MESSAGE PROPAGATION AND SOCIAL INFLUENCE IN TWITTER

By

VISHALI NARAYANA

Bachelor of Technology in Computer Science and Engineering

Mahatma Gandhi Institute of Technology

Hyderabad, India

2015

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 2017

MESSAGE PROPAGATION AND SOCIAL INFLUENCE IN TWITTER

Thesis Approved:

Dr. K.M. George

Thesis Adviser

Dr. N. Park

Dr. Johnson Thomas

ACKNOWLEDGEMENTS

I would first like to thank my thesis adviser Dr. K.M. George, Head of the Computer Science Department at Oklahoma State University for helping me choose the topic. The door to Professor George office was always open whenever I had a question about my research. He steered me in the right direction whenever I needed it. The weekly sessions held by Professor Dr. George helped me obtain new insights of the topic.

I would also like to thank professors Dr. J.P. Thomas and Dr. Nohpill Park for their support and guidance. I express my sincere gratitude for the contribution made by my senior Ashwin Kumar Thandapani Kumarsamy in helping me understand every minute in detail. I do appreciate the time he spent in reviewing my work through discussions.

Our family is a circle of strength and love. I would like to thank my family and friends for helping me reach my goals.

Acknowledgements reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

Name: VISHALI NARAYANA

Date of Degree: JULY, 2017

Title of Study: MESSAGE PROPAGATION AND SOCIAL INFLUENCE IN
TWITTER

Major Field: COMPUTER SCIENCE

Abstract: Twitter data has potentially unlimited value and numerous applications and is known for its increase in users over time. Twitter facilitates information diffusion at an exponential rate and also the creation of networks of users with a common interest. People reacting to the spread of an epidemic or a natural disaster are greatly influenced by the information diffusion in social media. Twitter, being a popular micro-blogging network provides an effective way to measure diffusion in terms of speed and strength. Our research is based on previous work on models related to topic diffusion and user influence. A topic is defined by a set of keywords.

This research concentrates on the implementation of algorithms for computation of diffusion of a topic in twitter. The degree of influence of the users who tweet on the topic is also addressed. We have presented two different approaches to compute user influence based on topic potential. We compare two diffusion models proposed in the literature, namely potentials and connections. For testing and empirical analyses we use tweets related to “flu”, “food poisoning”, and “politics”.

TABLE OF CONTENTS

Chapter Page

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

CHAPTER I

INTRODUCTION

Online social media generates a prodigious wealth of real-time data and allows millions of people to produce and consume content. It has become a standard platform for information diffusion. The most commonly used social website Twitter facilitates the information diffusion at an exponential rate. Twitter is a social network which allows users to exchange public messages of 140 characters or less, known as Tweets. Tweets can be text-based or they can contain multi-media such as images or video and links. In Twitter, tweets of a user are seen by other Twitter users who "_follow_". The "follow" feature allows one to build a network of peers with common interests [15]. Thus, tweets provide a large corpus of text for information mining.

There are several information diffusion models present in the literature. Each model attempt to capture specific information from the tweets. One of the models proposed in [20] addresses the propagation of topics and their strengths in Twitter as a function of time. The authors define a concept of potential (which can be interpreted in different ways) and present formulae to compute the potential. They also provide a formula to compute influence of users at a point in time from the potential. They used ad-hoc methods to compute the time-series and demonstrate the usefulness of the model. No algorithms were given to construct the time-series.

The objectives of this research are 1) review several diffusion models whose features are summarized in Appendix 4, 2) develop and test algorithms to build time-series based on the models given in [20] in a Hadoop environment, and 3) to compare the [20] model against other models.

This research contributes big data analytics from a micro-blog standpoint. It concentrates on implementation algorithms for computation of information diffusion of a topic in twitter. Degree of influence of the users who tweet on the topic is also addressed. A diffusion model for topics in twitter data (based on [20]) is outlined and implemented by a new algorithm in map-reduce framework. We outline two approaches to assigning influence measures to users based on their tweets/retweets and implement them. In [20], information diffusion is computed by introducing a weight associated to topic defined in Chapter II called topic potential. The tweets related to a topic form the nodes of a diffusion network. Edges of the network represent retweet relation. The potential of a topic is computed at regular intervals to build a time series. We implement algorithms to compute the time series from tweets. We also implement the diffusion network model proposed by [23] for comparison purpose.

We review literature related to this work in the next chapter. The methodology section, Chapter III, describes the data collection process from Twitter using Apache Flume. Chapter IV describes the implementation of programs to compute topic potential and user influence. The implemented programs are used to analyze three datasets i.e.

1. Flu data
2. Food poisoning data
3. Politics data.

Chapter V outlines the results of the analyses and Chapter VI provides conclusion and suggestions for future work.

CHAPTER II

REVIEW OF LITERATURE

Online networks are more focused on sharing information and have been studied extensively in the context of information diffusion [3]. The widely popular use of Twitter have created very large corpus of information. Several models assume that the information diffuses from one user to another such as spreading of epidemic [17]. This thesis focuses on the general topics of information diffusion and user influence and hence related research has been reviewed. We use the term topic in relation to diffusion which will be defined later in section 2.1.1. There are many investigations done on number of people influenced, the speed and geographic range of propagation [16] [6], diffusion flow model [18] [23], probability of influence [12]. In [13], the authors introduced a T-BaSIC model that models the information diffusion through a directed network of users with diffusion function and time-delay parameters for each arc (u_x,u_y). In [8], a random graph of users is first sampled from the distribution induced by a particular diffusion model and then a function is defined for node reachability in the sampled graph. The expectation of this function is the influence for the random graphs. The authors suggested that a sum over conditional probabilities is used to learn influence function.

Although some papers do exist to measure the 5 V's of big data, this research is based on a new approach to measure the velocity of topics in tweets proposed in [20].

2.1 INFORMATION DIFFUSION

The spread of information over time regarding a particular topic is called information diffusion [4]. The study of information diffusion is helpful in many applications such as election prediction [21], linking patterns of political bloggers i.e. interaction between conservative and liberal blogs [1], to estimate the expected time for information to reach a specific user in the network using a Diffusion Rank algorithm [18], detection of real-time emerging topics on Twitter implementing aging theory [5], movie box office [2].

Many models have been proposed so far to study the information diffusion in Twitter and other social networks as the topic diffusion model [20], predictive model T-BaSIC, diffusion networks model [23], by using social graph and cascade graph [19] etc. These models reviewed in the following sections. The primary contribution of this thesis is related to the topic diffusion model proposed in [20]. So it will be described in greater detail.

2.1.1 TOPIC DIFFUSION MODEL

The topic diffusion model proposed in [20] models topic propagation in tweets. The authors introduced a weight associated to topic called topic potential. A weight associated to the tweet is called tweet potential. The tweets related to a topic form the nodes of a diffusion network. Edges of the network represent retweet relation.

User interests, user familiarity, curiosity etc. affect the information diffusion [20]. The authors assumed that all the above features affecting the information diffusion are present in the tweets themselves. The authors defined the topic and potential as follows:

Topic

A topic is a set of keywords L = {α_1, α_2, α_3, α_4 …... α_k} associated with weights γ_1, γ_2, γ_3….. γ_k

with the normalizing condition $-1 <= \sum \gamma_i <= 1$. If $\gamma_i < 0$, then the keyword will influence the topic negatively.

Level

Each tweet is associated to a level. The levels of the tweets from the diffusion network are computed as follows.

The root tweet i.e. original tweet will have a level '0'. If the tweets with level '0' is retweeted, then the level of the retweet is 0+1. In general, if the retweet with level 'l' is retweeted, then the new level becomes l+1. In other words, the level captures the depth of a tweet measured by retweets and interpreted as velocity of diffusion. The tweet potential is computed based on the levels. The tweet potentials are summed up for detecting the topic potential for particular dates. Information diffusion is calculated by computing the topic's potential. The relevant definitions follow:

Potential

A weight called potential p (tw) of the tweet tw is associated relative to a topic L. It is computed as follows:

$$PL\ (tw) = \sum_{1}^{k} \gamma_i\ I(\alpha_i)$$

Where I (α_i) =1 if keyword α_i is present in the tweet tw, 0 otherwise

Topic Potential

The potential of a topic L at time t is defined as the sum of the two contributions as follows:

$$PL\ (t) = \sum p(tw)$$

Where l is the level of retweet of tw at time t.

Information Diffusion

The topic potential when plotted against time generates a tweet graph which captures the topic diffusion over time.

The retweets build a diffusion network that changes over time. The idea is illustrated in Figure 1.

In Fig 1, a diffusion network is built for tweet and its retweets. The root node in black color is the original tweet at time t0. All brown color nodes are the immediate retweets and are of level ‘1’ at times t1, t2, and t3. The blue color nodes are of level ‘2’ and so on.

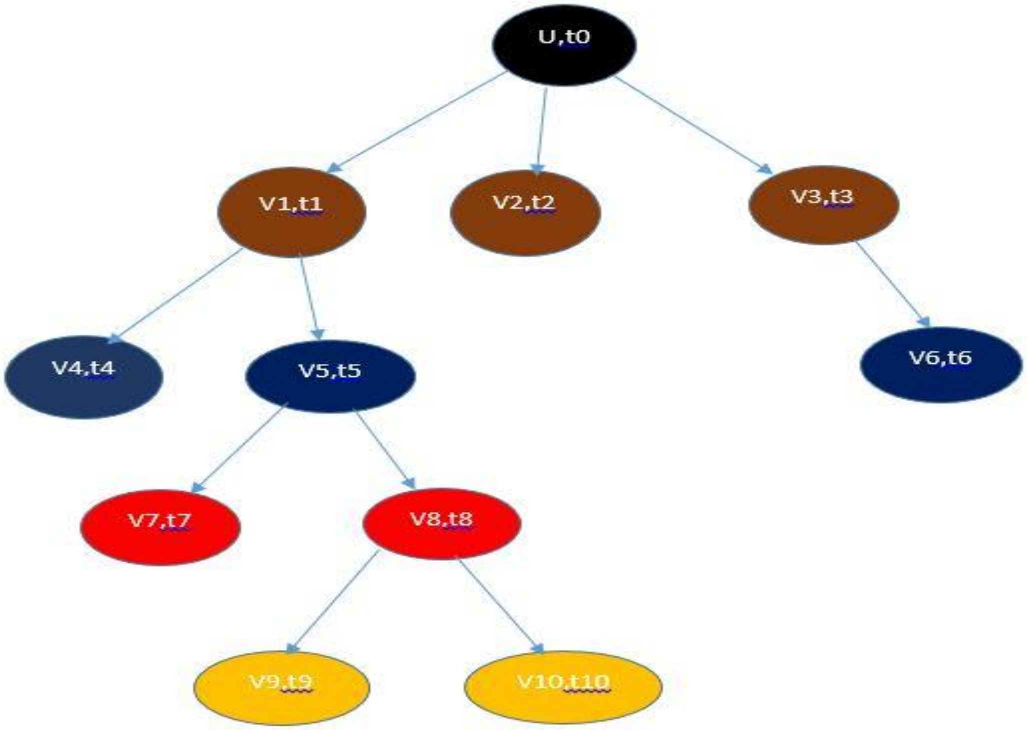


Fig 1: Diffusion Network indicating different levels of one tweet at different times

2.1.2 PREDICTIVE MODEL T-BaSIC

The graph based T-BaSIC (Time-Based Asynchronous Independent Cascades) predictive model proposed in [13] models information diffusion in online social networks.

A directed network G = (U, E) is considered where U is the set of all the nodes formed by users and E ($\subset$ U x U) is the set of all arcs. An edge (u_i, u_j) is in E if u_j is exposed to a message from u_i.

Fig 2 depicts the directed network G described above. Users are denoted as u_i and messages are denoted by m_j. An arc (u_x, u_y) means that u_x is exposed to messages published by u_y [13].

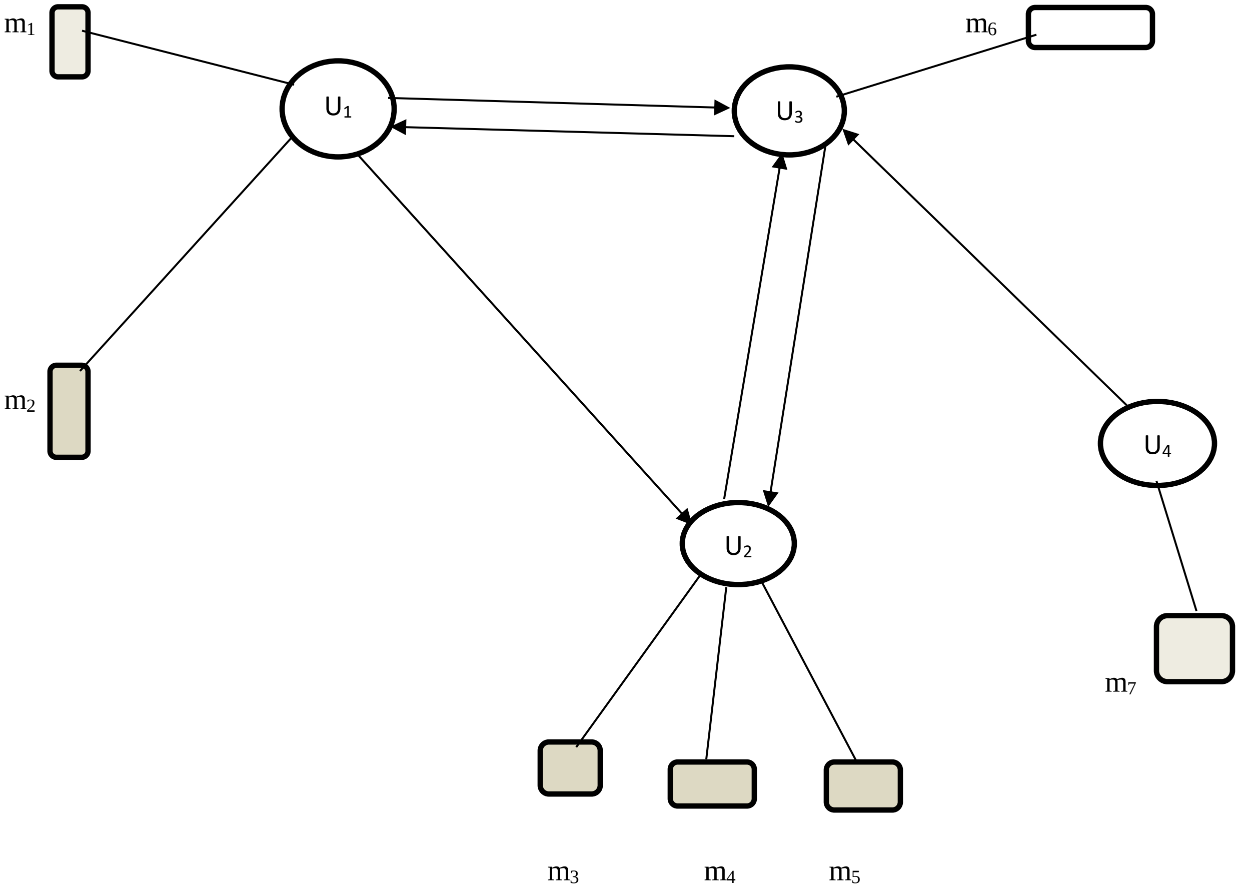


Fig 2: Directed Network with users (u_i) as nodes associated with messages m_i

T-BaSIC models the information diffusion through a directed network G = (U, E) of users. For each arc (u_x,u_y) there are two parameters 1) Diffusion Function 2) Time-Delay Parameter that are described below.

Diffusion Function

Diffusion function is a function of nodes, edge and exchanged content features. It gives the probability that u_x transmits information to u_y at a time t of the day.

$$0 < f\ (u_x,u_y)(t) \ < 1$$

Time - Delay Parameter

The time required by a node u_x to activate another node u_y is called Time – Delay parameter.

$$r\ (u_x,u_y) > 0$$

The author considered that each node u_x that becomes activated at time t is given a single chance to activate each of its inactive neighbors u_y with a probability of f (u_x,u_y)(t) .

If the activation is successful, u_y becomes active in t+ r (u_x,u_y). Stopping condition is reached when no more activations are possible [13].

Fig 3 illustrates the principle of T-BaSIC showing the input and output. Each arc have the two parameters time-delay and diffusion function. The diffusion function initially starts with a set of activated nodes. Over time, all the other nodes get activated predicting diffusion [13].

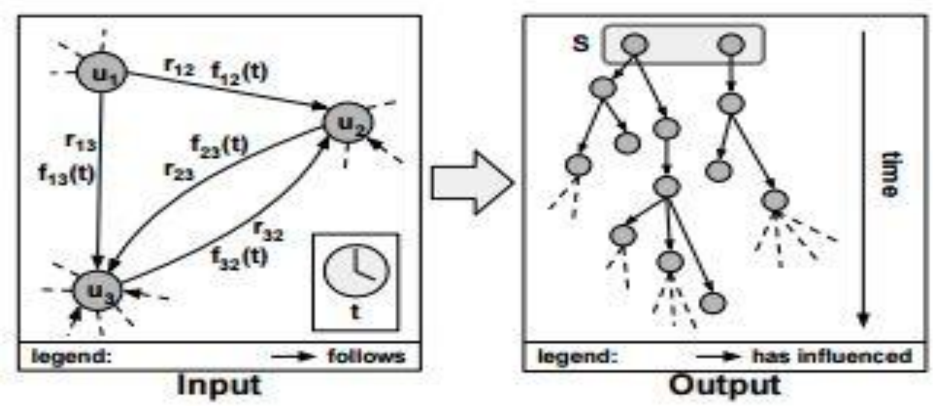


Fig 3: T-BaSIC model predicts the diffusion process along a continuous time-axis [13]

The prediction is based on the time-delay and diffusion function on each arc, starting from a set S of initially activated nodes proposed by [13].

The probability that a node u_x transmits a message/information to a node u_y computed by the model is a function of nodes, edges and topic features belonging to social, topical and temporal dimensions. These features described below are numerical values between 0 and 1 that are computed on past information diffusion traces.

Social Dimension Features

The rate of interaction between the nodes is measured by social dimension features.

The following measures are considered as social dimension features.

The rate at which each node publishes messages - I (u_x), I (u_y)

Jaccard coefficient between two sets of nodes u_x and u_y interaction - H (u_x, u_y)

Ratio of directed messages VS non-directed messages by each node - dTR (u_x), dTR (u_y)

Rate at which each node receives targeted messages - mR (u_x), mR (u_y)

Topic Dimension Features

Topic Dimension Features measure the interest of each user towards the topic.

Interest of each user for the information - hK (u_x, i), hK (u_y, i)

Topic Dimension Features

Topic Dimension Features measure the interest of each user towards the topic.

Interest of each user for the information - hK (u_x, i), hK (u_y, i)

Model Parameter Estimation

Diffusion Probability function is given by the following formula for all the 11 interpretable features that are described above. The formula below is given by [13]

$$P(\text{"diffusion"}|V) = \frac{1}{1+\exp(w_o + \sum_{a=1}^{13} w_a V_a\ )} \quad \text{(A. Guille 2013)}$$

V is the related vector of features.

wa coefficients are estimated using Bayesian logistic regression on data, describing how information was diffused in the past.

2.1.3 DIFFUSION NETWORK MODEL

The diffusion network model is proposed by [23]. The three major properties of information diffusion are speed, scale and range. These are captured by the diffusion network model. Properties of the users, the rate with which a user is mentioned historically are equal or stronger predictors.

An interaction network is built with a constraint of similar topic based on @username mentions to extract network structural properties, attributes of users and content that predicts diffusion within these structures. Mentioning (@) includes all uses like reply, retweet and forms an active interaction network.

Fig 4, illustrates the diffusion link between users A and B. User A has tweeted about the topic Iran election. B has mentioned A (@A) and talked about the same topic i.e. Iran election. Therefore, a link is built between A and B.

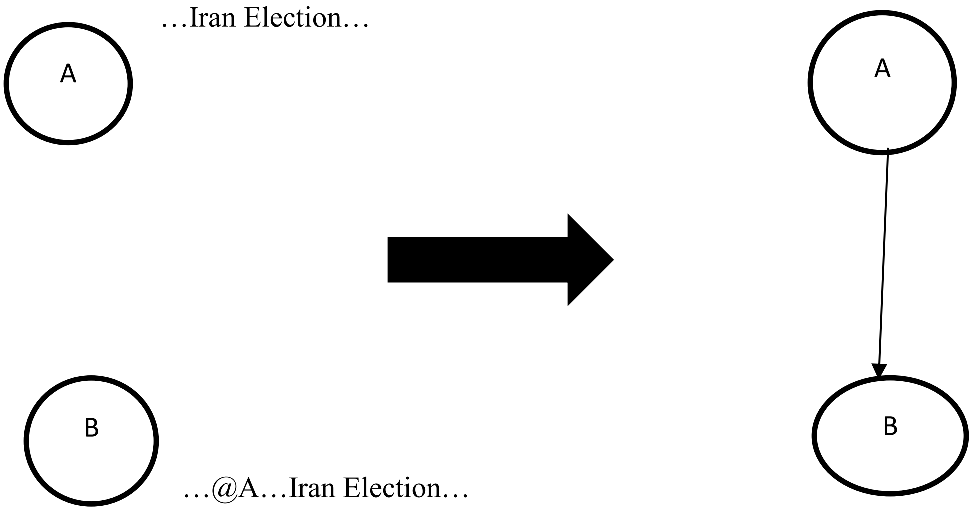


Fig 4: Diffusion link between users A and B when B mentions @A and talks about same topic i.e. Iran Election proposed by [23]

In fig 5, a diffusion network with timestamps is built. All the posts that contain the topic keywords are labeled with timestamps and diffusion links are built. The fully colored nodes are in the diffusion network while other light nodes with black outline are not counted. The black outlined nodes have mentioned the topic but without linking to any ancestor node.

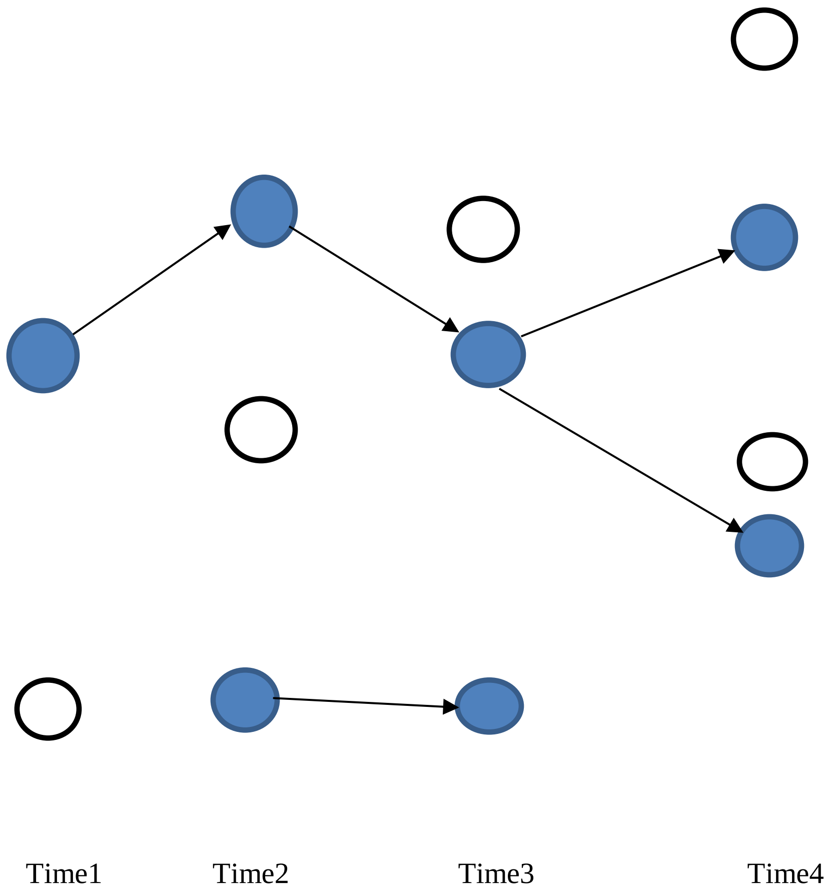


Fig 5: Diffusion Network over time

The authors developed models for three dimensions speed, scale and range in diffusion networks in Twitter. The relevant definitions follow:

Speed: Whether and when the first diffusion instance will take place

Scale: The number of affected instances at the first degree

Range: How far the diffusion network chain can continue on in depth.

Models to measure the above dimensions

Speed: When a post about a particular topic is seen, the most common question that occurs, is how the followers would be influenced, retweet and reply or mention the initial tweet in their tweet about the same topic. This question has two parts: whether one would mention at all and if so when will this mention happen [23]. The authors employed survival analysis to address both questions in a single model. Using this model, a prediction can be made when a tweet containing a topic is likely to be mentioned by another tweet also containing the topic. Cox proportional hazards regression model is used to quantify the degree to which a number of features of both users and tweets themselves predict the speed of diffusion to the first degree offspring.

The author considered few aspects of the twitter users and tweets to measure the speed, scale and range. The aspects of each individual author, such as their activity level in tweeting, mentioning and being mentioned may also predict the diffusion speed. Tweet characteristics whether a tweet contains a link, whether it itself is a mention, and stage: whether the tweet comes at an earlier stage or later stage in the topic lifespan. The authors simplified the stage variable by dividing the tweets into two sets: before and after 10 days after the first observation of the topic is made.

The author considered few aspects of the twitter users and tweets to measure the speed, scale and range. The aspects of each individual author, such as their activity level in tweeting, mentioning and being mentioned may also predict the diffusion speed. Tweet characteristics whether a tweet contains a link, whether it itself is a mention, and stage: whether the tweet comes at an earlier stage or later stage in the topic lifespan. The authors simplified the stage variable by dividing the tweets into two sets: before and after 10 days after the first observation of the topic is made.

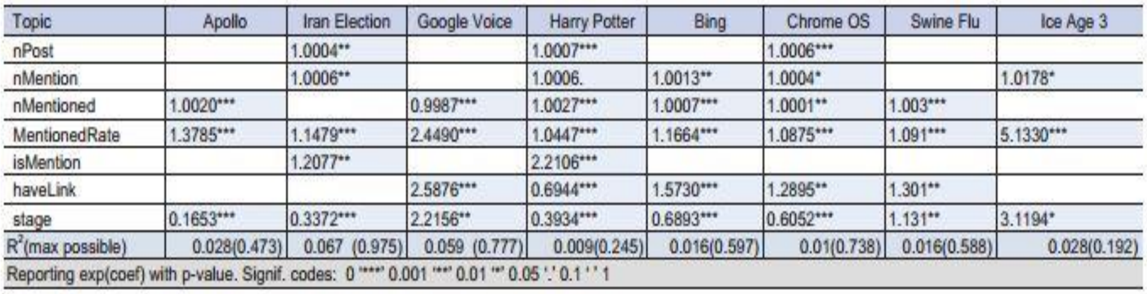
nPost: If the author is more active in posting the tweets

nMention: If the author is more active in mentioning other individuals

MentionedRate: The rate of the author being mentioned isMention: If the post is a mention

haveLink: Measures if the tweet has a link

If the value is greater than 1, a positive relationship exists between the predictor and speed of influence. Values less than 1 indicate negative relationship.



Topic	Apollo	Iran Election	Google Voice	Harry Potter	Bing	Chrome OS	Swine Flu	Ice Age 3
nPost		1.0004**		1.0007***		1.0006***		
nMention		1.0006**		1.0006.	1.0013**	1.0004*		1.0178*
nMentioned	1.0020***		0.9987***	1.0027***	1.0007***	1.0001**	1.003***	
MentionedRate	1.3785***	1.1479***	2.4490***	1.0447***	1.1664***	1.0875***	1.091***	5.1330***
isMention		1.2077**		2.2106***				
haveLink			2.5876***	0.6944***	1.5730***	1.2895**	1.301**	
stage	0.1653***	0.3372***	2.2156**	0.3934***	0.6893***	0.6052***	1.131**	3.1194*
R^2(max possible)	0.028(0.473)	0.067 (0.975)	0.059 (0.777)	0.009(0.245)	0.016(0.597)	0.01(0.738)	0.016(0.588)	0.028(0.192)

Reporting exp(coef) with p-value. Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

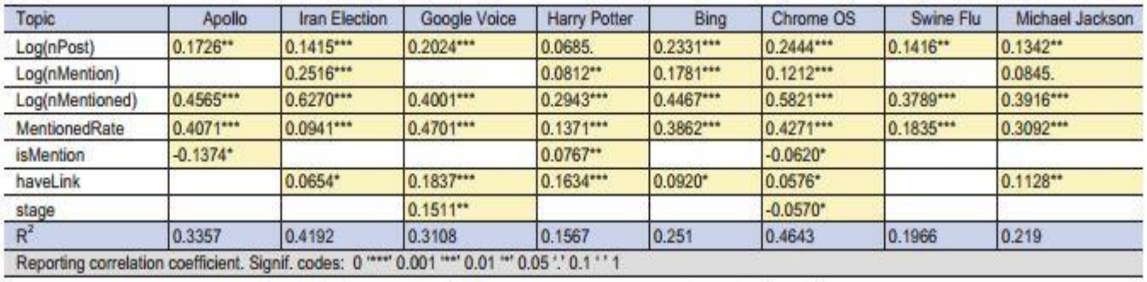
Fig 6: Predicting whether and when a post will get mentioned by an offspring node about the same topic. Only significant effects are shown [23]

A conclusion is made by [23] for the topic Iran Election, that when an author is more active in posting (nPost) and has a higher rate of being mentioned (MentionedRate), the present tweet gains an offspring in a shorter time. When a post is a mention, then it has a higher chance to continue diffusion. Whether a tweet contains a link does not affect the ability to generate the offspring nodes for the above topic Iran Election.

For all the topics in figure 6, MentionedRate is an important predictor to predict how fast a tweet on particular topic would be mentioned. Stage is also an important predictor. For some topics, earlier posts are more effective in producing an offspring where as for the topics Ice Age3 and Google Voice, tweets later in the observation period are more effective.

The above results suggest that a topic can have different diffusion efficiency at different stages of its life cycle.

Scale: For each tweet, how many people mentioned the same topic as first degree child nodes in the diffusion network? The author assumed that each user is only counted once for their first post about a given topic. The authors only predicted based on tweets that had at least one child node. Logarithm of these variables is measured. The following figure 7 shows the regression results on sample trending topics. R2 of the regression and correlation coefficient between the predictor and log (nChild) is mentioned in each cell [23].



Topic	Apollo	Iran Election	Google Voice	Harry Potter	Bing	Chrome OS	Swine Flu	Michael Jackson
Log(nPost)	0.1726**	0.1415***	0.2024***	0.0685.	0.2331***	0.2444***	0.1416**	0.1342**
Log(nMention)		0.2516***		0.0812**	0.1781***	0.1212***		0.0845.
Log(nMentioned)	0.4565***	0.6270***	0.4001***	0.2943***	0.4467***	0.5821***	0.3789***	0.3916***
MentionedRate	0.4071***	0.0941***	0.4701***	0.1371***	0.3862***	0.4271***	0.1835***	0.3092***
isMention	-0.1374*			0.0767**		-0.0620*		
haveLink		0.0654*	0.1837***	0.1634***	0.0920*	0.0576*		0.1128**
stage			0.1511**			-0.0570*		
R^2	0.3357	0.4192	0.3108	0.1567	0.251	0.4643	0.1966	0.219
Reporting correlation coefficient. Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1								

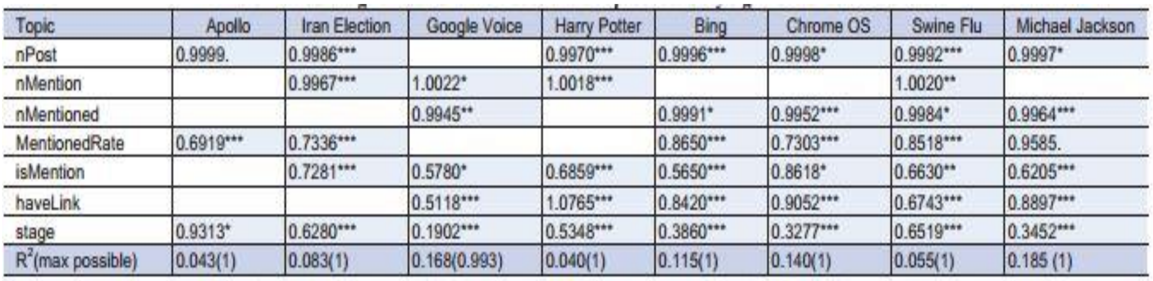
Fig 7: Predicting number of child nodes one can produce [23]

The author concluded that the activity level of the user and number of times she is mentioned are stable predictors for variance. A tweet having a link is often generates more children.

Range: The range of topic diffusion is measured by the number of hops in the diffusion chain [23]. The length of the original chain indicates how far the original node diffuses in depth. For most of the topics, half the ancestor nodes fail to produce the offspring nodes of the first degree and less than 30% has the second degree child nodes. After 5 hops, less than 5% of the ancestor nodes still continue to produce the child nodes [23].

In figure 8, the aspects of users and tweets are analyzed to predict the range of diffusion. The figure presents the predictors of the length of topic chain in a diffusion network [23].

The user activity in posting and being mentioned are greater predictors of the longer diffusion hops. A tweet being mentioned itself at a later stage is also a great predictor except for Harry Potter.



Topic	Apollo	Iran Election	Google Voice	Harry Potter	Bing	Chrome OS	Swine Flu	Michael Jackson
nPost	0.9999.	0.9986***		0.9970***	0.9996***	0.9998*	0.9992***	0.9997*
nMention		0.9967***	1.0022*	1.0018***			1.0020**	
nMentioned			0.9945**		0.9991*	0.9952***	0.9984*	0.9964***
MentionedRate	0.6919***	0.7336***			0.8650***	0.7303***	0.8518***	0.9585.
isMention		0.7281***	0.5780*	0.6859***	0.5650***	0.8618*	0.6630**	0.6205***
haveLink			0.5118***	1.0765***	0.8420***	0.9052***	0.6743***	0.8897***
stage	0.9313*	0.6280***	0.1902***	0.5348***	0.3860***	0.3277***	0.6519***	0.3452***
R^2(max possible)	0.043(1)	0.083(1)	0.168(0.993)	0.040(1)	0.115(1)	0.140(1)	0.055(1)	0.185 (1)

Fig 8: Predicting length of influence chain of ancestor nodes [23]

2.1.4 SOCIAL GRAPH AND CASCADE GRAPH MODEL

The social graph and cascade graph model is proposed by [19]. Information diffusion in real-time is studied using retweets on Twitter as starting point. The authors considered influencer as a friend/follower who exposes information to his/her followers and influence them to forward the piece of information [13]. The relationship of who was influenced by whom is determined by influence paths. The set of influence paths form a social graph, that share a common root is called information cascade.

Social Graph

The author considered a directed social graph SG = (V, F) of follower/friend relationships. V is the set of users and F denote the followers of V showing for each node/user from V who follows the user (F) is constructed. Each message has few attributes like timestamp t, user v € V and information item identifier i that are used [19].

Whenever a user forwards the same message, a cascade is formed. The author assigned two values for every node. The temporal order of sending retweets and the number of followers. A social graph is formed with friend follower relationship. From this social graph, a cascade graph is extracted that has the influence paths.

Cascade Graph

Cascade graph CG (U, E) with U ⊆ V is a directed graph of influence paths among users. U is the set of users and E represents the retweet relation between the users. CG is a subset of SG annotated with influence time on the edges. Cascade graph contains only those users as nodes who actually retweeted. Those who were exposed to the information and did not react are not included in the graph.

The author’s algorithm searches for these edges in the graph for all messages in Message stream. For real-time data, the edges shall be added incrementally whenever a message arrives.

For assessing connectivity of information cascades, two metrics are introduced that are described below.

Connectivity Rate

The connectivity rate assess whether there is a connection between two users/nodes in the cascade. It returns the percentage of users that have at least one connection and are thus influenced by another user. In the below formula, either (u_1, u) or (u, u_1) edges can belong to E.

Connectivity Rate=

$$\frac{|\{u|(u_1, u) \in E \lor (u, u_1) \in E\}|}{|U|}$$

Root Fragment Rate

The root fragment rate assess whether there is a path to the root user from every other user. It returns the percentage of users that are connected to the root directly or through an influence path over multiple users.

Root Fragment Rate=

$$\frac{|\{u_j \ \in \ U|\text{iff exists a path } u_r, . . , u_j \text{ in } C\}| \ |}{|U|}$$

The author used a subset of the dataset that contains cascades with more than 100 messages for testing. The cleaned dataset contains cascade with more than 90% of their messages acquired and having available more than 80% of the follower lists. For the cleaned dataset a connectivity rate of 85% and root fragment rate of 80% is obtained. The author then extended evaluations to the full dataset where he observed that the connectivity rate and root fragment rate are dropped.

The author concluded that for 20% of the cascades, we get more than 80% connectivity rate and 70% root fragment rate. In ideal cases which have message completeness 99% and follower lists 95%, a connectivity rate CR=93% and RFR=90% is obtained. From this, the author concluded that social links are indeed the predominant carriers of information. However, there are still 10% messages that cannot be assigned using social graph information. That means, either the user has no social connections available (deleted or private account), or the user forwarded a message without having a direct link to any of the previous (re)tweeters (forwarded it from the public Timeline where messages of non-followers are depicted).

2.1.5 PROBABILISTIC COLLABORATIVE FILTER MODEL: MATCHBOX

A probabilistic collaborative filter model is proposed by [24]. This predictive model was originally developed to predict the movie preferences of the users based on meta-data about movies. Using the data of who and what was retweeted, a probabilistic collaborative filter model is trained to predict future retweets [24]. The author named this model as Matchbox model. The input of the model is the tweeter, retweeter and content of the tweet. The output will be a p value that is the probability of a retweet of the tweet by the retweeter.

The author has used the following features

- Tweeter's features

  The tweeter's name and number of followers of the tweeter etc.

- Retweeter's features

  The retweeter's name and number of followers of the retweeter etc.

The above features are divided into item and user features by different methods and different models are trained to see which division works best. The binary feedback would be 1 if the retweeter retweets the tweet within a specific time window else 0. A time window of 1 hour is used because half of the retweets occur within an hour of the source tweet.

Positive and Negative Feedback

Positive binary feedback is required for training Matchbox which is obtained by collecting retweets. The negative feedback is generated by the followers who do not retweet the tweet within an hour.

By selecting unique tweet and retweeter pairs from the collected data, a network of users is obtained.

Generation of Training Data

For every tweet from the time of generation till an hour, all the retweets of that particular tweet are collected in that one hour time window which forms the positive binary feedback. Negative feedback is obtained from all the followers of the tweeter who did not retweet. The author's data has 99.8% negative feedback as most tweets were not retweeted.

2.2 USER INFLUENCE DETECTION

In real-world scenarios, an individual accepts any new piece of information based on his/her interest as well as friends influence [9]. Each information sender influences its neighbors with some probability [7]. If the probability is too high, the user can adopt the opinion [10].

The following are some of the models for influence computation.

2.2.1 INDEGREE, RETWEETS AND MENTIONS MODEL

Indegree, retweets and mentions are considered for influence computation in [6]. In Twitter, one way to measure influence of any user is based on the number of retweets that happen to their original tweets [6]. The author uses a part of this retweet consideration but not completely because it doesn't measure the influence of the root user for the grandchildren. Some other methods describe the influence based on the number of reply's he/she has got for his/her tweets. The relevant definitions follow: [6]

Indegree: The number of people who follow a user.

Retweet: The number of times others "forward" a user's tweet

Mentions: The number of times others mention a user's name

Indegree influence: The number of followers of a user is measured by indegree.

Retweet influence: The number of retweets containing one's name indicate value of the user generated content.

Mention influence: The number of mentions indicates the ability of the user to engage others in a conversation.

The author collected the data in august 2009 for all user IDs from 0 to 80 million. There is no single user connected to user ID greater than 80 million. Out of the collected 80 million IDs, only 54,981, 152 users are active which were connected by social links. The authors gathered information about user's follow links and all tweets ever posted by a user. The private accounts were ignored by [6].

The authors mentioned that the social link is based on final snapshot of the network topology at the time the data was collected and no idea about the links were made. Indegree measures the popularity of a user, retweets represent content value and mentions measure name value of a user [6]. The information about user's social links and tweets are collected and the value of each influence measure is computed which are then compared. The users are ranked based on the 3 influence measures. The relative ranking is given based on the Spearman's rank correlation coefficient.

$$\rho = 1- 6\sum (x_i-y_i)^2 / (N3-N)$$

xi and yi are the ranks of users based on two different influence measures in a dataset of N users.

The user who has highest ρ is said to be more influential.

2.2.2 INFLUENCE INDEPENDENT OF DIFFUSION MODEL

The author of [8] said that influence can be computed without calculating diffusion first. Avoiding the need to compute diffusion first, influence is learned directly from cascade data. A random graph of users is first sampled from the distribution induced by a particular diffusion model and then a function is defined for node reachability in the sampled graph [8]. The expectation of this function is the influence for the random graphs. The author suggested that a sum over conditional probabilities is used to learn influence function.

Random Reachability function

Each sampled random graph ‘G’ is represented by a binary reachability matrix

$R \in \{0, 1\}$ d×d

$$R_{sj} = \begin{cases} 1, & j \text{ is reachable from source } s, \\ 0, & \text{otherwise} \end{cases}$$

Sth row indicates the information that if s is the source, which nodes are reachable from it.

Jth column indicates if j is reachable from other nodes.

d×d indicates the dimension of the matrix

Compute node reachability

With a given set of sources S, whether a node j will be influenced or not in graph G is computed with a simple non-linear function ϕ defined as converge function below.

Firstly,

Given a set S of sources, S is represented as an indicator vector χS

$\chi S \in \{0, 1\}$ d , with ith entry

Where χS is an indicator vector of sources S

The below formula is given by [8]

$$\chi S\ (s) = \begin{cases} 1, s \in S \\ 0, \text{otherwise} \end{cases}$$

Inner product

Inner product tells us if the target node j is reachable from any of the source nodes in S.

$$\chi_S^\top \boldsymbol{R}_{:j} \in \mathbb{Z}_+$$

Target node j is reachable if $\chi_S^\top \boldsymbol{R}_{:j} \geq 1$

If the target node j is not reachable, the above value is ‘0’

Concave function

A concave function φ(u) is used to transform $\chi_S^\top \boldsymbol{R}_{:j}$ into a binary function.

$$\varphi(u) = \min\ \{u, 1\} : Z+ \rightarrow \{0, 1\}$$

Coverage Function

The coverage function is used to compute influence.

$$\phi\left(\chi_S^\top \boldsymbol{R}_{:j}\right) \quad : \quad 2^{\mathcal{V}} \mapsto \{0,1\}$$

Where V is the set of nodes.

The influence of ‘S’ in graph G is the number of target nodes reachable from the source set S.

$$\#(\mathcal{S}|\boldsymbol{R}) := \sum_{j=1}^{d} \phi\left(\chi_{\mathcal{S}}^{\top} \boldsymbol{R}_j\right)$$

d is the total number of nodes in the user graph.

Expectation for random functions

The overall influence of a source set ‘S’ in a diffusion model is the expected value of # (S|R)

i.e. σ (S) := ER~PR [#(S|R)]

PR is the distribution over binary matrix R.

Hence, the author confirms that the influence can be learned directly from cascade data without computing diffusion first.

2.2.3 GENERAL THRESHOLD MODEL

General threshold model is proposed by [12]. At any given timestamp, any node u is either active or inactive. Each node’s tendency to become active increases as most of its neighbors become active [12]. With time, more and more neighbors of u become active increasing the chances of u to become active which further triggers its neighbors to become active [23].

The activation threshold ø$_u$ is chosen independently and uniformly at random from the interval [0, 1]. In this model each node has an activation function:

fu: 2N(u) → [0, 1]

N(u) – Set of neighbors of u

Node u becomes active at t+1 if f$_u$(S) >= ø$_u$

S – Set of neighbors of u that are active at time t

The author computed the influence probability by considering the following graphs.

Social Graph

An undirected social graph G = (V, E, T) is considered by the author.

V is the set of users and E is the social ties between the users.

T is the timestamp for the edge i.e. at which the social tie was created between any two users.

Action Log

An action log is also considered which is in the following form.

Actions (User, Action, Time) which has a tuple (u, a, t)

This indicates a user u has performed a particular action a at time t.

Action Propagation

The author denoted universe of actions by A and social tie between users by E. An action a € A propagates from user v_i to v_j if and only if

(v_i, v_j) € E

$\exists(v_i, a, t_i),(v_j, a, t_j) \in$ Actions with $t_i < t_j$

There must be a social tie between v_i and v_j, both must have performed the action after their social tie has been created. This forms a propagation graph defined below.

Propagation Graph

The users who performed the action are included in the propagation graph with the edges connecting in the direction of propagation. When a user performs an action, he is activated and has the ability to activate all the inactive friends. The power to influence the neighbors is called influence probability. At any time, the user v tries to influence its inactive neighbor u has a fixed probability of making u active. The influence probability is the ratio of number of successful attempts over the total number of trials which is known as Jaccard index. The Jaccard index is often used to measure similarity between sample sets and is defined as the size of the intersection divided by the size of union of the sample sets [12]. The author adopted Jaccard index to estimate $P_{v,\,u}$ as follows:

$$P_{v,\,u} = A_{v2u} / A_v$$

The probability is given by the following function.

$$p_u(S) = 1 - \prod_{v \in S} (1 - p_{v,u})$$

P (v,u) is the probability of v influencing u.

CHAPTER III

DATA COLLECTION AND DATA STORAGE

This thesis research focuses on algorithm design and implementation to analyze information diffusion and determination of user influence. Also, two user influence models are compared. For testing and comparison, Twitter data sets are used.

3.1 DATA COLLECTION

In this research, the initial part is to collect the data from Twitter through Twitter Application Programming Interface (API) [11]. The streamed Twitter data is stored in HDFS. The data are collected in three domains, namely Flu, Food Poisoning and Trump Politics. The size of Flu data is 200GB, Food Poisoning data size is 20GB and Trump Politics data size is 200GB.

The Hadoop cluster has 24 nodes with node names hadoop1-hadoop24. Hadoop1 is the name node. Hadoop2-Hadoop24 are all data nodes. Each node has 2 CPU cores, 8GB RAM, 500GB Hard disk. Fedora 21 operating system is installed on all the nodes.

3.1.1 APACHE HADOOP

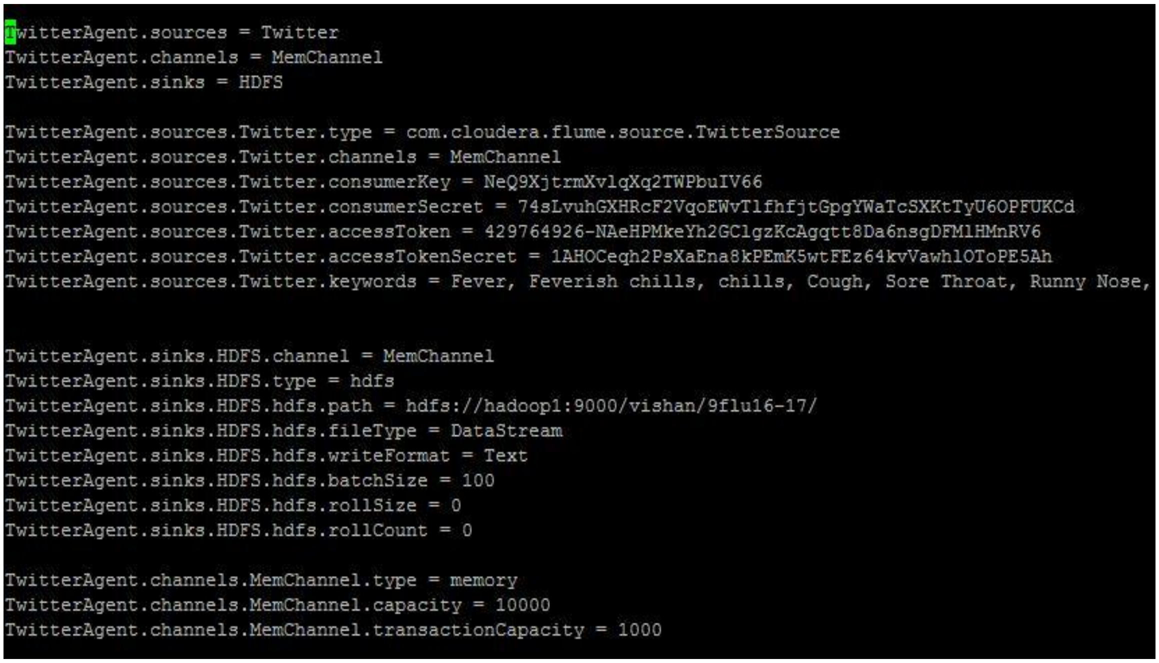
Apache Hadoop is an open-source software that provides scalable, reliable and distributed computing. For us to start the streaming of twitter data with flume, we should have Hadoop installed either in standalone mode or multi-node cluster.

3.1.2 APACHE FLUME

Apache Flume is an open-source software that helps to store the streaming data on HDFS. A flume agent should be created through which we can stream the data. The following steps describe the process to collect the twitter data.

1. Create an account for yourself in twitter and login with the credentials.
2. Navigate to https://apps.twitter.com/ and create a new app.
3. Get the consumer secret, consumer token, access token and access token secret for your application.
4. There are 3 components for a twitter agent namely source, sink and channel.
5. The flume source connects to Twitter API and receives data in JSON format which in turn are stored in HDFS.
6. Add the flume source to the flume class-path.
7. Now, create a configuration file for the flume agent by specifying the consumer key, consumer secret, access token and access token secret and keywords, hdfs sink path.

A sample configuration file which I have used is shown in Figure 9. It shows all the keys and keywords to be used to collect the twitter data.



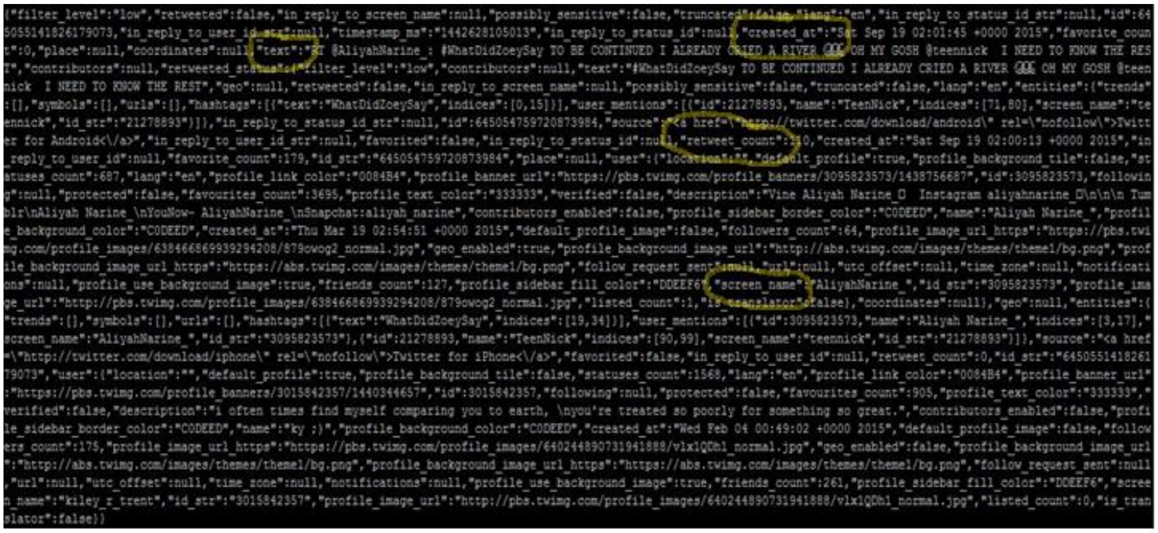
```
TwitterAgent.sources = Twitter
TwitterAgent.channels = MemChannel
TwitterAgent.sinks = HDFS

TwitterAgent.sources.Twitter.type = com.cloudera.flume.source.TwitterSource
TwitterAgent.sources.Twitter.channels = MemChannel
TwitterAgent.sources.Twitter.consumerKey = NeQ9XjtrmXvlqXq2TWPbuIV66
TwitterAgent.sources.Twitter.consumerSecret = 74sLvuhGXHRcF2VqoEWvTlfhfjtGpgYWaTcSXKtTyU6OPFUKCd
TwitterAgent.sources.Twitter.accessToken = 429764926-NAeHPMkeYh2GClgzKcAgqtt8Da6nsgDFMlHMnRV6
TwitterAgent.sources.Twitter.accessTokenSecret = 1AHOCeqh2PsXaEna8kPEmK5wtFEz64kvVawhlOToPE5Ah
TwitterAgent.sources.Twitter.keywords = Fever, Feverish chills, chills, Cough, Sore Throat, Runny Nose,


TwitterAgent.sinks.HDFS.channel = MemChannel
TwitterAgent.sinks.HDFS.type = hdfs
TwitterAgent.sinks.HDFS.hdfs.path = hdfs://hadoop1:9000/vishan/9flu16-17/
TwitterAgent.sinks.HDFS.hdfs.fileType = DataStream
TwitterAgent.sinks.HDFS.hdfs.writeFormat = Text
TwitterAgent.sinks.HDFS.hdfs.batchSize = 100
TwitterAgent.sinks.HDFS.hdfs.rollSize = 0
TwitterAgent.sinks.HDFS.hdfs.rollCount = 0

TwitterAgent.channels.MemChannel.type = memory
TwitterAgent.channels.MemChannel.capacity = 10000
TwitterAgent.channels.MemChannel.transactionCapacity = 1000
```

Fig 9: Twitter Configuration File

After starting the flume agent, check for data in hdfs sink path to see if the data is getting collected properly. Counting the number of files collected regularly is one of the methods to check if the data streaming is continuous. The other way is to check the log file for any errors occurred in the data streaming process. Sometimes, the data collection stops when there is a power outage as the Hadoop cluster gets shutdown. Once the power is back, we can restart the flume data collection process.

The data collected is in JSON format. A sample data file is shown in Figure 10.

There are many fields in the json data file which can be used for different types of analysis. I have circled few fields in yellow ink. The data associated with the respective fields is obtained by filtering the raw json data by using a python script.



```
{"filter_level":"low","retweeted":false,"in_reply_to_screen_name":null,"possibly_sensitive":false,"truncated":false,"lang":"en","in_reply_to_status_id_str":null,"id":64
5055141826179073,"in_reply_to_user_id_str":null,"timestamp_ms":"1442628105013","in_reply_to_status_id":null,"created_at":"Sat Sep 19 02:01:45 +0000 2015","favorite_coun
t":0,"place":null,"coordinates":null,"text":"RT @AliyahNarine_: #WhatDidZoeySay TO BE CONTINUED I ALREADY CRIED A RIVER □□□ OH MY GOSH @teennick  I NEED TO KNOW THE RES
T","contributors":null,"retweeted_status":{"filter_level":"low","contributors":null,"text":"#WhatDidZoeySay TO BE CONTINUED I ALREADY CRIED A RIVER □□□ OH MY GOSH @teen
nick  I NEED TO KNOW THE REST","geo":null,"retweeted":false,"in_reply_to_screen_name":null,"possibly_sensitive":false,"truncated":false,"lang":"en","entities":{"trends"
:[],"symbols":[],"urls":[],"hashtags":[{"text":"WhatDidZoeySay","indices":[0,15]}],"user_mentions":[{"id":21278893,"name":"TeenNick","indices":[71,80],"screen_name":"te
ennick","id_str":"21278893"}]},"in_reply_to_status_id_str":null,"id":645054759720873984,"source":"<a href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitt
er for Android<\/a>","in_reply_to_user_id_str":null,"favorited":false,"in_reply_to_status_id":null,"retweet_count":110,"created_at":"Sat Sep 19 02:00:13 +0000 2015","in
_reply_to_user_id":null,"favorite_count":179,"id_str":"645054759720873984","place":null,"user":{"location":"","default_profile":true,"profile_background_tile":false,"st
atuses_count":687,"lang":"en","profile_link_color":"0084B4","profile_banner_url":"https://pbs.twimg.com/profile_banners/3095823573/1438756687","id":3095823573,"followin
g":null,"protected":false,"favourites_count":3695,"profile_text_color":"333333","verified":false,"description":"Vine Aliyah Narine_□  Instagram aliyahnarine_□\n\n\n Tum
blr\nAliyah Narine_\nYouNow- AliyahNarine_\nSnapchat:aliyah_narine","contributors_enabled":false,"profile_sidebar_border_color":"C0DEED","name":"Aliyah Narine_","profil
e_background_color":"C0DEED","created_at":"Thu Mar 19 02:54:51 +0000 2015","default_profile_image":false,"followers_count":64,"profile_image_url_https":"https://pbs.twi
mg.com/profile_images/638466869939294208/879owog2_normal.jpg","geo_enabled":true,"profile_background_image_url":"http://abs.twimg.com/images/themes/theme1/bg.png","prof
ile_background_image_url_https":"https://abs.twimg.com/images/themes/theme1/bg.png","follow_request_sent":null,"url":null,"utc_offset":null,"time_zone":null,"notificati
ons":null,"profile_use_background_image":true,"friends_count":127,"profile_sidebar_fill_color":"DDEEF6","screen_name":"AliyahNarine_","id_str":"3095823573","profile_ima
ge_url":"http://pbs.twimg.com/profile_images/638466869939294208/879owog2_normal.jpg","listed_count":1,"is_translator":false},"coordinates":null},"geo":null,"entities":{
"trends":[],"symbols":[],"urls":[],"hashtags":[{"text":"WhatDidZoeySay","indices":[19,34]}],"user_mentions":[{"id":3095823573,"name":"Aliyah Narine_","indices":[3,17],"
screen_name":"AliyahNarine_","id_str":"3095823573"},{"id":21278893,"name":"TeenNick","indices":[90,99],"screen_name":"teennick","id_str":"21278893"}]},"source":"<a href
=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitter for iPhone<\/a>","favorited":false,"in_reply_to_user_id":null,"retweet_count":0,"id_str":"6450551418261
79073","user":{"location":"","default_profile":true,"profile_background_tile":false,"statuses_count":1568,"lang":"en","profile_link_color":"0084B4","profile_banner_url"
:"https://pbs.twimg.com/profile_banners/3015842357/1440344657","id":3015842357,"following":null,"protected":false,"favourites_count":905,"profile_text_color":"333333","
verified":false,"description":"i often times find myself comparing you to earth, \nyou're treated so poorly for something so great.","contributors_enabled":false,"profi
le_sidebar_border_color":"C0DEED","name":"ky ;)","profile_background_color":"C0DEED","created_at":"Wed Feb 04 00:49:02 +0000 2015","default_profile_image":false,"follow
ers_count":175,"profile_image_url_https":"https://pbs.twimg.com/profile_images/640244890731941888/vlx1QDhl_normal.jpg","geo_enabled":false,"profile_background_image_url
":"http://abs.twimg.com/images/themes/theme1/bg.png","profile_background_image_url_https":"https://abs.twimg.com/images/themes/theme1/bg.png","follow_request_sent":null
,"url":null,"utc_offset":null,"time_zone":null,"notifications":null,"profile_use_background_image":true,"friends_count":261,"profile_sidebar_fill_color":"DDEEF6","scree
n_name":"kiley_r_trent","id_str":"3015842357","profile_image_url":"http://pbs.twimg.com/profile_images/640244890731941888/vlx1QDhl_normal.jpg","listed_count":0,"is_tran
slator":false}}
```

Fig 10: JSON Data

CHAPTER IV

IMPLEMENTATION

There are several information diffusion models available in the literature with different perspectives. In the related work section we have reviewed several models and analyzed their differences. Summary of our analysis can be found in Appendix 4. This chapter is primarily concerned with implementation of models presented in [20].

4.1 TOPIC PROPAGATION

Social networks play an important role in information diffusion [8], understanding which is the objective of this research. We assume that information is represented by “topics” and the strength/volume of propagation is defined by a potential. In this work, algorithms are implemented based on a previously defined model for information diffusion.

The twitter data is used to investigate information propagation. This thesis is concerned with topic propagation analysis in twitter data. There are several models proposed in the literature to measure the intensity of topic propagation. In this thesis, we consider the Topic Diffusion Model proposed by [20]. We develop implementation methods for the potential model. We also implement the diffusion network model proposed by [23] for comparison purpose.

The data that has been collected from twitter into HDFS through Twitter API is in json format and contains many pieces of information. Certain fields are necessary for this research. In this research, tweet propagation is computed first which is followed by user influence. For computing both of them, the following five fields are used.

1. Tweet_Text

   Tweet_text is an important field that denotes topic. The topic for which we are computing the diffusion is based on the Tweet_text.

2. This_Timestamp

   This_timestamp denotes the time at which the user has tweeted/retweeted the tweet. It is the most critical field in this research for calculating the tweet propagation.

3. This_User

   This_user denotes the current user who has tweeted/retweeted the tweet. This helps in finding out how the tweet propagated from one to another.

4. Owner

   Owner represents the user from which this_user [current user] has retweeted the tweet. Both the field's owner and this_user helps in finding out the influence of the users.

5. Owner_Timestamp

   Owner_timestamp is the time at which the owner has created the tweet. Owner_timestamp is used to compute the tweet propagation.

Level of a Tweet

The level indicates the hops taken by a tweet as retweets. All the original tweets are considered to be of level '0'. Retweets have levels greater than zero. When a user retweets a tweet, the retweet's level is increased by '1' [20]. In general if the level of a tweet/retweet is l, then the level of its retweet will be l+1.

Potential of a Tweet

Once the level of the tweets are computed, potential of the tweets can be calculated.

$P\ (tw) = \sum (\text{keyword weight}).\ \beta^{l}$

Keywords define a topic. They are used to identify tweets related to a particular topic. Every keyword is given a weight in the topic.

‘β’ is a constant that serves as a scale factor.

‘l’ is the level of the tweet.

The summation of the tweet potentials at regular time intervals gives the topic potential in those time intervals.

Calculating potential of the topic based on the level and potential of the tweets.

$P\ (T) = \sum P\ (tw)$

Where,

$P\ (T)$ is the potential of the topic. $P\ (tw)$ is the tweet potential.

4.1.1 DATA DESCRIPTION

The extracted Twitter Data is in JSON format. A sample tweet in JSON format is shown below:

{"filter_level":"low", "retweeted":false, "in_reply_to_screen_name":null, "truncated":false, "lang":"en", "in_reply_to_status_id_str":null, "id":665536821493796864, "in_reply_to_user_id_str":null, …. }

In this research, tweet propagation is computed using 3 map-reduce jobs written in python. We use map-reduce framework here in order to handle the large input datasets. The first map-reduce job is to filter the raw json data. The second map-reduce job deals with computing the levels of the tweets. The third map-reduce job deals with computation of topic potential. The map-reduce jobs are described in the sections that follow.

4.1.2 FILTERING THE TWITTER DATA

The first job is to filter the data collected. Out of all the available fields in the JSON data, only the required fields for this research are extracted. The fields extracted in this research for tweet propagation are:

Tweet_text, This_User, This_Timestamp, Owner_Timestamp, Owner

Partitioning the tweets alphabetically

After extracting the above five fields from all the tweets, we need to partition them alphabetically. Partitioning is done by taking the first letter of the tweet_text. All the tweets starting with the same letter are grouped and sent to the same reducer. The tweets starting with special characters are grouped together. Each group is sent to a reducer for processing. Therefore, we need a total of 27 reducers.

Appendix 12 shows the flowchart for filtering the twitter data.

4.1.3 TWEET LEVEL COMPUTATION

The level computation job computes levels of tweets. The length of retweet chains of a tweet at a point in time is determined by computing the level of a tweet. The output obtained by filtering the raw json data done by the first job is taken as input by the second map-reduce job that has the tweets sorted by a custom partitioner. A tree called the User-Tree is constructed for the set of tweets/retweets with the same text. The date of tweet/retweet is stored in the node along with the user of the tweet. The root node of the tree is a dummy node. The children at the first level represent original tweeters called root user for that particular tweet and the rest of the users who retweet are connected as descendant nodes to the root user based on their retweets. The levels are calculated for the tweet from the User-Tree on the available dates.

A flowchart for the User-Tree construction is shown in **Appendix 13**. The User-Tree construction and level computation process is explained in Example 1.

EXAMPLE 1: User-Tree construction and level computation

This example illustrates the User-Tree construction and level computation.

INPUT DATA

The following is the input dataset for which level of all the tweets will be computed.

- if you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day   Thu Sep 24 16:58:40 +0000 2015,dxtarun N/A,N/A
- if you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day   Tue Sep 08 22:25:45 +0000 2015,diiy_hacks   N/A,N/A
- if you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day   Tue Sep 08 22:32:06 +0000 2015,lexi_corona   diiy_hacks,Tue Dec 27 22:30:56 +0000 2011
- if you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day   Thu Sep 10 03:00:22 +0000 2015,BethaniaG   TheDIYHacks,Sun Dec 02 11:20:07 +0000 2012
- if you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day   Thu Sep 11 03:00:26 +0000 2015,diiy_hacks   lexi_corona,Tue Sep 08 22:32:06 +0000 2015

USER-TREE CONSTRUCTION

(Root, Root) is the dummy node for initializing a tree

- The first tweet is an original tweet and hence we add the user and this_timestamp directly to the dummy node.
- The second tweet is also an original tweet and hence we add the user and this_timestamp to the dummy node again.
- The third tweet is a retweet of the second tweet. As the owner of the third tweet is already present in the tree, we add the user of third tree to its owner i.e. diiy_hacks
- The fourth tweet is a retweet but, its owner is not in the tree so far constructed. So, we first add the owner to the dummy node of the tree followed by its user.

The tree after adding all the users will be shown in Figure 11

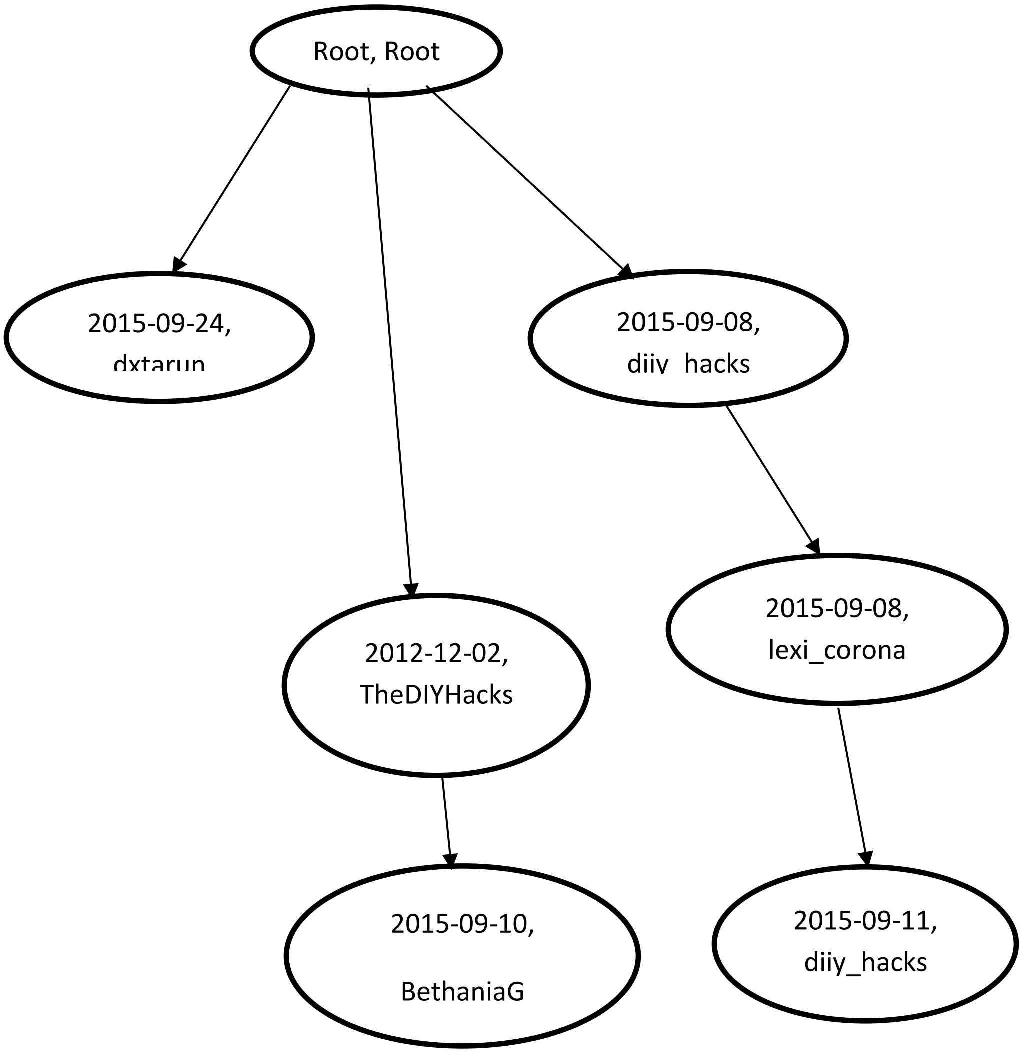


Fig 11: User-Tree illustration for example

TWEET LEVELS COMPUTATION BY TRAVERSING THE USER-TREE

A tree with all the users of a similar tweet is constructed as above. We utilize a data structure-dictionary “tweet_levels = {date: level}” where date represents the calendar dates and level represent a list of integers which are levels of tweets to store the levels of the tweets/retweets with date and levels on that particular date. We initialize the variable “level = 0”. While traversing the tree, we increase the variable level by ‘1’ for every child node. This way, we get the following levels:

{2015-09-24: 1
2015-09-08: 1, 2
2015-09-10: 2
2015-09-11: 3}

The date range is 09/01/2015 – 09/30/2015. The data shown in the sample input in example 1 is collected in this date range 09/01/2015 – 09/30/2015.

We initialize another list “this_propagation” with all 0’s in the dates from the given date range. We will have to replace the 0’s on particular days when there exists tweet levels on those dates.

```
this_propagation = ['0'] * (parser.parse(endDate) - parser.parse(startDate)).days
```

Now, the variable “this_propagation” is populated as follows with all 0’s in all the dates from 09/01/2015 to 09/30/2015

```
0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
0    0    0    0    0    0    0    0    0    0    0    0
```

Now, we update “this_propagation” with the tweet levels which are available in “tweet_levels” dictionary and attach the tweet_text before the levels.

FINAL OUTPUT

```
if you drink enough fluids in the morning you will feel happier sharper and more energetic
throughout the day   0    0    0    0    0    0    0    1,2    0    2    3    0    0    0
0    0    0    0    0    0    0    0    0    1    0    0    0    0    0    0
```

We will see the process of user-tree construction and level computation by traversing the tree in the below sections.

4.1.3.1 USER-TREE CONSTRUCTION

The user-tree construction takes the sorted tweets as input. All the similar tweets are grouped before the user-tree construction. All the users involved in similar tweets/retweets form the nodes of the tree. A variable previous_tweet is maintained to keep track of the tweets. When a first tweet is read from the input file, the variable previous_tweet is loaded with the tweet_text. For the rest of the tweets, tweet_text is compared with the previous_tweet variable value. All the users involved in similar tweets/retweets form a tree. Once, the first set of tweets are done and a different tweet occurs, tweet levels are computed from the tree so far formed for the first group of tweets that is discussed in the below sections. After the level computation is done, the tree is re-initialized to (root, root). This process continues for all the tweets

4.1.3.2 TWEET LEVELS COMPUTATION BY TRAVERSING THE TREE

From the above section, we have seen the construction of a user-tree. Every node contains the user who created the tweet and the date when the tweet is created. After a user-tree is formed, we compute tweet levels by traversing the tree from the root. A variable level_count is maintained to keep track of the tweet levels. While traversing the tree, each time a child node is encountered, level_count variable is incremented by one. Once, all the levels of the tweets/retweets for similar tweets/retweets are found, we re-initialize the tree and proceed with other group of tweets. A tweet_levels dictionary is maintained to store tweet levels every day.

The flowchart for level computation is shown as part of **Appendix 14**.

4.1.4 POTENTIAL COMPUTATION FOR TWEETS

The potential computation job is used to compute the potential of the tweets for each day. Here, we use a map-reduce job to compute the topic potential because we need to deal with large datasets. Also, the mapper job outputs the tweet potential on particular days while the reducer sums up the tweet potential giving us the topic potential. The output of the level computation map-reduce job is taken as input by the potential computation map-reduce job. The mapper here calculates the potential of tweets every day. The reducer sums up the potential of tweets occurred on same day giving the potential of topic on every day. A sample input line for this job is given below:

if you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day 0 0 0 0 0 0 0 1,1 2,2,1 3,3,2,1 4,4,3,2,1 5,5,4,3,2 0 0

The text “if you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day” is the tweet_text and the numbers are the levels on each day separated by a comma. The flowchart is shown in **Appendix 15.**

4.2 DIFFUSION NETWORK MODEL

The diffusion network model is proposed by [23]. Connected user count is used as diffusion measure. A time-series of connected users is built in the diffusion network model. The date range depends on when the data is collected. When a user ‘B’ retweets a tweet of user ‘A’, we say that the users ‘B’ and ‘A’ are connected as B → A. The number of such connections are counted on every day present in the time-series. These counts are plotted against the dates. This graph is compared with the topic potential graphs discussed in the above section which is an objective of this research.

The algorithm to compute the diffusion network model is described by the flowchart shown in **Appendix 16**.

Connections

We consider an example to explain the term connection. For example, a user “X” has tweeted/retweeted a tweet on day1. A user “Y” has retweeted X’s tweet/retweet on day2. We say a connection exists on day2 from Y → X and the count of the connections on day2 is one.

Flowchart description

In the above section 4.1.2, the first map-reduce job filters the raw json data and gives the five fields namely tweet_text, this_user, this_timestamp, owner, owner_timestamp. The output of the first map-reduce job is used as input to the diffusion network model. Ignore the original tweets. For retweets, check if there is a connection on this_timestamp. If there is no connection on this_timestamp, we add user → owner connection on this_timestamp. If some connections already exists on this_timestamp, we check if this_user is present in the connections. If the user doesn’t exist, we then add the user → owner connection. If the user already exists, we process the next input lines. Finally, we count the number of connections on every day.

4.3 USER INFLUENCE

User influence measures the influence of a user in Twitter. A twitter user is said to be influential if the user’s tweets are retweeted by other users. There are few researchers who compute the influence of a user based on the number of followers of a user, number of times that user is being mentioned in other users tweets. In this research, we assume a user is influential if more number of retweets occur to the tweets done by him/her.

ASSUMPTIONS

My assumption for this paper to compute the user influence is that all the twitter users are equal. User influence computed is the influence of the users who have participated in tweeting or retweeting the tweets only. We do not compute the influence of the users whose tweets are not retweeted. For example say, twitter user tweets:

"People suffering from flu are eligible for free treatment."

If no body retweets this tweet, we do not calculate influence for the user since the tweet has no contribution towards user influence for the flu topic. Based on our influence computation, such a user would not be considered as influential even if the user has high influence in terms of power and name recognition.

To compute user influence, based on retweets, we present two approaches:

1. Multi-Level Marketing Method
2. Root User Benefits Method

4.3.1 MULTI-LEVEL MARKETING

Multi-level marketing is also called network marketing or pyramid selling. Not only the direct investor gains profit but the middle marketers who are responsible for the sales also gain percentage of profit leading to multi-level marketing.

In this research, multi-level marketing strategy is applied for user influence computation over time. In this strategy, partial influences to all the parents starting from the root user is assigned.

The computation of user influence is done in 3 map-reduce jobs. The first map-reduce job filters the tweets as described in the section 4.1.2. The second map-reduce job computes the levels of tweets. The third map-reduce job computes the influence of every user.

The flowcharts given in Appendix 17, Appendix 18, and Appendix 19 denote the implementation of the three stages.

4.3.1.1 TWEET LEVEL COMPUTATION

We have used a different approach to compute the tweet levels and tweet potential individually because we need to have the tweet potential separately in order to use it in the user influence computation formula. After filtering of raw json data is done, we take the output and give it as input to the first job here. We group all the similar tweets by comparing the tweet_text in a tweet_list dictionary. Now, we sort all the tweets based on this_timestamp using an inbuilt sort function in python. For every tweet in the input file, we compare its owner with the user of other tweets. Whenever a match is made, we increase the level by one associated with the parent tweet. Finally, we get the levels of every tweet in the input file. The flow chart is given in Appendix 17.

4.3.1.2 TWEET POTENTIAL COMPUTATION

From the above tweet level computation, we will have a level for every tweet. Now, we have to compute the tweet potential. We start by checking the tweet_text with the words in the keywords list. We assign a default weight of one to all the keywords. We initialize a variable named “count” with zero. Whenever a keyword from the keywords list is found in the tweet_text, we increase the count variable by one. The flowchart is in Appendix 20. We then compute the tweet potential by using the formula

$$\text{Tweet\_Potential} = \text{count}*\text{pow}\ (\text{rho}, \text{level})$$

4.3.1.3 USER INFLUENCE COMPUTATION

From the previous computation, we get the tweet_potential of every tweet. Now, we need to compute the influence of all the users. We initialize a dictionary named “influence” to store the dates and influence of the users on those dates. We initialize a variable named “relative_level” to store the relative level of a tweet with respect to another tweet. We group the similar tweets based on the tweet_text. We then start by taking the first tweet and check its owner. If owner is none, proceed with the next input line. If owner exists, increment the relative_level variable by one. Compute influence of the owner of the tweet by using the formula:

Influence [owner] = tweet_potential*POW (rho, relative_level)

Now, check the influence dictionary if the owner entry is made. If the owner is not present in the dictionary, add this_timestamp, owner and their influence in the influence dictionary. If the owner already exists in the influence dictionary, update the influence value by adding new influence value to the existing influence value. The flowchart is given in **Appendix 19**.

4.3.2 USER INFLUENCE FORMULA FOR MULTI-LEVEL MARKETING METHOD

The following figure illustrates the sample network chain of tweeters and retweeters.

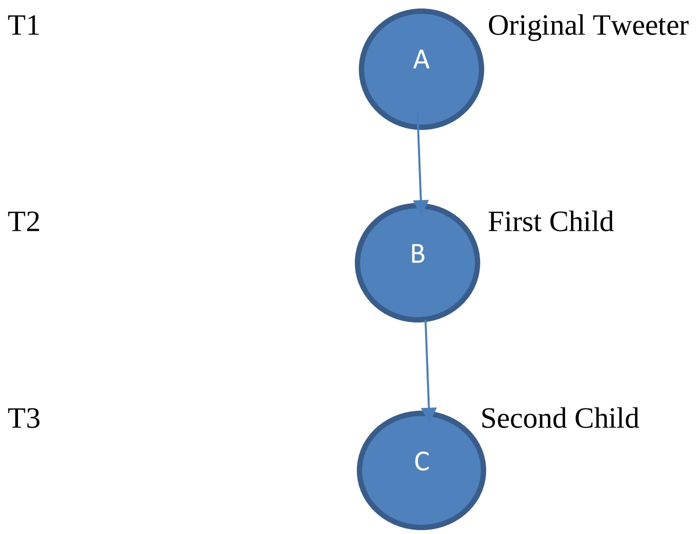


Fig 12: Sample network of users to show the levels

When B retweets from A at time T2,

Influence (A, T2) = $\sum P\,(Tw).\beta^{l}$ Level here is ‘1’

When C retweets from B at time T3,

Influence (B, T3) = $\sum$ P (Tw).β^{l} Level here is ‘1’

Influence (A, T3) = $\sum$ P (Tw).β^{l} Level here is ‘2’

4.3.3 ROOT-USER BENEFITS METHOD

Root-User Benefits Method is another method to compute the influence of Twitter users. As discussed in section 4.3.1, in multi-level marketing method, we compute the influence for both the root users and intermediate users. Root user benefits model computes the influence of the root users only. Only the root users gets the complete credit. Whenever a retweet is made, only the root user is said to be influential. None of the intermediate users is said to be influential.

As discussed in section 4.3.1, we follow the same steps for tweet level calculation in section 4.3.1.1 and tweet potential computation 4.3.1.2. The only difference in the user influence computation 4.3.1.3 is that we store the root users in a list and print the influence values for the root users only. The formula to compute the influence of the Root_User at time “T” is:

$$\text{User\_Influence (Root\_User, T)} = \sum \text{P (Tw)}.\beta^{l}$$

Where,

P (Tw) is the potential of the tweet

“l” is the level of the tweet and β serves as scalable factor.

Let us illustrate this with an example:

EXAMPLE 1

Let the input dataset be as follows:

```
if you drink enough fluids in the morning you will feel happier sharper and more energetic
throughout the day   Tue Sep 08 22:25:45 +0000 2015,A   N/A,N/A

if you drink enough fluids in the morning you will feel happier sharper and more energetic
throughout the day  Wed Sep 09 22:32:06 +0000 2015,B   A,Tue Sep 08 22:25:45 +0000 2015

if you drink enough fluids in the morning you will feel happier sharper and more energetic
throughout the day  Thu Sep 10 22:44:09 +0000 2015,C   B,Wed Sep 09 22:32:06 +0000 2015

if you drink enough fluids in the morning you will feel happier sharper and more energetic
throughout the day  Fri Sep 11 03:00:22 +0000 2015,D   C,Thu Sep 10 22:44:09 +0000 2015

if you drink enough fluids in the morning you will feel happier sharper and more energetic
throughout the day  Sat Sep 12 03:00:26 +0000 2015,E   D,Fri Sep 11 03:00:22 +0000 2015

if you drink enough fluids in the morning you will feel happier sharper and more energetic
throughout the day  Sat Sep 12 04:00:26 +0000 2015,Vishali D,Fri Sep 11 03:00:22 +0000 2015

if you drink enough fluids in the morning you will feel happier sharper and more energetic
throughout the day  Fri Sep 13 05:00:26 +0000 2015,Pinky   B,Wed Sep 09 22:32:06 +0000 2015
```

The influence of the users based on Multi-Level Marketing model will be as follows:

```
-------------------------------------------------------------------

        2015-09-09     A        4

-------------------------------------------------------------------

        2015-09-10     A        16

        2015-09-10     B        8

-------------------------------------------------------------------

        2015-09-11     A        64

        2015-09-11     C        16

        2015-09-11     B        32

-------------------------------------------------------------------
```

```
          2015-09-12        A          512

          2015-09-12        C          128

          2015-09-12        B          256

          2015-09-12        D          64

----------------------------------------------------------------------

          2015-09-13        A          16

          2015-09-13        B          8
```

Now, we will see the root user influence values based on Root-User Benefits model. The root user in the above input dataset is “A”. So, the root-user benefits model computes only the influence values for the root user over time.

```
          ----------------------------------------------------------------------

                    2015-09-09        A          4

          ----------------------------------------------------------------------

                    2015-09-10        A          16

          ----------------------------------------------------------------------

                    2015-09-11        A          64

          ----------------------------------------------------------------------

                   2015-09-12         A          512

          ----------------------------------------------------------------------
```

```
        2015-09-13        A          16

----------------------------------------------------------------------
```

Influence of the users in flu data, food poisoning data and politics data are computed by using both the methods i.e. multi-level marketing and root-user benefits method. A comparison of both the ranks is made and these results are compared with the user ranks based on followers, retweets and mentions. A comparison table of user influences can be found in Appendix 8 for flu data, Appendix 10 for food poisoning data and Appendix 11 for politics data.

CHAPTER V

RESULTS

In the previous chapter we presented several algorithms and their implementations for potential and user influence computations. This chapter presents the results of applying those to three data sets. The data sets are related to flu propagation, food poisoning and political discourse. Topics are defined in each dataset for analysis. The keywords defining the topics are given in the Appendices 1-3. The results provided in this section illustrate the applicability of the algorithms for information diffusion and user influence analysis. They also serve to compare the results to a similar model.

5.1 TWEET PROPAGATION RESULTS

Results of potential computation are shown for three datasets flu data, food poisoning data and politics data in sections 5.1.1, 5.1.3 and 5.1.4 respectively. Daily comparison of tweet_count and potential is also shown which indicate they are not similar. A deduction could be made that tweet_count is not necessarily the best indicator for information diffusion.

5.1.1 FLU DATA TWEET PROPAGATION RESULTS

Tweet propagation for flu data (flu topic) is computed in the date range September 9, 2015 to November 26, 2015 with different RHO values that serves as a scale factor. The RHO values used here are 1, 2, and -2. The keywords used for flu topic are listed in Appendix 1. All the keywords are given a default weight ‘1’.

Figure 13 denotes flu tweet propagation with RHO value equals to 2. The value of highest flu topic potential is 270.00k and is found on September 27, 2015.

Figure 14 denotes flu tweet propagation with RHO value equals to 1. The value of highest flu topic potential is 140.00k and is found on September 27, 2015.

Figure 15 denotes flu tweet propagation with RHO value equals to -2. The value of highest negative flu topic potential is -270.00k and is detected on September 27, 2015.

From the figures 13, 14 and 15, as the RHO value changes, the topic potential value varies. The topic potential is directly proportional to the RHO value used. But, the highest topic potential is found on September 27, 2015 in all the three cases with different RHO values. Here, RHO (ρ) serves only as a scale factor as the weight of all keywords are same.

The CDC website [refer to row 2 in Appendix 21] says that mid-September in the year 2015 is the high time for flu infection. The news say to take the flu vaccination in September. There is lot of activity taking place in September related to flu. There are announcements related to the change in vaccination shots [refer to row 2 in Appendix 21].

CDC recorded the highest hospitalization rates in the week 39 an estimated 313.8 per 100,000 people in the age group 65 were hospitalized from flu [refer to row 1 in Appendix 21]. That could be the reason the flu potential encountered a peak on September 27, 2015.

However, more evidence is needed to make any conclusive claims. As we were unable to obtain similar data for other topics considered in the sections that follow, such analysis could not be made for those topics.

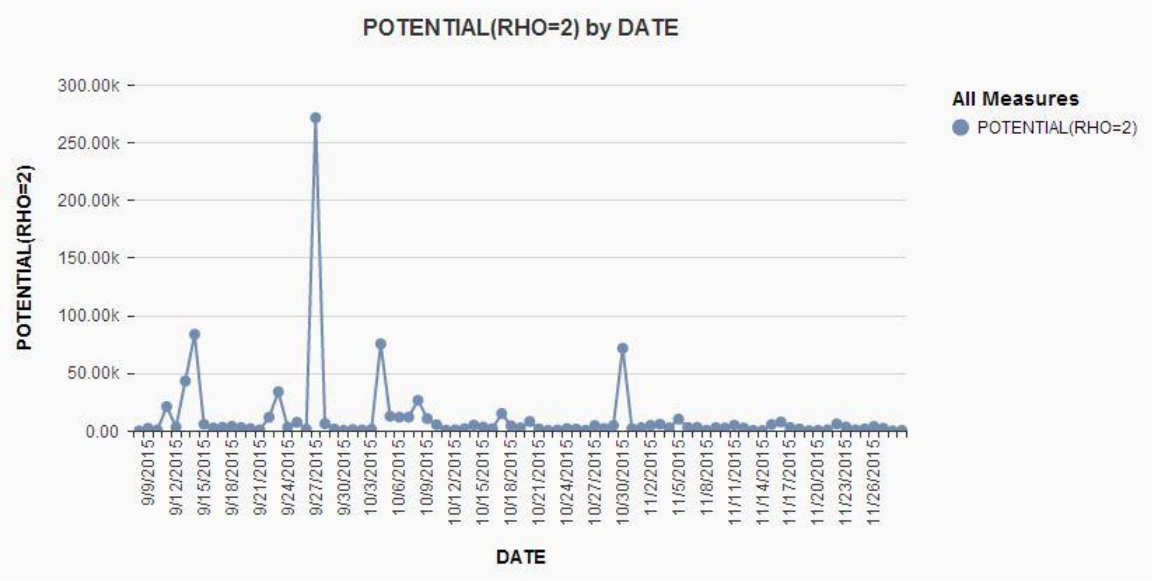


Fig 13: Topic Potential Series for the flu topic with RHO = 2

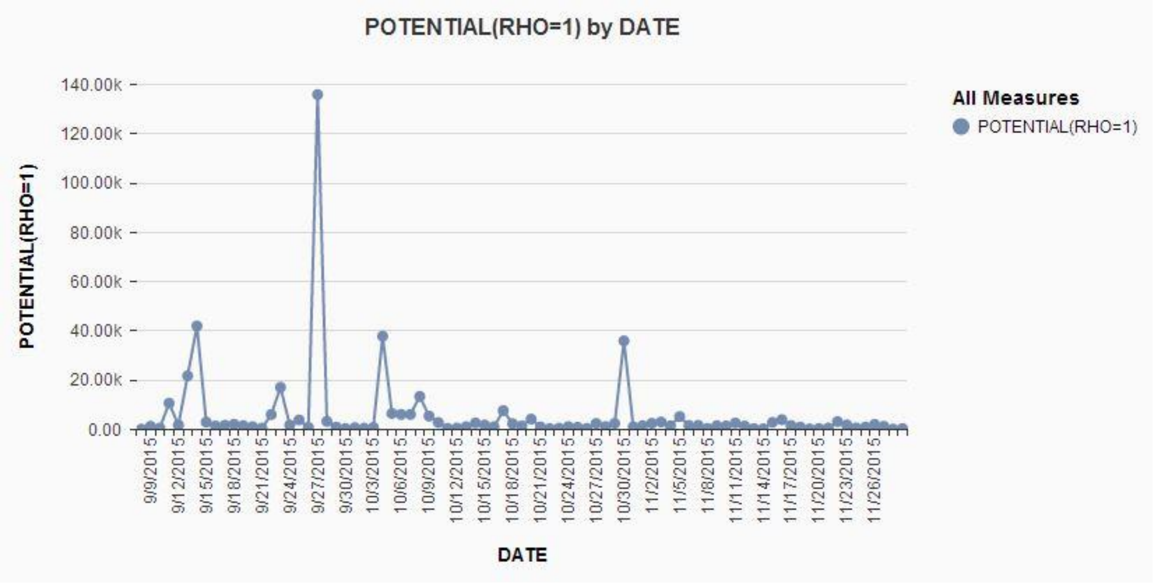


Fig 14: Topic Potential Series for the flu topic with RHO = 1

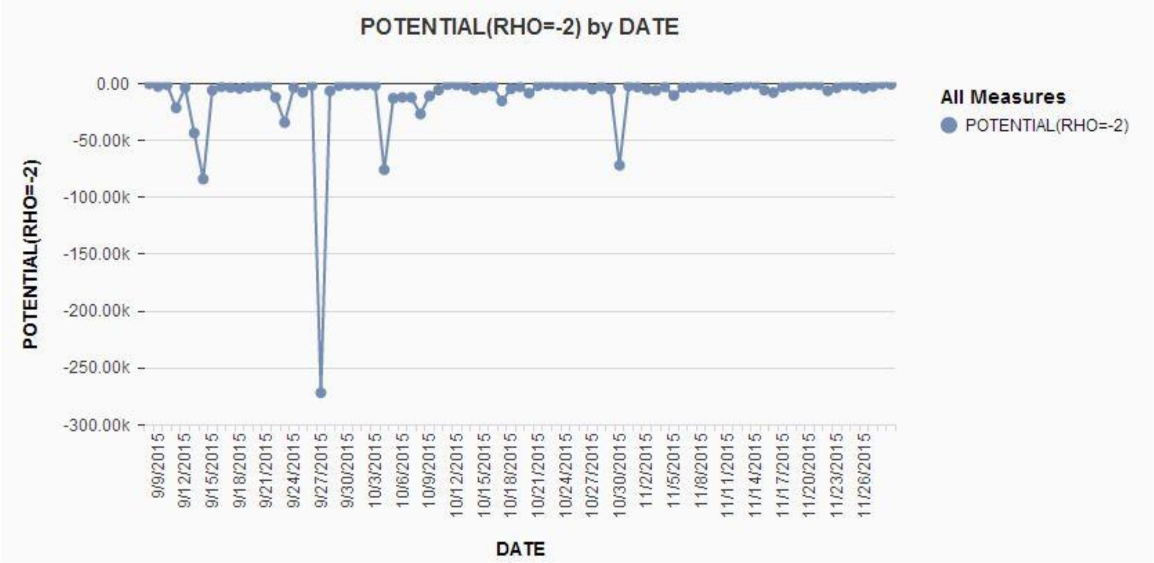


Fig 15: Topic Potential Series for the flu topic with RHO = -2

5.1.2 FLU TOPIC POTENTIAL VS FLU TWEET COUNT

We wanted to explore the relationship between the topic potential obtained for flu data and the tweet count by observing their values. The top 20 dates on which the flu topic potential is the highest are chosen as representative date for comparison. The flu tweet count is visualized in figure 16 against the potential. The blue graph indicates the potential values for top 20 dates and the pink graph indicates the tweet counts on those respective dates. Visual comparison of the graphs indicate no correlation between them. As our research computes the topic potential based on the retweets, the graphs of topic potential and tweet count may not be directly proportional to each other.

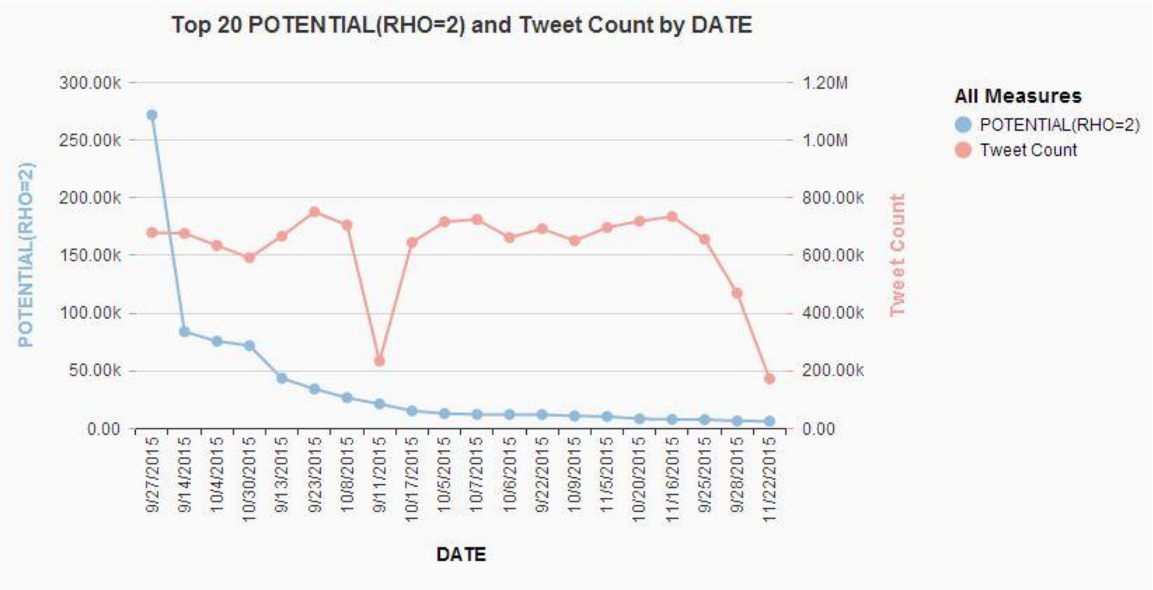


Fig 16: Tweet count vs Potential - flu data

5.1.3 FOOD POISONING DATA TWEET PROPAGATION RESULTS

Tweets related to food poisoning are collected in the date range 15 November, 2016 – 14 January, 2017. While all flu related data were lumped into one topic, in the case of food poisoning, we explore more sublevel information. Based on data collected from CDC, three topics are defined on food poisoning. The topic names and keywords are given in Appendix 2.

Food poisoning can be caused by viral, bacterial and parasitic organisms. Taking into account the keywords for the three topics, the total food poisoning data is divided into 3 partitions and tweet propagation is computed on all the topics separately.

Figure 17 visualizes tweet propagation for all the three topics of food poisoning data. The highest topic potential value is 8.5k and is encountered on January 2, 2017 with RHO value equals to 2. Figure 18 visualizes tweet propagation for all the three topics of food poisoning data. The highest topic potential value is 4.00k and is encountered on January 2, 2017 with RHO value equals to 1.

From the figures 17 and 18, as the RHO value changes, topic potential value changes. But, the highest topic potential value is found on 2 January, 2017 for all the three topics with different RHO values. We can say that RHO serves just as a scale factor in our research.

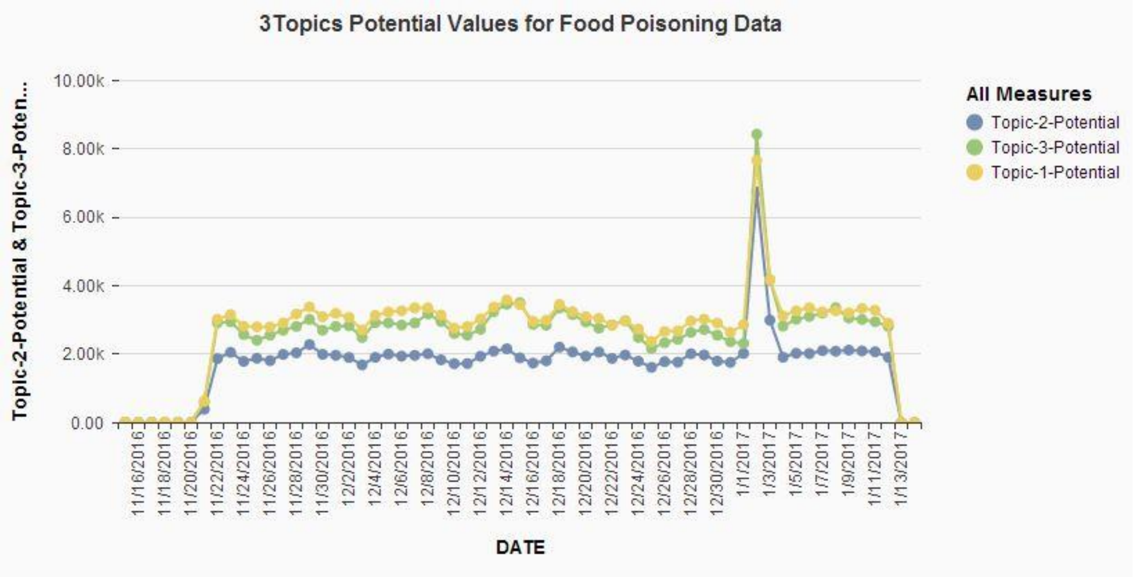


Fig 17: Potential time-series for food poisoning topics (RHO=2)

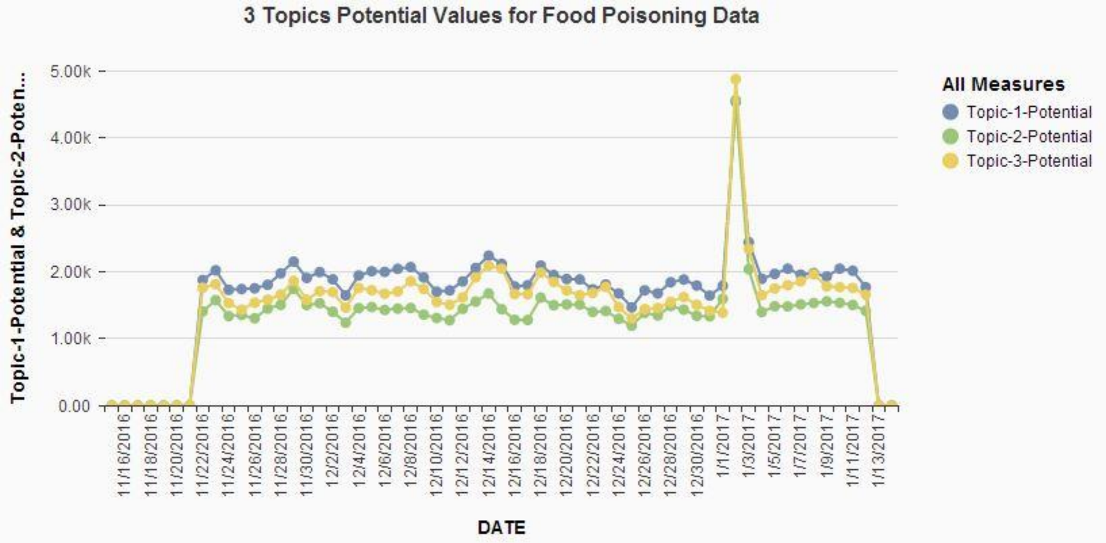


Fig 18: Potential time-series for food poisoning topics (RHO=1)

5.1.4 FOOD POISONING TOPIC POTENTIAL VS FOOD POISONING TWEET COUNT

Food topic potential on every day is compared with the tweet count on respective days. In this research, I have considered food topic potential on all days and tweet count on all days.

Figure 19 visualizes the food poisoning topic1, topic2, topic3 potential values with the tweet counts on respective days. Pink graph is the tweet count. Blue graphs indicate topic potentials. While not as obvious as the flu topic, visual analysis of the graph indicates that topic potential may not be directly proportional to tweet count.

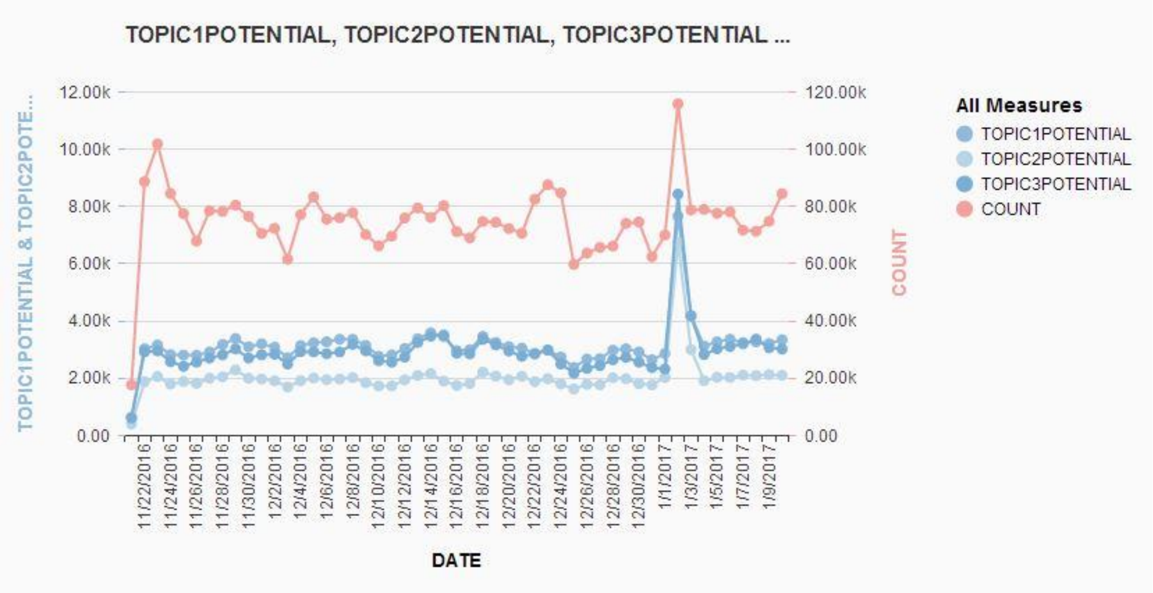


Fig 19: Tweet count vs Potential – food poisoning data

5.1.5 POLITICS DATA TWEET PROPAGATION RESULTS

The politics data is collected from Twitter between February 17, 2017 and March 17, 2017. Tweets are collected after the 2016 elections using the keywords Trump, Immigration, Muslim, Mexico and Terrorism. Politics data is divided into 2 topics namely civil and economic. The keywords for the respective topics are shown in Appendix 3.

The tweet propagation for topic1 i.e. civil and topic2 i.e. economic over time is shown in the below Figure 20. The highest topic potential for civil topic is found on 22 February, 2017.

On February 22, 2017, a revised travel ban for the countries Iraq, Iran, Syria, Sudan, Somalia, Yemen and Libya is released by the White House. There are discussions about the treatment of Syrian refugees, whose immigration to the U.S. in the original ban was indefinitely suspended. This comes under the civil rights topic that could serve as justification for the spike in the topic potential [refer to row 4 in Appendix 21].

The highest topic potential for economical topic is found on 23 February, 2017. The RHO value used here is 2.

On February 23, 2017, the CBS News poll results were released regarding the U.S economy if immigration is the most problem for Trump and Congress [refer to row 3 in Appendix 21].

February 23, 2017 is the time when homeland security secretary John Kelly assured Mexico that the U.S will not carry out any mass deportations of people illegally in the country and regarding immigration as the Mexico wall building was the news everywhere. These could provide the justification for the spike in topic potential.

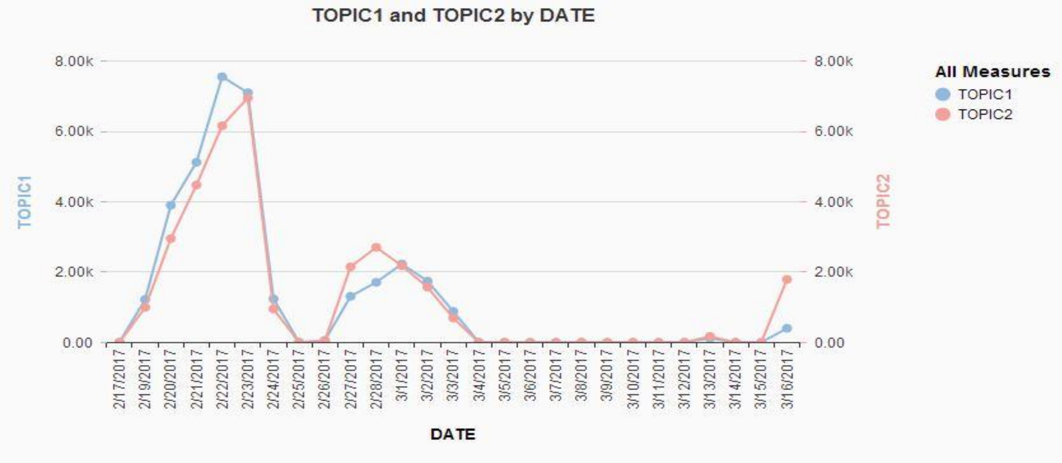


Fig 20: Potential time-series for political topics

5.1.6 POLITICS TOPIC POTENTIAL VS POLITICS TWEET COUNT

Politics topic potential on every day is compared with the tweet count on respective days. In this research, I have considered politics data topic potential on all days and tweet count on all days. Figure 21 visualizes the topic potential of politics data comparison with tweet count. It is clear that topic potential is not directly proportional to the tweet count. Based on the results we have presented we can claim that the topic potential and tweet count are not correlated except in the case of zero tweets.

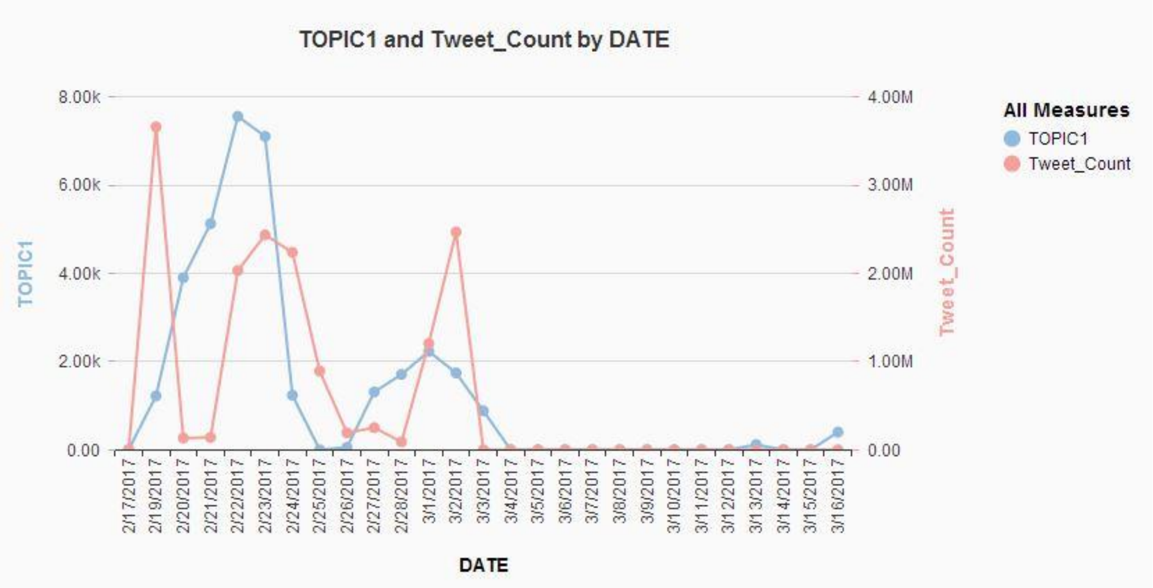


Fig 21: Tweet count vs Potential - politics data

5.2 DIFFUSION NETWORK MODEL RESULTS

The diffusion network model is described in (J. a. Yang 2010). Here we present the computational results of connections which is used as a measure in (J. a. Yang 2010).

5.2.1 FLU DATA DIFFUSION NETWORK MODEL RESULTS

The diffusion network model counts the number of connections on every day. Connections are the number of connected users on every day.

Figure 22 is a histogram that compares number of connections and topic potential for flu data everyday. The highest number of connections for flu data is found on September 27, 2015.

The highest topic potential is also found on September 27, 2015 from the flu topic propagation results in figure 13. This can be used as one of the verification methods to our developed model.

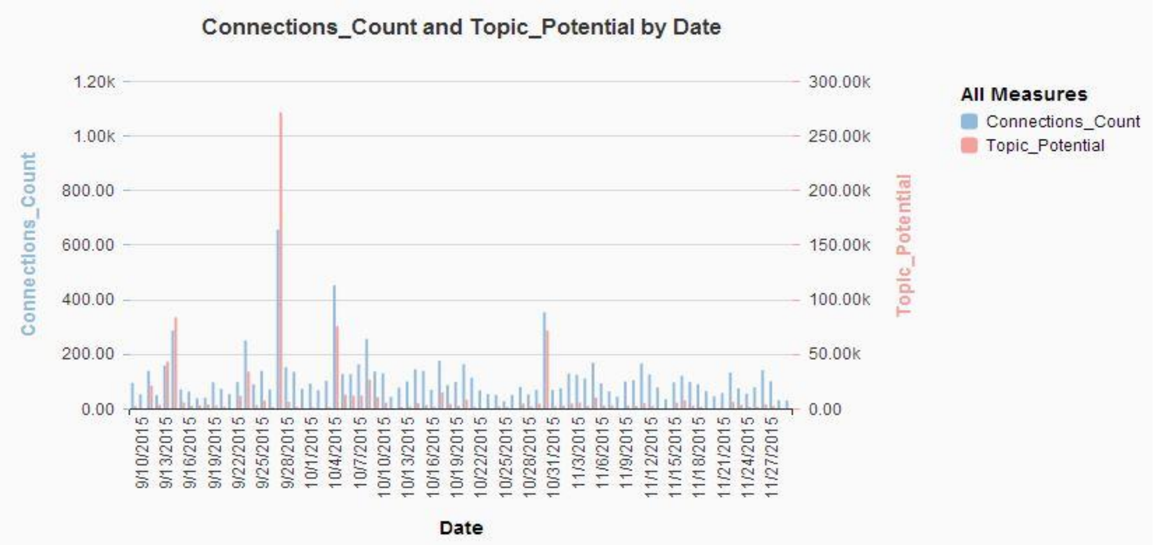


Fig 22: Connections vs Potential - flu data

5.2.2 FOOD POISONING DATA DIFFUSION NETWORK MODEL RESULTS

The highest number of connections is found on November 23, 2016 from the Figure 23. The second highest connections count is found on January 2, 2017. Figure 23 compares the number of number of connections and topic potential for food poisoning data every day.

The highest topic potential for food poisoning data is found on January 2, 2017 for all the three topics in the Figure 17. Due to New Year celebrations, January 2 is possibly the time where most people report fever, stomach problems, vomiting etc. which are caused due to food poisoning.

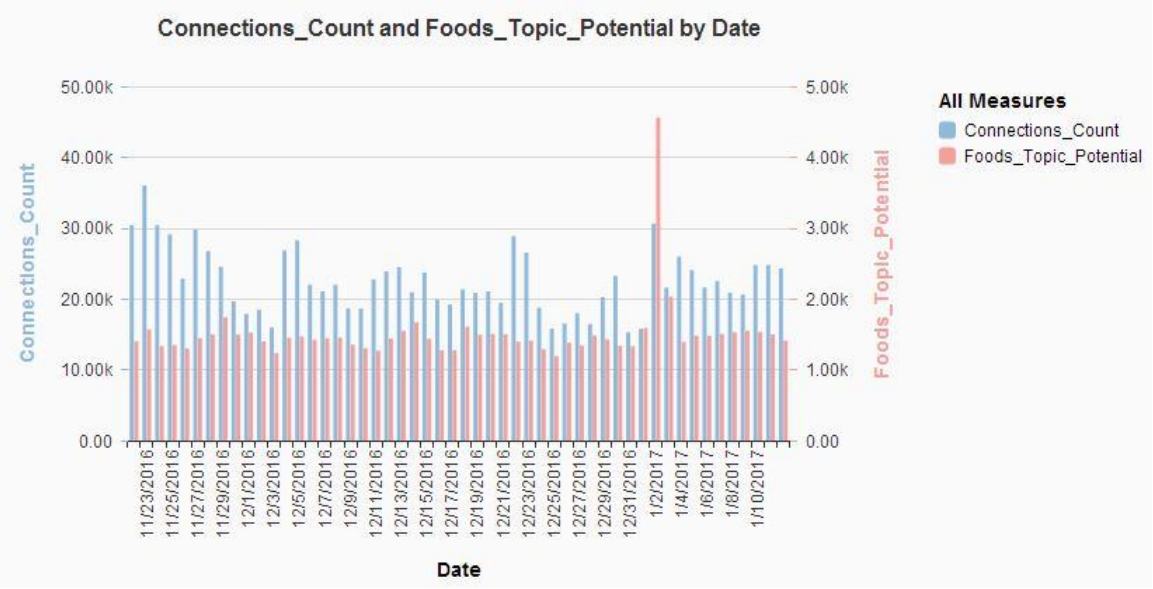


Fig 23: Connections vs Potential – food poisoning data

5.2.3 POLITICS DATA DIFFUSION NETWORK MODEL RESULTS

The figure 24 compares the number of connections and topic potential for politics data every day. The highest number of connections for politics data is found on February 23, 2017 from the Figure 24. The second highest connections count is found on February 22, 2017. The highest topic potential for topic 1 is found on February 22, 2017 and the second highest count is found on February 23, 2017 from the above tweet propagation results in Figure 21.

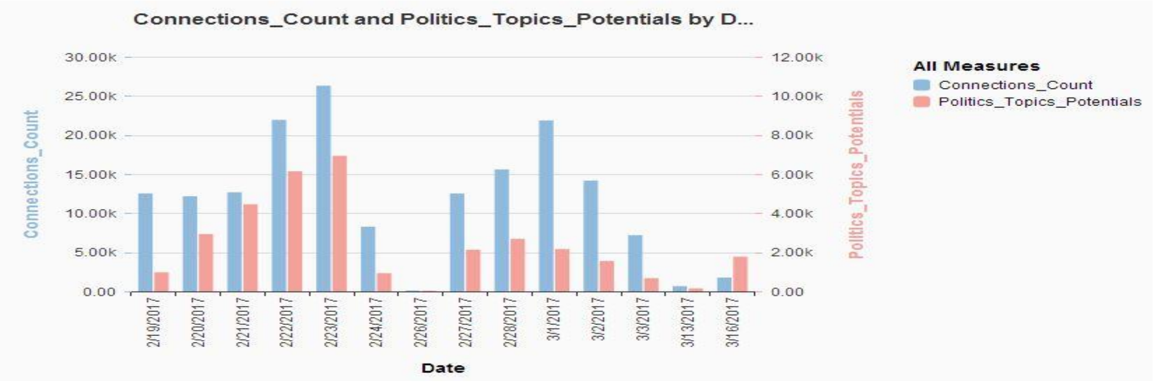


Fig 24: Connections vs Potential - politics data

5.3 USER INFLUENCE RESULTS

In this section we present the results related to user influence. Related algorithms and formulae can be found in chapters 2 and 4.

5.3.1 FLU DATA USERS INFLUENCE RESULTS

Figures 25 through 30 exhibit information related to user influence in the flu topic based on different measures listed below:

Followers: The number of followers to every user

FollowerRank: The rank of users based on the number of followers

Retweets: The number of retweets for the tweet posted by the user

RetweetRank: The rank of users based on the number of retweets every user’s tweet has.

MentionsCount: The number of times a user is being mentioned by the other users.

MentionsRank: The rank of the users based on the number of times an user is being mentioned.

Figure 25 visualizes the influence of all users of the highest retweeted tweet. The highest retweeted tweet of Flu Data has 1000 retweets:

“If you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day”

Figure 26 visualizes the top 10 influential users based on the number of followers they have.

Figure 27 visualizes top 10 influential users based on the mentions count i.e. the number of times a user is being mentioned by another user.

Figure 28 visualizes the top 10 influential users based on the number of times a user's tweet has been retweeted.

Figure 29 visualizes the top 10 influential users based on the follower rank, retweets rank, mentioned rank. The lower rank denotes the high influential user. If all the ranks are near to the X-axis, it denotes that the user is highly influential.

Figure 30 visualizes the comparison of the top 10 users based on user influence values and all the other measures followers, retweets and mentioned count. Influence rank is based on influence of the user computed in this research. These ranks are compared against the ranks of the users based on the followers rank, retweets rank and mentions rank.

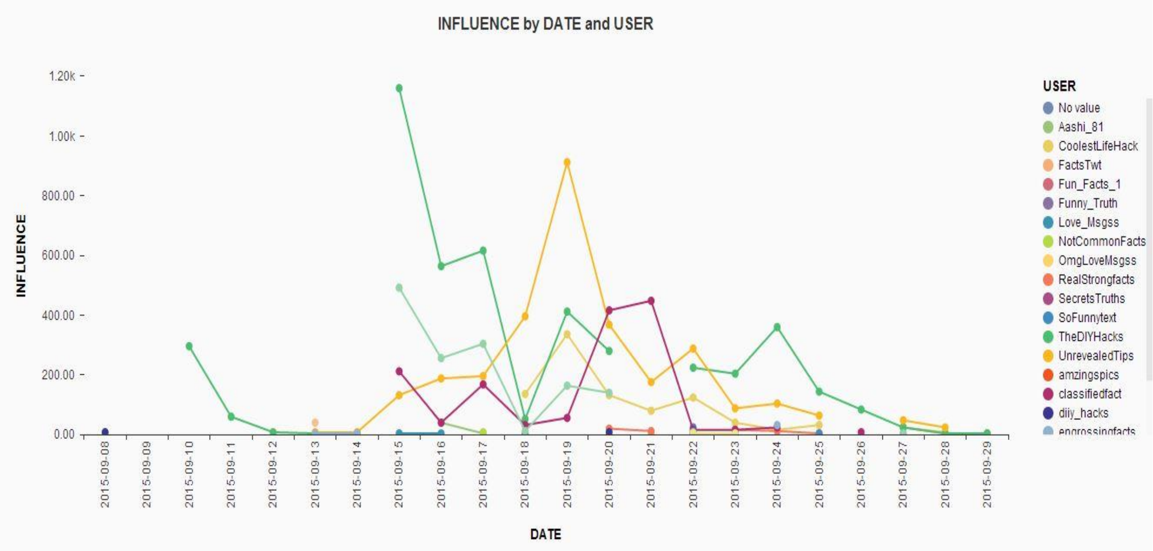


Fig 25: Influence of different users for Flu Data

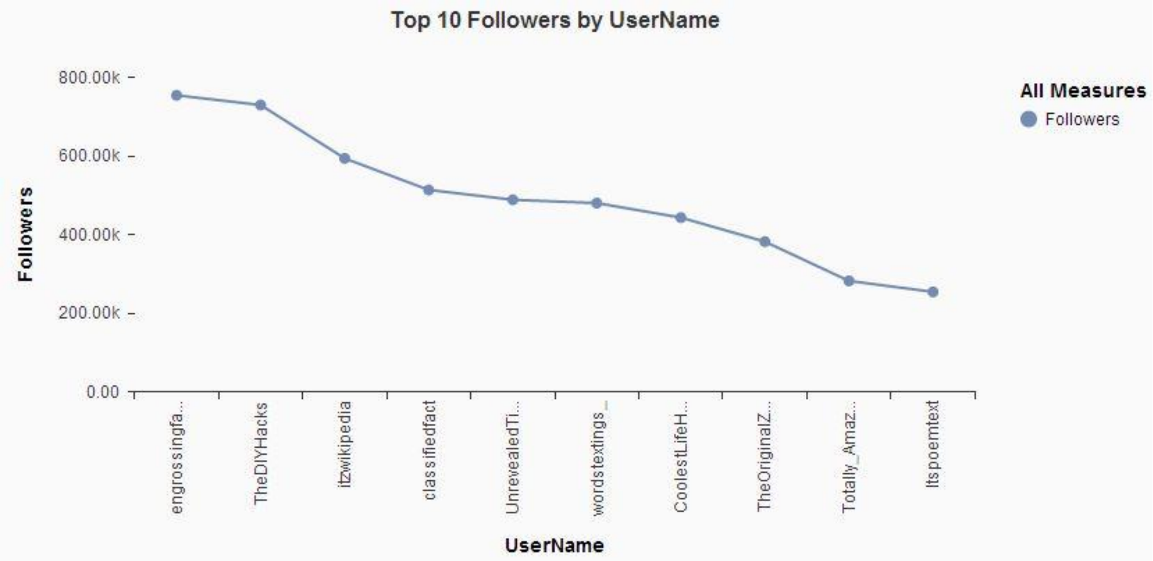


Fig 26: Top 10 users based on number of followers

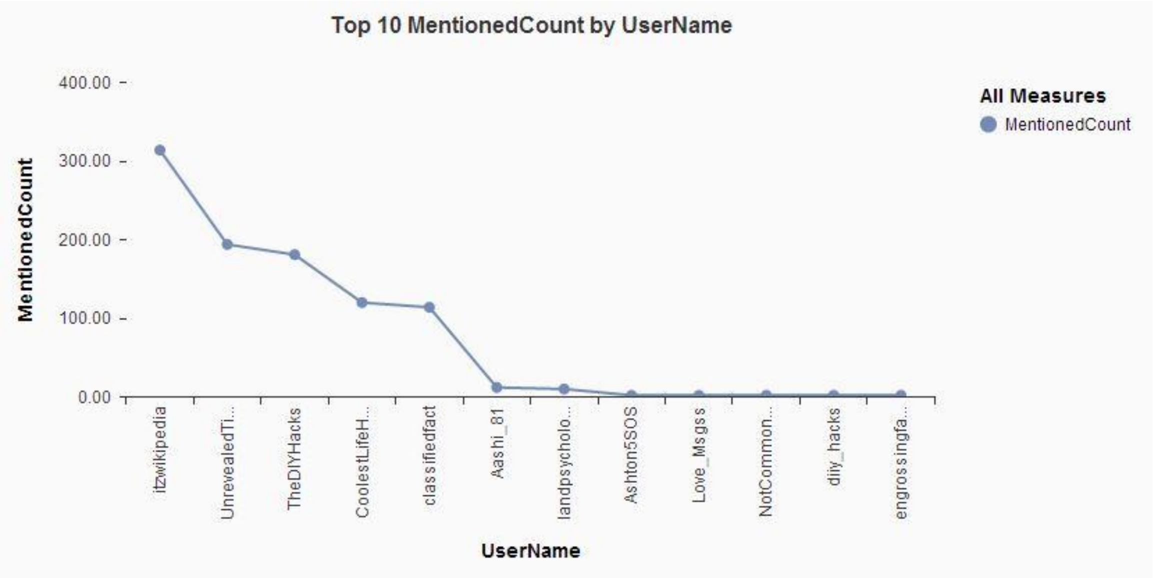


Fig 27: Top 10 users based on number of times a user is mentioned

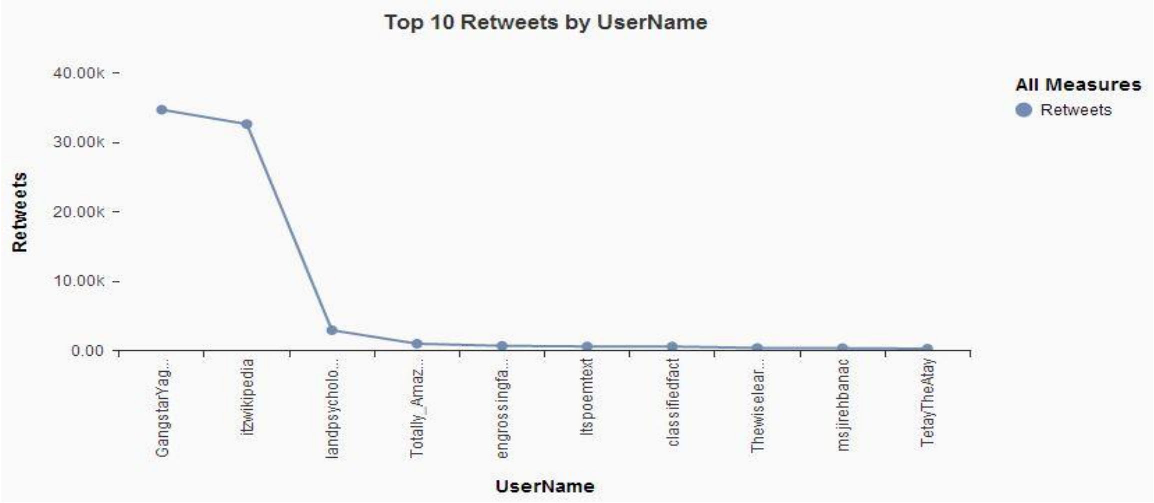


Fig 28: Top 10 users based on number of retweets done

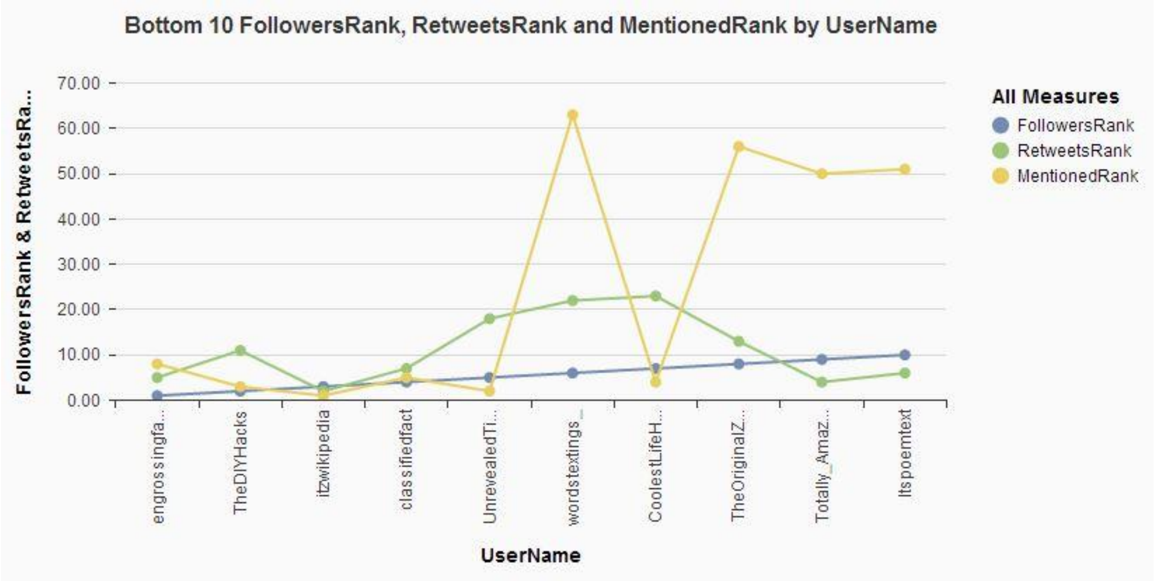


Fig 29: 10 Influential users based on all the measures followers, mentions and retweets

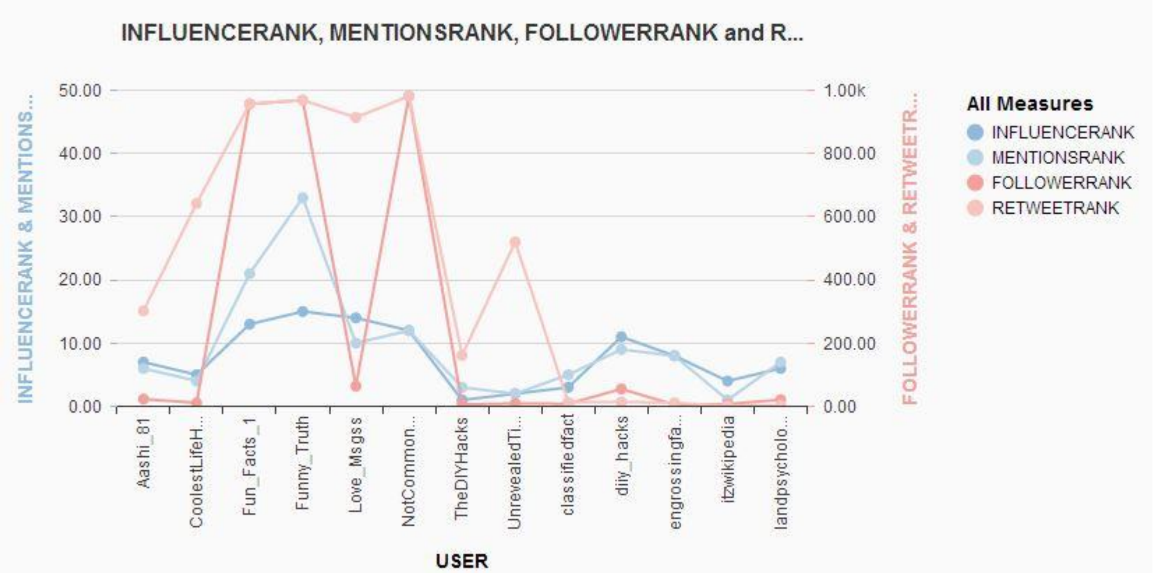


Fig 30: Users ranks based on user potential, followers rank, retweets count and mentions count

Appendix 7 compares user influence values and corresponding ranks of users associated to the highest retweeted tweet based on different measures.

Appendix 8 shows the top 10 users among all the users in the complete dataset with highest user influence ranks compared to all the other measures.

5.3.2 FLU DATA CORRELATION

Spearman Rank Correlation Coefficient is used in this research to find correlation between the retweets, followers and mentions of every user. From the correlation results below shown in Figure 31, it is clear that the retweets for a user’s tweets are mostly done by his/her followers. The correlation table can be found in Appendix 5.

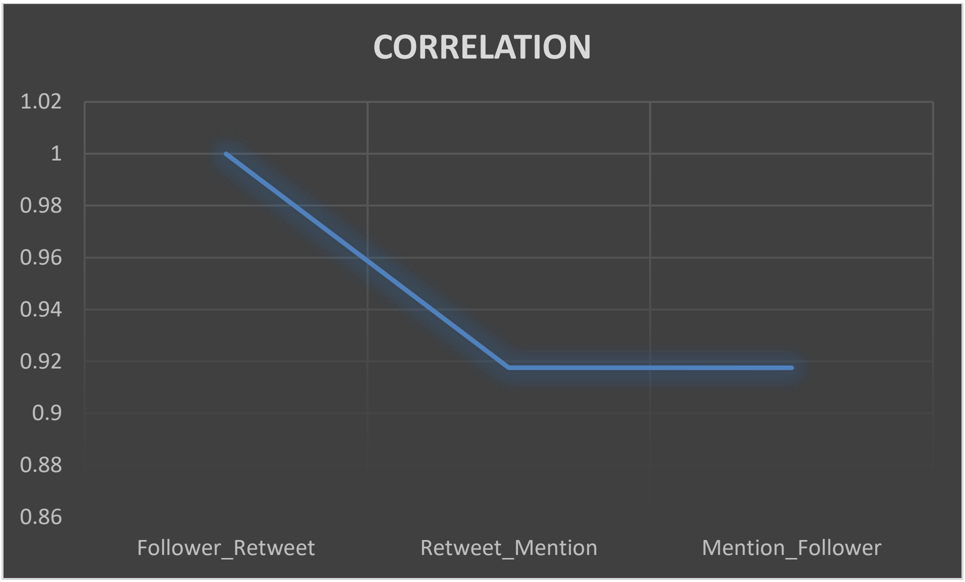


Fig 31: Correlation for flu data

5.3.3 FOOD POISONING DATA USER INFLUENCE RESULTS

Figure 32 visualizes the influential users based on the followers rank, mentions rank and retweets rank for the tweet "**this gives me chills**" which is one of the highest retweeted tweet.

Figure 33 visualizes the users influence rank against follower's rank, mentions rank and retweets rank. If all the ranks of a user are low i.e. near to the X-axis, then that user is said to be influential.

Appendix 9 compares the users influence rank against the followersrank, retweetsrank and mentionsrank for the highest retweeted tweet.

Appendix 10 shows the top 10 users among all the users in the complete dataset with highest user influence ranks compared to all the other measures.

From Appendix 9 and Appendix 10, we can see that there are not any common users having the same influence. This proves though a single tweet of an user is retweeted many times, he might not be the most influential user.

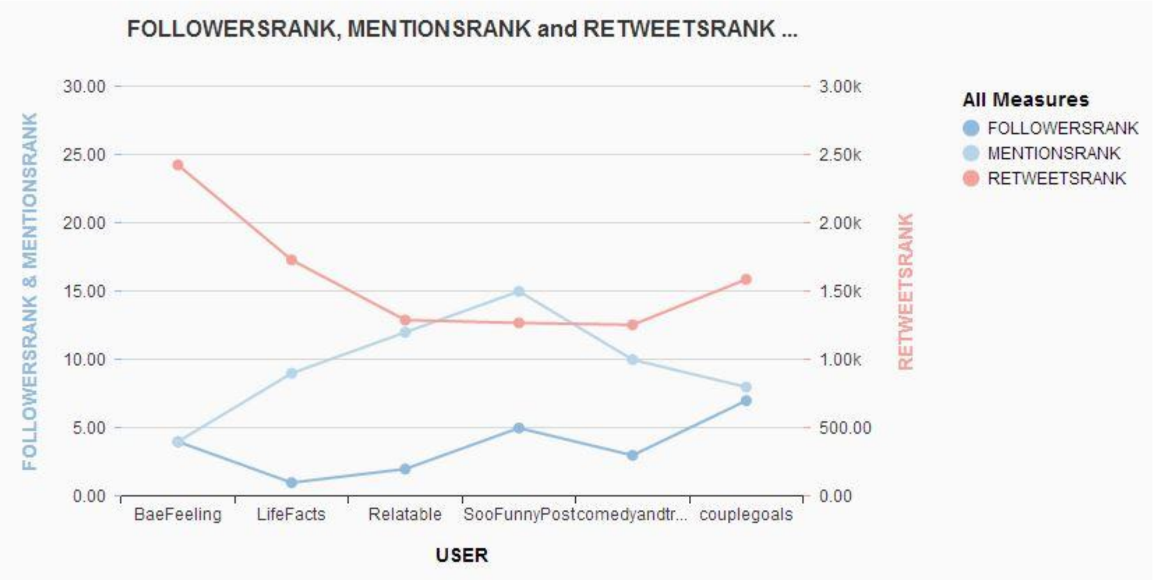


Fig 32: Users Rank Based on Followers, Retweets, Mentions

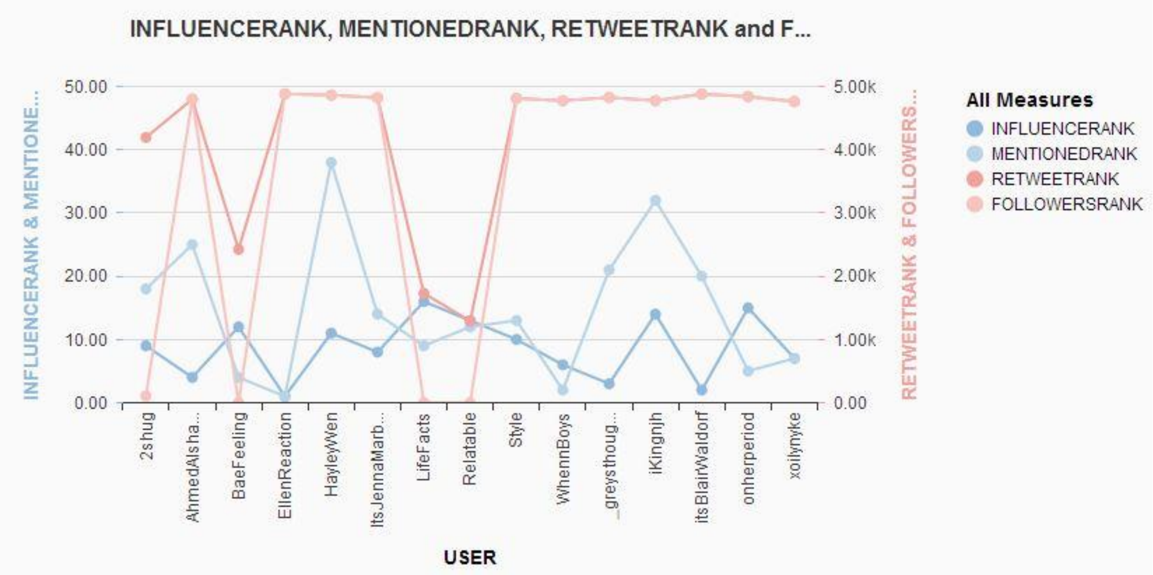


Fig 33: Users Rank Based on User Potential, Followers, Retweets, Mentions

5.3.4 FOOD POISONING DATA CORRELATION

Spearman Rank Correlation Coefficient is used in this research to find correlation between the retweets, followers and mentions of every user. From the correlation results in Figure 34, it is clear that the retweets for a user's tweets are mostly done by his/her followers. The correlation table can be found in Appendix 6.

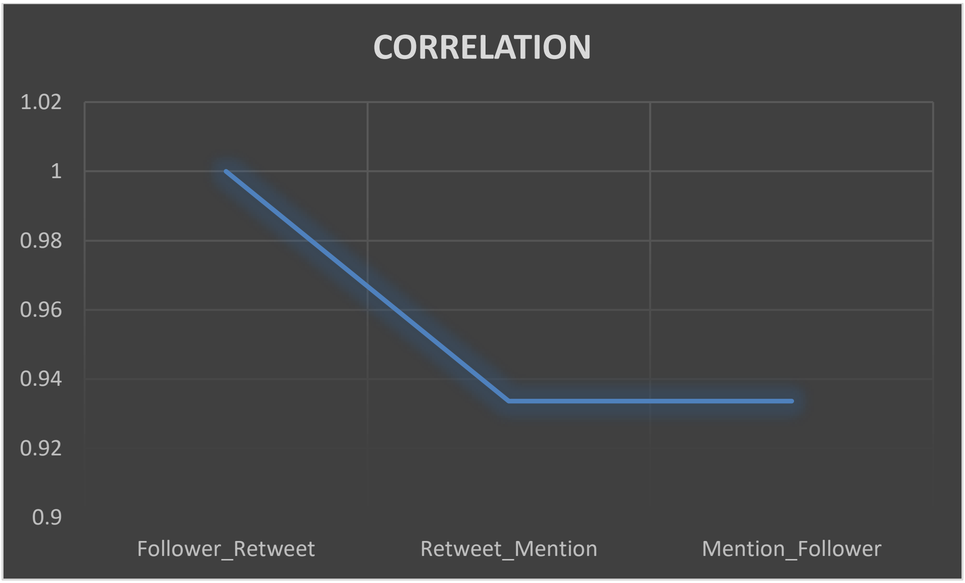


Fig 34: Correlation for Food Poisoning Data

5.3.5 POLITICS DATA USER INFLUENCE RESULTS

Politics data is collected from 02/17/2017 to 03/17/2017. All the users in the dataset are considered for user influence computation based on the formula from multi-level marketing strategy. The obtained user ranks are compared against the ranks based on the number of followers, retweets and mentioned count. The results can be found in the Appendix 11

CHAPTER VI

CONCLUSION

SUMMARY

The objective of this research is to develop software tools for the analysis of information propagated via the microblog medium which can be termed big data due to their volume and velocity of propagation. We focus on the Twitter platform. Our work is based on information diffusion models and user influence models based on Twitter data proposed in the literature. We have developed and implemented algorithms to compute topic potential and user influence measure proposed in [20] and the diffusion measure (number of connections) proposed in [23]. Based on the user influence measure defined in [20], we developed and implemented two models named Multi-Level Marketing and Root User Benefits. We have also considered influence measures proposed in [6]. We collected data related to “flu”, “food poisoning”, and “politics” to test the algorithms.

OBSERVATIONS

Based on the results obtained by applying our algorithms to the data collected we can make the following claims:

1) Implementation of algorithms work properly and can be applied to any Twitter data collected in Json format.
2) The two diffusion models have similar performance in identifying peak points.

3) User influence rankings vary with the models.
4) The flu topic potential series have some similarities with published flu data at the CDC website.
5) More empirical analysis is required for validation which is considered future work.

FUTURE WORK

In this research, we have used map-reduce framework to compute the topic diffusion and user influence as the data is already collected in HDFS (Hadoop Distributed File System). In future, this work can be extended by using Apache Spark through which analysis can be done directly on the data collected facilitating real-time analysis.

REFERENCES

References

[1] Adamic, Lada A and Glance, Natalie. 2005. "The political blogosphere and the 2004 US election: divided they blog." In *Proceedings of the 3rd international workshop on Link discovery*.

[2] Arias, Marta and Arratia, Argimiro and Xuriguera, Ramon. 2013. "Forecasting with twitter data." *ACM Transactions on Intelligent Systems and Technology (TIST)* 8.

[3] Asur, Sitaram and Huberman, Bernardo A and Szabo, Gabor and Wang, Chunyan. 2011. "Trends in social media: Persistence and decay." *Available at SSRN 1755748.*

[4] Bakshy, Eytan and Rosenn, Itamar and Marlow, Cameron and Adamic, Lada. 2012. "The role of social networks in information diffusion." In *Proceedings of the 21st international conference on World Wide Web*, 519--528.

[5] Cataldi, Mario and Caro, Luigi Di and Schifanella, Claudio. 2013. "Personalized emerging topic detection based on a term aging model." *ACM Transactions on Intelligent Systems and Technology (TIST)* 7.

[6] Cha, Meeyoung and Haddadi, Hamed and Benevenuto, Fabricio and Gummadi, P Krishna. 2010. "Measuring User Influence in Twitter: The Million Follower Fallacy." 30.

[7] Domingos, Pedro and Richardson, Matt. 2001. "Mining the network value of customers." In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*.

[8] Du, Nan and Liang, Yingyu and Balcan, Maria-Florina and Song, Le. 2014. "Influence Function Learning in Information Diffusion Networks." In *ICML*, 2016--2024.

[9] Fan, Lidan and Lu, Zaixin and Wu, Weili and Bi, Yuanjun and Wang, Ailian and Thuraisingham, Bhavani. 2014. "An individual-based model of information diffusion combining friends' influence." *Journal of Combinatorial Optimization* 529--539.

[10] Fang, Xiao and Hu, Paul Jen-Hwa and Li, Zhepeng and Tsai, Weiyu. 2013. "Predicting adoption probabilities in social networks." *Information Systems Research* 128--145.

[11] Gabielkov, Maksym and Legout, Arnaud. 2012. "The complete picture of the Twitter social graph." In *Proceedings of the 2012 ACM conference on CoNEXT student workshop*, 19--20.

[12] Goyal, Amit and Bonchi, Francesco and Lakshmanan, Laks VS. 2010. "Learning influence probabilities in social networks." In *Proceedings of the third ACM international conference on Web search and data mining*, 241--250.

[13] Guille, Adrien and Hacid, Hakim and Favre, Cecile and Zighed, Djamel A. 2013. "Information diffusion in online social networks: A survey." *ACM SIGMOD Record* 17--28.

[14] Guille, Adrien. 2013. "Information diffusion in online social networks." In *Proceedings of the 2013 SIGMOD/PODS Ph. D. symposium*.

[15] Kwak, Haewoon and Lee, Changhyun and Park, Hosung and Moon, Sue. 2010. "What is Twitter, a social network or a news media?" In *Proceedings of the 19th international conference on World wide web*, by Haewoon and Lee, Changhyun and Park, Hosung and Moon, Sue Kwak.

[16] Li, Cheng-Te and Kuo, Tsung-Ting and Ho, Chien-Tung and Hong, San-Chuan and Lin, Wei-Shih and Lin, Shou-De. 2013. "Modeling and evaluating information propagation in a microblogging social network." (Springer).

[17] Rogers, Everett M. 2010. "Diffusion of innovations." In *rogers2010diffusion*. Simon and Schuster.

[18] Song, Xiaodan and Chi, Yun and Hino, Koji and Tseng, Belle L. 2007. "Information flow modeling based on diffusion rate for prediction and ranking." In *Proceedings of the 16th international conference on World Wide Web*, 191--200.

[19] Taxidou, Io and Fischer, Peter M. 2014. "Online analysis of information diffusion in twitter." In *Proceedings of the 23rd International Conference on World Wide Web*, 1313--1318.

[20] TK, Ashwin Kumar and George, KM and Thomas, Johnson P. 2015. "An Empirical Approach to Detection of Topic Bubbles in Tweets." *In Proceedings of 2015 IEEE/ACM 2nd International SYmposium on Big Data Computing.* 31--40.

[21] Tumasjan, Andranik and Sprenger, Timm Oliver and Sandner, Philipp G and Welpe, Isabell M. 2010. "Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment." *ICWSM* 178--185.

[22] Yang, Jaewon and Leskovec, Jure. 2010. "Modeling information diffusion in implicit networks." In *2010 IEEE International Conference on Data Mining,* 599--608.

[23] Yang, Jiang and Counts, Scott. 2010. "Predicting the Speed, Scale, and Range of Information Diffusion in Twitter." *ICWSM* 355--358.

[24] Zaman, Tauhid R and Herbrich, Ralf and Van Gael, Jurgen and Stern, David. 2010. "Predicting information spreading in twitter." In *Workshop on computational social science and the wisdom of crowds, nips,* 17599--601.

APPENDICES

APPENDIX 1

KEYWORDS FOR FLU DATA

Keywords
fever, feverish chills, chills, cough, sore throat, runny nose, stuffy nose, body ache, muscle ache, headache, fatigue, tiredness, tired, vomiting, diarrhea, joint aches, pain around eyes, watery eyes, flushed skin, exhaustion, sneezing, dry cough, persistent cough, weakness, nasal congestion, oseltamivir, zanamivir, tamiflu, relenza, permavir, rapivab, rest, drink fluids, damp cloth on forehead, using humidifier, gargling salt water, warm blanket, decongestants, cough medicine, cough drops, throat lozenge, acetaminophen, tylenol, ibuprofen, advil, motrin, nuprin, antihistamine, pseudoephedrine, phenylephrine, aspirin, naproxen, aleve, anti viral meds – nausea & vomiting oseltamivir – delerium, self-harmful behavior, anti viral drugs - dizziness, runny nose, stuffy nose, cough, diarrhea, headache and some behavioral side-effect, antihistamine – drowsiness, decongestants – hyper activity, increased blood pressure, increased heart rate

APPENDIX 2

KEYWORDS FOR FOOD POISONING DATA

TOPIC 1	TOPIC 2	TOPIC 3
headache,nausea,vomiting,body aches,cough,dizziness,tiredness,sweats,hoarseness,fainting,abdomen swelling,flushing,fainting,sore throat,malaise,anorexia,fatigue,muscles pain,joint pain,back pain,depression,low blood pressure,thirst,muscle cramps,restlessness,rapid heart rate,loss skin elasticity,dry mucous membranes,abdominal cramps,diarrhea,weakness,anemia,rash,red eyes,jaundice,loss balance,stiff neck,confusion, tenesmus	diarrhea,throwing up,nausea,stomach pain,fever,headache,body aches,dry mouth,dry throat,feeling dizzy,sleepy,fussy,cry,fatigue,abdominal pain,dark urine,jaundice,vomiting,loss of appetite,clay colored bowel movements,clay colored stool	stomach pain,stomach cramping,bloody stools,fever,abdominal pain,nausea,vomiting,abdominal distention,diarrhea,mucus stools,abdominal discomfort,weight loss,dehydration,stomach cramps,stomach pain,watery diarrhea,bloating,loss appetite,gas,greasy stools,reduced vision,blurred vision,pain bright light,redness eye,muscle pains,itchy skin,constipation,heart problems,breathing problems,face swelling,eyes swelling,cough,chills

APPENDIX 3

KEYWORDS FOR POLITICS DATA

TOPIC1	TOPIC2
abortion, civil rights, education, families, children, welfare, poverty, principles, values	budget, economy, corporation, government reform, tax reform, social security, jobs

APPENDIX 4

DIFFUSION MODEL FEATURES

MODEL	GRAPH-BASED	NON-GRAPH BASED	REGRESSION	DIFFUSION MEASURE	INFLUENCE MEASURE	PREDICTIVE
Topic Diffusion Model	YES	NO	NO	YES	YES	NO
Predictive Model T-Basic	YES	NO	NO	YES	NO	YES
Diffusion Network Model	YES	NO	YES	YES	NO	NO
Social Graph and Cascade Graph Model	YES	NO	NO	YES	NO	NO
Probabilistic Collaborative Filter model	NO	YES	NO	YES	NO	YES
Indegree, Retweets	NO	YES	NO	NO	YES	NO

and Mentions Model						
Influence Independent of Diffusion Model	YES	NO	NO	NO	YES	NO
General Threshold Model	YES	NO	NO	YES	YES	YES

APPENDIX 5

FLU DATA CORRELATION

RELATION	**CORRELATION COEFFICIENT**
Follower_Retweet	0.999977817
Retweet_Mention	0.917570189
Mention_Follower	0.917579591

APPENDIX 6

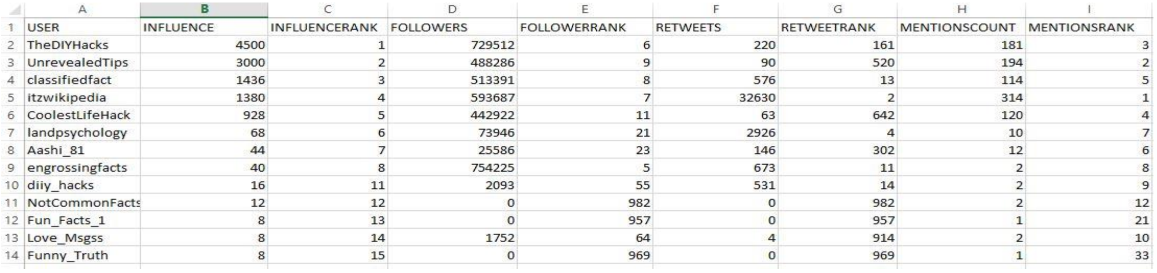
FOOD POSIONING DATA CORRELATION

RELATION	**CORRELATION COEFFICIENT**
Follower_Retweet	0.999999984
Retweet_Mention	0.93364918
Mention_Follower	0.933649164

APPENDIX 7

TABLE OF INFLUENTIAL USERS FLU DATA FOR THE HIGHEST RETWEETED TWEET

"if you drink enough fluids in the morning you will feel happier sharper and more energetic throughout the day"



	A	B	C	D	E	F	G	H	I
1	USER	INFLUENCE	INFLUENCERANK	FOLLOWERS	FOLLOWERRANK	RETWEETS	RETWEETRANK	MENTIONSCOUNT	MENTIONSRANK
2	TheDIYHacks	4500	1	729512	6	220	161	181	3
3	UnrevealedTips	3000	2	488286	9	90	520	194	2
4	classifiedfact	1436	3	513391	8	576	13	114	5
5	itzwikipedia	1380	4	593687	7	32630	2	314	1
6	CoolestLifeHack	928	5	442922	11	63	642	120	4
7	landpsychology	68	6	73946	21	2926	4	10	7
8	Aashi_81	44	7	25586	23	146	302	12	6
9	engrossingfacts	40	8	754225	5	673	11	2	8
10	diiy_hacks	16	11	2093	55	531	14	2	9
11	NotCommonFacts	12	12	0	982	0	982	2	12
12	Fun_Facts_1	8	13	0	957	0	957	1	21
13	Love_Msgss	8	14	1752	64	4	914	2	10
14	Funny_Truth	8	15	0	969	0	969	1	33

APPENDIX 8

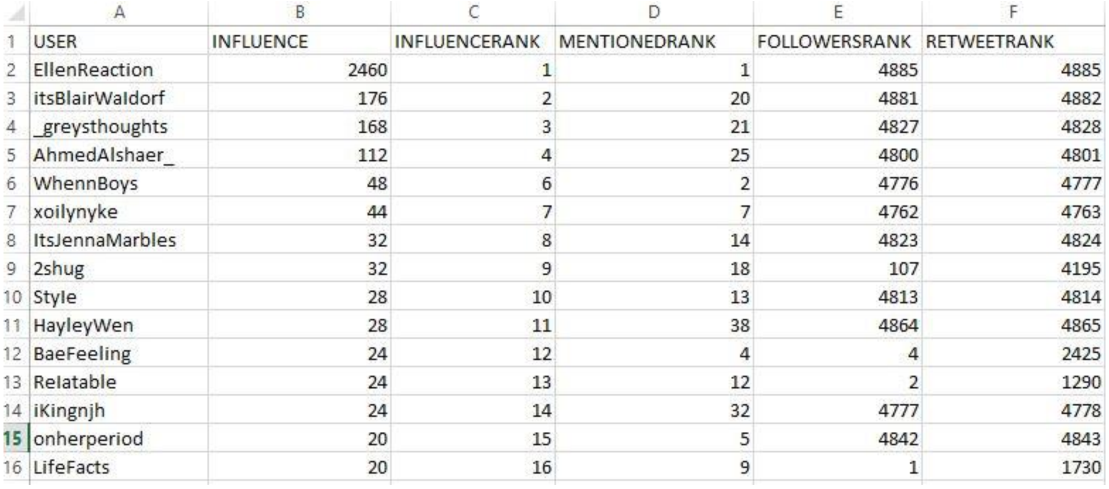
USER INFLUENCE FLU DATA

USER	MULTI-LEVEL MARKETING INFLUENCE RANK	ROOT_USER INFLUENCE RANKS	MENTIONED RANK	RETWEETS RANK	FOLLOWERS RANK
U1	1	1	1458	8490362	8419574
U2	2	2	1145	61524	1473
U3	3	3	4105	8815444	8769553
U4	4	4	1819	9288452	9278728
U5	5	5	345	2959	949
U6	6	-	8696	8739369	8687569
U7	7	6	5010	9304832	9296450
U8	8	7	1703	7745397	79338
U9	9	8	1718	8465764	8393058
U10	10	9	10722	8743050	8691537

APPENDIX 9

TABLE OF INFLUENTIAL USERS FOOD POISONING DATA FOR THE HIGHEST RETWEETED TWEET

“this gives me chills”



	A	B	C	D	E	F
1	USER	INFLUENCE	INFLUENCERANK	MENTIONEDRANK	FOLLOWERSRANK	RETWEETRANK
2	EllenReaction	2460	1	1	4885	4885
3	itsBlairWaldorf	176	2	20	4881	4882
4	_greysthoughts	168	3	21	4827	4828
5	AhmedAlshaer_	112	4	25	4800	4801
6	WhennBoys	48	6	2	4776	4777
7	xoilynyke	44	7	7	4762	4763
8	ItsJennaMarbles	32	8	14	4823	4824
9	2shug	32	9	18	107	4195
10	Style	28	10	13	4813	4814
11	HayleyWen	28	11	38	4864	4865
12	BaeFeeling	24	12	4	4	2425
13	Relatable	24	13	12	2	1290
14	iKingnjh	24	14	32	4777	4778
15	onherperiod	20	15	5	4842	4843
16	LifeFacts	20	16	9	1	1730

APPENDIX 10

USER INFLUENCE FOOD POISONING DATA

USER	**MULTI-LEVEL MARKETING INFLUENCE_ RANK**	**ROOT-USER BENEFITS INFLUENCE RANK**	**MENTIONED_ RANK**	**RETWEET_ RANK**	**FOLLOWERS_ RANK**
U1	1	1	13	997	27074
U2	2	7	6	102	4604
U3	3	8	11	4	3353
U4	4	11	1871	218828	28349
U5	5	15	2821	362138	30373
U6	6	16	1590	485125	121665
U6	7	19	1026	9638	17341
U7	8	21	2859	334375	9269
U8	9	22	719	307898	51295
U9	10	23	39562	49558	68

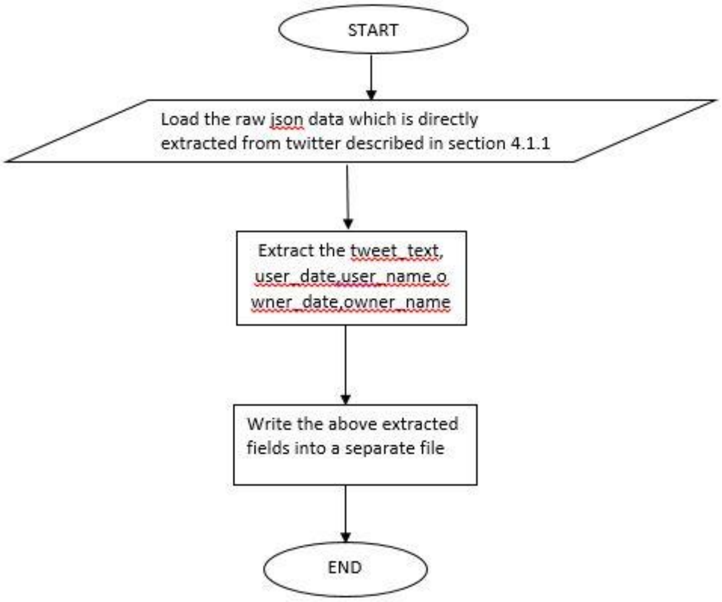
APPENDIX 11

USER INFLUENCE POLITICS DATA

USER	**MULTI-LEVEL MARKETING USER INFLUENCE_RANK**	**ROOT-USER BENEFITS USER INFLUENCE RANK**	**MENTIONED_RANK**	**RETWEET_RANK**	**FOLLOWERS_RANK**
U1	1	1	1191	11023	25743
U2	2	2	1414	553933	27235
U3	3	-	659	5023	3591
U4	4	3	1410	1153	6974
U5	5	-	3021	4055	88203
U6	6	4	82	92757	17240
U7	7	5	797	113663	55092
U8	8	6	46	229552	11333
U9	9	7	764	108401	60734
U10	10	8	42150	1812277	18217

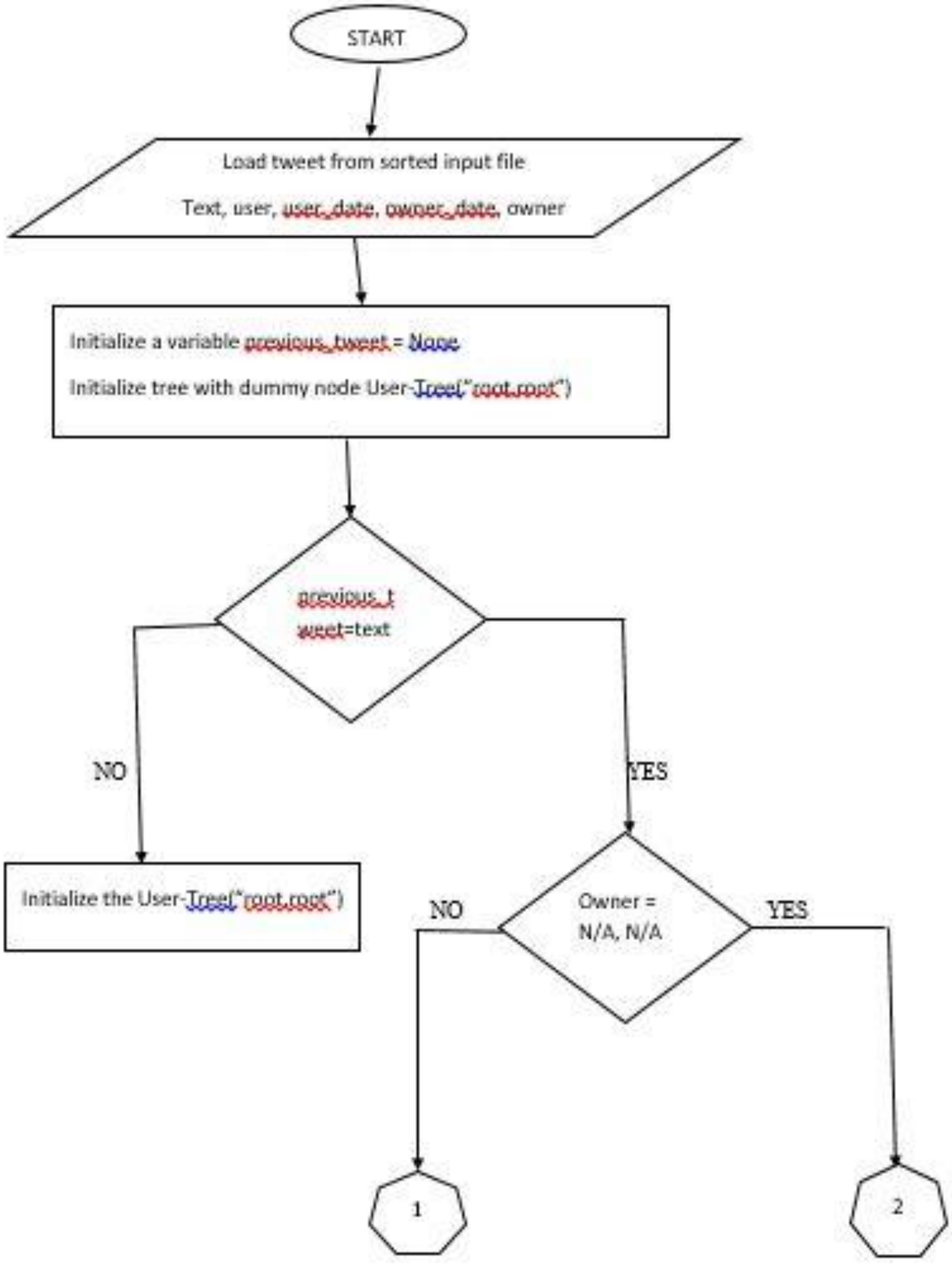
APPENDIX 12

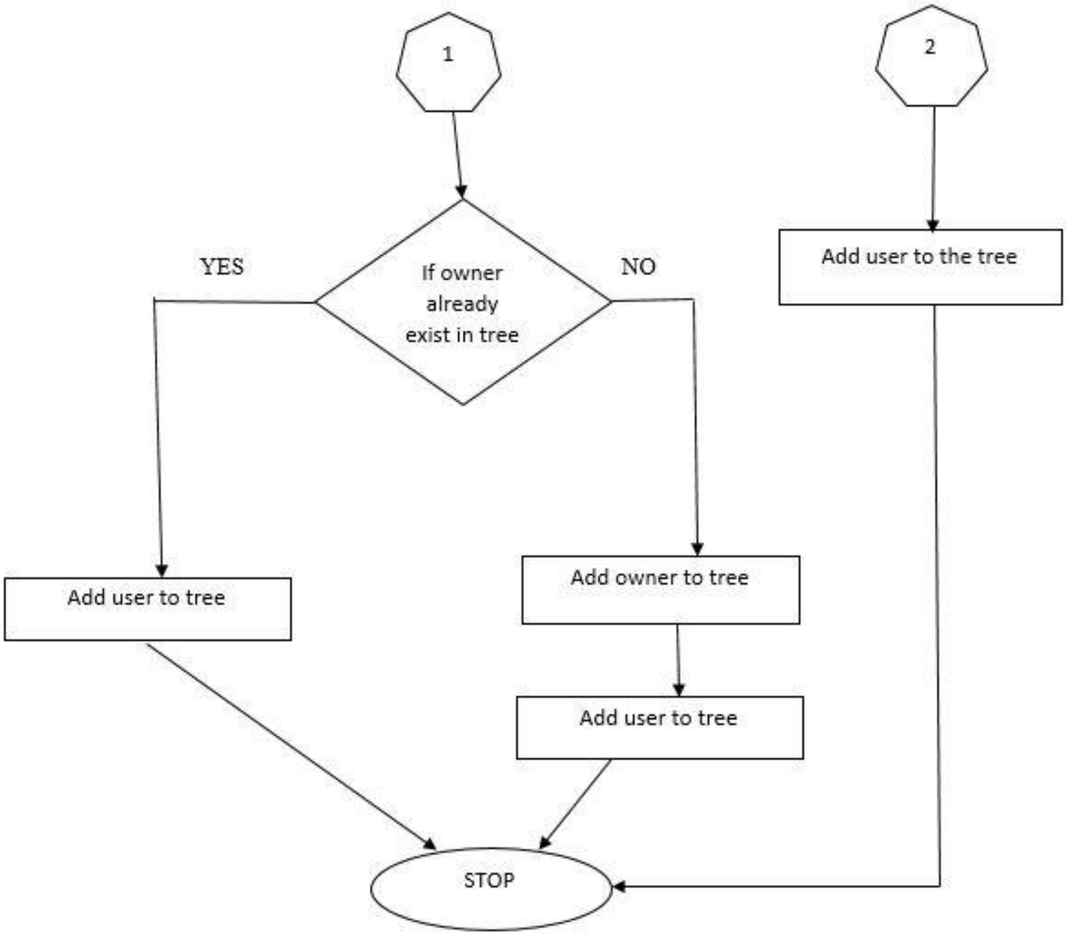
FLOWCHART TO FILTER THE RAW DATA


START
Load the raw json data which is directly extracted from twitter described in section 4.1.1
Extract the tweet_text, user_date,user_name,owner_date,owner_name
Write the above extracted fields into a separate file
END

APPENDIX 13

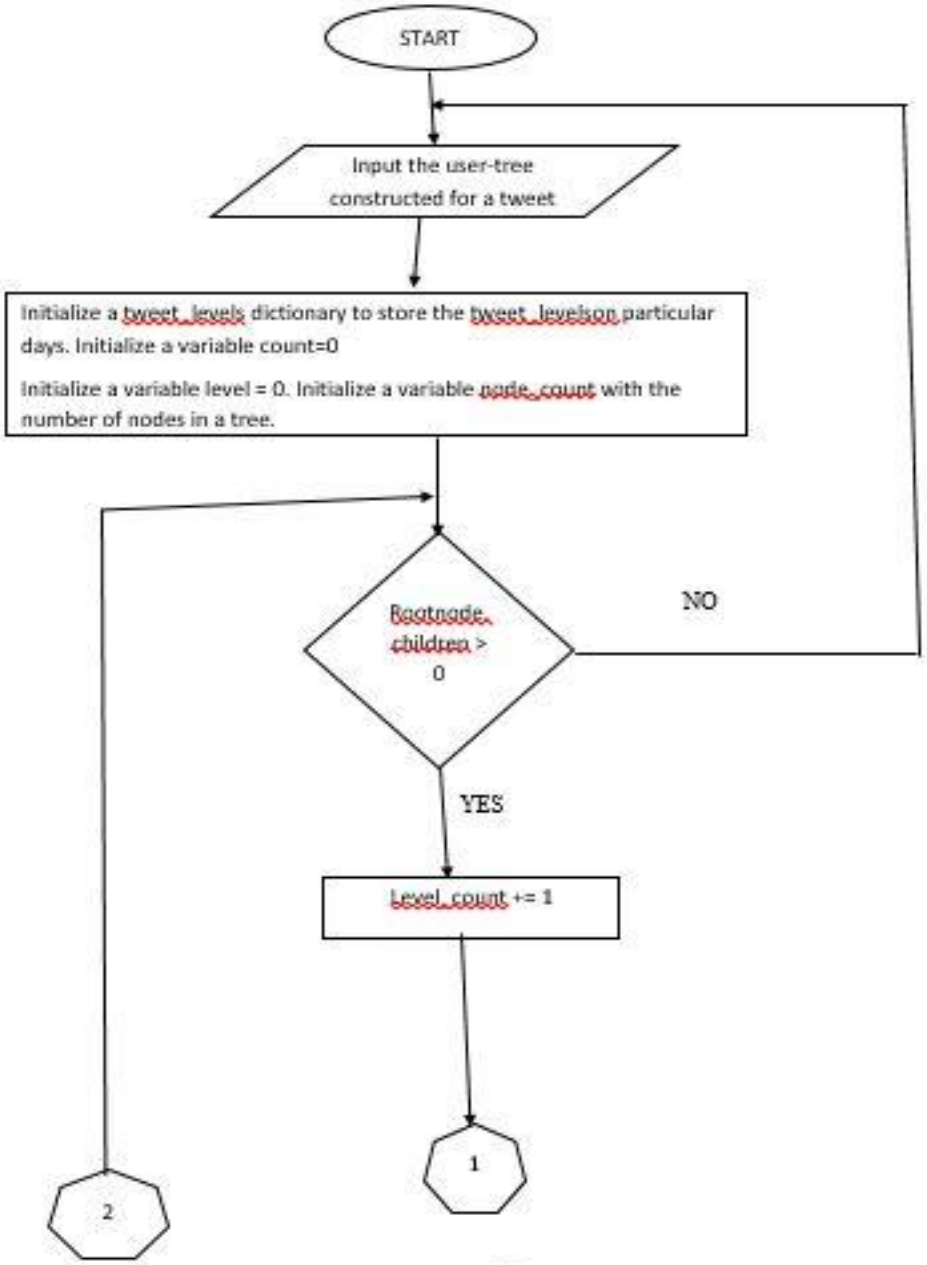
FLOWCHART TO CONSTRUCT USER-TREE

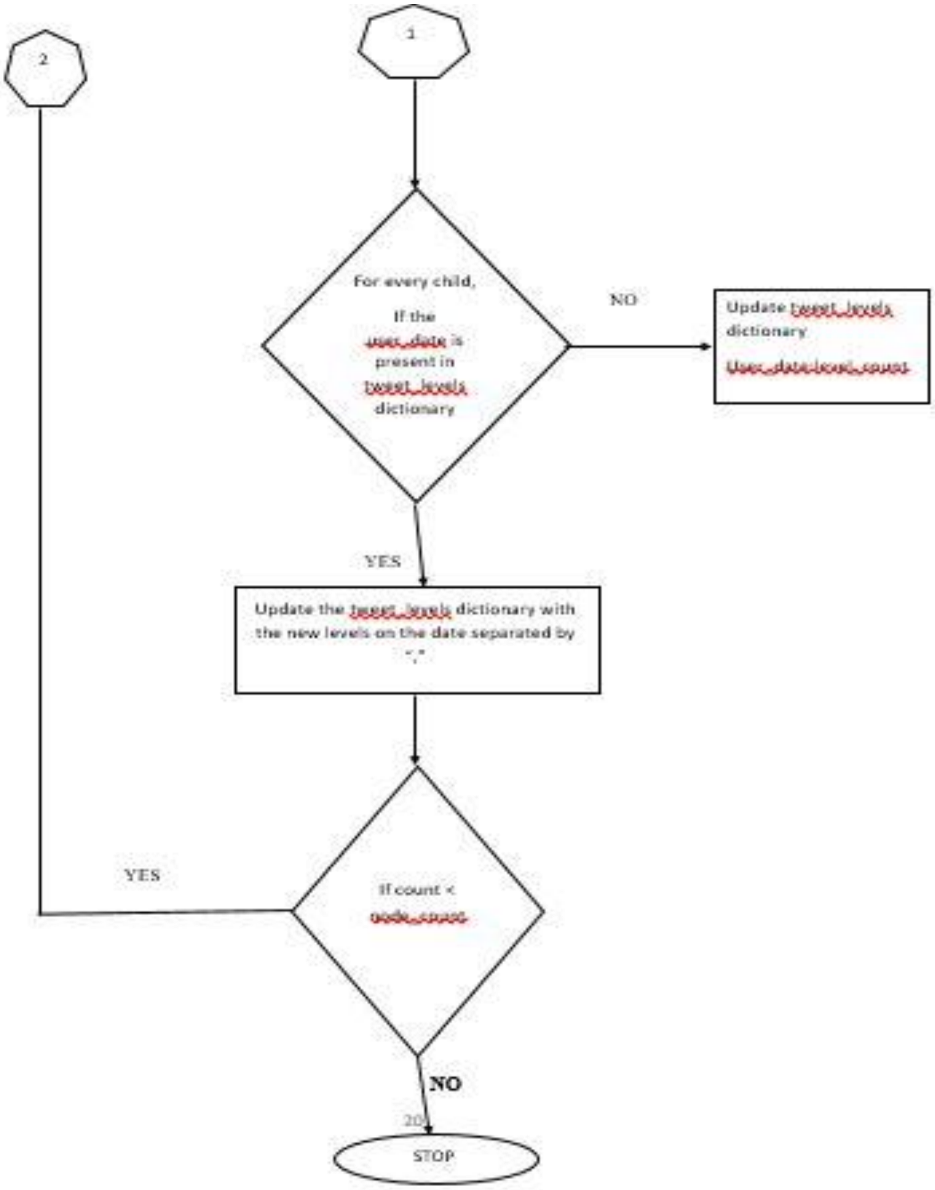

START
Load tweet from sorted input file
Text, user, user_date, owner_date, owner
Initialize a variable previous_tweet = None
Initialize tree with dummy node User-Tree("root.root")
previous_t
weet=text
NO
YES
Initialize the User-Tree("root.root")
NO
Owner =
N/A, N/A
YES
1
2


1
2
If owner already exist in tree
YES
NO
Add user to the tree
Add user to tree
Add owner to tree
Add user to tree
STOP

APPENDIX 14

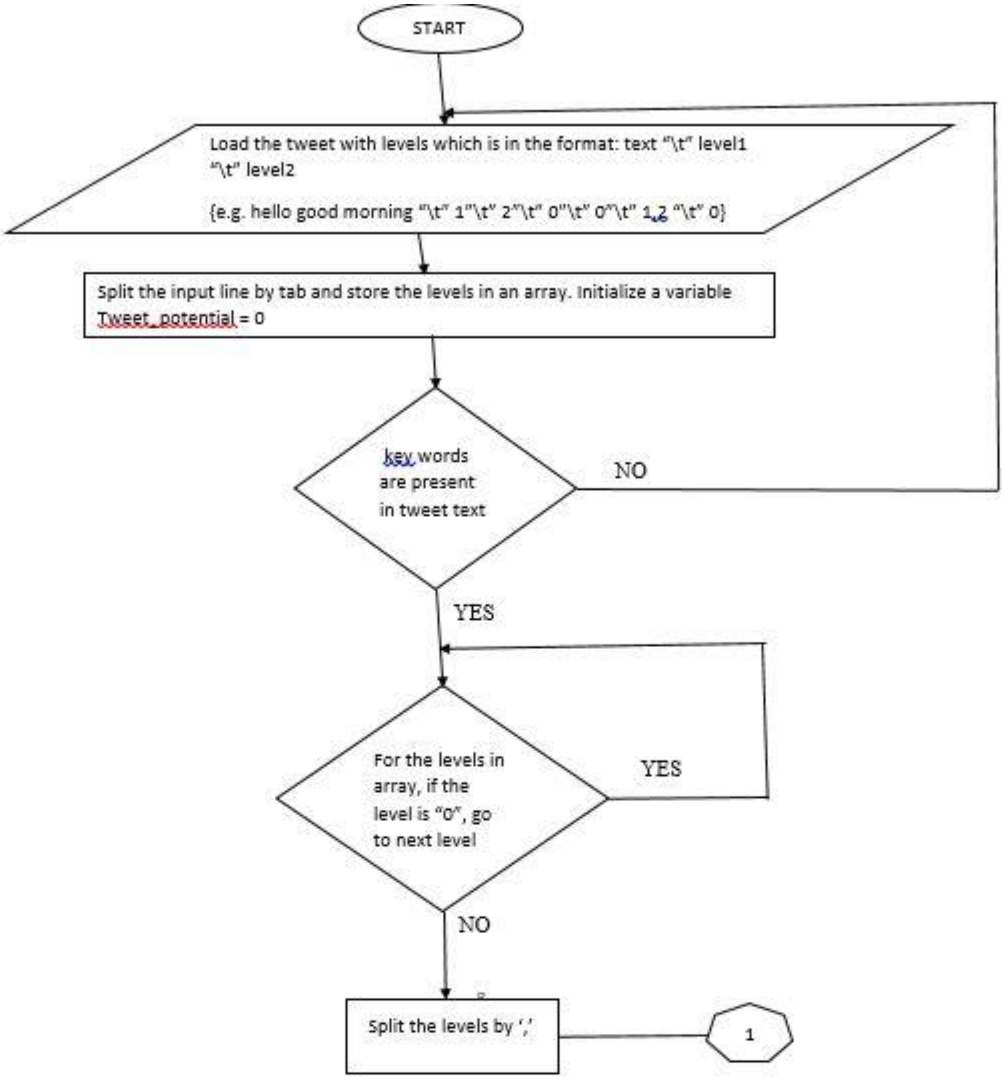
FLOWCHART TO COMPUTE TWEET POTENTIAL BY TRAVERSING USER-TREE

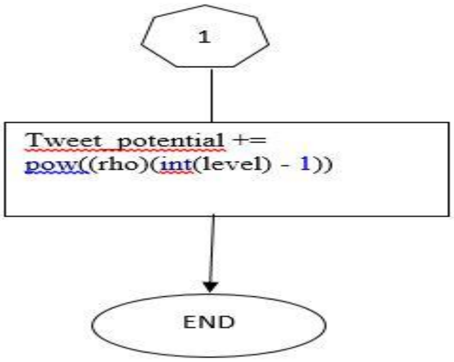

START
Input the user-tree constructed for a tweet
Initialize a tweet_levels dictionary to store the tweet_levelson particular days. Initialize a variable count=0
Initialize a variable level = 0. Initialize a variable node_count with the number of nodes in a tree.
NO
Rootnode_children > 0
YES
Level_count += 1
1
2


2
1
For every child,
If the user_date is present in tweet_levels dictionary
NO
Update tweet_levels dictionary
User_date:level_count
YES
Update the tweet_levels dictionary with the new levels on the date separated by ","
YES
If count < node_count
NO
STOP

APPENDIX 15

FLOWCHART TO COMPUTE TWEET POTENTIAL

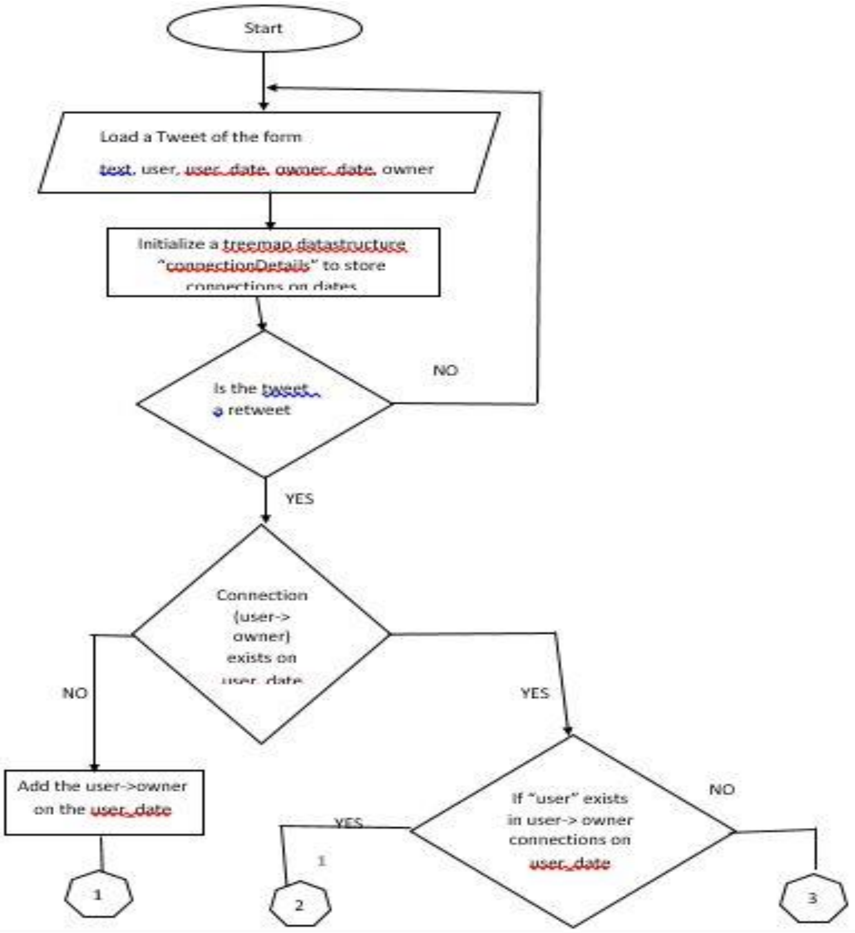

START
Load the tweet with levels which is in the format: text "\t" level1 "\t" level2
{e.g. hello good morning "\t" 1"\t" 2"\t" 0"\t" 0"\t" 1,2 "\t" 0}
Split the input line by tab and store the levels in an array. Initialize a variable Tweet_potential = 0
key words are present in tweet text
NO
YES
For the levels in array, if the level is "0", go to next level
YES
NO
Split the levels by ','
1

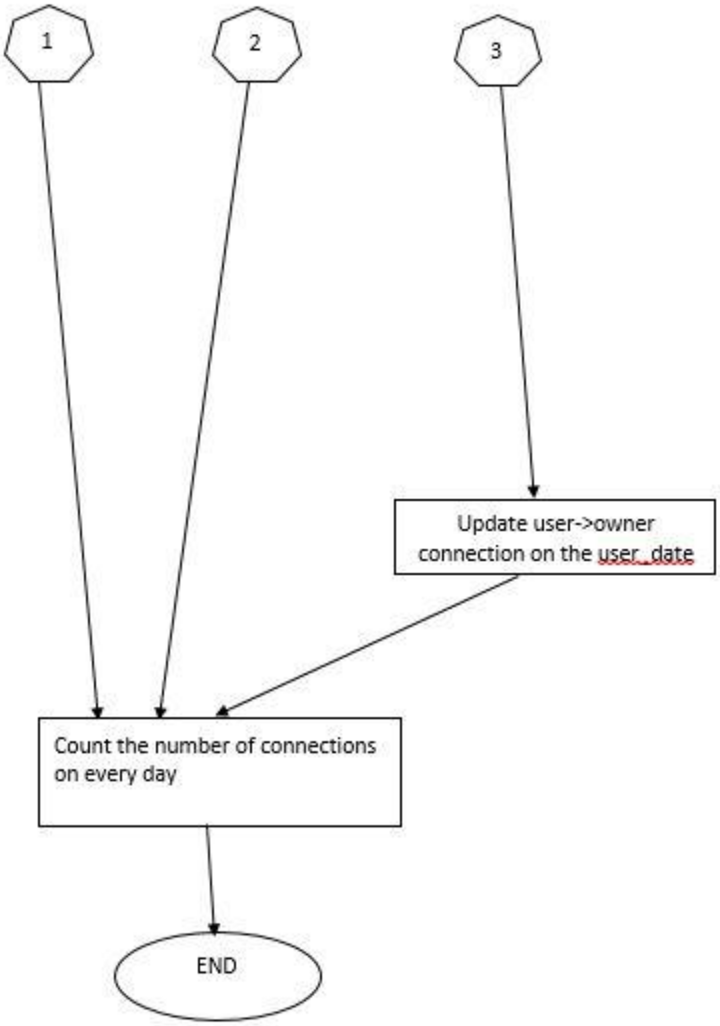


```
Tweet_potential +=
pow((rho)(int(level) - 1))
```

APPENDIX 16

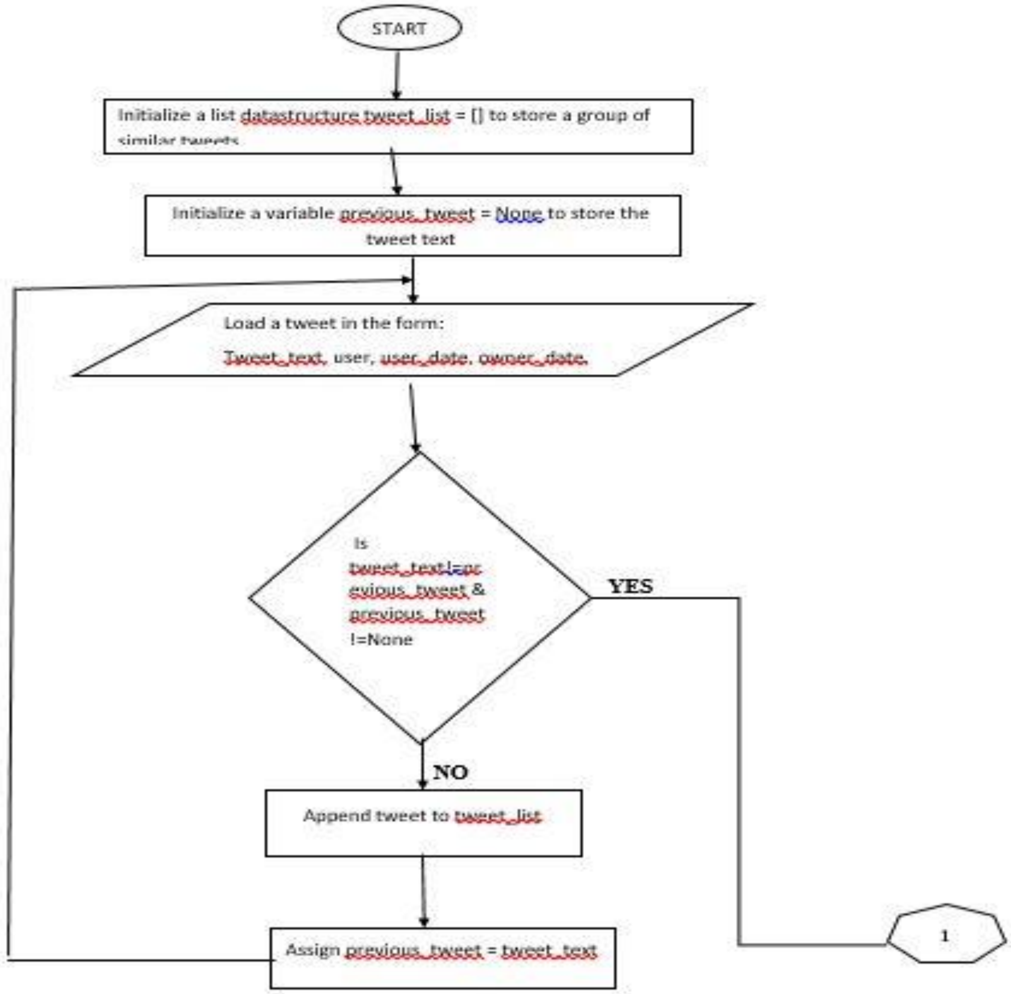
DIFFUSION NETWORK MODEL FLOWCHART

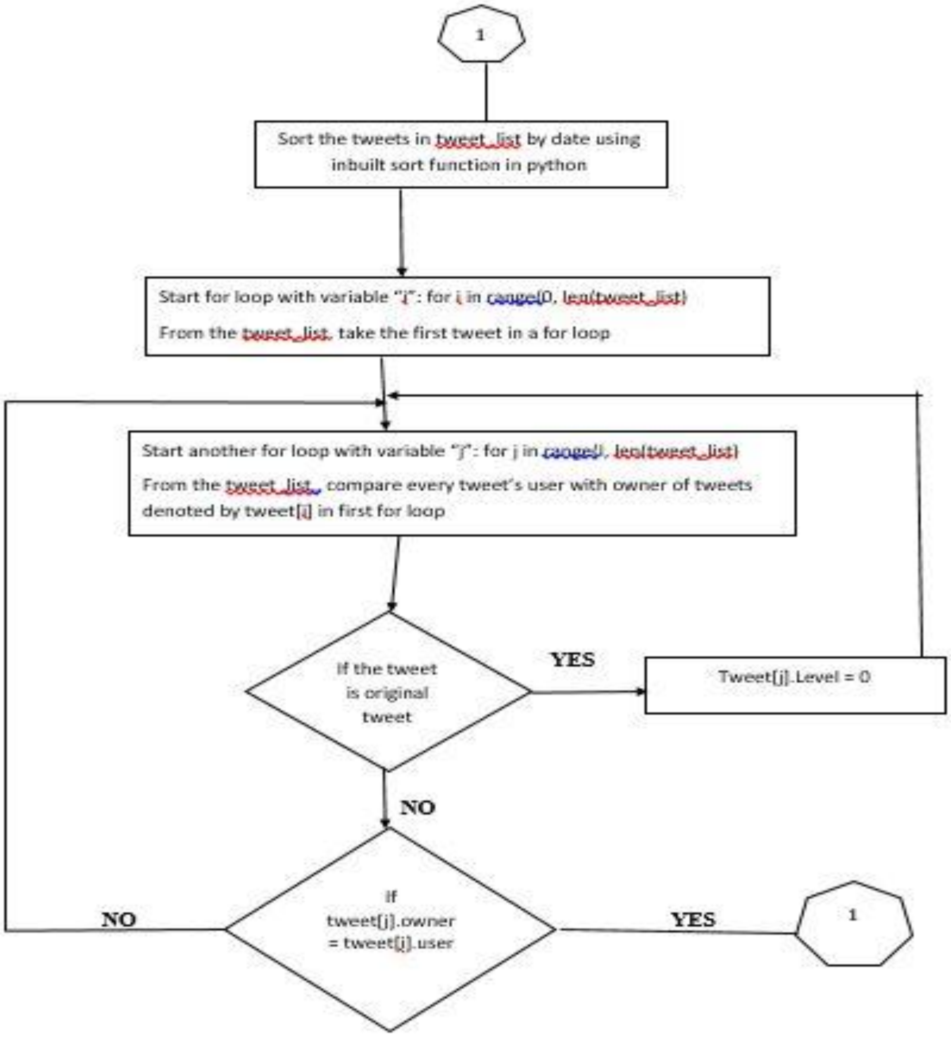


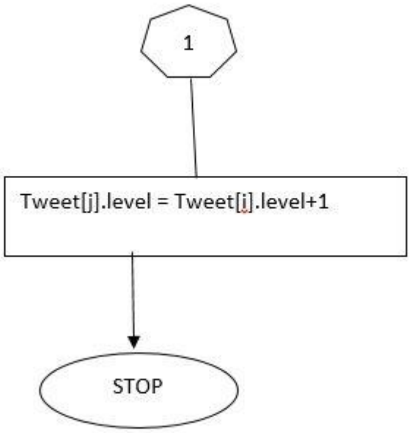

1
2
3
Update user->owner connection on the user_date
Count the number of connections on every day
END

APPENDIX 17

TWEET LEVEL COMPUTATION FLOWCHART FOR USER INFLUENCE

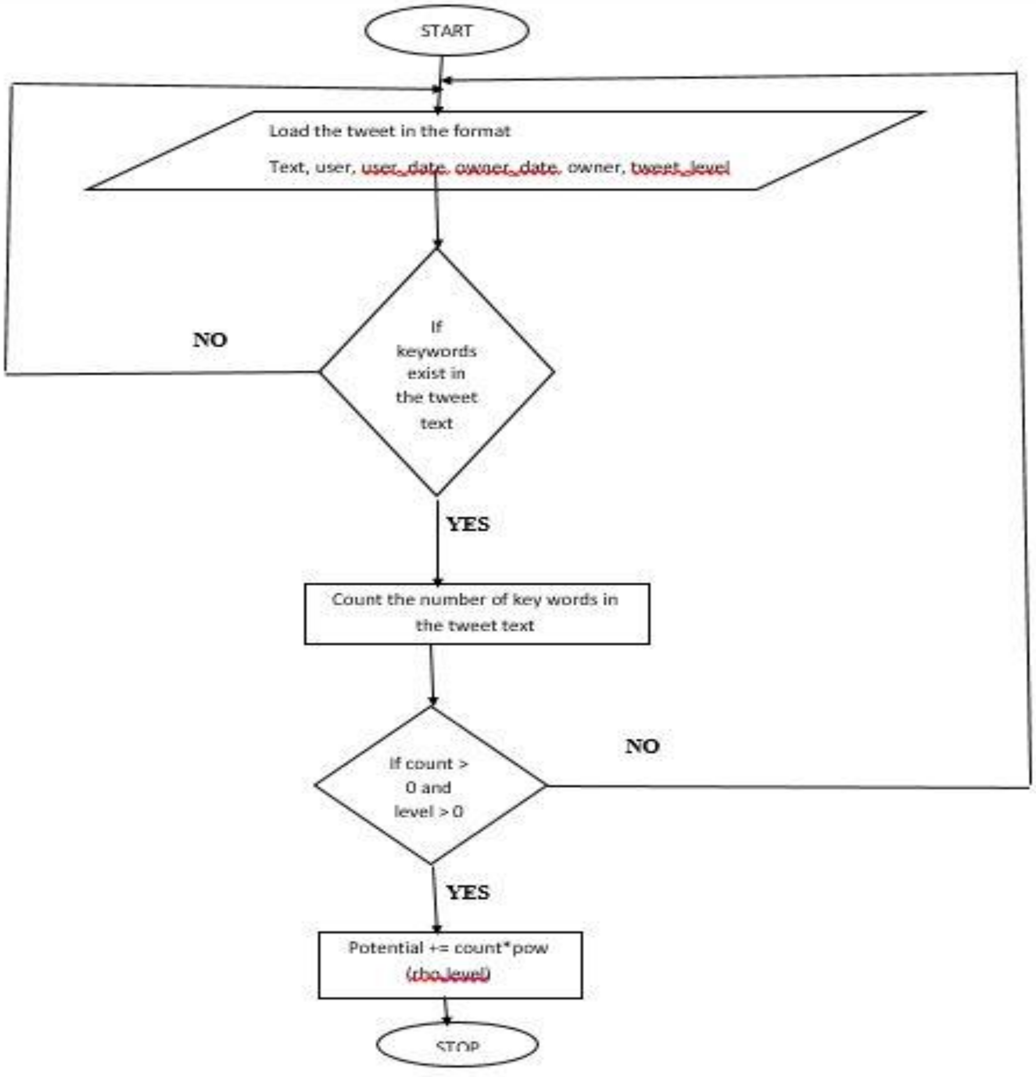

START
Initialize a list datastructure tweet_list = [] to store a group of similar tweets
Initialize a variable previous_tweet = None to store the tweet text
Load a tweet in the form:
Tweet_text, user, user_date, owner_date,
Is tweet_text!=previous_tweet & previous_tweet !=None
YES
NO
Append tweet to tweet_list
Assign previous_tweet = tweet_text
1


1
Sort the tweets in tweet_list by date using inbuilt sort function in python
Start for loop with variable "i": for i in range(0, len(tweet_list)
From the tweet_list, take the first tweet in a for loop
Start another for loop with variable "j": for j in range(i, len(tweet_list)
From the tweet_list, compare every tweet's user with owner of tweets denoted by tweet[i] in first for loop
If the tweet is original tweet
YES
Tweet[j].Level = 0
NO
If tweet[j].owner = tweet[i].user
NO
YES
1


1
Tweet[j].level = Tweet[i].level+1
STOP

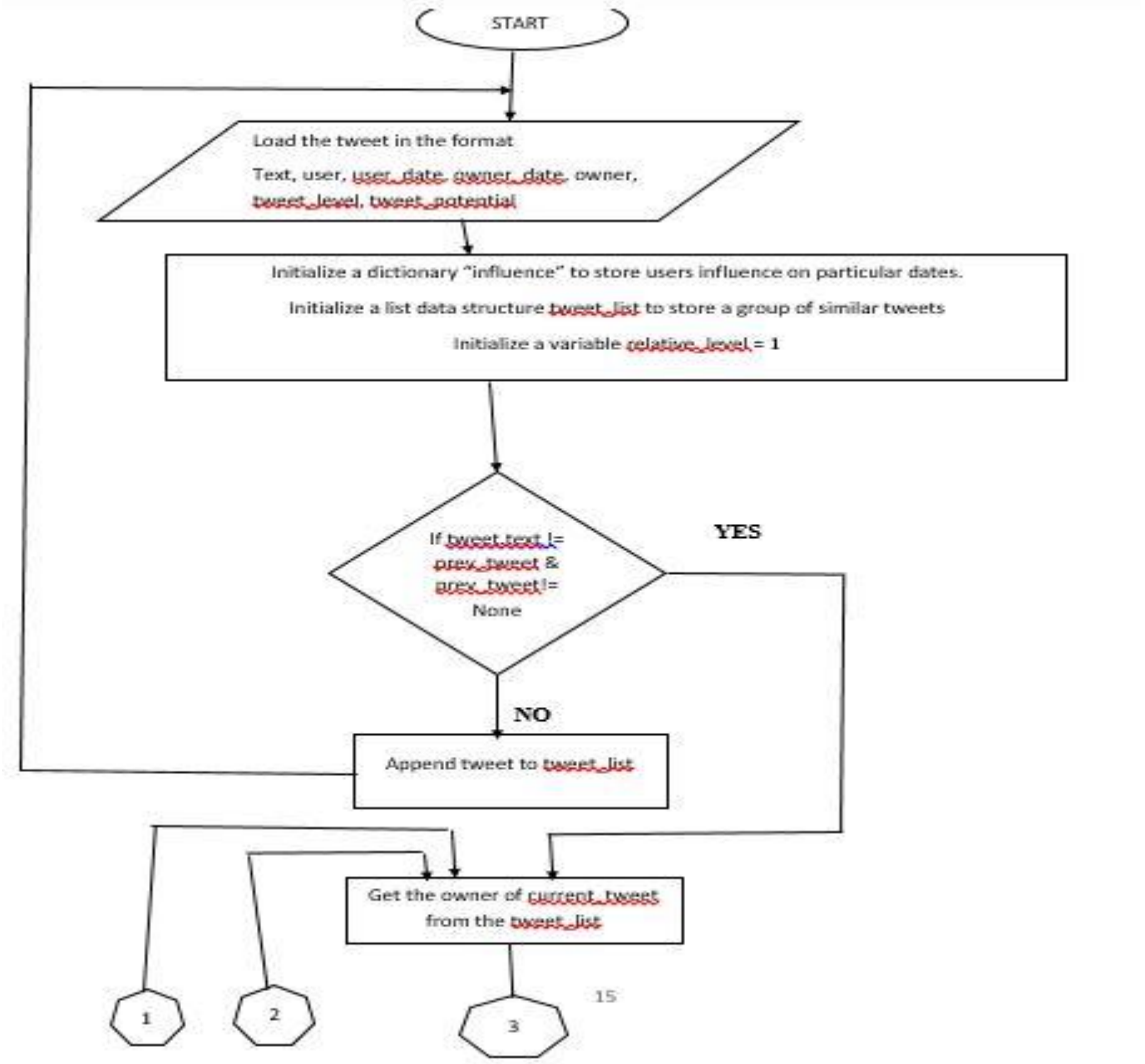
APPENDIX 18

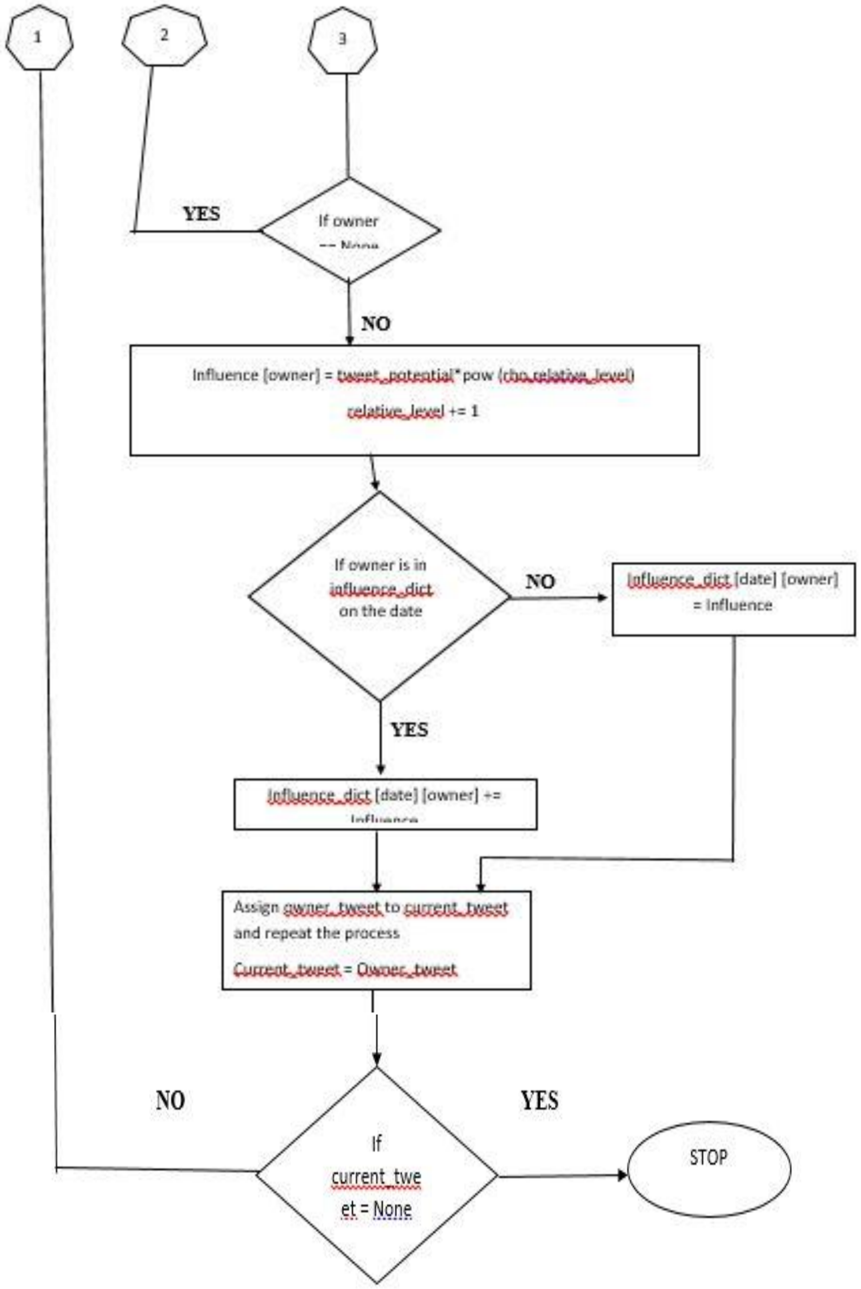
TWEET POTENTIAL COMPUTATION FLOWCHART FOR USER INFLUENCE


START
Load the tweet in the format
Text, user, user_date, owner_date, owner, tweet_level
NO
If keywords exist in the tweet text
YES
Count the number of key words in the tweet text
If count > 0 and level > 0
NO
YES
Potential += count*pow (rho,level)
STOP

APPENDIX 19

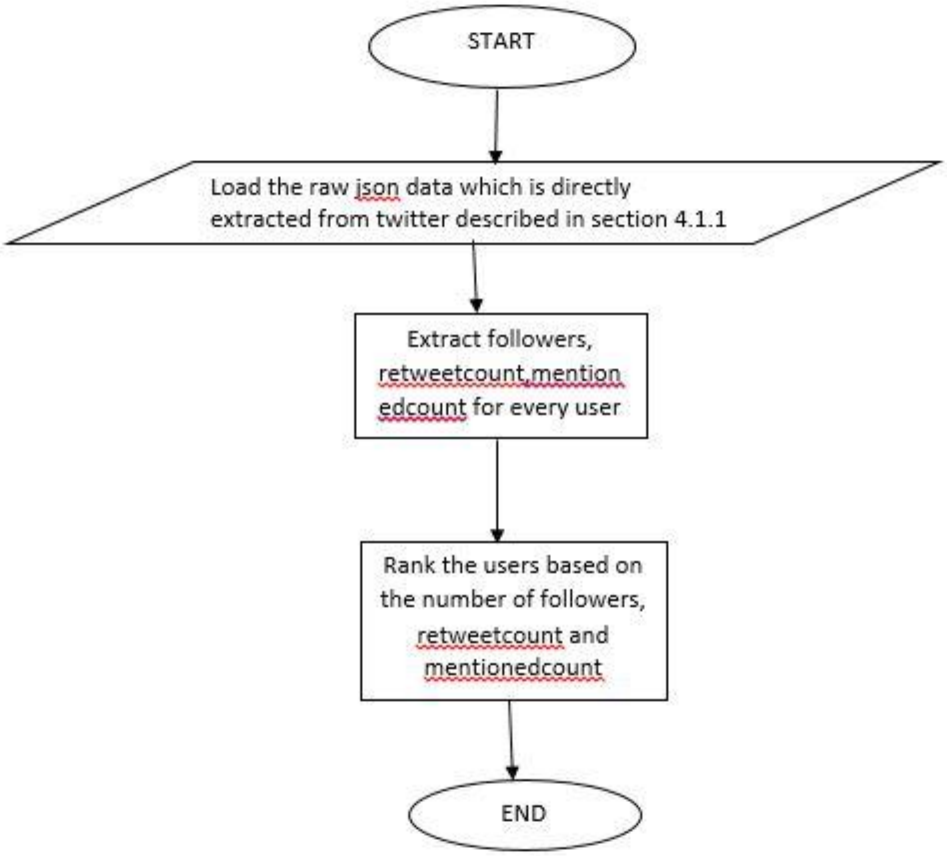
USER INFLUENCE COMPUTATION FLOWCHART


START
Load the tweet in the format
Text, user, user_date, owner_date, owner, tweet_level, tweet_potential
Initialize a dictionary "influence" to store users influence on particular dates.
Initialize a list data structure tweet_list to store a group of similar tweets
Initialize a variable relative_level = 1
If tweet.text != prev_tweet & prev_tweet != None
YES
NO
Append tweet to tweet_list
Get the owner of current_tweet from the tweet_list
1
2
3
15


1
2
3
YES
If owner
== None
NO
Influence [owner] = tweet_potential* pow (rho,relative_level)
relative_level += 1
If owner is in
influence_dict
on the date
NO
Influence_dict [date] [owner]
= Influence
YES
Influence_dict [date] [owner] +=
Influence
Assign owner_tweet to current_tweet
and repeat the process
Current_tweet = Owner_tweet
NO
YES
If
current_twe
et = None
STOP

APPENDIX 20

RETWEETS, INDEGREE, MENTIONS MODEL FLOWCHART


START
Load the raw json data which is directly extracted from twitter described in section 4.1.1
Extract followers, retweetcount,mentionedcount for every user
Rank the users based on the number of followers, retweetcount and mentionedcount
END

APPENDIX 21

EXTRENAL LINKS

Sr No	**EXTERNAL LINKS**
1	https://www.cdc.gov/flu/news/2014-2015-flu-season-wrapup.htm
2	http://www.huffingtonpost.com/2015/06/06/2016-flu-shot_n_7521344.html
3	http://www.cbsnews.com/news/latest-trump-news-today-february-23-2017/
4	http://www.cbsnews.com/news/today-in-trump-february-22-2017/

VITA

Vishali Narayana

Candidate for the Degree of

Master of Science

Thesis: MESSAGE PROPAGATION AND SOCIAL INFLUENCE IN TWITTER

Major Field: Computer Science

Biographical:

Education:

Completed the requirements for the Master of Science in your Computer Science at Oklahoma State University, Stillwater, Oklahoma in July, 2017.

Completed the requirements for the Bachelor of Technology in your Computer Science at JNTUH, Hyderabad, India in 2015.

Experience:

Professional Memberships: