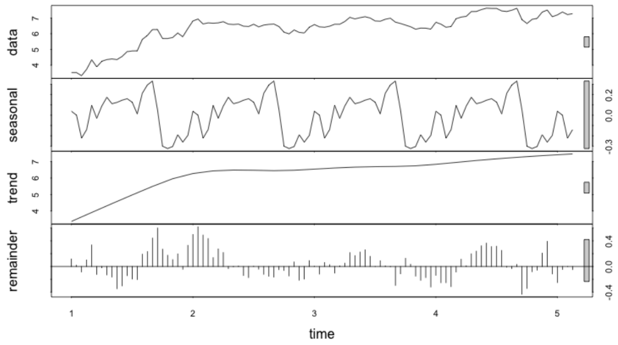


The mean absolute percentage error for the ARIMA model: 0.452.

Here is the CDF of the residuals for this model.

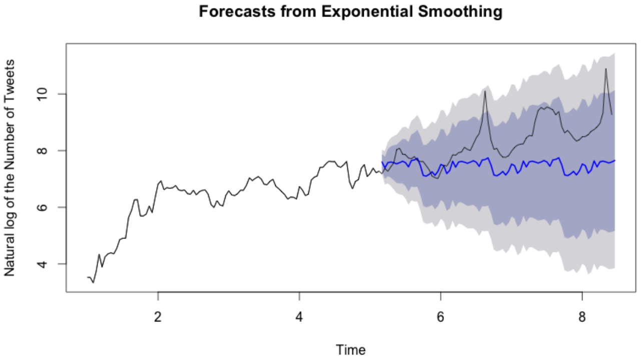


Here we attempt exponential smoothing. To begin, consider a decomposition for this time series.



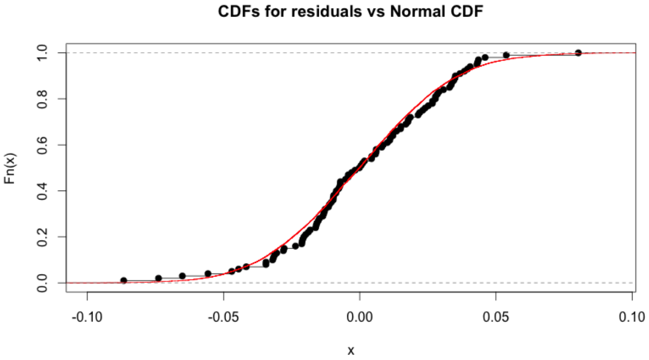
From this decomposition, we assume that there is a linear trend (though it is slight), we assume an additive seasonality component (because the amplitude is constant), and a multiplicative error term (because the noise is more prevalent in some areas than others).

Here are the forecasts generated from this model.

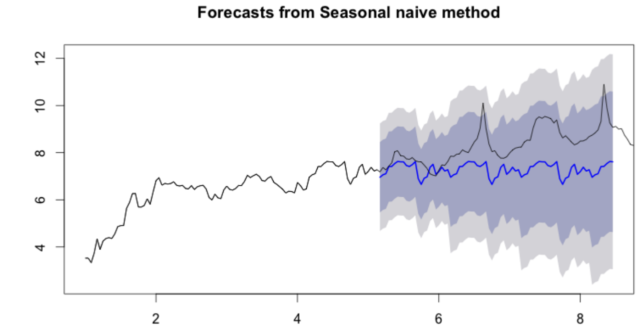


The mean absolute percentage error for the exponential model was 0.515.

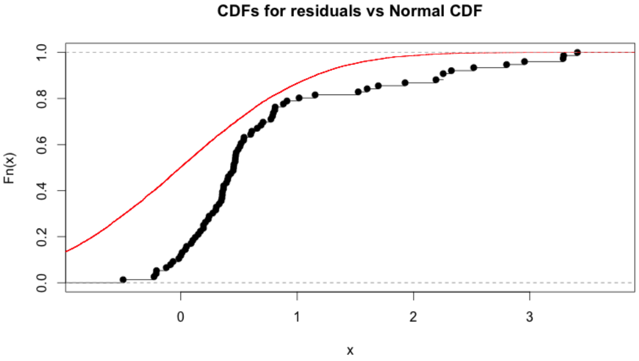
Here is the CDF of the residuals for this model.



Here is our null model, the forecasts generated by the model and the mean absolute percentage error for this model.



The mean absolute percentage error was 0.568. Here is the CDF of the residuals for this model.



E. Discussion of Second Attempt

All in all, every model we tried was less accurate at predicting this part of the series (mainly because this part is more volatile). We see that both the ARIMA and exponential model outperform the naive model, but that the ARIMA model outperforms all the models. This would suggest that exponential smoothing does not always outperform ARIMA and vice versa. In addition, both the ARIMA and Exponential models have residuals that are closely normally distributed, while the naive model has residuals that are not distributed normally. This means that not only are the ARIMA and exponential models more accurate than the naive model, but the prediction intervals from these two models are also more valid than those produced by the naive model. It is worth noting that, while our point forecasts are highly inaccurate, that the increase in activity is actually contained in our 95 % prediction intervals.

F. Discussion of Tweet Activity

The modeling we attempted to do suggests that Twitter activity is highly dynamic. In particular, for this time period, there is a definite overall trend present at the hourly level; first, overall activity increases, and then plateaus over time. In addition, there is a distinct seasonal trend, which suggests that Twitter activity is seasonal from day to day. Thus, this modeling also suggests that it is indeed possible to forecast tweet activity, based on historical activity.

While the results of our modeling definitely suggest the possibility of successful forecasting, there is also evidence that the features we model for forecasting are, in a sense, unstable. For example, while there is seasonality in the tweet time series, we also notice that the amplitude of this seasonality is not constant. Indeed, this aspect of tweet activity would appear to be highly associated with when significant events occur. Thus, any one world event has the potential to completely alter the behavior of this part of the time series, rendering forecasting difficult. Any overall trends in activity will also be subject to similar issues; a major event or catastrophe could cause a massive overall increase in Twitter activity, rather than just a momentary increase may just destabilize the seasonality of the time series.

However, if a major event is anticipated, or it is known that it will occur in the future, then the change in Twitter activity may be more gradual, thus making forecasting of Twitter activity easier, since time series analysis accounts for not only the current behavior of a phenomenon, but also how the behavior of the phenomenon is changing. However, sudden, significant events are much harder to account for, since these events will cause a “shock” in the activity, and the behavior of the time series may or may not return to what historical activity has been, depending on the specific nature of the event. All in all, our attempts at modeling the overall tweet activity in this dataset suggest that, while forecasting of activity from historical data is certainly possible, that our point forecasts can be misleading due to the fact that Twitter activity is very sensitive to ’shocks’ that can destabilize the time series of activity, rendering point forecasts highly inaccurate. Thus, rather than considering only these forecasts, we should also consider our prediction intervals for these forecasts when trying to make statements about future tweet activity.

IV. Rumor Propagation

A significant area of interest in analyzing tweet content is how to discern credibility as news diffuses across social media. Intuitively, the propagation of a rumor through time will differ from that reflecting a true event as it is discredited and revealed to be false. In order to test this hypothesis, we begin by using regular expressions to extract sets of tweets using keywords, comparing discussions across the web of two major topics within our time frame.

One of the most prominent rumors during the Arab Spring claimed that Gaddafi had fled Libya to Venezuela under long time political ally, Hugo Chavez. Propagated through social media throughout the entirety of our time frame, this rumor first arises from speculation and is later perpetuated through external news sources. Within this dataset, we extract all tweets containing the keywords ’Muammar Gaddafi’ and ’Venezuela’, with flexibility in spelling to account for colloquialism and misspelling. As a basis for comparison, we extract a separate data set of tweets containing keywords ’Tripoli’ and ’police’ in reference to a series of major protests that occurred in Green Square, Tripoli resulting in hundreds of deaths as police opened fire on protesters. For each of these datasets, we extract original tweets and retweets or replies separately using the same method.

A comparison of time series in Figure 1 and Figure 2 reveals similarities between both data sets as well as to that

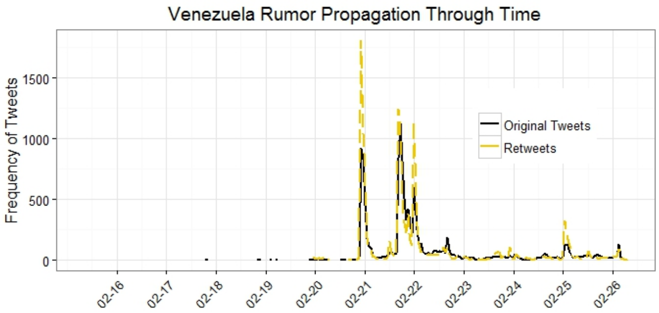


Fig. 1: Time series of Venezuela tweets.

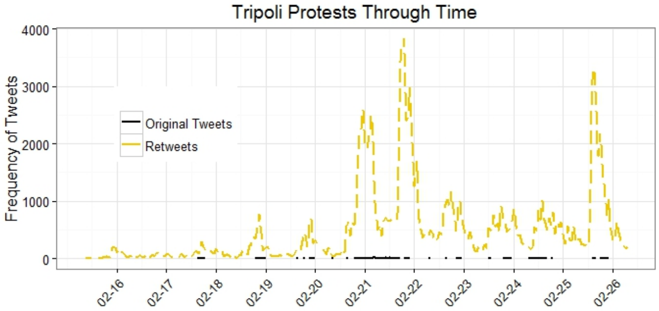


Fig. 2: Time series of Tripoli Protest tweets.

generated above for the original data set. This recurring pattern suggests a high correlation in tweet frequency to major events during the time. Therefore, to understand the implications of the tweets surrounding these topics and the resulting degree of credibility attached, we focus closely on analyzing the tweet content within the Venezuelan rumor time series through segmentation and sampling. We first begin by segmenting the time series into 16 distinct segments of peaks and plateaus based on tweet frequency broken down by hour, and conduct a simple sample size calculation determining the amount of tweets necessary to obtain a representative sample with a bound of error of 0.1 on the proportion of rumors and truths.

Within each of these samples, we translate and categorize the content of each tweet as detailed in Table 1.

TABLE I: Categorization of Sample Tweets

Category	Description
1	States rumors as truth or cites credible sources that perpetuate rumor
2	Speculates about rumors but neither confirms nor denies validity
3	Unrelated to the specific target rumor
4	Discredits rumor as false or cites other sources that discredit rumor

Figure 3 shows a visualization of tweet credibility over our time frame tracking the changes in proportion of tweets falling under categories 1 and 2 that perpetuate the rumor either through speculation or statement of fact–categorized as ’Rumor’–and those that refute the rumor–categorized as ’Truth’.

In the time frame of the data set, peaks 2 and 3 correspond to two major events that sparked controversy over Gaddafi’s intentions and whereabouts. Interestingly, several counter-intuitive trends are reflected across the Venezuelan tweets as a result of these events, offering insight into the idiosyncratic nature in the way people tend to respond to major events across the web through social media.

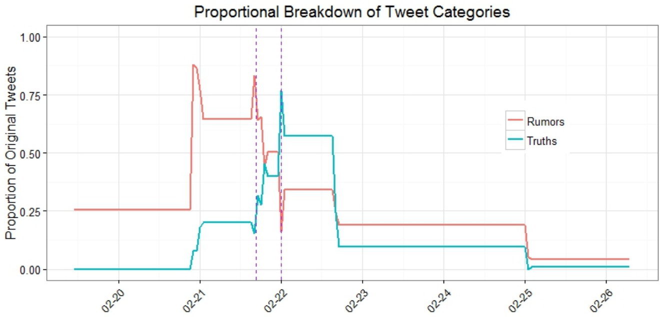


Fig. 3: Time series of tweet credibility.

February 21, 5:30 PM UTC: Foreign Secretary William Hague announces his possession of information indicating that Gaddafi is on his way to Venezuela.

“I have seen some information that suggests he is on his way there at the moment.” [1]

This first major event acts as a substantiation of the rumor as a truth. The proportional distribution of rumor and truth during this event shows, as expected, an overall spike in the frequency of tweets as well as a peak in the discussion of rumors. However, this announcement of a rumor as a fact instigates an upward trend of truth propagation – in this case referring to discussions discrediting the rumor. This suggests that this seemingly credible source in fact instigates a trend of more refutation across users, who begin to cite Venezuelan officials denying the legitimacy of the rumor.

February 22, 12:00 AM UTC: Gaddafi appears on TV broadcast ensuring that he is in Tripoli. [2]

“I am in Tripoli and not in Venezuela. Don’t believe those misleading dog stations.”

This second major event discredits the rumor as false, substantiated from the subject of the rumor himself. In this case, rumors experience an increase that is sustained for a significant period of time.

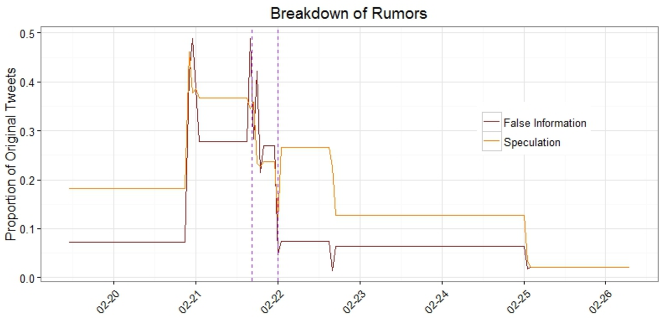


Fig. 4: Time series of tweet credibility broken down by rumor components.

Figure 4 dissects the propagation of ’Rumor’ into its components of ’False Information’, or category 1 tweets – in which false information is stated as fact – and ‘Speculation’, or category 2 tweets in which the tweet propagates the rumor but neither confirms nor denies it as factual information.

Upon closer examination of this breakdown, we see as expected a secondary spike in false information and increase in speculation following Hague’s announcement. Similarly, following Gaddafi’s announcement, the spread of false information as truth drops off and flat lines. However, speculation in this case rises and plateaus, suggesting counter-intuitively that users exhibit an increased level of skepticism as a result of the fact that subject of the rumor himself refutes it.

Examination of peaks 4 and 5 suggest decreasingly relevant tweets to the rumor of interest. This unrelated noise, along with the underlying implications behind credibility within the distribution of tweets in the Venezuelan data set therefore indicate that a deeper level of understanding is necessary in order to systematically analyze tweets. In order to isolate and distinguish credible tweets by tracking the propagation of specific events, we must both isolate these tweets by topic as well as seek to understand the sentiments behind them in order to discern their overall implications in the context of the time period. We therefore proceed with an analysis from a machine learning approach.

V. Sentiment Analysis

A. Overview

In order to understand the underlying sentiment behind a tweet, our sentiment analysis algorithm attempts to assess the feelings of an individual through the content of their tweet. For this algorithm, we first consider a dictionary of words. This dictionary associates certain common words with emotions. For example, the word “death” may be associated with anger or sadness, while the word “perfect” may be associated with happiness. A textual analysis is then performed, where each tweet is given a score based on the emotions it appears to be associated with. For this sentiment analysis, we use very basic categories for emotions: anger, anticipation, disgust, fear, joy, sadness, surprise, trust. Similarly, we designate tweets a polarity–that is, whether the tweet is positive, negative, or neutral.

After these analyses are done, we attempt to associate the topics with certain emotions. This can be done by relating the outputs of both analyses to the tweet the results are associated with. For example, we may notice that a lot of the tweets strongly associated with topic one may also be tweets that are negative in emotion.

B. Breakdown of Naive Bayes Classifier

We implement the sentiment algorithm, which heavily relies upon the Naive Bayes algorithm, when allocating the emotions to each of the tweets in the data set. The Naive Bayes algorithm is a conceptually basic but effective classifier that utilizes the fundamentals of the Bayes theorem and is frequently applied to classify various texts, including email spam detection, language detection, sentiment detection, etc.

One major assumption to note is that Naives Bayes heavily relies on an independence assumption. In other words, the classification of one tweet does not make an impact on the classification of any other tweets. There are two primary methods employed by the Naives Bayes classifier: lexicon-based and learning based algorithm. We choose to apply the lexicon-based method, which will rely on dictionaries of words that list words by sentiment (emotion) and polarity (positive or negative; weak or strong). The latter involves machine learning and training a classifier. We will use the lexicon method because it is less computationally intensive and moderately successful in its classification. [12]

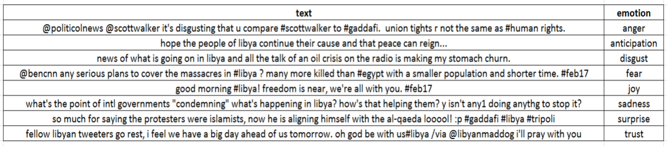
An extensive lexicon ensures that text can be properly classified. To do this, we modified the existing algorithm from the package sentiment [16] and the corresponding classify_emotion function in R. We chose to use R over Python because we could specify and edit the particular emotions we wanted. To improve computational speed, we employed parallel processing when running the sentiment algorithm. When conducting our initial sentiment analysis, many tweets failed to be classified and were “NA”. This was due to a sparse lexicon of only about 1,000 words and therefore unfeasible to classify over 2 million tweets. Therefore, we downloaded the emotion lexicon provided by National Research Council Canada, with about 14,000 lexicons with the associated sentiments: anger, anticipation, disgust, fear, joy, sadness, surprise, and trust. After manipulating the lexicon and converting it into a tidy format, we edited the function to aggregate the previous lexicon and add the two emotions “surprise” and “trust”, which were previously not in the function. Note that the Arab Spring Twitter dataset contains many Arabic tweets. Initially, we attempted to incorporate arabic tweets–ubiquitous in our data–and apply the appropriate data-cleaning techniques to work with unicode data. However, we decided to stick with English and Spanish tweets, for time-constraint reasons and greater interpretability. Overall, performing this expansion of the lexicon tremendously improved our results so no tweet was left uncategorized.

For each tweet the algorithm will calculate:

$$P(c|t) = \frac{P(c)P(t|c)}{p(t)}$$

where c is the value (for sentiment, the value of c will be either anger, anticipation, disgust, fear, joy, sadness, surprise, trust), t is the text in the tweet. $P(c)$ and $p(t)$ are the prior probabilities of the class and text, respectively. We do not have information on the distribution of the emotions so we will assume $p(c) = 1$. $P(t|c)$ is the probability that the particular text will appear if it is a certain class.

We then proceed by calculating $logP(c|t)$, the log likelihood for each emotion. Finally, the highest emotion will be selected to categorize the tweet.



text	emotion
@politicolnews @scottwalker it's disgusting that u compare #scottwalker to #gaddafi. union tights r not the same as #human rights.	anger
hope the people of libya continue their cause and that peace can reign...	anticipation
news of what is going on in libya and all the talk of an oil crisis on the radio is making my stomach churn.	disgust
@bencnn any serious plans to cover the massacres in #libya ? many more killed than #egypt with a smaller population and shorter time. #feb17	fear
good morning #libya! freedom is near, we're all with you. #feb17	joy
what's the point of intl governments "condemning" what's happening in libya? how's that helping them? y isn't any1 doing anythg to stop it?	sadness
so much for saying the protesters were islamists, now he is aligning himself with the al-qaeda looool! :p #gaddafi #libya #tripoli	surprise
fellow libyan tweeters go rest, i feel we have a big day ahead of us tomorrow. oh god be with us#libya /via @libyanmaddog i'll pray with you	trust

For the above example, we see that intuitively, the tweets have been correctly categorized. It is highly probable that the lexicon associates the word “disgusting” with both “anger” and then “disgust” (the lexicon often associates a word with multiple emotions). It then became ultimately narrowed down to anger because of the other words in the tweet. Furthermore, in the second line, “hope” and “continue” are often linked to “anticipation” and thus was categorized accordingly.

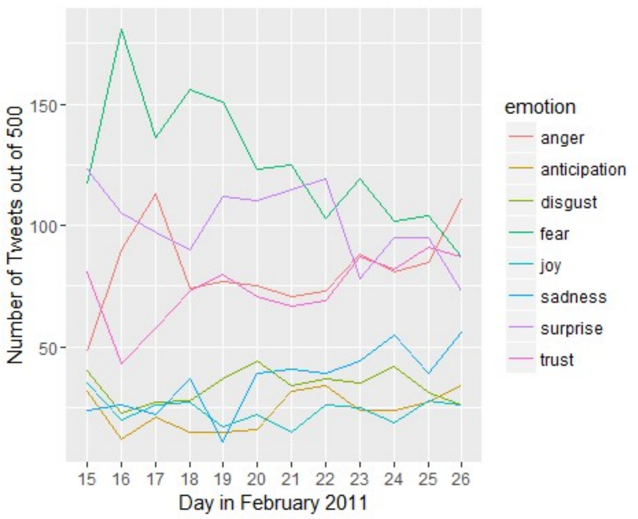


Fig. 5: The above plot consists of a stratified sample over the 12 days in our Arab Springs dataset. We drew a sample of 500 over the 12 days, thereby comprising a sample of 6,000 tweets. Because of computation limitations, such a small sample size may result in misleading conclusions but we will assume representativeness for our samples. In this plot, we see that prior to February 17, known as the “Day of Rage”, interestingly, there was a peak in fear. Then, on the actual Day of Rage, there was a subsequent peak in anger and slight increaes in surprise.

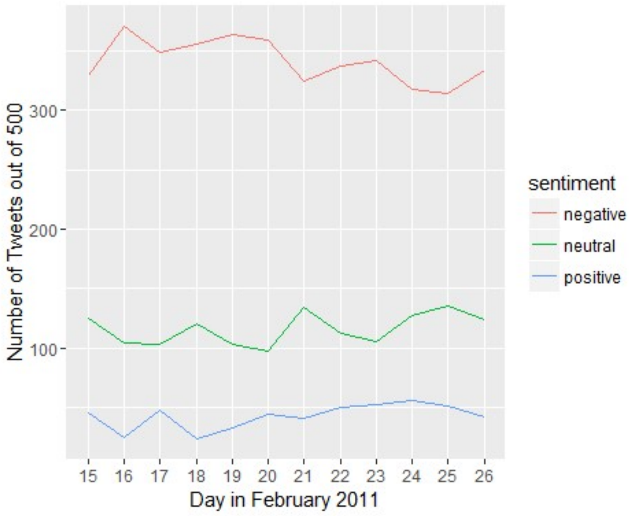


Fig. 6: Similarly, the same plot is produced while categorizing by positive, negative, or neutral. Most notably, an overwhelming majority of the tweets are categorized as negative.

.

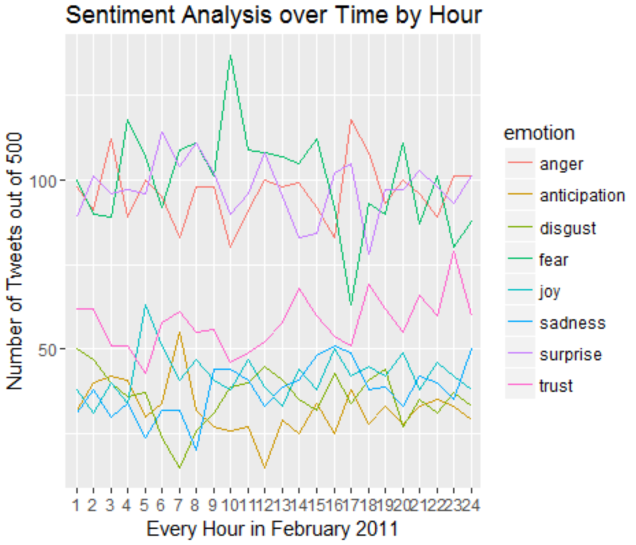


Fig. 7: This plot is stratified by time of day from February 15 to February 26. We see that there was a peak of fear at 10 a.m. and a subsequent nadir of peak at 5 p.m.

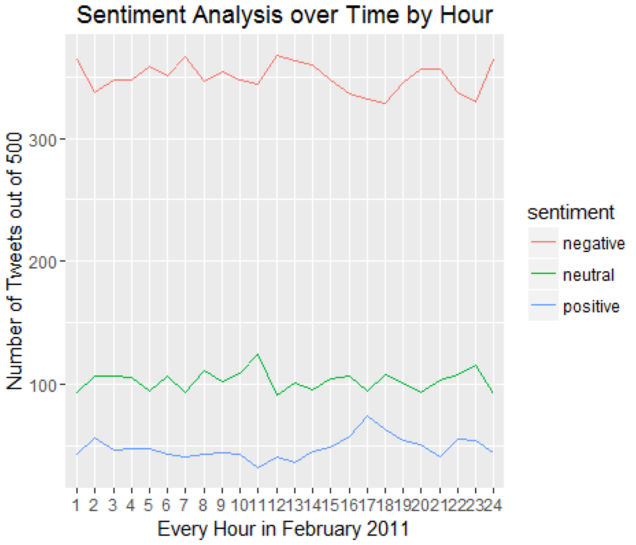


Fig. 8: The polarity of the tweets do not seem to fluctuate too much throughout the day.

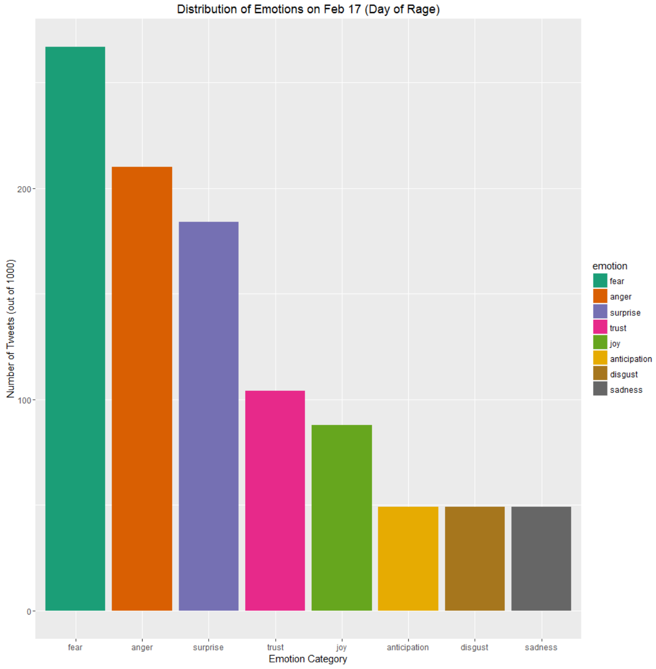


Fig. 9: Here is a barplot on the distribution of emotions for a sample of the tweets on February 17th, the Day of Rage. The three highest emotions are fear, anger, and surprise, which make sense intuitively.

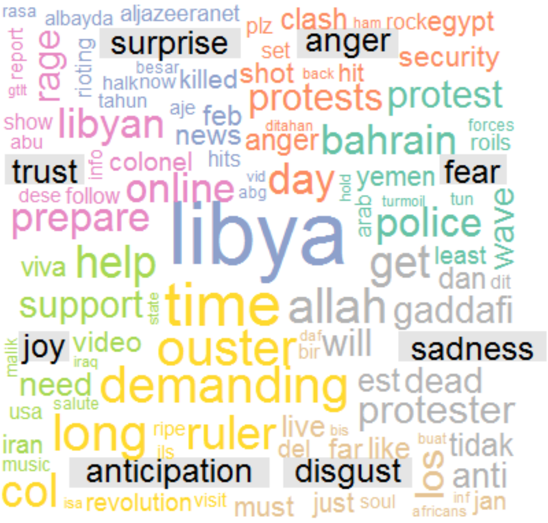


Fig. 10: The following is a word cloud displaying frequently used words on the Day of Rage. Only the top few hundred words are displayed in the cloud, with size being proportional to frequency. Each tweet is categorized an emotion, and the word chosen to be displayed on the word cloud is dependent on word count.

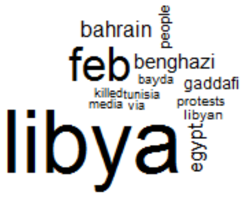


Fig. 11: This commonality cloud emphasizes words shared across all tweets on February 17.

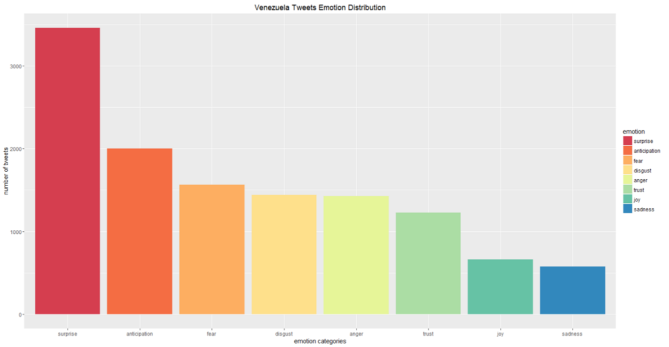


Fig. 12: Focusing on the Venezuela Rumor Propagation, we decided to further subset our data on specifically tweets pertaining the Venezuela. From the bar plot below, we see that a huge majority of the tweets are categorized by surprise and anticipation, as opposed to fear and anger in the February 17 subset.

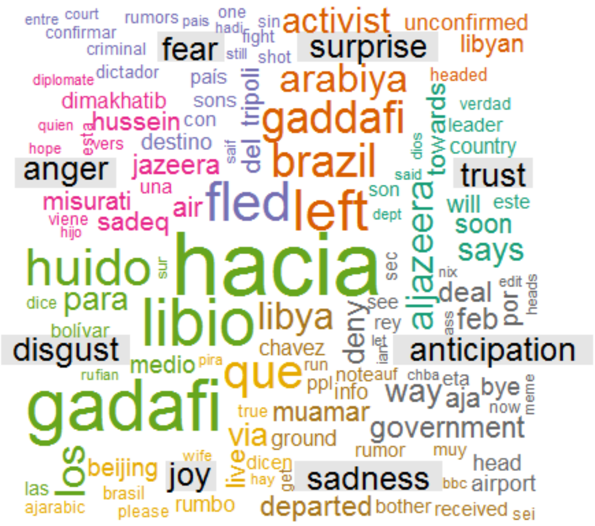


Fig. 13: In this word cloud, we see the prevalence of Spanish tweets, which made it essential to incorporate the Spanish lexicon to properly classify each tweet. This consists of strictly original tweets.

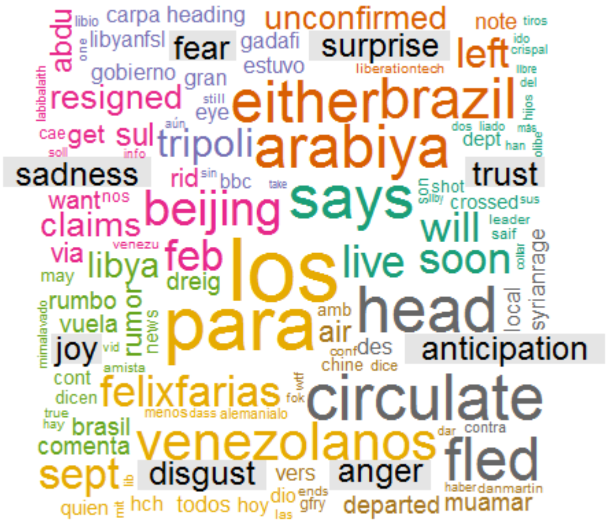


Fig. 14: Similar to the previous cloud, this word cloud consists of only retweets about the Venezuela incident. Noteworthy differences include “circulate”, “claims”, “unconfirmed” (bigger in the retweet cloud, indicating higher frequency), and “rumor”. These words listed in the retweets but not necessarily in the original tweets imply greater skepticism in the retweets than the orignal tweets.

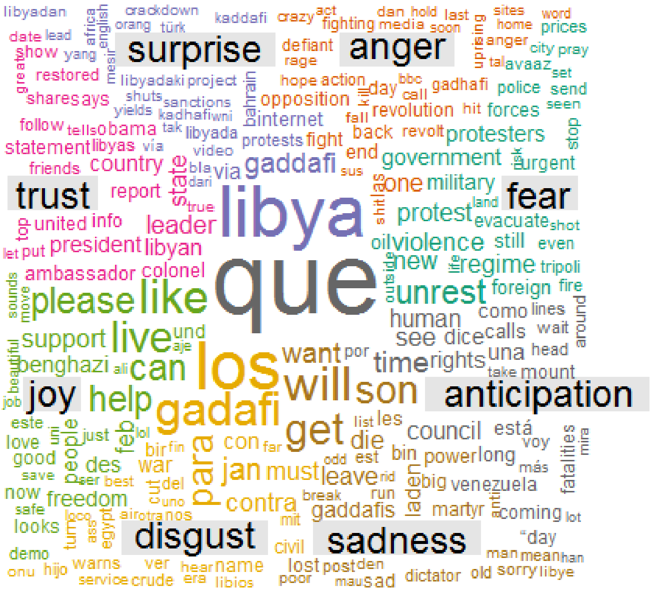


Fig. 15: The following word cloud is done on a much larger dataset of a sample of 20,000 and attempts to generalize the whole 2 million tweets.

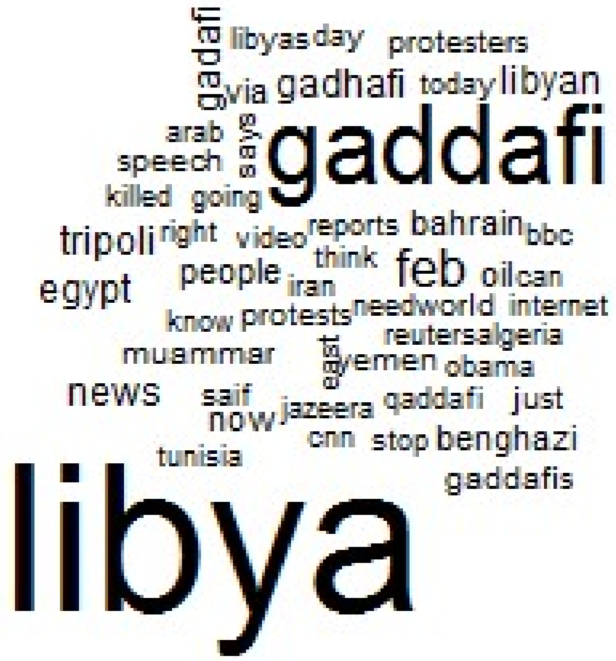


Fig. 16: This commonality cloud on a more comprehensive sample indicates the vast amount of tweets with libya and gaddafi in them. Note that for word cloud purposes, words and hashtags were treated the same.

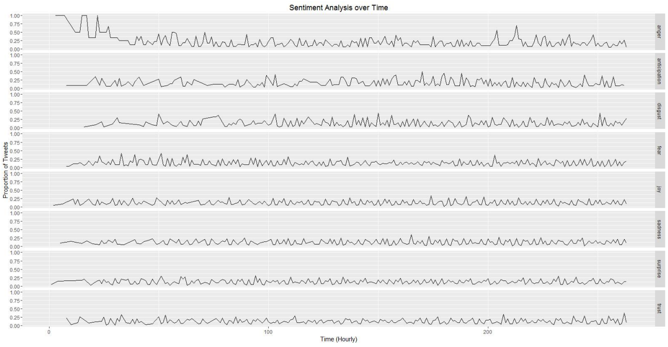


Fig. 17: The above is a time-series of the proportion of each emotion. Overall, we see slight variability in the emotions throughout the 12 days in our data. The relative stagnant behavior of the graph may indicate a weak relationship between time and emotion, which may need to be further probed in subsequent research.

C. Aftermath

Overall, our sentiment analysis was primarily descriptive. Ideally, if we had the resources and computational power, we would like to have incorporated the emotions and polarities into our predictive model in distinguishing rumor from truth. Research by previous projects involving Twitter data indicated the significance and relevance of positive versus negative sentiment in detecting false rumors. However, this algorithm was unsuccessful in classifying all 2 million tweets in a reasonable timeframe.

VI. PageRank

Developed by Google Search to recommend search results, PageRank is an algorithm to rank different nodes within a graph. In our case, the nodes are individual users. A directed edge pointing from node A to node B indicates that user A retweeted user B. The algorithm counts all of these connections, and weights each edge depending on the corresponding user's rank within the graph. In this way, PageRank accounts for the relative importance of an item's connection. If a node has a higher PageRank score, it gives a higher boost in rank to the connected nodes (e.g., the users that the node retweeted). The algorithm starts with all nodes having the same value, and as it iterates through the connections, values are transferred and ranks are updated. In general, the PageRank formula for a node u is:

$$PR(u) = \Sigma_{v \in B_u} \frac{PR(v)}{L(v)}$$

PR is PageRank. v is any other node in the pool that might be connecting to page u. $L(v)$ is the number of connections from node v. B_u is a set of nodes that connect to node u in the graph given to the algorithm. This formula tells us that the resulting rank of a page is determined by the rank of the pages that connect to it.

The original PageRank algorithm produces probability values as PageRank scores, but newer versions such as ours do not restrict the scores to be between 0 and 1.

We want to have a score for each Twitter user in our data, so we perform PageRank analysis at the user level. In our case, an item is a user, and a retweet is a connection. Pagerank is an appropriate method, because when determining a user's rank, it does not simply look at how many retweets the user gets but also who is retweeting the user. For example, if two users both only have one connection, but one of them is connected a high rank user while the other to a low rank user, then the first one would get a higher rank even though they both have only one connection. We believe that having many retweets does not necessarily suggest higher rank, so this logic of PageRank's helps us bring sophistication to how we determine the rank of the users in our data.

Our PageRank analysis is done in Python using a module called NetworkX. The output is a list of users and their corresponding scores. We have around $150,000$ users, and the score ranges from 0 to around $13,000$. While we know PageRank certainly does not cover everything for ranking a user's reliability, it is a measure that will prove to be helpful for us later in modeling.

VII. Topic Modeling

A. Latent Dirichlet Allocation

In natural language processing, topic models aim to uncover hidden semantic structure within a collection of documents. The most widely-used topic model, latent Dirichlet allocation (LDA), employs a three-level hierarchical Bayesian model to analyze text, representing each document as a random mixture over a fixed number of underlying topics. Each topic is, in turn, represented as a collection of words with certain probabilities associated with them. [10]

This generative approach utilizes an EM algorithm, and each document in the collection undergoes a two-stage procedure [10]:

1. Choose a distribution over the topics at random.
2. For each word in the document:
   a) Randomly choose a topic from the distribution in step 1.

   b) Randomly choose a word from the words associated with the topic.

We used the gensim package [9] in Python to create a LDA model for the original tweets in our dataset (retweets excluded). This totaled to about 1 million tweets. After processing the text, each tweet was separated into a list of stemmed words using a special tokenizer for handling Arabic and unicode text.

B. Assumptions

Most of the tweets analyzed here concern the Arab Spring, which might be considered a relatively specific topic on its own. In addition, certain keywords are used in the collection of these tweets, so these keywords are likely to be found within many of the topic representations. However, in our model we were able to successfully identify subtopics within the Arab Spring as a whole.

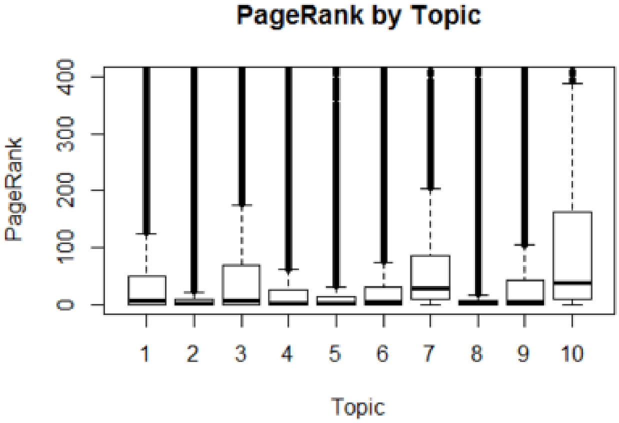
C. Results

After trying several parameters for latent Dirichlet allocation, with 5, 10, 15, and 20 topics, the results with 10 topics seem to give the best results:

TABLE II: Topic Descriptions

Topic	Description
1	English, mixed content
2	News sources, economic reports
3	Shootings, dozens killed during protests
4	World reactions (White House, EU, etc.) and human rights
5	Spanish, relationship between Chavez and Gaddafi
6	Rumor: Gaddafi fleeing for Venezuela, others fleeing, Turkish
7	Arabic, news on aftermath of protests
8	German and other European languages
9	Protest locations, pushes for nonviolence, injury treatment
10	Arabic, outcries against Gaddafi

To visualize the distribution of PageRank scores within each topic, we take a tweet's topic to be the one which has the highest probability assigned through LDA. From the boxplots below, we observe that Topic 10 has the highest median PageRank score, while Topics 2 and 8 tend to contain tweets from lower PageRank users. Intuitively, this agrees with intuition, as Topic 10 consists of Arabic tweets against Gaddafi's reign, whereas Topic 2 consists of news and economic reports, and Topic 8 contained tweets in European languages, where the Arab Spring was not as central of an issue (however, this may also imply sampling bias from the keywords used to pull the dataset).



We conduct statistical testing on PageRank scores with respect to topics. The distribution of PageRank scores is skewed within each topic. However, the one-way ANOVA is considered a robust test against the normality assumption for large sample sizes. Levene's test indicated that the difference in variances between topics is significant at the 1% level. Thus we used Welch's one-way ANOVA, which does not assume equal variances. We found that the difference in PageRank scores between topics was signifant at the 1% level, with an F-score of 1611.8.

VIII. PageRank Model

In order to help discern truth from fiction, we use each Twitter handle's PageRank score as a measure of user credibility. The user with the highest PageRank in our dataset is @sultanalqassemi (Sultan Sooud Al-Qassemi, a prominent Emirati commentator on Arab affairs).

We employ a generalized boosted model [7] to predict these scores, using features that have found to be related to credibility in prior research [8], such as the fraction of tweets containing links, and the fraction of tweets containing first-person pronouns.

The below table describes the features used for modeling:

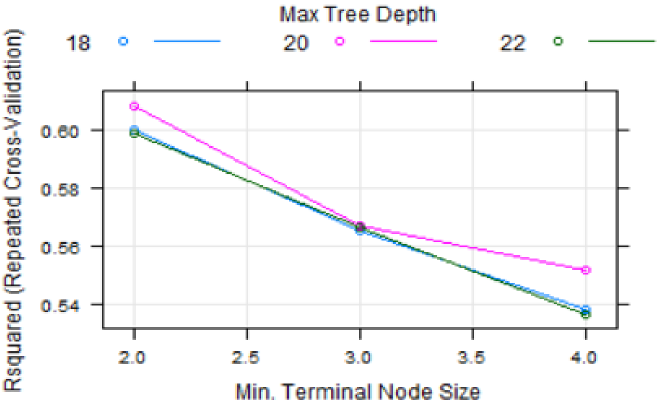
TABLE III: Feature descriptions

Features	Description
topics (1-10)	Average of ten topic proportions (see Table II)
total_impressions	Sum of the Twitter impressions of all tweets
time	Average time of day user tweeted
num_tweets	Total number of tweets
length	Average number of characters
has_links	Fraction of tweets with links
thirdperson	Fraction of tweets with a third person pronoun
firstperson	Fraction of tweets with a first person pronoun
mention	Average number of mentions
exclamation	Fraction of tweets with an exclamation point
smiley	Fraction of tweets with a smiley emoticon

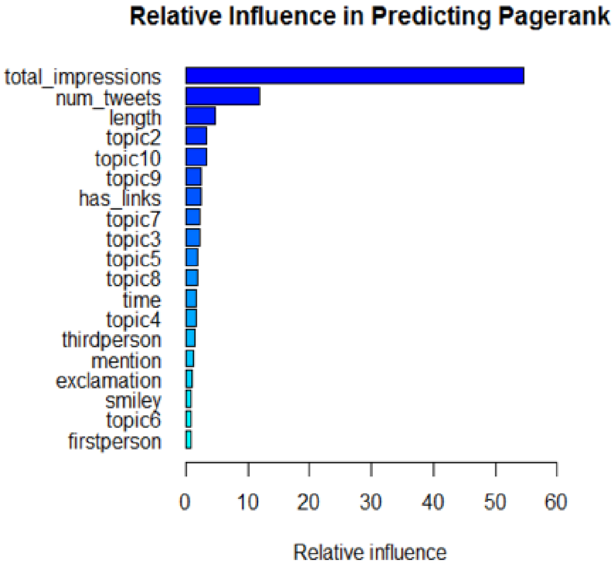
We are also interested in how the different topics a user discusses may be related to their PageRank score. As each tweet is a mixture of 10 topics, we aggregated the tweets of each user to obtain the average distribution of topics among them.

We chose the generalized boosted model because of its predictive power, and because it has fewer assumptions than methods such as linear regression. (footnote: Although linear regression indicated that all features were significant at the 1% level, the assumptions of the model were not satisfied. Despite the removal of bad leverage points and several transformations, subsequent models still showed non-normality and patterns in the residuals, leading us to believe that regression was not appropriate for the data.)

We honed the parameters of the model through repeated 5-fold cross-validation on the training data. This process was repeated with several sets of parameters. The performance with the final set of tested parameters is shown below:

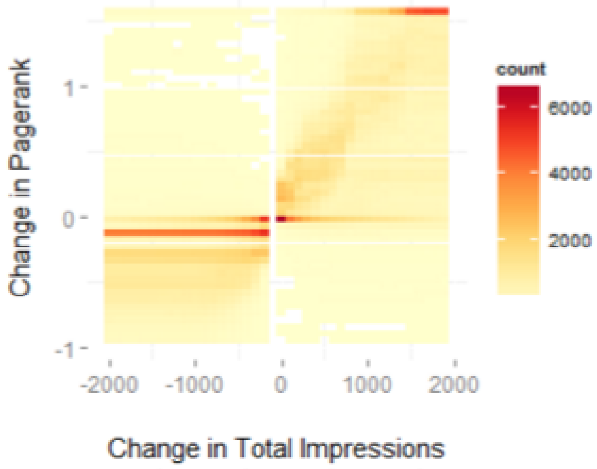


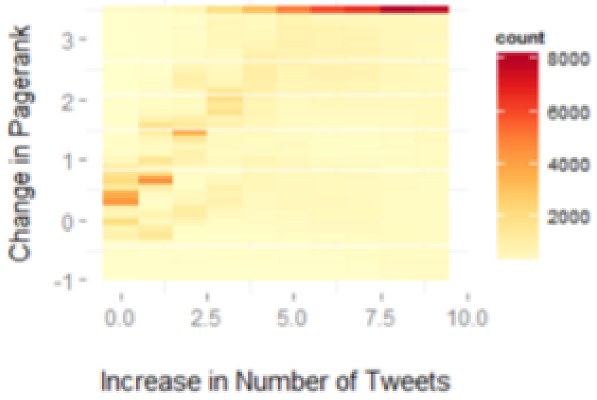
After tuning the parameters, the best model had an R-squared value of 0.655 on the test data. The features with the greatest relative influence are shown below.

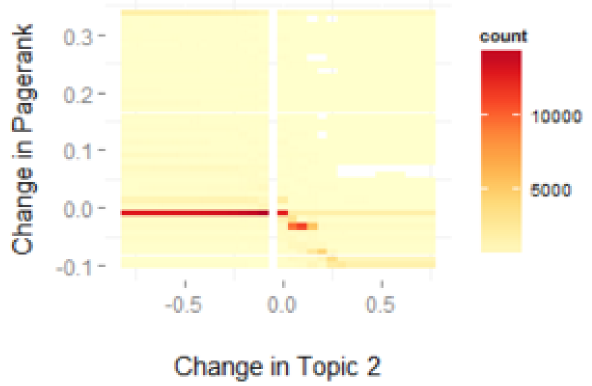


A. *Exploration of Variable Influence*

We would like to be able to visualize how changes within important variables affect model prediction. Taking a ten percent sample of users from our training data, we change the variable of interest by a specified amount, fix all other variables to their original value, and record the change in the model's predicted PageRank score. Using this method, we are able to see how different variables influence the model, and though it does not capture interactions, we still keep the effects of other variables into account. The following visualizations are gridded density plots of these changes and their effects on model prediction.



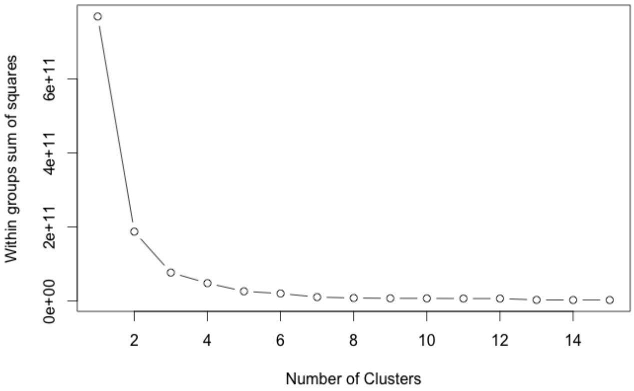




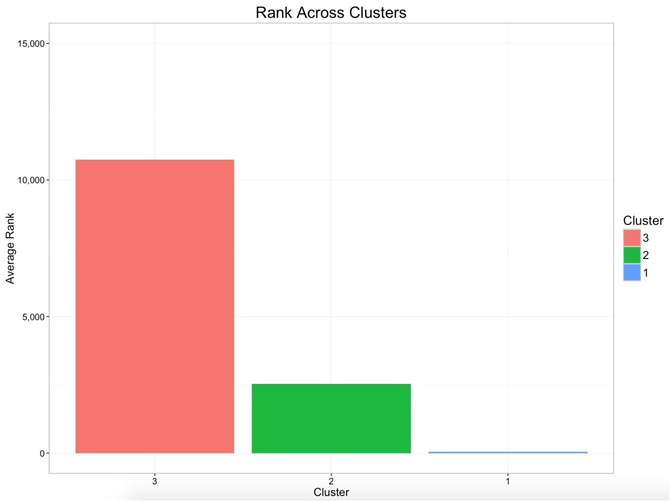
In the case of the user's total impressions [fig] and the user's total number of tweets [fig], the high density areas show that an increase in either of these variables often lead to an increase in the predicted PageRank score. On the other hand, an increase in average Topic 2 proportions among tweets led to a lower predicted PageRank score. As Topic 2 mainly quoted news sources and economic reports, this suggests that 'drier' or less 'sensational' tweets may be retweeted less often - evidence that PageRank is influenced by a number of factors in addition to credibility, such as shock or entertainment value.

IX. Clustering

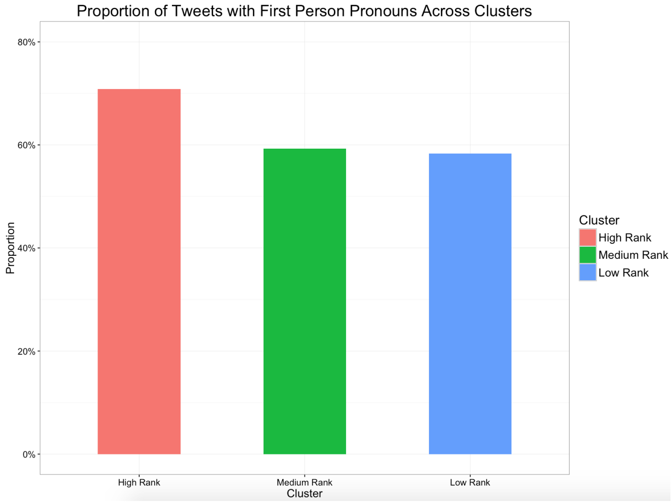
Further investigating how we can categorize the Libya tweets, we perform k means clustering at the tweet level. To find the appropriate k, the number of categories, we plot the within-group variance against number of categories. As we can see, the variance starts decreasing at a much slower rate around $k = 3, 4$, so we decide to cluster on 3 groups.



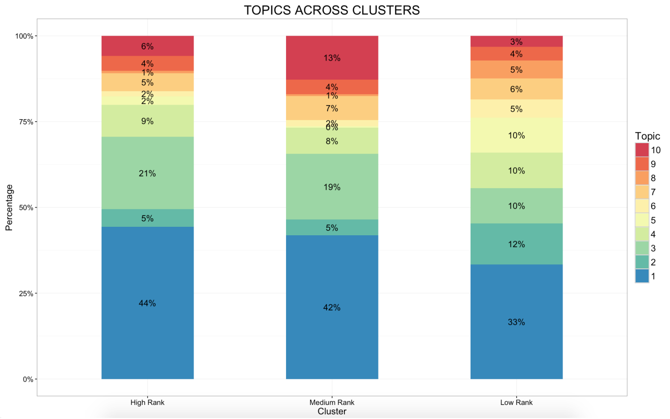
We use variables from our analyses done throughout the process, including results from topic modeling, the user rank from PageRank network analysis, and extra features created at the tweet level based on our research. These extra features include punctuations, pronoun usage, tweet length, whether the tweet contains links and so on. Our clustering results cover 90% of differences.



The most distinct difference between groups is the average PageRank, which is clearly shown in the plot. The red group has an average PageRank of $11,742$, the green group has $3,529$ and the blue has 54. From now, the red group will be referred to as "high rank group," the green as "medium rank" and the blue as "low rank."



One of the other characteristics that differ between groups is the proportion of tweets using first person pronouns. 71% of the high rank tweets, 59% of the medium rank tweets and 58% of the low rank tweets use first person pronouns. We find that tweets classified as high rank are about 12% more likely to have first person pronouns in them.



To tie back to topic modeling, we now look at how our topic results are distributed within each group. We have some interesting findings:

- Topic 5 and topic 6 tweets, which consist of the rumor about Gaddafi fleeing from Libya to Venezuela, are mostly classified as low rank, implying they are less reliable compared to other tweets.
- Topic 3, which has many reports of killings and massacres, appear to be mostly classified as medium to high rank, showing these reports were more reliable than not.

While we are aware that our clustering results were highly dominated by PageRank, since it has a wider range of values, we are able to use the clustering results to analyze the characteristics or signals that tend to be seen in tweets classified in higher rank compared to lower rank, which can suggest a potential rumor, and vice versa. Our results also lead us to connect back to the topics of these tweets we are clustering, because we see that the topics that are mainly rumors are successfully classified as lower rank.

X. Shortcomings and Future Research

The Arab Spring Twitter dataset presented numerous challenges, particularly in deciphering foreign tweets and incorporating them in our modeling and analysis. During the topic modeling process, we encountered an interesting pattern where several topics had been segregated by language, suggesting a certain degree of bias based on these language restrictions. Furthermore, there were significant computational limitations due to the magnitude of the data, for example inhibiting our analytical ability to perform sentiment analysis on the full dataset. Because the data itself was a sample of 2 million tweets–from which we then further sampled–our results may not be reflective of the true population. Ultimately, our aim to decipher credibility by systematically separating truth from fiction incorporates numerous factors that have yet to be investigated. One such factor of interest is the the discrepancy in retweet ratios between the propagation of a rumor versus a truth.

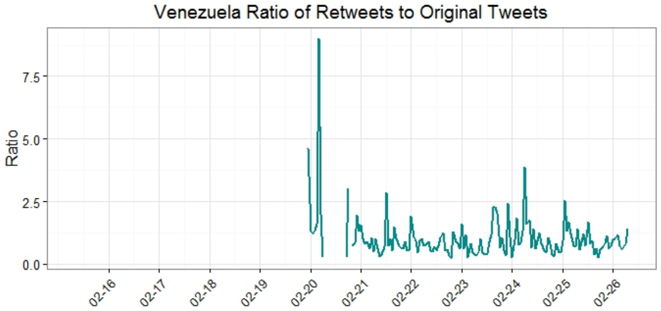


Fig. 18: Ratio of Retweets to Tweets in Venezuela rumor dataset.

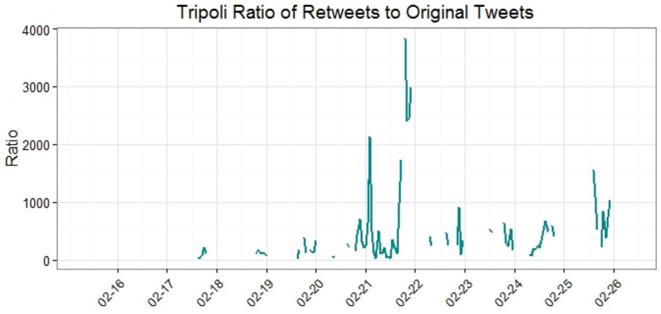


Fig. 19: Ratio of Retweets to Tweets in Tripoli dataset.

Figure 18 and 19 show distinct patterns in both the magnitude highlighted by the scale in the y axis as well as the occurrence of retweets following certain peaks corresponding to events. A further analysis can be conducted to understand the source of the discrepancy.

References

[1] *UK Hague: some information Gaddafi on way to Venezuela.* Reuters: 21 February, 2011. http://www.reuters.com/article/us-libya-venezuela-idUSTRE71K3S620110221

[2] *Gadhafi: Im in Tripoli, not Venezuela.* NBCNews.com: 22 February, 2011. http://www.nbcnews.com/id/41700027/ns/world_news-mideast_n_africa/t/gadhafi-im-tripoli-not-venezuela/

[3] Carlos Castillo, Marcelo Mendoza, Barbara Poblete. *Predicting information credibility in time-sensitive social media.* Emerald Insight: Internet Research (2012): 1-29. http://chato.cl/papers/castillo_mendoza_poblete_2012_predicting_credibility_twitter.pdf

[4] Aditi Gupta and Ponnurangam Kumaraguru. *Credibility Ranking of Tweets during High Impact Events.* Proceedings of the 1st Workshop on Privacy and Security in Online Social Media (2012): 1-8. http://precog.iiitd.edu.in/Publications_files/a2-gupta.pdf

[5] Rich Calaway, doParallel: Foreach Parallel Adaptor for the 'parallel' Package https://cran.r-project.org/web/packages/doParallel/index.html, version 1.0.10 (2015).

[6] Rob J Hyndman, forecast: Forecasting Functions for Time Series and Linear Models https://cran.r-project.org/web/packages/forecast/index.html, version 6.2 (2015).

[7] G. Ridgeway, gbm: Generalized Boosted Regression Models. https://cran.r-project.org/web/packages/gbm/index.html, version 2.1.1 (2015).

[8] C. Castillo, M. Mendoza, B. Poblete. *Predicting information credibility in time-sensitive social media.* Emerald Insight: Internet Research (2012): 560-588. http://chato.cl/papers/castillo_mendoza_poblete_2012_predicting_credibility_twitter.pdf

[9] R. Rehurek and P. Sojka. *Software Framework for Topic Modelling with Large Corpora.* Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks(2010): 45-50. http://is.muni.cz/publication/884893/en

[10] D. Blei, A. Ng, M. Jordan. *Latent Dirichlet Allocation.* Journal of Machine Learning Research (2003): 993-1022. https://www.cs.princeton.edu/~blei/papers/BleiNgJordan2003.pdf

[11] Edwin Chen. *Introduction to Latent Dirichlet Allocation* echen blog(2011). http://blog.echen.me/2011/08/22/introduction-to-latent-dirichlet-allocation/

[12] Pablo Gamallo and Marcos Garcia. *Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets* Proceedings of the 8th International Workshop on Semantic Evaluation (2014): 1-5. http://alt.qcri.org/semeval2014/cdrom/pdf/SemEval2014026.pdf

[13] P Cheng-Jun Wang. *Sentiment analysis with machine learning in R* datascience+ (2016). http://datascienceplus.com/sentiment-analysis-with-machine-learning-in-r/

[14] Vivek Narayanan,Ishan Arora,Arjun Bhatia3. *Fast and accurate sentiment classification using an enhanced Naive Bayes model.* Department of Electronics Engineering (2013). http://arxiv.org/ftp/arxiv/papers/1305/1305.6143.pdf

[15] NetworkX Developers. *NetworkX* https://pypi.python.org/pypi/networkx, version 1.11 (2016).

[16] Timonthy Jurka, sentiment: Sentiment. https://cran.r-project.org/web/packages/sentiment/index.html, version 0.2 (2012).

[17] Daniel Jurafsky and James H. Martin. *Lexicons for Sentiment and Affect Extraction*. Speech and Language Processing(2015): 1-22. https://web.stanford.edu/~jurafsky/slp3/21.pdf

[18] Will Oremus. *Building a Better Truth Machine.* Future Tense (2012). http://www.slate.com/articles/technology/future_tense/2012/12/social_media_hoaxes_could_machine_learning_debunk_false_twitter_rumors_before.html

[19] Tetsuro Takahashi and Nobuyuki Igata. *Rumor detection on twitter.* Soft Computing and Intelligent Systems Conference (2011): 1-6. https://www.researchgate.net/publication/261264390_Rumor_detection_on_twitter

[20] Ian Fellows, wordcloud: Word Clouds. https://cran.r-project.org/web/packages/wordcloud/index.html, version 2.5 (2014).