A Novel Design Framework For Rumour Analysis In Twitter

P.SuthanthiraDevi, S.Karthika

Abstract:In social networking, a large amount of information can be disseminated to different social media users. Many times the social media users spread this information in unverified manner. The unverified information or misinformation is referred as rumors. Identifying source of rumor in a social media is a critical problem. Our main aim is to develop a rumor source detector, which will assist, in real time to determine the source of rumor in social media. In this paper the authors present, two methodologies to detect the source of rumor, first one is Graph Theoretical Framework and the second one is Veracity Assessment Framework. Graph Theoretical approach, the authors suggest to use regular trees, graphs and rumor centrality to estimate the source of the rumor. In Veracity Assessment framework, data coupled with real-time streaming and to understand the lifecycle of rumor. It is designed for automatically predicting rumor veracity at different time spans based on the conversation patterns.

Keywords: Rumor, Diffusion, Centrality, SIR, Ingest, Veracity, RSS

I. INTRODUCTION

Social media have progressively attained fame in the past decade, making possible for the people to have the 'in touch feel' among their personal circle, and also to stay aware of minute to minute ongoing, upcoming events and breaking news as they are disclosed. The most significant feature of social media is its capability for propagating information rapidly through a large community of users. Information can be spreaded in unverified manner in the form of a rumor. Despite being unverified, it propagates as epidemic to a large crowd, influencing their insight about the event. The instances of rumors like Barack Obama being injured in White House bombing and financial aid scheme during Hurricane Sandy has forced US Federal Emergency Management Agency to set up a Web page specifically for rumor control. The rumors of reoccurrence of earthquake in Nepal provoked chaos in the rehabilitation centers for few days. Currently, with the increasing reach of social media, there is a need for a system to keep track, filter and curb the spreading of rumor. Construction of rumor source detector has two key challenges 1) how to construct it and 2) how to fix the limits for source detector. Graph Theoretical Framework used to identify the source of the rumor in large scale networks. Rumors are initiated by single source node and spread over the social network.

Revised Manuscript Received on April 10, 2019
P.SuthanthiraDevi, Associate Professor,SSN College of Engineering
S.Karthika, Associate Professor,SSN College of Engineering

The main objective of this graph model is to find who originates the rumor at the specific time. Veracity Assessment Framework is used to identify and predict veracity of rumor. It is automatically envisaging rumor veracity at different time spans based on the conversation Patterns . This model will catalogue the source of the posted tweet like government officials, new channels, and members of private sector, etc., to validate the authority. It also tracks the sources that authenticate or deny the tweet. All these input parameters will be used to observe the evolution of the rumor based on the conversation in the social networkSection 2, discusses the various works related to rumor and describes existing design models related to these two frameworks. Section 3, presents the graph model to construct a regular tree in a natural and computationally efficient manner to identify rumor source in a particular graph. Each node in a tree assigned using positive values, and this value is called as rumor centrality [1].The rumor centrality act as a source detector of the rumor in social networks. Section 4, the design of Veracity Assessment Framework aims to identify and expose rumors as they spread across social network. The main objective of this framework is to model rumor ontology using the spread reaction sub-classes to overcome the factuality issue in existing message crafting model for identifying the influential source in propagating the rumor.

II. RELATED WORK

We started our work, by identifying the source of rumor from the set of social media posts. To analyze the rumor, the first problem is to finding rumor source. The authors of [1] to follow the SIR model to design a rumor source identifier and also define rumor centrality with ML estimator. In contrast, the authors [2] using small amount of origin information, first they find whether that information is rumor or not. The authors of [3] use method of two-stage algorithms, stage one uses cluster to identify the source of rumor and stage two uses search algorithm within cluster. Rumor source with partial identification focuses to recover the misinformation using three approaches, 1) Compressed Sensing Base 2) DN Matrix Completion 3) DN Matrix Completion with Renewal theory based analysis [4][5]. Castillo et al. presents automatic tool based on credibility twitter posts. They proposed machine leaning the model, used to analyze vast amount of features and to group the source of rumor based on user wise events or topic-wise events. They also present tweet propagation pattern to segregate rumor and non-rumor [6] [7]. Some framework focuses metrics like trust and quality of social media information.

The authors [8] [9] build, a machine language framework used to evaluate the trustworthiness of information based on policy based approach.

III. RUMOR SOURCE DETECTION - GRAPH THEORETICAL FRAMEWORK

Most of the source detectors based on the Susceptible-Capable-Recovered (SCI) network model, where **Error! Reference source not found.** is the susceptible nodes, **Error! Reference source not found.** is the capable of being infected nodes that can spread the rumour and **Error! Reference source not found.** is the recovered nodes are curable [9].In social networking the authors assume, each node is defined as time invariant and undirected graph. The authors design an undirected graph**Error! Reference source not found.**, where **Error! Reference source not found.**the set of infinite nodes is and **Error! Reference source not found.** is the set of edges of the form **Error! Reference source not found.** for**Error! Reference source not found.**. Initially **Error! Reference source not found.** node is the source of the rumour. Each node in v £ V, has three possible nodes, susceptible (**Error! Reference source not found.**, infected (**Error! Reference source not found.**), recoverable (**Error! Reference source not found.**).If x node has a rumour, it is possible to spread to another node y only if **Error! Reference source not found.**. Let **Error! Reference source not found.** be the time interval between the x and y to send and receive the rumour. The following section presents rumour source detector using Regular trees and graphs.

A. REGULAR TREE ANALYSIS

Assume rumor spread at time 0, at the **Error! Reference source not found.** node, over the social network of the graph **Error! Reference source not found.**.After some time, all the infected nodes form a connected sub graph of**Error! Reference source not found.**. Rumor detector denoted by**Error! Reference source not found.Error! Reference source not found.**, it is estimated based on the observation of the G_N and original source node s

Error! Reference source not found.
(1)

Where **Error! Reference source not found.** is the probability of observing **Error! Reference source not found.** under SIR model, where s**Error! Reference source not found.** is the source of the s**Error! Reference source not found.***.The authors note that **Error! Reference source not found.** may not be compute rumor source node tractable. For Regular tree, **Error! Reference source not found.** is proportionally related to the quantity and structure of **Error! Reference source not found.**Consider **Error! Reference source not found.** as a regular tree, we need to find the probability of all infected nodes in a tree.

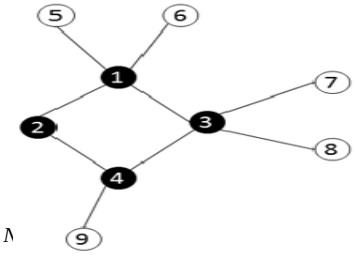

Fig. 1: Rumor Graph with four nodes

In Figure 1 shows the tree structure with N=4. If node1 detected as a source node, we need to calculate **Error! Reference source not found.**the possibilities of rumor spreading nodes path are (1,2,4,3) and (1,3,4,2) based on the network structure nodes (1,4,3,2) are not infected. To evaluate the**Error! Reference source not found.**, find all permitted permutations and their corresponding probabilities. The set of all permitted permutation **Error! Reference source not found.** form the rumor graph**Error! Reference source not found.**, starting with source node **Error! Reference source not found.Error! Reference source not found.**. To obtain the probability **Error! Reference source not found.** for each σ ∈ **Error! Reference source not found.** where **Error! Reference source not found.** = **Error! Reference source not found.**.This is denoted by

Error! Reference source not found. (2)

This can be rewritten as**Error! Reference source not found. Error! Reference source not found.**) **Error! Reference source not found.**

Error! Reference source not found. (3)

B. GRAPH ANALYSIS

The rumor source detector is computed by using general graph with spanning tree. Each node receives rumor, the authors first identify, which spanning tree was involved in the rumor spreading process. Based on the spanning tree they compute regular tree detector. Assume a node s **Error! Reference source not found.**N was the source node to spread rumor, then bfs tree rooted between **Error! Reference source not found.** and $T_{bfs}(s)$. The general graph source detector is given by

Error! Reference source not found. (4)

(4)

C. RUMOR CENTRALITY: INFORMATION DIFFUSION

Regular trees used to observe only the rumor spread nodes, **Error! Reference source not found. Error! Reference source not found.**and edges between them. Our main goal is to find the rumor source in an undirected graph **Error! Reference source not found.**.Rumor Centrality used to overcome the limitations of the trees and graphs. In Rumor

Centrality, let **Error! Reference source not found.**be**Error! Reference source not found.**, counts the number of the rumors spreading in a graph, where s**Error! Reference source not found.** as a source node. Each node of **Error! Reference source not found.** has a graph-score value and assigns a non-negative number for each of the vertices. The maximal score value of the source node is called as Rumor Center. The rumor centrality for graph **Error! Reference source not found.** is denoted by

Error! Reference source not found. Error! Reference source not found.
(5)

Where **Error! Reference source not found.** is the size of the sub tree of graph, where **Error! Reference source not found.Error! Reference source not found.** is root node and **Error! Reference source not found.** as a source node.

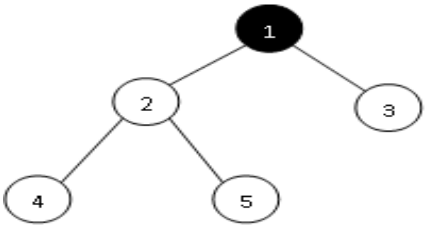

Fig. 2: Rumor Centrality for source node

Let **Error! Reference source not found.**be the node 1, based on the sub tree values **Error! Reference source not found.** is 5.All the nodes started spreading from the root node 1, then the rumor centrality of **Error! Reference source not found.** equal to 8. The spreading sequence is {1,3,2,4,5},{1,2,3,4,5}, {1,2,4,5,3}, {1,2,4,3,5}, {1,3,2,5,4}, {1,2,3,5,4}, {1,3,2,5,4},{1,2,5,3,4},{1,2,5,4,3}.

IV. DESIGN OF RUMOR VERACITY ASSESSMENT FRAMEWORK A STEP BEFORE THE FINAL SUBMISSION

The authors suggest this model is used to assess the veracity of rumors. The major emphasis of this model will be storing, processing and analyzing large volumes of historical data coupled with real-time streaming new data to understand the lifecycle of rumor which is evolution, propagation and demystifying. This model follows the data value chain for veracity checking. It encompasses seven steps namely discover, ingest, process, persist, integrate, analyze and expose. In the proposed framework these steps are presented as three major modules namely capturing of data, model building, and visualization.

A.DATA VALUE CHAIN – INTEGRATION PERSPECTIVE

The data value chain in the proposed model will perform two types of integration namely data integration and software integration to process the historical and streaming data for analyzing and predicting veracity. In Data Integration, the veracity assessment model will integrate data sources like social media and knowledge db. The social media content will acquire and store data from the open links. This framework adapts the novel data-flow and stream oriented approach for handling voluminous data in timely fashion. This stored data will be used for training the models for later rumor classification in real-time data. The streaming data can be stored in NoSQL and the batch processing can be efficiently performed using Apache HBase. The other pre-processing and analytical tasks like text indexing, retrieval based on multiple dimensions can be performed using Apache Solr. The knowledge content repository will present the data for analytics using annotation and concept extraction methodology. The Linked Open Datasets (LOD) including DBPedia, Wiki-data, RSS and GeoNames will be used for building the ontology, identifying and tagging the concepts. Software Integration follows the concept of pipe-lines. It should be able to increase the throughput and decrease the latency as much as possible. Generally in real time processing, more processing units can be added either nodes to a system or running the whole pipeline on a high speed machines. Apache Kafka is a high-throughput publish/subscribe messaging system and it is frequently used to enable data pipelining. The real-time computation can be performed in Storm due to its easy integration with Apache tools. It has excellent scalability, tolerance to failures and integration capabilities. The data processing can use the Flick libraries and REST API for executing basic and multi-dimensional based queries from users. The following Figure 3 shows the data and software integration of various modules in veracity assessment framework. The User Generated Content (UGC) across the media should be clubbed to ease the analysis. They should be associated, indexed and stored. The multiple tweets or posts from the data source should be collected using Data Collectors (DC). It accumulates data within a start and end time interval using configurable user queries like keyword or hashtag. The data source will be recognized based on the Source ID and the query definition type. The capture module will perform the following three steps, 1) The DC and data sources are created to retrieve and store the collected data from the sources specified.2) The likely rumorous tweets will be sampled using the novel methodology of selecting the most highly retweeted tweets based on reply/respond, retweet and mention patterns.3) The complete conversations for the sample of likely rumorous tweets will be collected. The conversation includes all the tweets that reply/respond to each of the tweets. The conversation collection step will be performed by using the tweet ID of each of the selected tweets.

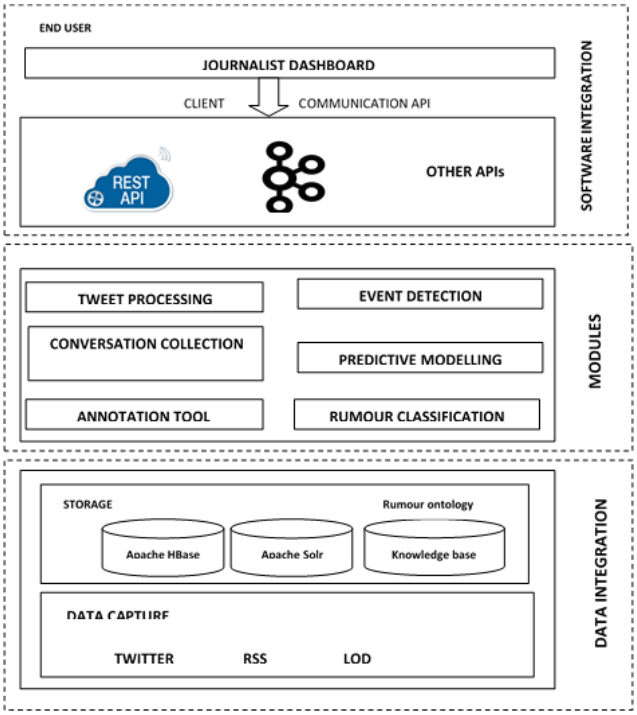

Fig. 3: Overview of modules and integration approach

The annotation schema will be used for training the models for the given search query using the existing rumor ontology as training set and the data collected from the social database as test set. This model classifies the tweets as rumors after several iterations. It depicts the training and evaluation process in the model building module. Conversation collection module contains all tweets and retweet data. Tweet Processing used for feature extraction, Cleaning, and Normalization. Event Detection module used to find particular event during emergency periods.

The above framework shows the veracity assessing used to overcome the chaos during crisis time. The disaster management will be made effective by tracking the data, reducing the cost of manual content curation and presenting the corrected data.

V. CONCLUSION

In Social Networking system, one of the biggest challenges is to find the accuracy of the information. In order to find veracity of data, the authors collect disaster time based tweets and analyze the rumor. This paper presented two methodologies for identifying the source of the rumor. In Graph Theoretical Framework, the authors design a rumor source detector by using regular trees and graphs. The authors suggest, rumor centrality will provide a better solution for rumor source identification. The Veracity Assessment model quickly verifies information and track its origin which would enable government and other sectors to respond more effectively for crisis and disaster management.

REFERENCES

1. Shah D, Zaman T, "Rumors in a network: Who's the culprit?", IEEE Transactions on information theory, 57(8), 5163-5181. (2011).
2. Seo E, Mohapatra P, Abdelzaher T, "Identifying rumors and their sources in social networks", SPIE defense, security, and sensing, 83891I-83891I. (2012).
3. Louni A, Subbalakshmi KP, "A two-stage algorithm to estimate the source of information diffusion in social media networks", In Computer Communications Workshops (INFOCOM WKSHPS), IEEE Conference on (pp. 329-333). (2014, April).
4. Donoho DL, "Compressed sensing" ,IEEE Transactions on information theory, 52(4), 1289-1306. (2006).
5. Louni A, Santhanakrishnan A, Subbalakshmi KP, "Identification of source of rumors in social networks with incomplete information", arXiv preprint arXiv: 1509.00557. (2015).
6. Castillo C, Mendoza M, Poblete B," Information credibility on twitter", In Proceedings of the 20th international conference on Worldwide Web (pp. 675-684).ACM. (2011).
7. Castillo C, Mendoza M, Poblete B," Predicting information credibility in time-sensitive social media", Internet Research, 23(5), 560-588. (2013).
8. Nurse JR, Agrafiotis I, Goldsmith M, Creese S, Lamberts, K" Two sides of the coin: measuring and communicating the trustworthiness of online information", Journal of Trust Management, 1(1), 5. (2014).
9. Nurse JR, Creese S, Goldsmith M, Rahman SS, "Supporting human decision-making online using information-trustworthiness metrics", In International Conference on Human Aspects of Information Security, Privacy, and Trust (pp. 316-325). Springer, Berlin, Heidelberg. (2013, July).
10. Nausheen Azam, Jahiruddin, Muhammad Abulaish , "Twitter Data Mining for Events Classification and Analysis", In Second International Conference, Delhi, IEEE,pp.978-1-4673-9819-0,2015.
11. Chua AYK, Banerjee S "Analyzing Users Trust for Online Health Rumors", Springer International Publishing Switzerland, pp.33-38, 2015.
12. Hannak A, Margolin D, Keegan BI, Weber,"Get back! You don't know me like that: The social mediation of fact checking interventions in twitter conversations", In: Proceedings of the AAAI Conference on Weblogs and Social Media, 2014.