Rumor Detection in Online Social Networks

Kale Ashish Anil

Roll No. 14IT60R03

Department of Computer Science & Engineering

Indian Institute of Technology, Kharagpur

West Bengal, India

April 2016

Rumor Detection in Online Social Networks

A Thesis submitted in partial fulfilment of the

requirements for the degree of

Master of Technology

in

Information Technology

by

Kale Ashish Anil

Roll No. 14IT60R03

under the supervision of

Prof. Shamik Sural

Department of Computer Science & Engineering

Indian Institute of Technology, Kharagpur

West Bengal, India

April 2016

DECLARATION

I, **Kale Ashish Anil**, Roll No. **14IT60R03**, registered as a student of M.Tech. program in the Department of Computer Science & Engineering, Indian Institute of Technology, Kharagpur, India (hereinafter referred to as the ’Institute’) do hereby submit my thesis, title: **Rumor Detection in Online Social Networks** (hereinafter referred to as ’my thesis’) in a printed as well as in an electronic version for holding in the library of record of the Institute.

I hereby declare that:

1. The electronic version of my thesis submitted herewith on CDROM is in PDF format.

2. My thesis is my original work of which the copyright vests in me and my thesis does not infringe or violate the rights of anyone else.

3. The contents of the electronic version of my thesis submitted herewith are the same as that submitted as final hard copy of my thesis after my viva voice and adjudication of my thesis on 05-05-2016.

4. I agree to abide by the terms and conditions of the Institute Policy on Intellectual Property (hereinafter ’Policy’) currently in effect, as approved by the competent authority of Institute.

5. I agree to allow the Institute to make available the abstract of my thesis in both hard copy (printed) and electronic form.

6. For the Institute’s own, non-commercial, academic use I grant to the Institute the non-exclusive license to make limited copies of my thesis in whole or in part and to loan such copies at the Institute’s discretion to academic persons and bodies approved of from time to time by the Institute for non-commercial academic use. All usage under this clause will be governed by the relevant fair use provisions in the Policy and by the Indian Copyright Act in force at the time of submission of the thesis.

7. Furthermore

(a) I agree to allow the Institute to place such copies of the electronic version of my thesis on the private Intra-net maintained by the Institute for its own academic community.

(b) I agree to allow the Institute to publish such copies of the electronic version of my thesis on a public access website of the Internet should it so desire.

8. That in keeping with the said Policy of the Institute I agree to assign to the Institute (or its Designee/s) according to the following categories all rights in inventions, discoveries or rights of patent and/or similar property rights derived from my thesis where my thesis has been completed:

a. with use of Institute-supported resources as defined by the Policy and revisions thereof,

b. with support, in part or whole, from a sponsored project or program, vide clause 6(m) of the Policy.

I further recognize that:

c. All rights in intellectual property described in my thesis where my work does not qualify under sub-clauses 8(a) and/or 8(b) remain with me.

9. The Institute will evaluate my thesis under clause 6(b1) of the Policy. If intellectual property described in my thesis qualifies under clause 6(b1) (ii) as Institute-owned intellectual property, the Institute will proceed for commercialization of the property under clause 6(b4) of the Policy. I agree to maintain confidentiality as per clause 6(b4) of Policy.

10. If the Institute does not wish to file a patent based on my thesis, and it is my opinion that my thesis describes patentable intellectual property to which I wish to restrict access, I agree to notify the Institute to that effect. In such a case no part of my thesis may be disclosed by the Institute to any person(s) without my written authorization for one year after the date of submission of the thesis or the period necessary for sealing the patent, whichever is earlier.

Kale Ashish Anil
Department of Computer Science & Engineering,
Indian Institute of Technology, Kharagpur.

CERTIFICATE

This is to certify that this thesis entitled **Rumor Detection in Online Social Networks**, submitted by **Kale Ashish Anil** to Indian Institute of Technology, Kharagpur, is a record of bonafide research work carried under my supervision and I consider it worthy of consideration for award of the degree of Master of Technology of the Institute.

Dated:

Prof. Shamik Sural
Department of Computer Science & Engineering,
Indian Institute of Technology, Kharagpur.

ACKNOWLEDGEMENT

First and foremost, I would like to express my deepest gratitude and sincere thanks to my advisor, ***Prof. Shamik Sural*** for his inspiration, encouragement and able guidance all throughout the course of my M.Tech Project. It is because of his valuable advice from time to time and constant support that today I have been able to give shape to my work. It is indeed an honour and a great privilege for me to have worked under his guidance which has made my research experience productive. I learned an approach of humanity, perseverance and patience from him.

I sincerely acknowledge my deepest gratitude towards all faculty members of Department of Computer Science & Engineering for providing in-depth knowledge on various subjects over the past two years. It was a pleasure to learn and work with their co-operation.

Dated:

Kale Ashish Anil
Department of Computer Science & Engineering,
Indian Institute of Technology, Kharagpur.

ABSTRACT

The spread of incorrect information on Twitter can lead to harmful effect on individuals as people may consume and derive wrong inference from the information that reaches to them.There is thus a need to design a framework to limit the spread of such information.This thesis develops a model by harnessing the power of information retrieval algorithms to detect rumors.The tweets are collected and clustered using semantic information extracted from them.Each cluster has unique properties which can facilitate in the decision to consider it as rumor or non rumor.The work done extracts temporal properties from the clustered tweets and uses it to categorize rumors.Non Temporal properties are also considered for evaluation and they also play an important role in rumor detection.This ability to track the rumors has many real-time applications in various domains and ultimately systems can be designed to react to rumors once they are detected using such a framework.

Key words: Rumor Propagation, Information Retrieval,Tweet Clustering,Semantic Information,Temporal Characteristics,Rumor Detection

Contents

List of Figures

List of Tables

List of Algorithms

Rumor Detection in Online Social Networks

Chapter 1

Introduction

The Internet is a major source of all information which includes news as well.It is also the next important source of information after television as per [7].Nowadays, social media sites have had an impact on news journalism.People use primarily social media to interact with their friends and family but also to share news [6].This is done to convey updated information to people around them.The ease of use of social media has made this information flow much faster than ever before.In emergency and disaster situations,Twitter has proven to be great aid [12].

Online social networks allow users to post any information that the users intends to convey to others.Due to the advent of social networks, anyone can express their views on any of the available platforms.This facility allows information to be propagated in real time rapidly to a large audience.This helps in certain situations where other media such as news require more time to convey the same information.But along with the fast pace of information diffusion in social networks comes the problem of the reliability of the information.Due to the open and uncontrolled nature of online social networks anyone can post any information either deliberately or inadvertently without verifying the authenticity of the facts involved.This kind of information can lead to the spread of rumors in the network.A rumor is considered as a statement whose truth can be verified at the current instant of time given all the relevant evidence.Rumor also has a component which is controversial i.e. some people may not accept the fact directly and will raise questions about its correctness.Rumors can cause incorrect facts to be propagated and may cause panic in the population.

Twitter is a online social network where user can post small messages called Tweets which can contain up to 140 characters.Some users also retweet - which is a repost or forward of a tweet by another user. It is indicated by the characters RT.The ubiquity, accessibility,speed and ease-of-use of Twitter have made it an invaluable communication tool.People turn to Twitter for a variety of purposes, from everyday chatter to reading about breaking news.Users can explicit write new messages or they can re-tweet tweets which are written by other Twitter users.Due to the re-tweet facility available,the rate of information propagation is increased in some cases as some tweets will become viral and many users will re-tweet it.The increase in the number of smart phones and the number of people using those to write/read tweets has exploded the amount of information being generated.Detection of rumors plays an important role in such context.

Before we proceed with the explanation of our algorithms for rumor detection and verification,we need to examine what exactly a rumor is. We state a rumor to be an unverified assertion that starts from one or more sources and spreads over time from node to node in a network.On Twitter,a rumor is a collection of tweets which is unverifiable which are tweeted and then re tweeted subsequently to form an cascade as the information flows through the Twitter network.Rumors can be concluded in may ways - it may either turn out to be true, false or remain uncertain till a certain verifiable account of information is available.There can be many rumors about some specific object in various context.The final outcome of one rumor with regards to a specific object can help to reach a decision to decide the other rumor in some other context.Hence it is important to identify the exact topic of the rumor which is being propagated through the network.

Given the huge rate at which tweets are generated,it is impossible for any human to track down all the rumors that are currently present.There is therefore a need for an automated tool which can provide the list of potential rumors.As this list will be comprehensible by a human,the output provided by such a tool will be helpful to take corrective actions against the rumor.For example,if the fact in a rumor has an relevant authority, the authority can either vouch for or disapprove the rumor using the same social network.This will limit any false rumor from propagating to a large audience thereby limiting the spread of misinformation caused by it and other effects it may cause in the population.

The overall thesis is organized as follows -

- In Chapter 2, we include a survey of related literature.
- In Chapter 3, we present our methods for the tweet retrieval and tweet filtering.
- In Chapter 4, we present the tweet clustering results and the analysis of the distinct properties of rumors.
- Finally, we conclude the work in Chapter 5 and provide directions for future research.

Chapter 2

Review of Literature

There has been significant research on information diffusion in the context of social networks.A broad discussion of these is available in [5].In this article, they have analyzed multiple methods related to information diffusion analysis in online social networks, ranging from popular topic detection to diffusion modeling techniques, including methods for identifying influential spreaders.They have indicated that bursts are a good signal to identify popular topics.Information diffusion may be modeled using both graph and non graph based methods.There are many ways to identify influential spreads in a network including pure topological approaches, such as k-shell decomposition or HITS. It provides various ways to model information spread in a social network.The work done here does not deal the information content that is being flowed through the network.To detect rumors,we not only need the information related to the information diffusion but also the content of the information that is being propagated.

The work done in [2] is the most extensive work related to the credibility of tweets posted on Twitter.They first build a dataset of tweets using detection of bursts within Twitter.Using manual annotators the tweets are divided into bins corresponding to the amount of confidence that the annotator has in the tweet.They extract user,message,topic and propagation based features which is then used to build a classifier.Then supervised learning is used to identify the set of tweets of unverified information.They also perform best feature selection to determine the features which have maximum impact and thus play a crucial role in the classifier that is built.They emphasize the importance of validation of credibility of the information posted on Twitter as a safeguard for inexperienced users who can be misled by incorrect information.

The approach used in [10] attempts to formulate the model for rumor detection and then build a classifier out of it.They divide their work into two parts namely-rumor retrieval and belief classification.Rumor retrieval step deals with the task of finding a set of controversial tweets.The next step i.e. belief classification finds the set of users who suspect the statement to be rumor and raise question on it.They annotate a set of tweets as 1 or 0 depending upn whether a user believes whether it is a rumor or not.Using these annotated tweets,they extract features from the tweet dataset of different types such as content,network and some Twitter specific features.They manually annotate a set of tweets and then train a classifier to predict whether a new tweet contains a rumor or not.The work is mainly targeted to retrieve a set of related rumors in the dataset but it does not detect new types of rumors that are not contained in the training dataset.

The work as described above takes into account only the temporal and linguistic properties of the Twitter dataset.The work done in [8] is the first work to distinguish the temporal properties of rumor v/s a non rumor dataset.They have successfully demonstrated that temporal features have greater impact in decision of a rumor tweet.Rumors show bursty fluctuations over time unlike other random tweet chatter occurring over Twitter.The difference in spike behavior has been demonstrated as a good signal to differentiate the non rumor tweets.The work done in this thesis thus extends this notion and works primarily on temporal properties.This work has done the relative ranking of importance of various features and also mentioned which features dominate the decision to formulate a rumor.

Another work done in [11] uses Sina Weibo rather than Twitter as dataset due to the large number of users compared to Twitter.In addition to the standard linguistic and user features used in the previous work they have used multimedia features - one of them which is timespan.The timespan is calculated based upon the posted date of the new microblog and the posted date of the old image. If a microblog does not contain any picture, then value of timespan to be 0. If the timespan between the text and the picture is bigger than the threshold, then the value is 1.Otherwise the value is 2.The point to conclude is that rumors mostly contain images which are earlier posted on the Internet.Thus this work also depicts the importance of temporal properties when judging for a rumor.

The work done in [14] propose a real-time rumor detection procedure that has the five steps.The algorithm first finds enquiry tweets using a set of regular expressions .This set of tweets are called as signal tweets.Then they use Jaccard similarity to cluster these signal tweets.They calculate the Jaccard similarity by taking 3-ngrams of each tweet. Jaccard similarity is a commonly used indicator of the similarity between two sets.It is calculated as follows.

$$Jaccard(a,b) = \frac{|ngram(a) \cap ngram(b)|}{|ngram(a) \cup ngram(b)|} \tag{2.1}$$

They find the the most frequent and continuous substrings(3-grams that appear in more than 80% of the tweets) and output them in order as the summarized statement.Then they use this summary statement to compare with each of the tweets which are not signal tweets.This is done so that more tweets are included in the cluster which will help to build a better classifier.This is necessary because the signal tweets detected is very less compared to the non signal tweets.Using statistical features of the clusters that are independent of the statements content, they rank the candidate clusters in order of likelihood that their statements are rumors.

Chapter 3

Information Extraction

3.1 Problem Definition

We first define what constitutes a rumor.As per [3] a rumor is defined as -

A rumor is a controversial and fact-checkable statement.

We make certain observations related to the above definition:-

- Fact-checkable: It means that one can verify the correctness given the concrete evidence for the same.Statements for which truth value will be determined in future are not included.
- Disputable: People will not accept and will question the belief discussed in the information.

Any statement will includes a reference to a statement which has the above 2 properties can be also considered as rumor.

Most of the work done above in the context of rumor detection considers the data using only it's lexical and syntactic features.For example, work done in [14] treats tweets using the Bag-of-Words model to calculate the tweet similarity.For the problem of rumor detection it is necessary to identify the subjects that the rumor is spreading.Using the Bag-of-Words model,we don't have any sufficient information to identify the subject(s) that is/are being discussed in tweet data involved. The accurate subject identification of tweets however is a challenging task since the limited number of tokens in a post often implies a lack of sufficient contextual information.There is a certain need to analyze the problem by identifying its semantic features.

The problem is formulated as follows.Given a set of tweets,we need to group them into rumor clusters.Each cluster is identified by the subject discussed in the rumor.Then we enrich the semantic context of each cluster by extracting subjects contained in them.After this step,rumor analysis of each cluster is to be done to indicate the probability that it represents a rumor.The subjects which are extracted using Part-of-Speech tags are used to extract semantic information from the tweets.The semantic information is then used to cluster the tweets as a similarity measure.

The below diagram shows the overall flow of the working of the system proposed.

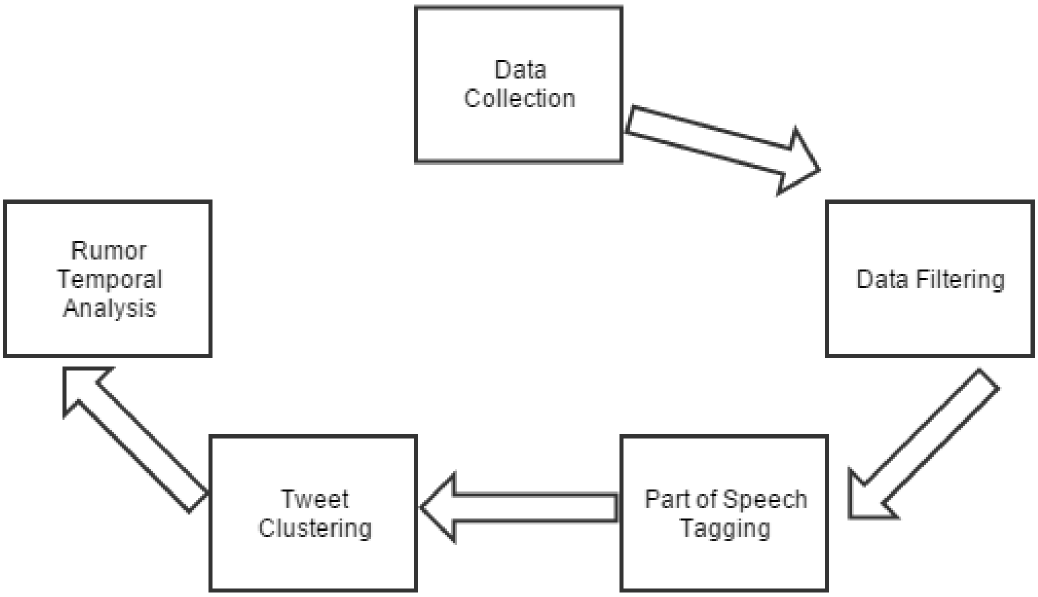

Figure 3.1: Flow diagram

3.2 Terminology

We first define some Twitter specific terms that will be used:-

- Tweet - Each message written on Twitter is called a tweet.
- Feed - A feed is any constantly-updating list of tweets or other updates, usually sorted chronologically with the most recent updates appearing at the top.
- Follower - On Twitter, you follow another user to see his or her updates on your Twitter home page.
- Mention - Twitter allows user to mention other user in a tweet using @ symbol.
- Hashtag - Using # symbol a user can enrich the subject being discussed in the tweet.
- Retweet - Using retweet we can make someone else's tweet appear in stream of our followers.

3.3 Data Collection

The tweets are collected using the Twitter Streaming API which provides 1% sample of the real-time tweets.We use the data available from Twitter for the month of July 2015.Given the huge volume of the data,the tweets are stored into a NoSQL MongoDB database in JSON format.A NoSQL (originally referring to "non SQL" or "non relational") database provides a mechanism for storage and retrieval of data which is modeled in means other than the tabular relations used in relational databases.Storing in the JSON format allows the meta-data to be updated over the time to add new features to the tweet data as well as enable efficient retrieval of large amount of data.The JSON data is then exported to HDFS for Map-Reduce operations on this large scale data.

The current dataset is limited to tweets which are written only in the English language.The information contained in the tweet meta-data is used to select only the English tweets.The data is filtered to remove the tweets which are written in multiple lines.This is done to facilitate processing on HDFS which considers one line as a single record.As the number of tweets having multiple lines is a small fraction of the total tweets,it is assumed that removing these does not impact the correctness of further operations.The tweets were also more or less evenly divided between each day of week, with each day having somewhere between 14% and 15% of the tweets. Similarly, the tweets were almost evenly divided between each hour, with each having somewhere between 3% and 5% of the tweets.The overall distribution of the tweets posted per day for the month of July 2015 of the 1% dataset is shown below.

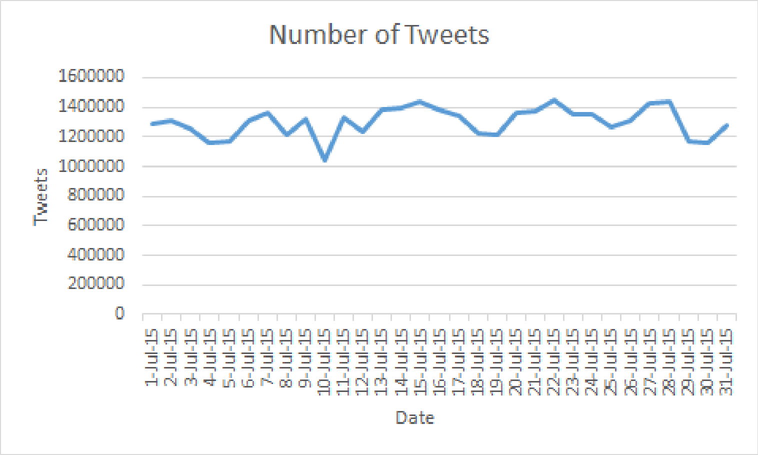

Figure 3.2: Number of Tweets

3.4 Data Filtering

The set of tweets collected is divided into two sets - signal tweet and non signal tweet.Signal tweets are those which contain enquiry pattern such as "really?","is it true" as identified in [14].These signal tweets are used as signal element to identify a rumor.All others tweets other than enquiry tweets are collected into the non signal tweet set. This is done to identify the first set of signal that may be used to indicate that a tweet may be a rumor.As per figure 3.2 we have seen that the number of tweets per day is very large and the average is around 1.3 million tweets per day.To process such data,we need certain type of distributed computation to scale up the processing time and get results within a minimum stipulated time.We have used Hadoop framework to process this "big" data.Hadoop used HDFS as the filesystem to store files i.e. the tweet data in current context.The Map-Reduce framework is then used to filter the tweets wherein the work is distributed to multiple machines in the cluster and partial results are then aggregated to form a unified dataset of filtered tweets.

3.5 Part-Of-Speech Tagging

A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads text in some language and assigns parts of speech to each word (and other token), such as noun, verb, adjective.The current approach uses POS tagging to extract syntactic features from each tweet rather than the lexical model used in previous works.The syntactic features will be later used to derive semantic information about each subject discussed in the tweet cluster. Standard POS tagging tools are used to extract POS tags from sentences in long documents.Tweets are online conversations which are short and it's word usage is different than of long text documents.Standard POS tagging tools are trained on data sets of large words,hence do not work on tweets.A specialized tool for tweets is needed for perform POS tagging efficiently.So an existing efficient and specialized Twitter POS tagging tool [4] is used for extracting POS features.The tool decomposes the tweets into various components such as proper noun,common noun,verb,adjective,adverb and interjection.Some Twitter specific POS tags such as hash tag,user-mention,URL and emoticon are also retrieved.

For example, the following tweet

"RT @brownjenjen : Ben Affleck denies affair rumors #rumor http://t.co/qwrfe"

is decomposed in POS tags as follows:-

RT	Re-tweet
@brownjenjen	At-mention
:	Discourse marker
Ben	Proper Noun
Affleck	Proper Noun
denies	Verb
affair	Common Noun
rumors	Common Noun
#rumor	Hashtag
http://t.co/qwrfe	URL

Table 3.1: Tweet Part-of-Speech Tagging

3.6 Tweet Clustering

This process involves dividing a set of tweets into subsets, where elements in each subset are considered related by some similarity measure.It is very rare that tweets which contain different subjects will contain any similarity on a semantic basis.Using a string similarity measure may lead to a spurious similarity for tweets even if they are not related.So the current approach avoids use of a string similarity measure and instead represents each tweet by proper nouns and URLs contained in it.These are obtained by the POS tagging step done earlier.The number of similar proper nouns and URLs are used a measure of similarity.

Each tweet is added as a node in the graph.An edge is added between two tweets only if they represent some common subject i.e. proper noun/URL.The weight of the edge determines the degree of similarity which is number of matching POS tags.Any POS tag that is matched is assigned a score of 1.The edge accumulates the scores of individual matching POS tags.An undirected graph of tweets is built by including an edge joining any tweet pair with a similarity score of at least 2. This is done to ensure that the rumors clusters are accurate.The accuracy of the clusters is determined by the subject discussed within them but also by its context.It may happen that one subject is being discussed with two different contexts in the same temporal dimension.For example,"Obama" may be discussed related to two other subjects such as "nuclear deal" and "war".It is necessary to form two candidate rumor clusters separately for each of them.By adding an edge only when similarity score is 2,such dataset will have two clusters - one related to "Obama" POS tags with "nuclear deal" related POS tags and the other with "Obama" POS tags with "war" related POS tags.This splitting of clusters also allows each cluster to be analyzed independently of each other.This improves the coherent quality of each cluster as each of them has the exact context discussed and does not mix tweets with other context.

A connected component in an undirected graph is a group of vertices, every pair of which are reachable from each other through paths.The connected components in such a graph will contain the tweet cluster which discuss a common subject.Next,we calculate the summary features of each cluster by including the POS tags which occur in more than 25% of the tweets in the cluster.This step ensures that only the subjects that are actively being discussed will be brought in the tweet cluster summary.The summary statement consisting of POS tags is now compared with each tweet in the non signal cluster set to increase the cluster size.

The entire algorithm is outlined below.

Algorithm 1 *Tweet clustering algorithm from Part-of-Speech tags - Single Day*

```
Input: Raw tweets from Twitter 1% sample dataset
Output: Clusters of candidate rumors
for All tweets in dataset do
    if Tweet contains enquiry pattern then
        Add tweet to signal tweets set
    else
        Add tweet to non signal tweets set
    end if
end for
for All tweets in the signal set do
    Add tweet as node in graph
    Extract Part-Of-Speech tags from each tweet
    Assign proper noun and/or URL as features for each tweet
    Compare with rest of the signal tweets
    if Number of features matching is greater than or equal to 2 then
        Add edge between the tweets
    end if
end for
Get connected components from the graph
if Size of connected component is less than 2 then
    Discard connected component
else
    Consider connected component as a candidate rumor cluster
end if
for Each candidate cluster do
    for Each tweet in the cluster do
        if Feature of tweet is contained in more than 25% percent of the tweets in the cluster then
            Add feature to summary feature set of the cluster
        end if
    end for
end for
```

Algorithm 1 *Tweet clustering algorithm from Part-of-Speech tags - Single Day*

```
for Each candidate cluster do
    for Each tweet in the non signal tweets do
        if Number of matching features between candidate cluster summary and the non signal tweet is greater than or equal to 2 then
            Add tweet to candidate rumor cluster
        else
            Discard the tweet for further processing
        end if
    end for
end for
```

3.7 Results

To process such large data in a distributed manner, the algorithms were executed out on a Hadoop cluster having 15 nodes.The data filtering results for signal tweets that were carried out are explained below.

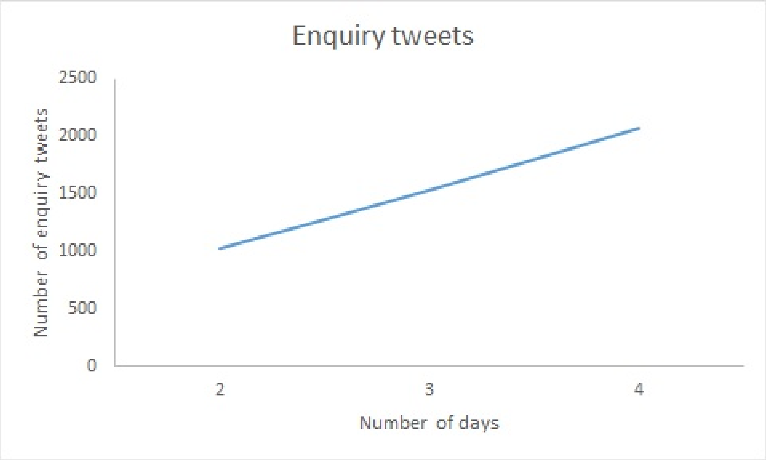

Figure 3.3: Days vs Signal Tweet Size

Figure 3.3 shows the number of enquiry tweets collected over days.The number of enquiry tweets is directly proportional to the size of the dataset.This concludes that everyday there exists a proportion of the users whose post enquiry tweets.This linear increase also proves that each day there is more or less a fixed amount of Twitter chatter which consists of the portion of signal tweets.

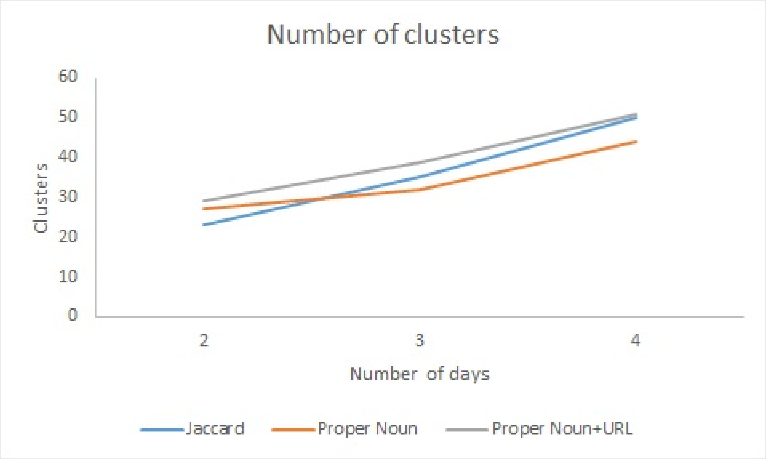

Figure 3.4: Number of clusters

Figure 3.4 shows that the number of clusters we get using tweet similarity measure as Jaccard distance is similar to the number of clusters we get using the proper noun POS tweet similarity measure.The proper noun+URL POS tweet similarity measure provides the best performance in all cases.Thus,we stick to this measure for all further operations.

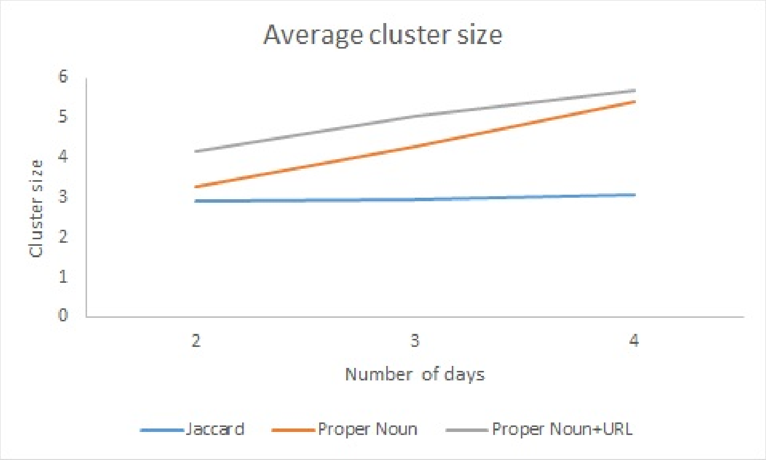

Figure 3.5: Average cluster size

Figure 3.5 shows the average cluster size for varying amount of tweet data.The average cluster size does not increase for Jaccard distance as larger datasets containing the same concept contains tweets which will contain diverse words. The diversity of the words involved in larger datasets will reduce the Jaccard similarity between tweets leading to small clusters being formed.When using POS tags, diverse words wont affect the result as they are not taken into account when determining the cluster similarity.In fact more the data, POS tag approach will lead to bigger clusters being formed.If we add URLs extracted from the POS tag in addition to proper noun, we get more bigger clusters as many tweets which contain a rumor which share a URL which contains more information about the fact being claimed.

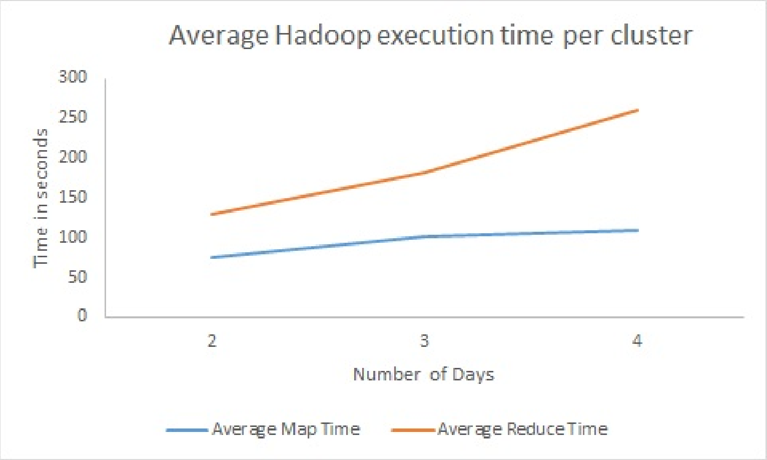

Figure 3.6: Hadoop execution time

While increasing the dataset shows improvement in the number of clusters and the average cluster size as shown in the Figures 3.4 and 3.5,the corresponding time required to process the non signal tweets to be assigned to a signal cluster also increases as shown in figure 3.6.In fact,the time required for Reduce operation starts to increase non-linearly.This metric is crucial when we need to detect rumors as quickly as possible.

3.8 Summary

This chapter has focused on the defining the rumor detection problem in a broad context.The data collection and filtering for the rumor detection purpose has been explained in detail.The semantic information extraction from a single day of Twitter data and its results using Hadoop cluster have been examined.We have also found that Proper Noun+URL Part-of-Speech similarity measure perform the best when clustering the tweets.In the next chapter,we deal with combining the individual Hadoop results of single days and performing rumor analysis.

Chapter 4

Rumor Analysis

4.1 Combining Overlapping Clusters

The earlier results show that increasing data to produce more as well as bigger clusters has a drawback since the time required to process such data increases proportionally with the number of days for which data is taken and thus is not scalable.If we need to see if any rumor is active from last ’n’ days, we need to process the ’n’ day’s data plus the current day’s data.Clearly this approach is not scalable as it requires re-computation of the last ’n’ days data.An efficient way to do this is store the result for last ’n’ days and combine it with the current day’s data.By this technique,the unit of data that should be processed can be kept to one day.This each one day’s processed data can be combined with earlier data to accurately represent the cluster.

The earlier algorithm clusters tweets into candidate rumor clusters for each day to expedite the processing on the Hadoop cluster.But this may lead to disjoint clusters created for each day,where rumors may spread over multiple days.Such rumor clusters which are divided into separate clusters over multiple days must be combined into single cluster so that we can accurately model the cluster for it’s temporal and quantitative characteristics.Such a framework also allows us to add newer data to existing clusters and check if any of the older rumors still exist or whether they have ceased.

To enable faster processing of clusters,we process only the signal tweets as they form a small portion of the overall tweet volume.Jaccard distance is used to calculate the distance between two tweets.Prior to clustering the tweets, the tweet data is processed using standard text preprocessing techniques so that the clustering performance is not impacted due to noise in the text data.The tweets are then clustered using hierarchical clustering.Average linking method is used to cluster the tweets is due it's advantages over the other methods.Average-link clustering is a compromise between the sensitivity of complete-link clustering to outliers and the tendency of single-link clustering to form long chains that do not correspond to the intuitive notion of clusters as compact, spherical objects.Average-link clustering merges in each iteration the pair of clusters with the highest cohesion.

The dendogram is cut at height 0.75 which is empirically the best value among all other values at which cut is done.Such a clustering will still lead to large number of small clusters.This is primarily due to the sparsity of the words contained in the tweets and the 140 character limit of each tweet.But these small clusters are indeed disjoint components of a bigger cluster.The algorithm 1 produces a list of clusters along with its summary features.These summary features can be used as semantic features to combine the clusters.For example, if a cluster obtained using the algorithm 2 contains summary features as f1,f2..fn, we can search other clusters where tweet has summary feature f1,f2,..fn.If such a tweet is found in another cluster of algorithm 1,the tweet and all other tweets containing in it's cluster are combined with the original cluster.This process is carried out transitively until no cluster produced by algorithm 1 can be further merged into a cluster produced by algorithm 2.

The entire algorithm is outlined below.

Algorithm 2 *Tweet clustering algorithm - Multiple days*

```
Input: Signal tweets with summary features from output produced by Algorithm 1
Non Signal tweets with summary features from output produced by Algorithm 1
Output: Clusters of candidate rumors spanning multiple days
for All tweets in input-signal-tweets do
    Remove punctuation from the tweet
    Remove all the extra white space between words
    Convert all the words in tweet to lowercase
    Remove stopwords from the tweet
end for
for Tweet ’a’ in input-signal-tweets do
    for Tweet ’b’ in input-signal-tweets do
        Calculate word-level Jaccard between tweet ’a’ and tweet ’b’ and store the result in the distance matrix at location distance[a][b]
    end for
end for
Use hierarchical clustering using average linkage to produce the dendogram using the distance matrix
Cut the dendogram at height determined empirically to determine the clusters
Rank the clusters based on decreasing order of their size
for Each cluster ’c2’ produced by algorithm 2 do
    Extract unique summary features contained in the cluster
    if A tweet in cluster ’c1’ produced by algorithm 1 contains these summary features then
        Combine cluster ’c1’ with ’c2’
    end if
    Repeat above steps until no cluster ’c1’ can be further merged into cluster ’c2’
    for All input-non-signal-tweets do
        if tweet summary matches to any unique summary features contained in the cluster then
            Add non-signal tweet to cluster
        end if
    end for
end for
```

4.2 Results - Clustering

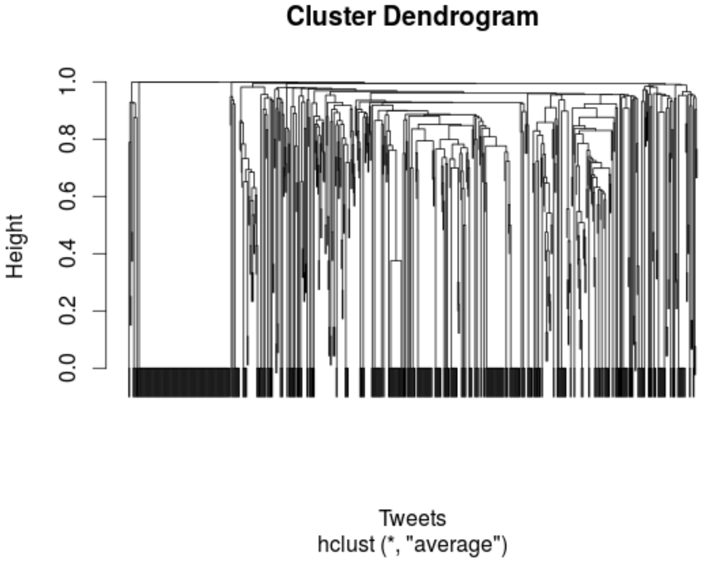

Figure 4.1: Cluster Dendrogram

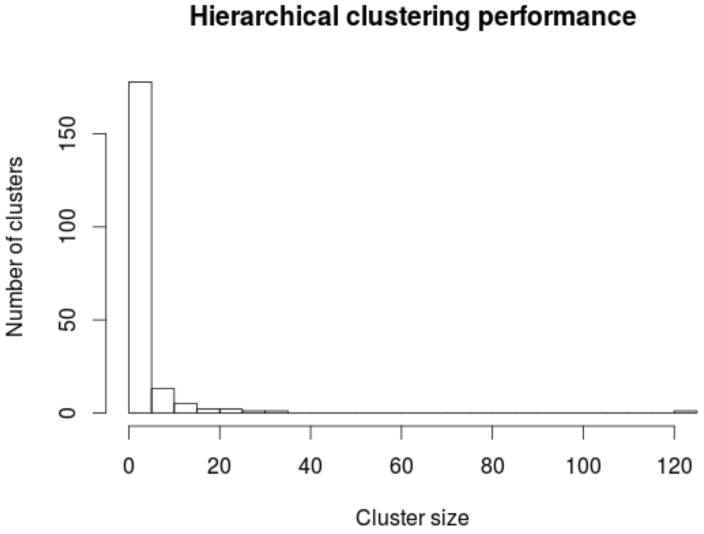

Figure 4.2: Cluster size histogram

Figure 4.1 shows the dendogram using hierarchical clustering of signal tweets using word level Jaccard distance.When this tree is cut at height 0.75, we get the clusters along with their size as shown in figure 4.2.The histogram shows that we have a very large number of small clusters and a very small number of large clusters.Therefore, we need to combine these clusters into larger ones using the common semantic features that were obtained from algorithm 1.

4.3 Temporal analysis

The next step involves the classification of the rumor clusters as rumor/non-rumors.The first focus is on the temporal aspects of the cluster.Other quantitative properties of the rumor will be explored later.We examine the rate of growth of tweet volume in a tweet cluster as a feature to classify the cluster.To calculate the rate of growth,the tweets are first sorted according to their tweet date.A new attribute rank 'i' is added to each tweet indicating that it is the 'i'th ranked tweet in the cluster according to its create date.A plot of tweet created date vs the rank of tweet in the cluster helps to determine the rate of growth of the tweet volume in the cluster.Analyzing the slope of distinct segments in the plot of can help to distinguish between rumor /non rumor.Each rumor cluster may have different size,hence we perform min-max normalization on the rank attribute to restrict it's values between 0-1 for any range of data.The min-max normalization is carried out as follows,where

$$x = (x_1, ..., x_n) \tag{4.1}$$

and z is now the ith normalized data :-

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)} \tag{4.2}$$

4.4 Results - Temporal Analysis

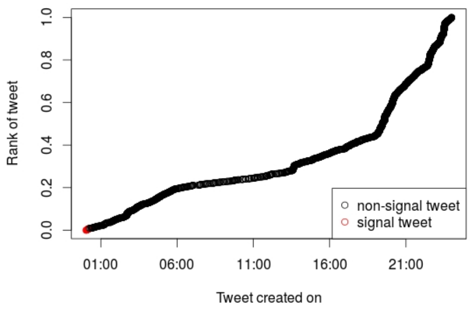

Figure 4.3: Tweet Date vs Rank of Tweet - Example 1

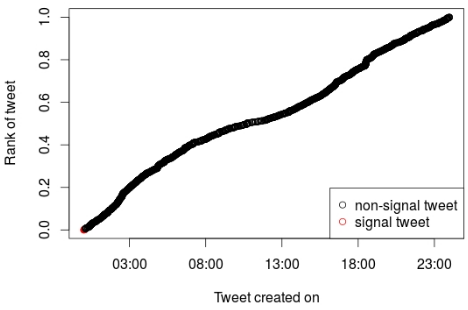

Figure 4.4: Tweet Date vs Rank of Tweet - Example 2

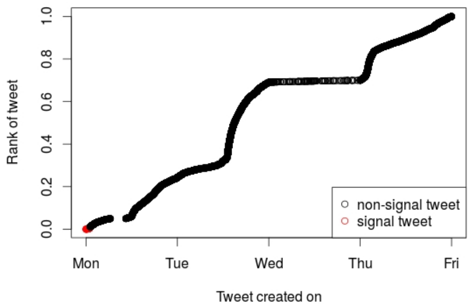

Figure 4.5: Tweet Date vs Rank of Tweet - Example 3

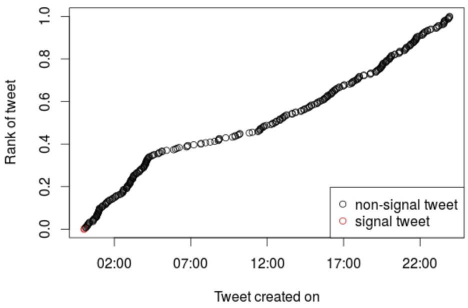

Figure 4.6: Tweet Date vs Rank of Tweet - Example 4

In Figure 4.3 we plot the time of tweet v/s the rank of the tweet(i.e. sequence number of the tweet when ordered according to it's create time). We can say that the growth of a non-rumor is fairly constant throughout it's lifetime.Since inception till it's end,the number of people posting about it does not deviate much.This can be seen by absence of sparse dots on the plot(we see this pattern in rumor when it's starts to fade away).Due to such behavior,this plot is a non-sparse curve.Owing to less number of distinct segments than a rumor plot,we can fit a linear line through a non-rumor cluster plot with less sum-of-square error than a rumor plot.Figure 4.4,4.5 and 4.6 show another similar instance of a clusters which do not constitute a rumor.

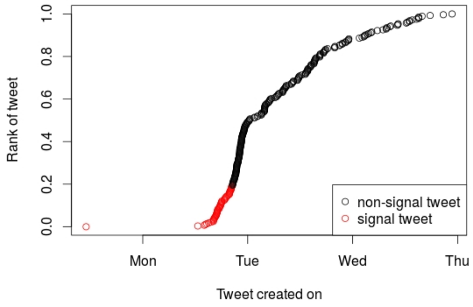

Figure 4.7: Tweet Date vs Rank of Tweet - Example 1

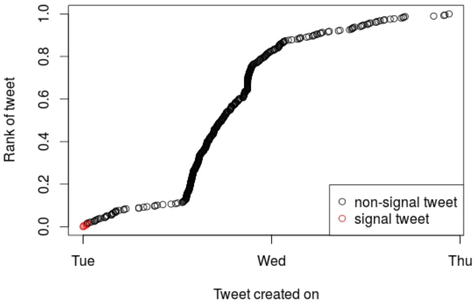

Figure 4.8: Tweet Date vs Rank of Tweet - Example 2

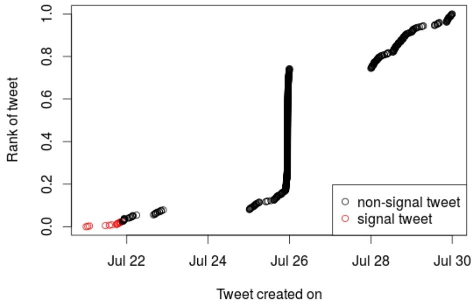

Figure 4.9: Tweet Date vs Rank of Tweet - Example 3

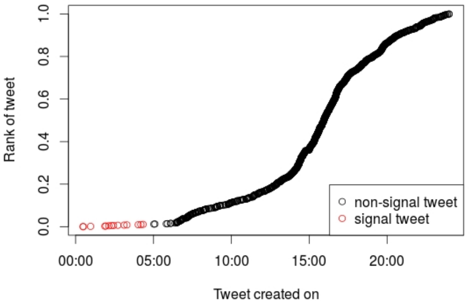

Figure 4.10: Tweet Date vs Rank of Tweet - Example 4

In Figure 4.7 we plot the time of tweet v/s the rank of the tweet(i.e. sequence number of the tweet when ordered according to it's create time).It shows that the growth of a rumor is non-linear.We find different phases where the rumor starts to spread,then increases at a rapid rate and ultimately decaying after a certain point in time.Figure 4.8,4.9 and 4.10 shows another similar instance of rumor growth rate.These different stages can be found by estimating the number of segments that are required to plot the curve.A rumor has many stages where rate of growth will be different at each stage thereby each of them resulting in a distinct segment on the plot.

4.5 Results - Burst Analysis

We now show the difference in burst patterns that are visible in rumors and non rumors.Here we divide the tweets in a rumor clusters in 30 bins divided by equal time intervals and plot the frequency of tweets in the corresponding bins.

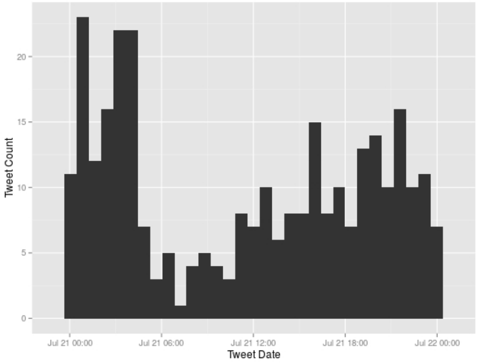

Figure 4.11: Tweet Date vs Frequency - Example 1

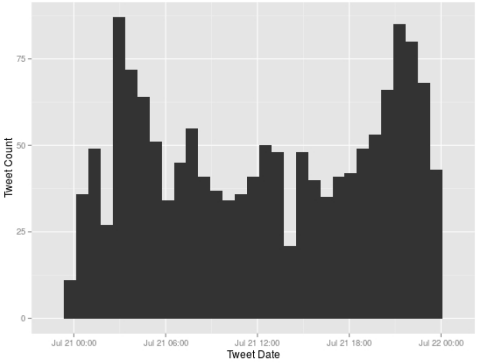

Figure 4.12: Tweet Date vs Frequency - Example 2

Figures 4.11 and 4.12 show the pattern observed in tweets which are non rumors.The distribution shows that number of tweets in every bin does not differ much.There is no presence of spikes in the plot which dominate every other bin in the plot.

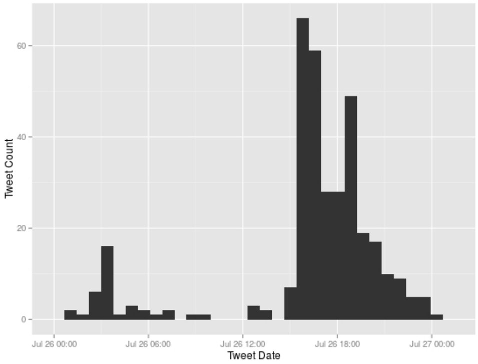

Figure 4.13: Tweet Date vs Frequency - Example 1

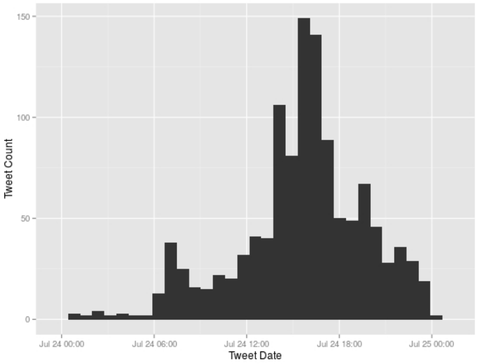

Figure 4.14: Tweet Date vs Frequency - Example 2

Figures 4.13 and 4.14 show the pattern observed in tweets which are rumors.The distribution shows that number of tweets in every bin differs by a significant quantity.There exists at least one significant spike in the plot towards end of the plot when the rumor starts to become viral.Also,the initial bins seems to be less frequent as the rumor has still not propagated through a large number of users.

4.6 Results - Data Analysis

The following figures 4.15 and 4.16 show the Tweet Date vs Tweet Is-Signal scatter plot.We plot these values for all combinations of values of number of user mentions and number of hashtags thus resulting in several sub plots. The Tweet-Is-Signal property can have a value of 1 or 0 depending on whether the tweet contains enquiry patterns.

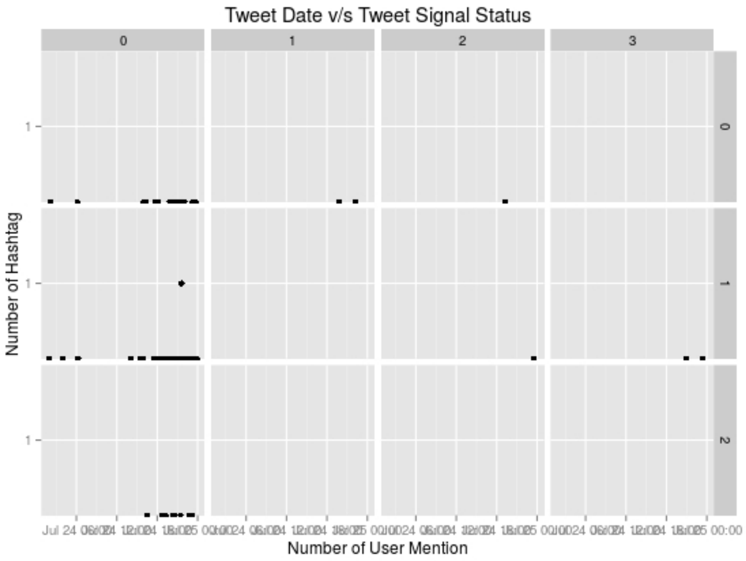

Figure 4.15: Tweet Date vs Tweet-Is-Signal

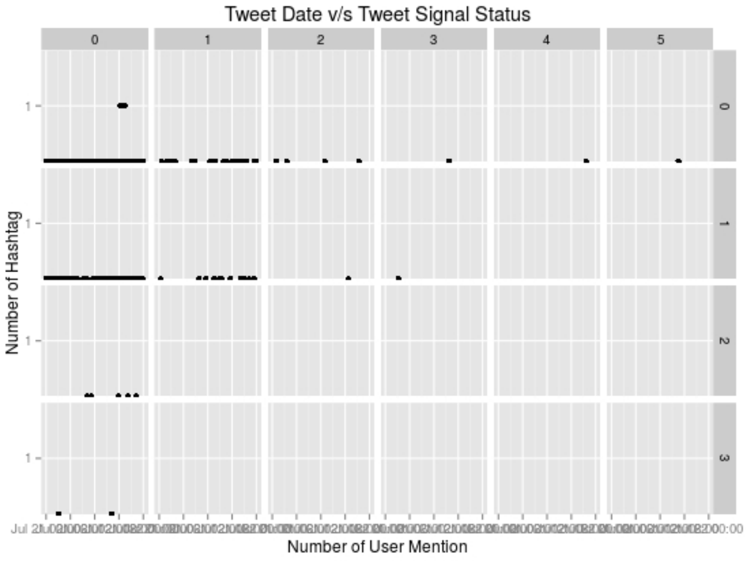

Figure 4.16: Tweet Date vs Tweet-Is-Signal

Figures 4.15 and 4.16 show the behavior observed in tweet clusters that are non rumors.We see that signal tweets in such a case do not contain usermentions as well as hashtags except barring a few exceptions.

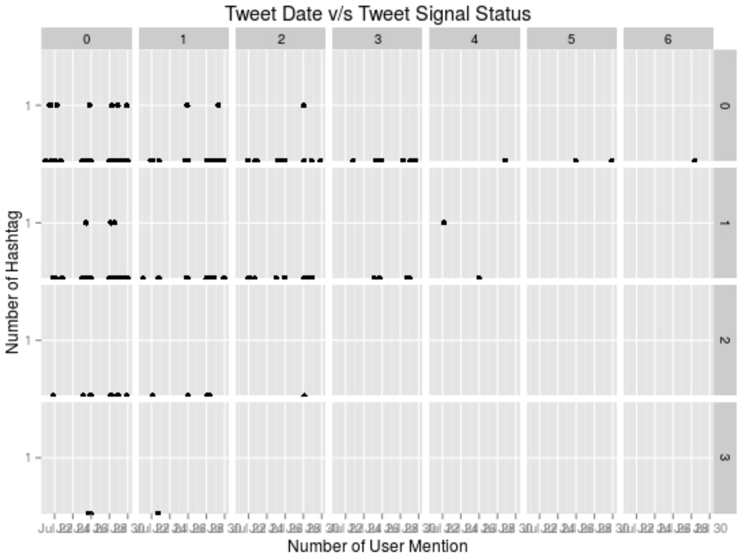

Figure 4.17: Tweet Date vs Tweet-Is-Signal

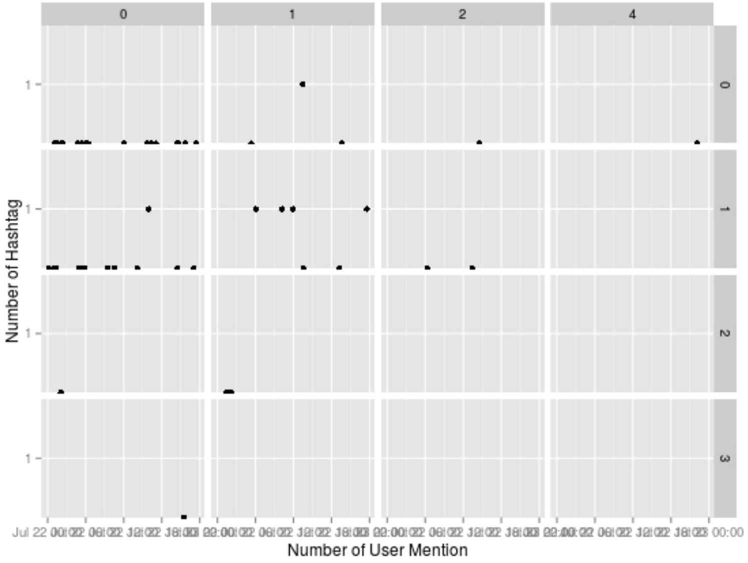

Figure 4.18: Tweet Date vs Tweet-Is-Signal

Figures 4.17 and 4.18 show the behavior observed in tweet clusters that are rumors.We see that signal tweets in such a case at least one contain user mentions or hashtags or a combination of both.

4.7 Summary

This chapter has focused on tweet clustering to accurately gather all tweets related to rumor.The dendrogram after initial clustering showed that clusters are not formed accurately due to sparse words in tweets and these clusters need to be combined further using another clustering algorithm.This helps in better prediction towards decision in the analysis part.The analysis part covered the difference in temporal and non-temporal properties of rumors which helps us to distinguish the both of them accurately.

Chapter 5

Conclusion and Future Direction

This thesis described a system for automatic detection and verification of rumors about real-world events on Twitter. Here we will summarize the contributions of this thesis and explore possible future directions for extending this work.

5.1 Contribution of the Thesis

The work described in this thesis describes how to create a system for detection and verification of rumors on Twitter.The major contributions are as follows:-

- We have used the NoSQL framework effectively to store the tweets given the huge volume of the tweets that are posted everyday.
- The Hadoop framework has been effectively utilized to run the information retrieval algorithm given the size of the data involved for filtering and processing the necessary subset of tweets.
- The extraction of semantic information to enrich the context of the rumor using Part-of-Speech tags
- Extending the clustering algorithm of candidate rumor clusters for finding overlapping rumor clusters is a key component that has been developed.
- A significant contribution of this thesis has been to find the essential temporal characteristics of the data and other Twitter specific meta data about tweets that constitute a rumor.

5.2 Future scope

There are many ways to extend the works presented in this thesis, several of which have been mentioned through out this document. Here, we will discuss what we believe to be the the five most fruitful directions for future work. These directions are:

- Linking to semantic web
- Design of classifier using temporal as well as non-temporal properties
- Extend system to other media platforms (social and traditional)
- Predict the impact of rumors
- Strategies for dampening the effects of rumors

There are already a wide variety of linked data sources incorporated in the semantic web as mentioned in [1] .The main advantages of these data sources are that they provide plentiful amount of data on a growing number of topics and they contain factual information about a large number of entities,covering these topics. The main goal is to exploit this semantic contextual information about entities contained in tweets by linking them to the data sources.This will aid in the data enrichment of the tweet data.Ultimately, this will help in the decision to identify rumors.

We have seen that the data properties of rumors and non rumors are different.The current work required manual intervention of looking at tweet content and verifying the same with the properties that the data exhibits.Using an exhaustive training data set which is annotated by an expert group of users who are trained to identify the correct rumors,we can build a classifier which will automatically classify and therefore identify new rumors.

The current system is primarily designed for Twitter.Similar systems cam be extended for other social networks such as Facebook,Reddit and LinkedIn.Though some of the features described for this work are Twitter-specific,many of the features are platform-agnostic and can readily be extracted and processed from different platforms.

In addition to detecting rumor,the system can predict the rate at which the rumor will spread and ultimately cease based on the recent data.This would be helpful to understand how many and up to what time people will be affected by the rumor.This is especially relevant for emergency services who might want to respond to false rumors that might have a large negative impact.

Finally, a system that can detect rumors may be used as a tool to counter attack the spread of rumor by administering an "antidote" in the form of content which spreads verified information about the rumor.This again would be something that would have the most relevance to the emergency services dealing with real-world emergencies as they are the ones that have to deal with the consequences and the fallout of rumors on social media.

Chapter 6

Bibliography

[1] Kalina Bontcheva and Dominic Rout. Making sense of social media streams through semantics: a survey. *Semantic Web*, 1:1–31, 2012.

[2] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684. ACM, 2011.

[3] Nicholas DiFonzo and Prashant Bordia. *Rumor psychology: Social and organizational approaches.* American Psychological Association, 2007.

[4] Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A Smith. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics, 2011.

[5] Adrien Guille. Information diffusion in online social networks. In *Proceedings of the 2013 SIGMOD/PODS Ph. D. symposium*, pages 31–36. ACM, 2013.

[6] Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65. ACM, 2007.

[7] Andrew Kohut and Michael Remez. Internet overtakes newspapers as news outlet. *Pew Research Centre*, 2008.

[8] Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. Prominent features of rumor propagation in online social media. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1103–1108. IEEE, 2013.

[9] O Muñoz-García, Andrés García-Silva, Oscar Corcho, M Higuera Hernández, and Carlos Navarro. Identifying topics in social media posts using dbpedia. In *Proceedings of the NEM Summit*, pages 81–86, 2011.

[10] Vahed Qazvinian, Emily Rosengren, Dragomir R Radev, and Qiaozhu Mei. Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1589–1599. Association for Computational Linguistics, 2011.

[11] Shengyun Sun, Hongyan Liu, Jun He, and Xiaoyong Du. Detecting event rumors on sina weibo automatically. In *Web Technologies and Applications*, pages 120–131. Springer, 2013.

[12] Sarah Vieweg. Microblogged contributions to the emergency arena: Discovery, interpretation and implications. *Computer Supported Collaborative Work*, pages 515–516, 2010.

[13] Tom White. *Hadoop: The definitive guide*. " O'Reilly Media, Inc.", 2012.

[14] Zhe Zhao, Paul Resnick, and Qiaozhu Mei. Enquiring minds: Early detection of rumors in social media from enquiry posts. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1395–1405. International World Wide Web Conferences Steering Committee, 2015.