UDC 004.021:004.78

Malicious Information Source Detection in Social Networks

V. V. Melnyk[1], I. V. Styopochkina[1]

[1]*National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute»,*
Institute of Physics and Technology

Abstract

The paper is devoted to investigation of the problem of malicious information source detection among the users of popular social network. The advantages and disadvantages of existing algorithms of rumor source detection were analyzed. The practice-oriented algorithm of malicious source detection have been proposed, it differs from the existing ones with additional verification procedure of the authorship. Appropriate software have been developed.

Keywords: social network, source, malicious information, social graph, algorithm

Introduction

The problem of malicious information distribution have become actual among users of popular social networks. As malicious information we can understand the following types of information:

- spam;
- informational messages, photos, publications that contain misinformation, or directed on implanting of enmity, agression and other harmful influences in social network users media;
- malware.

When protecting users from spam, there is a number of purely technical features that allow it to be filtered out. The spam protection is rather effective, as evidenced in paper [1]; also a large number of investigations is devoted to malware propagation prevention, for example [2]. However, the problem of control over malicious information influences in form of information messages (sometimes it is the result of organized malicious activity) still remains actual. The degree of harmfulness of the information block that is distributed by the social network and its origin source should be established by special services that deal with the security of the state and citizens.

The problem of malicious information source detection in social networks has common features with rumor source detection problem [3],[4] but is not identical to it. The question of rumors source detection in environments that can be represented in form of graph was studied in a number of papers, but the classical algorithms that were proposed for this problem (for example, in [2],[3]) did not deal with real social networks, and were mainly focused on the abstract graph-like structure or computer network. Modifications of these algorithms continue to be actively developed, but existing studies mainly focus on reducing the computational complexity of algorithms. Also in [5] is shown, that it is hard to separate real author of the rumor from the most influential users, that means there are no guarantee of exact identification of the source (which is acceptable for rumor tracking algorithms, but is not desirable for malicious information source detection). It should be noted that the basic methods of estimating the source of the rumors may not be applicable to arbitrary graphs, or be oriented primarily to the exchange of data in a computer networks rather than a social networks ([2],[4]). On the other hand, the analysis of social networking messages is reflected in the work [6], which focuses on the analysis of lexical characteristics of rumor messages of the social network. This is perspective way for exact detecting of the source. Taking into account mentioned problems, the analysis of the properties of classical algorithms from the point of view of their applicability in the real social network is an actual task that will eliminate the practically inappropriate algorithms. That is why an social network oriented algorithm, which is focused on unambiguous identification of the source of malicious information, using additional lexical verification of the results of the previous detection of the source of harmful information have been proposed on this paper. Also a problem of data gathering in social networks have been considered.

1. «Social network»

Social network is a structure that consists of large number of nodes (graph vertices) and social relations (edges), that are connections between nodes. Depending on network type, relations can be unilateral (for example, Twitter social network), or bilateral (for example, Facebook social network). So we can define such a structure: we have a set of users $U = \{u_1, u_2, u_3, \ldots, u_k\}$. Every user u_i, $i \in [1, k]$ has a profile p_k, that can contain attributes $A = a_1, a_2, a_3, \ldots, a_j$. As attributes can be considered: user identifier, name, age, gender, number of friends and etc. $S = \{(a_i, a_j)|i, j \in [1; k]\}$ – appropriate social relationships.

Depending on social network platform, user can connect with anybody who has a relationship with him, or use contact information to make a relation.

1.1. Popular social networks review

Because the most of social networks are similar by the principles of action, we have chosen the most popular as examples for consideration. The selected social networks (Facebook and Twitter) are different by idea and construction. So, for users security, every resource has its own security policy. For workability testing of algorithms of data searching it is necessary to access user information in social network. It can be a problem because of some reasons, for example Facebook security policy prohibits extraction and analysis of the personal attributes of profile, friend list, updates and messages without personal user agreement. The more accessible among popular social networks for testing the algorithms and solving the problems of this paper is Twitter.

Let us review existing tools for information gathering in Facebook and Twitter.

Facebook SDK include:

- Facebook SDK for JavaScript;
- Facebook SDK for PHP;
- Microsoft Windows SDK for Facebook;
- Java (Spring) from Spring Social;
- Django-Facebook.

Twitter SDK include:

- Twitter4J by @yusuke — a Twitter API library(Java >5, Android and GAE ready);
- TwitterJSClient by @BoyCook — Twitter client library written in Javascript and packaged as a node module;
- python-twitter maintained by @bear — this library provides a pure Python interface for the Twitter API;
- twitcurl by @swatkatsrants — Twitcurl is a C++ twitter API library based on cURL;
- codebird-php by @jublonet — a Twitter library in PHP.

1.2. Social network as a graph

We use social network model in form of oriented (for social networks like Twitter) or non-oriented (for social networks like Facebook) graph $G = (V, E)$, which consists of vertices (nodes) V and edges (arcs) E. In case of oriented graph, edges will be directed. The vertices are users of social network, and edges mean social relations. For non-oriented graph relationship means «friendship», and for oriented graph relationship means «tracking». Each arc is represented as ordered (for oriented graphs) or disordered (for non-oriented graphs) pair of two different edges. Thus the network is modeled as a graph.

2. Malicious information in messages

The definition of «malicious information» is rather ambiguous, since the content of this concept is not explicitly disclosed in the Ukrainian legislation. However, according to article 34 of Constitution of Ukraine, there are some restrictions on the right to free distribution of information. Namely: this right may be limited by law in the interests of state security or for public order maintaining, prevention of offenses such as social unrest or crime [3]. Hence, the main features of malicious information may be the following [4]:

- false or distorted information;
- unprofitable, compromising information for an object of attack;
- information and psychological impact aimed for stimulation of offenses;
- information bears the risk of causing damage for information attack object;
- information has hidden harmful direction.

Thus, it may be interpreted as malicious information:

- information denying public morals;
- information that harms person's honor and dignity (it may also be discriminatory);
- information that adversely affects the health of society.

Every active user of social network can freely share information blocks, links, video and photo. Information, which contains some details (including false ones) of interest to the broad mass, is widely distributed not only in the social network, but also in the mass media, which use social networks for latest news and ideas. Thus, malicious rumors or misinformation can quickly spread through existing social networks and cause harmful influence on other people and whole society. The importance of this work is conditioned by the need for timely detection of sources that regularly disseminate harmful information (informational messages) in social networks, for their neutralization or close observation of their activity.

Hence, malicious information psychologically harms the interests of a person, his rights and freedoms. However, the procedure for establishing the fact of the harm and the criteria for the degree of harm should be performed by experts of relevant security services of the society.

3. Approaches for problem solution

3.1. An approach based on measurements collected from a small number of observers

The basis of the approach is a non-oriented graph $G = (V, E)$, where set V – set of vertices, the number of vertices is N, E – set of edges. Graph G is known. Any node $s^* \in V$ – diffusion initiator. Diffusion modeling: in some time t each vertex u takes one from two possible states – infected or healthy. Note $F(u)$ – neighbors u. In moment t_u node u gets information firstly from one of neighbors s. Then u will redirect information to neighbors $v \in F(u)$ with $t_u + \theta(uv)$, where $\theta(uv)$ – time delay between nodes. Set $\{\theta(uv)\}$ – has free common distribution. Let set $O = \{o_k\}$, $k \in [1; K]$ – set of observers, the localization of observers in graph is known. Each observer gets information about arrival time and author of message. If $t_{v,o}$ – absolute time of message receiving from node v to node o, then the set of observations consists of the number of vectors $T = \{(o, v, t_{v,o})\}$. Thus, applying maximum likelihood criterion, we obtain that estimation of maximal probability of localization criterion equals to:

$$q(T) = \arg\max_{s\in G} P(T|s^* = s) =$$
$$= \arg\max_{s\in G} \sum_{W_s} P(\Pi_s|s^* = s)\times$$
$$\times \int ... \int g(\theta_1, ..., \theta_L, T, \Pi_s, s) d\theta_1, ..., \theta_L$$

where W_s – all possible paths between source and observer in graph, $\theta_1, \ldots, \theta_L$ – all possible delays, g – deterministic function, which depends on common distribution of delays [3].

Disadvantages of the algorithm:

- estimation has a complexity that increases exponentially with the number of nodes in the graph and therefore is difficult to resolve;
- there are difficulties with control of messaging process between nodes in real social networks;
- algorithm rather oriented on computer network than on social network.

Conclusion: because of combinatorial origin of estimation expression algorithm is practically inappropriate for our problem.

3.2. Centrality indicator algorithm

Let there be an infected source among a set of suspicious nodes. The problem is as follows: to locate this node among all nodes of a general non-oriented graph, taking into account the SI model of information distribution (that is, each infected node always tries to infect another).

We have information about nodes and relations between them on the input of algorithm. Let us fix node $s^* \in S$ and in some time moment we observe infected set of n nodes, which create connected graph. As infected node we mean that one, which accepted malicious information (distributed in its own blog etc). The aim is to build an estimate for identification of node $\hat{s}$ as estimate of rumor source s^*. Using Bayesian rule, maximal a posteriori estimate of node s^*, which maximizes probability of true detection, equals to:

$$\hat{s} \in \arg\max_{s\in\{S\bigcap G\}} P_G(s|G) =$$
$$= \arg\max_{s\in\{S\bigcap G\}} \frac{P_G(G|s) \times P_s(s)}{P_G(G)} =$$
$$= \arg\max_{s\in\{S\bigcap G\}} P_G(G|s),$$

where $P(G|s)$ is a probability of observation of graph G in condition that s is infected node [4]. As the calculation of $P(G|s)$ is too complex, it is possible to use another conception of centrality for needed estimate. Thus we obtain:

$$\hat{s} \in \arg\max_{s\in\{S\bigcap G\}} P_G(G|s) = \arg\max_{s\in\{S\bigcap G\}} R(s, G),$$

where $R(s, G)$ – centrality indicator of s in graph G.

«Rumor Centrality». For common non-oriented graph it is proposed the following solution of the problem of rumor source detection [4]:

$$\hat{s} \in \arg\max_{s\in\{S\bigcap G\}} P_G(G|s) = \arg\max_{s\in\{S\bigcap G\}} R(s, G),$$

where $R(s, G) = n! \prod_{u\in G} \frac{1}{|T_u^s|}$,
T_u^s – is subtree, rooted in node u with s as source in G,
$|T_u^s|$ – power of the subtree.

Whereas the information propagates with a minimum distance between neighbors of the source, therefore, it is relevant to construct the trees for the general graph using the technology «search in width».

Disadvantages:

- algorithm cannot be applied to oriented graph and has exponential complexity;
- the movement between nodes is realized as «search in width», that increases complexity of algorithm.

Another modification of this algorithm is named «Betweenness Centrality». It analyses the number of shortest paths through the node. Advantages:complexity of «Betweenness Centrality» is $O(nm)$, where n is a number of vertices, and m is a number of edges in graph.

Eigenvector centrality. Centrality of eigenvector is measure of node impact in the network. The algorithm assigns relative estimates for all nodes in network, basing on the rule that connection to highly popular nodes provides more high rating than connection to low popular nodes. For given $G = (V, E)$ with number of vertices $|V|$ let A be the matrix of adjacency. The centrality value equation is: $Ax = \lambda x$ where λ is resulting vector of eigenvalues.
Advantages: algorithm can be used for the problem of malicious source detection if infected graph is not very large, because its complexity is $O(log(n^3))$. The movement between nodes is realised as random wandering.

PageRank. The main idea of the Page Rank technology is: more important web-pages have to be estimated with higher score. When random user visits pages in any time, the pages with higher impact are visited more often. This technology can be used for social networks. Namely, the pages will be users, and edges of the link graph in PageRank will be user relations. The result will be a vector of user importance. If we give the infected graph to the input of the PageRank, we will obtain an estimate of malicious information source. PageRank is a kind of eigenvector centrality algorithm.
Advantages: algorithm is suitable for using in large networks, its complexity estimate is $O(log(n))$. The movement between nodes is realized as random wandering.
Thus, we can see that some centrality algorithms are suitable for the problem of detection of malicious information source in social network.

4. Proposed algorithm

Let us show the algorithm scheme for oriented graph (fig. 1). And for non-oriented graph algorithm differs, as is shown on fig. 2. Taking into account advantages and disadvantages of reviewed algorithms of rumor source detection, it have been recommended to use one of centrality indicator algorithms (unless Rumor centrality algorithm for oriented graph) as possible estimation algorithms for malicious information source. This algorithm have been implemented in structure of general algorithm, as it is shown on fig. 3 and fig. 2. The general algorithm includes branches of «infected» social graph building, estimation of information source stage and refinement procedure.
Note, that proposed algorithm can be applied as for Twitter-like networks, so for Facebook-like networks.

5. Experiment results

For implementation of constructed algorithm, which is based on the centrality indicator approach, it is appropriate to use Twitter API. With official methods of Twitter API we are able to collect detailed information about social graph. , In particular, we can collect: user identifier and relations (subscriber list, friend list), which are needed for algorithm.

However, there are some restrictions, namely: methods of Twitter API return maximum 5000 users per hour for one request at the present. It means, for example, if user have more than 5000 of subscribers, method will return only 5000 of the most popular users. Other restriction of API methods is number of requests per hour. Currently permitted number of requests per hour reaches 150, that essentially increases the time of social network analysis.

For social graph visualization we have chosen Gephi software. Gephi is an interactive platform for investigation all types of networks, dynamical and hierarchical graphs. So it can be applied for social networks. Also Gephi realizes PageRank and Eigenvector centrality algorithms. For other centrality algorithms and information gathering an applied software have been developed.

An example of social Twitter subgraph is given on fig. 3. At the practice, sometimes we have no need to build full version of social graph, because malicious messages can be propagated inside some social group, that is conforming to some subgraph of the social network. And the problem is to find the source among the users of this social subgraph. At the beginning, we have to notice any user, which accepted malicious message, and then we start to seek who could be the source of this message according to the algorithm. Also we build the graph of user relations as the parallel task. An experiment have been performed on the real Twitter network data, so user identifiers we show in the table of results (table 1,table 2) with several low positions commented by * for confidentiality. The social graph is shown on the fig. 4 for the oriented graph, and fig. 5 for non-oriented graph (built using only friendship relations).

After algorithms applying we can see that in case of oriented graph (table 1) results of PageRank, Eigenvector Centrality and Betweenness centrality coincide (Rumor Centrality is not applicable in this case): the most probable source is node with $8149430*$ Id (first position), also rather probable is node with $22248882*$ Id (second position).

For the case of non-oriented graph (table 2) we can see that source estimates of Eigenvector Centrality and PageRank coincide (Id $2224882*$), and estimates of Betweenness and Rumor Centrality coincide too (Id $819430*$). That can be explained by relativeness of these algorithms. Note, that vertices $2224882*$ and $819430*$ are directly related (fig. 4),(fig. 5). As in example of oriented graph and in the example of non-oriented graph the nodes $22248882*$ and $8149430*$ are suspicious as possible source of malicious message.

Then we have to perform an additional check of user, which was identified as the source according to the algorithms. The methods used in consist of the additional check contain analysis of virtual language style of messages. If we have detected the suspected user, we can analyze usual messages of the user and check the virtual style similarity for these messages and malicious sample. The algorithm of the additional check for each suspected user consists of the stages:

1) Calculate typical attributes of user messages: message length, hashtag presence (yes/no), hashtag fingerprints, web link presence (yes/no), link fingerprints, fuzzy hash of the message.
2) Construct the vectors of message attributes for recent messages in user profile.
3) Calculate the same attributes for malicious sample and construct the vector.
4) Submit all the vectors of the suspected users in series to the input of classification method (for example, with the use of neural network) to form the classes (each class consists of the messages of one suspected user). This is the stage of classifier learning.
5) Submit malicious sample vector to the input of classifier, identify its class. This is the stage of classification.
6) Make a conclusion about the authorship for the malicious sample.

Another way of the additional check consists in the following:

1) Collect a grammar of the suspected user messages.
2) Calculate frequencies of used words in texts that are exactly belong to the suspected author with the use of natural language processing tools.
3) Calculate $TF - IDF$ indices (term frequency - inverse document frequency) for the malicious sample and other messages of the user. Note, that for calculation according to this method we need to have user language corpus with typical frequencies of used words, where TF is frequency of word w in text d, and IDF is logarithm of reverse frequency of word w in language corpus C.

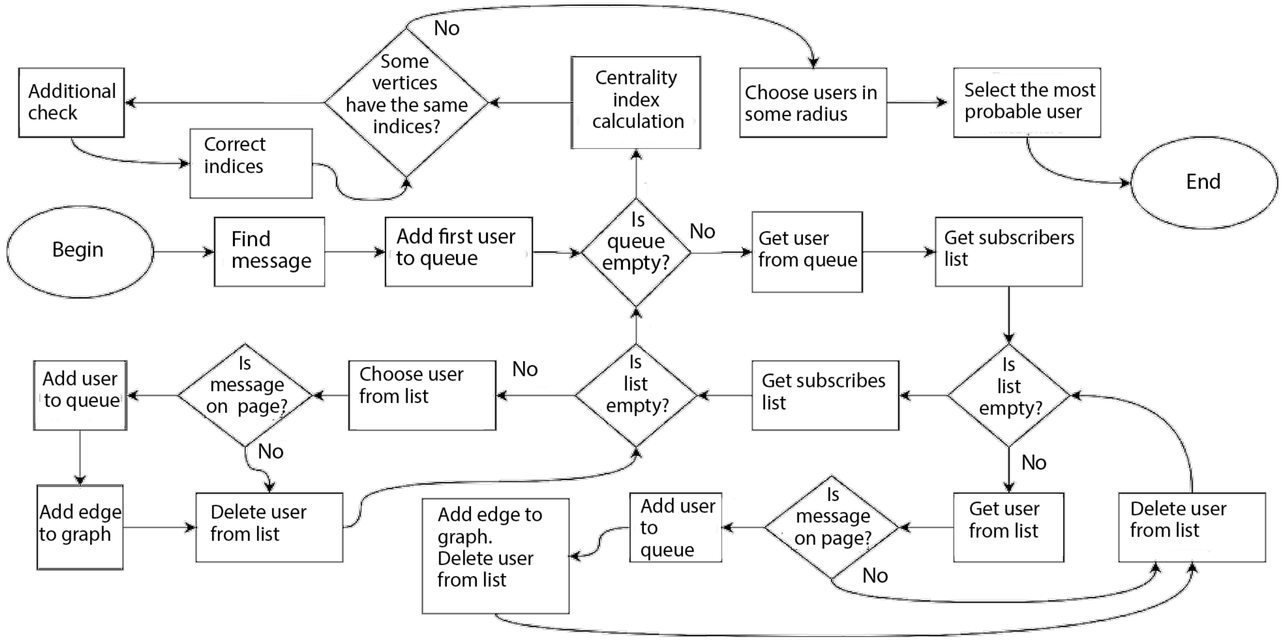

Fig. 1. Algoritm scheme for oriented graph

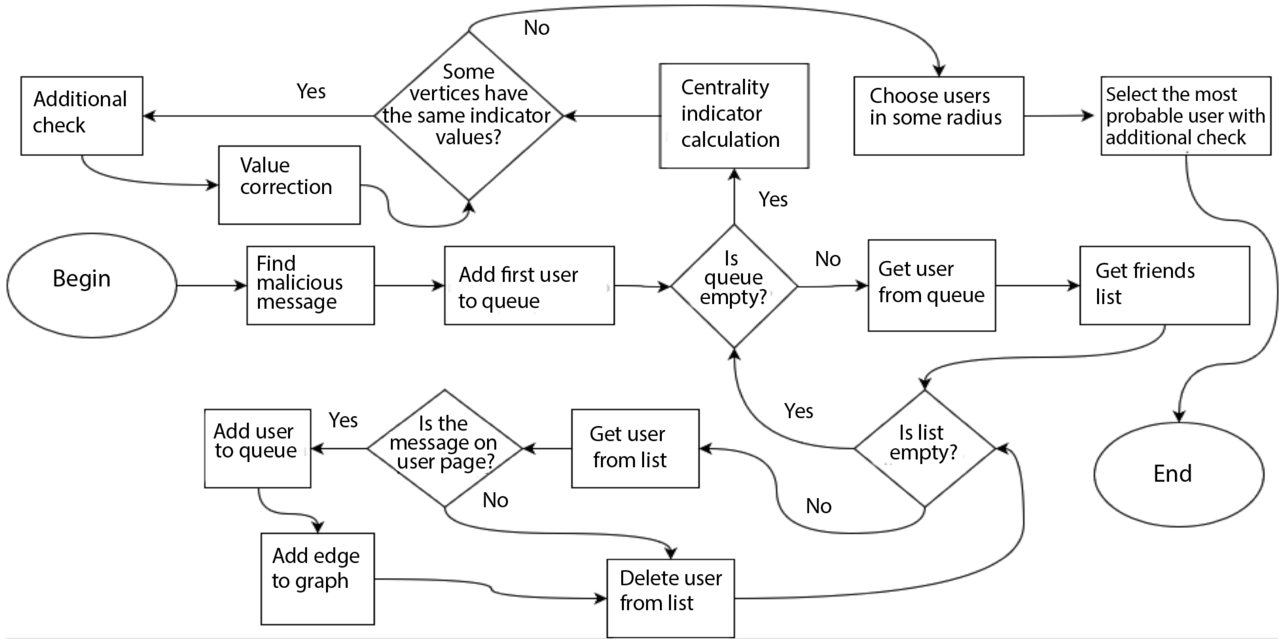

Fig. 2. Algoritm scheme for non-oriented graph

Table 1. Centrality algorithms results for oriented graph

ID	Eigenvector Centrality	ID	PageRank	ID	Betweenness Centrality
8129230∗	1.0	8149430∗	0.08059	8149430∗	0.03221
2224882∗	0.99982	2224882∗	0.06994	2224882∗	0.02271
3804207∗	0.99693	3804207∗	0.04689	2613887∗	0.01889
2644527∗	0.995246	6309956∗	0.03520	2365665∗	0.01344

Table 2. Centrality algorithms results for non-oriented graph

ID	Rumor	ID	Eigen	ID	PageRank	ID	Beetweenness
8129230∗	0.39	2224882∗	1.0	2224882∗	0.068	8149430∗	0.828
9680543∗	0.39	6309956∗	0.67	6309956∗	0.064	9680543∗	0.539
2224882∗	0.069	8149430∗	0.49	2613887∗	0.055	2224882∗	0.27
2613887∗	0.054	2613887∗	0.47	8149430∗	0.033	2613887∗	0.16

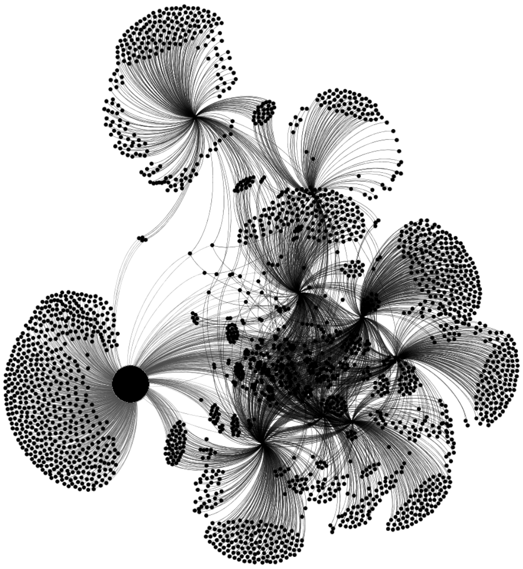

Fig. 3. Social subgraph

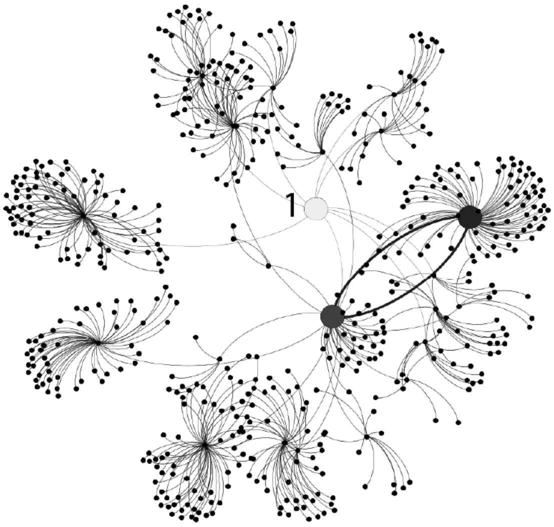

Fig. 4. Oriented subgraph of «infected» users. 1 - start node; probable sources of malicious information are related with bold arcs

.

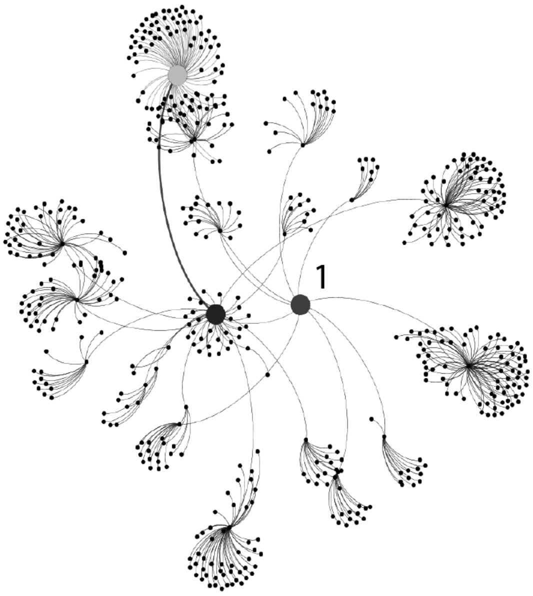

Fig. 5. Non-oriented subgraph of «infected» users. 1 - start node; probable sources of malicious information are related with bold arc

.

4) Use $TF - IDF$ indices as measure of closeness for malicious sample and other texts of suspicious users.

Note that sometimes it is enough to analyze technical attributes: average message length, typical message publication time, hashtag contents etc.

After using the additional check with the use of word frequencies technique we have selected the author of the message (node 8149430∗ in example before).

Additional check gives a possibility to make a decision about malicious information authorship, when algorithms have detected the several different users as possible source. We have to use additional check in case of concurrent results of centrality indicator algorithms also to ensure it, because of probabilistic nature of estimates.

Conclusion

On the base of analysis of existing approaches to the rumor source detection in social networks, it has been established that not all algorithms for detecting a rumor source can be applied in practice to malicious information source detection. The reasons are: computational complexity of the algorithms for calculating relevant estimates, as well as the complexity or impossibility of obtaining the source data for their successful work. The most effective are the algorithms based on the centrality indicator. However, the estimates of the algorithm have probabilistic nature and do not give 100% of guarantee of a true source detection. Therefore it is necessary to take into account and check the vertices with close to the maximum centrality values, using additional checks of lexical characteristics and virtual style. The appropriate algorithm was proposed, practical experiments have demonstrated its workability. As the «malicious» messages the benign information blocks were used, and proposed algorithm have detected its authors exactly. Basing on the received results we can confirm, that the higher the popularity of the user the most effective malicious influence he can provide.

References

[1] Kabakus Abdullah Talha, Kara Resul, "Survey of Spam Detection Methods on Twitter", *T. :International Journal of Advanced Computer Science and Applications,* pp. 29–38, 2017.

[2] D. Shah, T. Zaman, *Detecting sources of computer viruses in networks: theory and experiment,* Sigmetrics, (38), pp. 203–214, 2010.

[3] P.C. Pinto, P. Thiran, M. Vetterli,"Locating the Source of Diffusion in Large-Scale Networks", *Physical Review Letters,* Available: http://www.pedropinto.org.s3.amazonaws.com/publications/locating_source_diffusion_networks.pdf. Accessed on: 2012.

[4] W. Dong, W. Zhang, W. Tan Chee,"Rooting out the Rumor Culprit from Suspects", *IEEE International Symposium on Information Theory,* Available: http://www.cs.cityu.edu.hk/cheewtan/ DongZhangTan_ISIT2013.pdf. Accessed on: 2013.

[5] D. Król, K. Wiśniewska, "On Rumor Source Detection and Its Experimental Verification on Twitter", *Intelligent Information and Database Systems,* (vol. 10191), pp. 110–119, 2017.

[6] S. Vosoughi, "Automatic detection and verification of rumors on Twitter", *Massachusetts Institute of Technology,* Available: http://lsm.media.mit.edu/papers/Soroush_Vosoughi_PHD_thesis.pdf. Accessed on: 2015.

Malicious information source detection in social networks

Vadym Melnyk, Iryna Styopochkina

The existing algorithms of rumor source detection in networks have been analyzed, the most appropriate ones for detecting of malicious information source in social network were defined. It have been revealed that existing algorithms do not guarantee accurate detection of the source or are not oriented on practical implementation. Considering this, the practice-oriented algorithm for social networks have been proposed. It includes social graph construction, source detection using centrality indicator approach, and refinement procedure using lexical and virtual style features of malicious message. Software for data gathering and algorithm testing have been developed, practical experiments with Twitter users have shown the workability of the algorithm.

Keywords: social network, source, malicious information, social graph, algorithm