

I.	Project Title	Finding the Source of Rumor in Large Online Social Networks.
II.	Abstract	<p>We address the problem of estimating the source of a rumor in large-scale social networks. Previous works studying this problem have mainly focused on graph models with deterministic and homogenous internode relationship strengths. However, internode relationship strengths in real social networks are random. This uncertainty is modeled by using random, non-homogenous edge weights on the underlying social network graph. Also a novel two-stage algorithm has been proposed that uses the modularity of the social network to locate the source of the rumor with fewer sensor nodes than other existing algorithms.</p> <p>strengths. Simulations show that the proposed algorithm can determine the actual source within two hops, 69%–80% of the time, when the diameter of the networks varies between 7 and 13. We implement this proposed two stage algorithm and we will propose a new algorithm which will improve the efficiency of finding the source to more than 80%.</p>
III.	Requirement Specification	<p>Purpose:</p> <p>The purpose of this Software Requirement Specification is to give an overview of the functional and non-functional requirements of our project titled “Finding the source of the information in large social network” where we address the problem of estimating the source of rumor in large scale social networks.</p> <p>Product Scope:</p> <p>Nowadays online social networks are good platform to spread the news or information. Studies demonstrate that information spreads much faster in social networks than any other communication media. This ease of information dissemination can also be used to spread rumors. So the purpose of the project is to find the source of the rumor thereby halting the spread of the rumor.</p> <p>The goal is to find the first person who spread the potential misleading information and it is beneficial for government officers to find the culprit</p>

Product Perspective:

Recently countries are facing cases like mob lynching, fights and kidnapping due to spread of wrong information or rumor through social media. There comes a need for finding the cause of rumor and stop it. This product is follow-on member of already existing product which is less efficient i.e. it finds node within few hops of actual source node. Besides it provides scope for improvement. So this is dependent product and will be the first release of the product.

Product Functions

Our software uses the fact that large-scale social networks are modular i.e. made from different small clusters having some weak relationships. We represent the social network in graphical form.

So the major function is to find the first node who initiated the spread of information. We propose a novel two-stage algorithm that uses the modularity of the social network to locate the source of the rumor with fewer sensor nodes than other existing algorithms.

We model the function in following stages:

- **Find the Most Likely Candidate Cluster (First Stage)** : In this stage we find the sub-graph or the cluster where the probability of finding the source is more.
- **Find the Source of Rumor (Second Stage)** : In this stage using the sub-graph of above stage we find the actual source.

Operating Environment

Hardware Requirements:

- 2.40 GHz processor
- 8 GB memory RAM
- 100 MB of disk space

Software Requirements:

- Linux operating system
- Programming language : Python 3.0
- NetworkX library for network analysis

Design and Implementation Constraints

- Input Dataset should contain deterministic and homogenous internode relationship strengths.
- The software will support Linux Ubuntu machines with 2.40 GHz CPU and 8 GB memory.
- Specific technologies used are NetworkX, python 3.0, matplotlib, etc
- Varying internode strengths may limit the performance which is major constraint.

Assumptions and Dependencies

- Input dataset is assumed to be in specific format which gives relationship between nodes and corresponding strengths of edges among them.
- Social networks are assumed to be modular which can be divided into clusters.
- Dependent libraries are : NetworkX, matplotlib, etc.
- Information spread is assumed to be just like epidemic spread (SI, SIS or SIR models).
- The user has basic knowledge of computer and command line.

Functional Requirements

System Features:

1. Estimating the source of a rumor on a dataset provided by Twitter

- Parsing the essential attributes from JSON dataset of twitter handles/accounts.
- Converting the dataset into a discrete graph.
- Finding the sensor nodes and clusters in the graph.
- Implementing a two-stage algorithm on the graph to predict the source of a given tweet.

2. Adding Machine Learning capabilities to the algorithm

- Improvising the existing algorithms to improve speed and accuracy using various ML techniques.
- Designing a training dataset containing a set of tweets and their source in the form of a graph.
- Creating a model that can be trained using the designed training dataset to predict the source of a tweet on a test dataset.

3. Documenting the improvised algorithm in the form of an IEEE research paper

Other Nonfunctional Requirements

1. Performance Requirements:

- **Scalability** : System should be able to handle large-scale networks. For e.g. handling datasets containing around thousands of node
- **Speed** : The application should be fast. It should not slow down exponentially with increase in the size of database. Search functionality should be fast to enable better user experience.
- **Reliability** : System should be reliable enough to enable great experience

2. Software Quality Attributes

- **Usability** : User interface should be simple and clear to be understood by any users.
- **Availability** : System should be available at all time. It should be ensured that there should be minimum no downtime to ensure better user experience.
- **Testability** : Software should be testable. A separate test environment should be set up where testers and Quality Assurance engineers can test the application for bugs and/or incomplete or missed requirements.
- **Maintainability** : The software should be developed in such a way that it is extensible. It should be easy to incorporate new features, requirements or accommodate a change in existing requirements.

References:

IEEE paper :

- Alireza Louni and K. P. Subalakshmi, "Who Spread That Rumor: Finding the Source of Information in Large Online Social Networks With Probabilistically Varying Internode Relationship Strengths" IEEE transactions on computational social systems, vol. 5, no. 2, June 2018.
- D. Shah and T. Zaman, "Detecting sources of computer viruses in networks: Theory and experiment," SIGMETRICS Perform. Eval. Rev., vol. 38, no. 1, pp. 203–214, Jun. 2010.
- K. Zhu and L. Ying, "Information source detection in the SIR model: A sample path based approach," in Proc. Inf. Theory Appl. Workshop,

		<p>Feb. 2013, pp. 1–9.</p> <ul style="list-style-type: none"> • C. H. Comin and L. da Fontoura Costa, “Identifying the starting point of a spreading process in complex networks,” Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 84, p. 056105, Nov. 2011.
--	--	---

Group ID	MIS #	Student Name
IT-7	111508075	Jayant Kalani
	111508045	Ganesh Landge
	111508005	Akash Patil
Industry Mentor(if any)	Name of External	Organization
-	-	-
Internal Mentor	Name of Internal mentor	Signature of Internal mentor
AK	Anishraj Khobragade	