The fake news graph analyzer: An open-source software for characterizing spreaders in large diffusion graphs

Amirhosein Bodaghi, Jonice Oliveira, Jonathan J.H. Zhu

PII: S2665-9638(21)00078-6
DOI: https://doi.org/10.1016/j.simpa.2021.100182
Reference: SIMPA 100182

To appear in: Software Impacts

Received date: 12 November 2021
Accepted date: 19 November 2021

Please cite this article as: A. Bodaghi, J. Oliveira and J.J.H. Zhu, The fake news graph analyzer: An open-source software for characterizing spreaders in large diffusion graphs, Software Impacts (2021), doi: https://doi.org/10.1016/j.simpa.2021.100182.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).
The Fake News Graph Analyzer: An open-source software for characterizing spreaders in large diffusion graphs

Amirhosein Bodaghi* 1, Jonice Oliveira2, Jonathan J.H. Zhu1

1 Department of Media and Communication, City University of Hong Kong, Hong Kong
2 Department of Computing Science, Centre of Mathematical and Natural Sciences – CCMN, Federal University of Rio de Janeiro, Rio de Janeiro, Brazil

* Corresponding author,
+852 5532-9851
abodaghi@cityu.edu.hk

Abstract
In the study of fake news spreading, it is essential to know how different types of spreaders differ in terms of their characteristics, interconnections, and cascading flow. The fake news graph analyzer (FNGA) is an open-source software that provides the required computations for such extended analyses on large graphs. Moreover, FNGA generates data for graph visualizations. Also, FNGA is designed to consider the spreading of both fake and true news simultaneously in the graph, leading to a variety of confrontational patterns. FNGA facilitates future research on fake news and the diffusion of any contagion within a graph of entities.

Keywords
Fake News; Social Network; Spreading Process; Graph Analysis; Twitter

Code metadata

| Nr. | Code metadata description | Please fill in this column |
|-----|--------------------------|---------------------------|
| C1  | Current code version     | v1                        |
| C2  | Permanent link to code/repository used for this code version | https://github.com/AmirhoseinBodaghi/P9 |
| C3  | Permanent link to Reproducible Capsule | https://codeocean.com/capsule/7614966/tree/v1 |
| C4  | Legal Code License       | MIT License               |
| C5  | Code versioning system used | GIT                      |
| C6  | Software code languages, tools, and services used | Python |
| C7  | Compilation requirements, operating environments & dependencies | Python packages: numpy, scipy, snap-stanford, xld |
| C8  | If available Link to developer documentation/manual | Not provided |
| C9  | Support email for questions | abodaghi@cityu.edu.hk    |
1. Introduction
Following the 2016 US Presidential Election, fake news has gained attention in academic scholarship [1]. The fast-growing literature on fake news has studied this phenomenon in various aspects in which investigations on user-based features play a significant role. The analysis based on user characteristics has roots in a variety of research contexts, from sports [2] to social media [3]. However, when it comes to spreading fake news or viral contagions, the underlying graph of users and the characteristics driven from it would play a major role in the analysis. Particularly when it comes to modeling the spreading process of fake news on social media, considering the graph-based characteristics of users appear to be a necessity [4,5].

When an external source, particularly a junk news website, publishes a fake news post for the first time, some Twitter users fetch and tweet it [6]. Then retweeters come along and diffuse the fake news. The exact process occurs for spreading the truth, which encounters that fake news. Therefore, the play’s four main characteristics are the tweeter of fake news, retweeter of fake news, tweeter of truth, and retweeter of truth. However, some tweeters earn lots of retweets, making them Super Spreader, while many other tweeters of the same content find no success to have their tweets retweeted by anyone and end up as Unwelcome Spreaders. A fake news graph of diffusion is constituted of ego networks of all these different types of spreaders. Each ego network itself is made of all followers and followings of a single spreader as its nodes while connected to the spreader by directed edges.

2. Description
The FNGA" is written in python and uses libraries such as Snap-Stanford [7], Numpy, and Scipy. The inputs for each fake news are: 1 - Dataset of Diffusion (DD), an Excel file containing all the tweets related to the fake news. For each tweet that occupies one row in the file, various information is provided for the user who published it and the tweet itself. This information for the user includes ID, #tweets, #followings, #followers, data and time of the account creation, language, and description sentence. While the information for the tweet include date and time of publishing, #favorite, type of tweet, which can be a retweet, quote, reply, or original, the tweet and user IDs of the referred tweet in case the tweet be a retweet, quote, or reply, frequency of tweet occurrence in the dataset, and finally the last feature is the nature of tweet which can be fake news (marked as ‘r’), truth (marked as ‘a’), or just questioning about the fake news (marked as ‘q’). 2 - Dataset of Graph (DG), a Graph file containing the nodes and edges of the fake news graph. Indeed, this file could be achieved by applying the Snap.py a Python interface of SNAP, which confers an agile and light environment for big graph analysis [7], to run a process of compression and conversion upon the raw list-type data of followers and followings for each spreader of the fake news, to attain the agile graph dataset. In this way, for each dataset, we acquired a directed network in which each node represents a user, and a link from user i to user j is established if i follows j. 3 - The ID of users in DG, which is brought into a Jason file. After reading the inputs, six analysis steps will be done on each fake news by the software. Step1: Extract the information of retweeters of the superspreaders' tweets at different circles of distance around the super spreaders. Indeed, first a set of categories based on the distance to the super spreader are defined, then for members of each category, the following characteristics are calculated: 1- ratio of followings to followers (FwFr), 2- ratio of mutual followings to followings in general (MFw), 3- number of followings who had spread the most popular tweet before, as the antecedent spreaders of the given member (NA), 4- the number of passed seconds from the time the super spreader had spread the most popular tweet until the time the given member retweeted this tweet (TDS), 5- the number of passed seconds from the time the first antecedent spreader had spread the most popular tweet until the time the given member retweeted this tweet (T DFA), 6- number of published tweets and retweets per year from the given member (NTPY), which is calculated based on the overall activities since the creation of the account, 7- number of antecedent spreaders for the opposite contagion i.e. users who have been a following of the given member and had spread the truth/fake news before the given member spread fake news/truth (NAO). The software at this step calculates the mean and standard deviance of the above characteristics for users in each category, and the correlations of those characteristics with the T DFA at each category are computed.

Step2: Extract the information of unwelcome spreaders, including calculating a set of characteristics such as FwFr, MFw, and NTPY. Step3: Generating the input files for graph visualization by Cytoscape. Step4: Extracting the information of interconnections between tweeters and retweeters of fake news and truth. First,

---

1 https://github.com/AmirhoseinBodaghi/P9
2 https://codeocean.com/capsule/7614966/tree/v1
the software divides users into four groups; Fake News Tweeters (FT), Fake News Retweeters (FR), Truth Tweeters (TT), and Truth Retweeters (TR). The members of these groups extract a series of information, including 1- The fraction of followings (for the given member) who are members of the other groups. 2- The fraction of whole members of each group that are the followings of the given member. 3- The fraction of followers (for the given member) who are members of the other groups. 4- The fraction of whole members of each group that are the followers of the given member. Eventually, we compute the mean and standard deviation of these variables for members of each category. Step5: Extracting the general information of each fake news graph, including the number of nodes and edges of different types, triangles, triads, connected components, and length of the effective diameter. Step6: Extracting the information of centralities for tweeters and retweeters of fake news and truth. These centralities are calculated at the singular level (for single nodes) and plural level (for all nodes of a category). The former includes centralities such as Page Rank, Closeness, Hubs, and Authorities score, while the latter encompasses Modularity and the ratio of intra- to inter-links for each of the groups i.e. FT, FR, TT, and TR. Each of these six steps provides a unique output file (a .text file) in which the results are presented.

3. Impact
The FNGA enables an extensive range of analysis over fake news graphs driven from social media. A summary of these analyses can be seen in Fig. 1.

The analyses conducted by this software in recent research [8] proved its ability to reveal new findings towards a better understanding of fake news diffusion on social media. Furthermore, findings of different categories’ centralities and interconnections of spreaders depict a practical perspective that can guide practitioners to build better strategies against fake news spreading on social media. Moreover, the investigations on collective behaviors such as the wave-like forms of diffusion and their associations with micro behaviors are still in their infancy [9-11]. In fact, research on the impact of cyberpsychology at the aggregate level remains open to yield more profound insights into the diffusion process of fake news in social media. This software may pave the way for such research by providing an extensive range of analyses.

Declaration of Competing Interest
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. No potential conflict of interest was reported by the authors.

Acknowledgements
The study was partially funded by City University of Hong Kong Centre for Communication Research (No. 9360120) and Hong Kong Institute of Data Science (No. 9360163).

References
[1] Tandoc, E., Jenkins, J., & Craft, S. (2019). Fake news as a critical incident in journalism. Journalism Practice. 13(6), 673–689.
[2] Bodaghi, A. (2016). A novel pervasive computing method to enhance efficiency of walking activity. Health and Technology. 6, 269–276.
[3] Zhang, X., & Ghorbani, A. (2019). An overview of online fake news: Characterization, detection, and discussion. Information Processing and Management. 57(2), 102025.

[4] Bodaghi, A. & Goliaei, S. (2018). A novel model for rumor spreading on social networks with considering the influence of dissenting opinions. Advances in Complex Systems. 21, 1850011. DOI: 10.1142/S021952591850011X.

[5] Bodaghi, A., Goliaei, S., & Salehi, M. (2019). The number of followings as an influential factor in rumor spreading. Applied Mathematics and Computation. 357, 167-184.

[6] Bondielli, A., & Marcelloni, F. (2019). A survey on fake news and rumor detection techniques. Information Sciences. 497, 38–55.

[7] Leskovec, J. & Sosic, R. (2016). Snap: A general-purpose network analysis and graph-mining library. ACM Transactions on Intelligent Systems and Technology. 8(1), 1.

[8] Bodaghi, A., & Oliveira, J. (2021). The Theater of Fake News Spreading, Who Plays Which Role? A Study on Real Graphs of Spreading on Twitter. Expert Systems with Applications. 2021, 116110. https://doi.org/10.1016/j.eswa.2021.116110.

[9] Bodaghi, A., & Oliveira, J. (2020). The Characteristics of Rumor Spreaders on Twitter: A Quantitative Analysis on Real Data. Computer Communications. 160, 674-687.

[10] Meel, P., & Vishwakarma, D.K. (2021). A temporal ensembling based semi-supervised ConvNet for the detection of fake news articles. Expert Systems with Applications. 177(1), 115002.

[11] Choudhary, A., & Arora, A. (2021). Linguistic feature based learning model for fake news detection and classification. Expert Systems with Applications. 169(1), 11417.
Highlights

- The software enables to characterize different types of spreaders in a fake news diffusion graph.
- The software provides various computations for an extended range of analyses on diffusion graphs at scales of hundreds of millions of nodes.
- The software is designed to consider the spreading of both fake and true news simultaneously in the graph, leading to various confrontational patterns.
- The software generates input data for graph visualizations.
The Fake News Graph Analyzer: An open-source software for characterizing spreaders in large diffusion graphs

Amirhosein Bodaghi\textsuperscript{1*}, Jonice Oliveira\textsuperscript{2}, Jonathan J.H. Zhu\textsuperscript{1}

\textsuperscript{1}Department of Media and Communication, City University of Hong Kong, Hong Kong
\textsuperscript{2}Department of Computing Science, Centre of Mathematical and Natural Sciences – CCMN, Federal University of Rio de Janeiro, Rio de Janeiro, Brazil

* Corresponding author.
+852 5532-9851
abodaghi@cityu.edu.hk

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.