Predicting the Robustness of Large Real-World Social Networks Using a Machine Learning Model

Hindawi
Complexity
Volume 2022, Article ID 3616163, 16 pages
https://doi.org/10.1155/2022/3616163

Research Article

Predicting the Robustness of Large Real-World Social Networks Using a Machine Learning Model

Ngoc-Kim-Khanh Nguyen,[1] **Quang Nguyen**,[2,3,4] **Hai-Ha Pham**,[5] **Thi-Trang Le**,[4] **Tuan-Minh Nguyen**,[4] **Davide Cassi**,[6,7] **Francesco Scotognella**,[8,9] **Roberto Alfierif**,[7] **and Michele Bellingeri**[6,7,8]

[1]*Faculty of Basic Science, Van Lang University, Ho Chi Minh, Vietnam*
[2]*Institute of Fundamental and Applied Sciences, Duy Tan University, Ho Chi Minh 700000, Vietnam*
[3]*Faculty of Natural Sciences, Duy Tan University, Da Nang 550000, Vietnam*
[4]*John von Neumann Institute, Vietnam National University Ho Chi Minh City, Ho Chi Minh, Vietnam*
[5]*Vietnam National University, International University, Department of Mathematics, Thu Duc, Ho Chi Minh, Vietnam*
[6]*Dipartimento di Scienze Matematiche, Fisiche e Informatiche, Università di Parma, Parco Area Delle Scienze 7/A 43124, Parma, Italy*
[7]*INFN, Gruppo Collegato di Parma, I-43124 Parma, Italy*
[8]*Dipartimento di Fisica, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano, Italy*
[9]*Center for Nano Science and Technology PoliMi, Istituto Italiano di Tecnologia, Via Giovanni Pascoli 70/3, 20133 Milan, Italy*

Correspondence should be addressed to Quang Nguyen; nguyenquang29@duytan.edu.vn

Received 30 June 2022; Revised 24 September 2022; Accepted 3 October 2022; Published 9 November 2022

Academic Editor: Andrea Murari

Computing the robustness of a network, i.e., the capacity of a network holding its main functionality when a proportion of its nodes/edges are damaged, is useful in many real applications. The Monte Carlo numerical simulation is the commonly used method to compute network robustness. However, it has a very high computational cost, especially for large networks. Here, we propose a methodology such that the robustness of large real-world social networks can be predicted using machine learning models, which are pretrained using existing datasets. We demonstrate this approach by simulating two effective node attack strategies, i.e., the recalculated degree (RD) and initial betweenness (IB) node attack strategies, and predicting network robustness by using two machine learning models, multiple linear regression (MLR) and the random forest (RF) algorithm. We use the classic network robustness metric R as a model response and 8 network structural indicators (NSI) as predictor variables and trained over a large dataset of 48 real-world social networks, whose maximum number of nodes is 265,000. We found that the RF model can predict network robustness with a mean squared error (RMSE) of 0.03 and is 30% better than the MLR model. Among the results, we found that the RD strategy has more efficacy than IB for attacking real-world social networks. Furthermore, MLR indicates that the most important factors to predict network robustness are the scale-free exponent α and the average node degree $<k>$. On the contrary, the RF indicates that degree assortativity a, the global closeness, and the average node degree $<k>$ are the most important factors. This study shows that machine learning models can be a promising way to infer social network robustness.

1. Introduction

The study of the social network from a complexity science perspective has attracted much interest recently [1]. Especially, the study of dynamic processes that take place in these complex networks can have various applications. For example, the study of network robustness, i.e., "network robustness" is the capacity of a network to hold its functionality when a proportion of nodes/edges are removed, can help attack a network efficiently, or inversely

design a more robust network structure in practice [2–7]. On the other hand, the study of epidemic processes that take place in the network can be used to spread the news [8–12], optimize vaccination strategy [13–15], or define a better social-distancing rule [16–19].

Besides a few simple model networks where analytical models can be developed [20–24], most of the studies rely on computer simulations. For example, for the study of the network's robustness, node/edge removal Monte-Carlo simulations are usually employed. In such a process, nodes/edges are sequentially removed from the network using computer simulations. A "robustness" metric is then recorded at each step of the removal process. The most commonly used robustness metric is the largest connected component (LCC) of the remaining network [25].

The way nodes/edges are selected to be removed is called the removal strategy or attack strategy. One can classify attack strategies into two types, initial and recalculated attack strategies. For an initial attack strategy, nodes/edges are removed according to a node/edge ranking that is computed ahead of the removal simulation. In contrast for a recalculated attack strategy, the ranking is updated after each node/edge removal [4].

For node removal attack strategies, the node ranking is usually computed using node centrality measures such as degree [26, 27], closeness [4], and betweenness [7, 30]. It was found that for social networks, the recalculated betweenness node attack strategy (RB) is, on average, the most effective node attack strategy to dismantle the network [2, 7, 28, 29]. Other effective strategies are the recalculated degree (RD) and the initial betweenness (IB) [7, 28, 30].

Because of the sequential nature of the removal process, the node removal simulation is computationally costly, especially for recalculated strategies. For example, a simulation using an RD attack strategy has a time complexity of $O(N \times E)$, where N is the number of nodes and E is the number of edges of the network. The reason is that the node removal process has an N step, and at each step, a degree ranking is computed taking a time that scales with E. However, for RB, the computation of the whole network's betweenness is known to be very computationally costly, due to the definition of the network's node betweenness [31, 32]. The most efficient known algorithm for calculating network betweenness is the Brandes algorithm [33], which has a time complexity of O $(N \times E)$. In consequence, the whole node removal process using IB and RB attack strategies can have a time complexity of O $(N \times E)$ and O $(N^2 \times \mathrm{E})$, respectively. Although the IB attack strategy has the same time complexity as the RD attack strategy, the RB's time complexity is much higher. For illustration, in Figure 1, we present the total simulation time tIB and tRD for the corresponding attack strategies IB and RD, respectively, for all our studying social networks (48 networks see Section 2). In addition, we present the total simulation time tRB for the attack strategy RB for 4 networks (insert graph) as an example, as a function of the product $N \times E$. We found a good linear relationship between t_{IB} and t_{RD} and $N \times E$ for all networks as expected, and tRB is about two orders of magnitude higher than t_{IB} and t_{RD}, for networks of equal $N \times E$.

The simulation time can become an issue for the cases of social networks because their size can be extremely large. In fact, to our knowledge, most studies of dynamic processes on social networks that use an RB attack strategy only consider small-size real-world social networks of less than 100,000 nodes [7, 28, 30]. For very large social networks, the RB node attack strategy can take an unrealistic amount of time. Therefore, RB is not suitable for large social networks for an average computer station. One possibility is to use the alternative betweenness-based attack strategy with only one betweenness calculation, namely, the initial betweenness attack strategy IB, together with other recalculated strategies that use another node centrality metric that is less computationally costly. In consequence, in this work, we consider two candidate attack strategies for breaking large real-world social networks, IB and RD attack strategies. Besides the comparative study between different network node attack strategies, other works focused on the relationship between network robustness and network structural indicators (NSIs). Iyer et al. [4] studied network robustness as a function of the node clustering coefficient (or node transitivity). The research on model networks with tunable clustering coefficients demonstrates that networks with higher clustering coefficients are more robust, with the most important effect for the node degree and node betweenness attack [4]. Nguyen and Trang [34] studied Facebook social networks and found that those networks with higher modularity Q have lower robustness to node removal. The modularity indicator Q introduced by Newman and Girvan [35] measures how well a network breaks into communities, (i.e., a community or module in a network is a well-connected group of nodes that have sparser connections with nodes outside the group). In [29], the authors empirically analyzed how the modularity of scale-free models and real-world social networks affects their robustness and the relative efficacy of different node attack strategies. The abovementioned studies analyzed the relationship between network robustness and a single NSI.

On the other hand, machine learning (ML) is a technique that has seen a huge breakthrough in the last decade, beating state-of-the-art results in many prediction applications [36]. It initially solved technical problems in computer vision and natural language processing [37–39] and then expanded into many other fields such as health care, finance, manufacturing, energy, and environment. The key characteristic of an ML model is the ability to intelligently learn nonlinear relationships between the input and output without explicitly knowing them.

In this work, given such a complex relationship between network robustness and NSIs, we adopted a method from machine learning in order to learn such a complexity. Our main contribution is the application of the ML model to predict real-world social network robustness with acceptable errors. We develop ML models to predict network robustness under two main attack strategies, the IB and RD attack strategies, independently. We also implemented three popular ML models, single-variable linear regression, multiple-variable linear regression, and random forest models. Our results demonstrate that a data-driven method

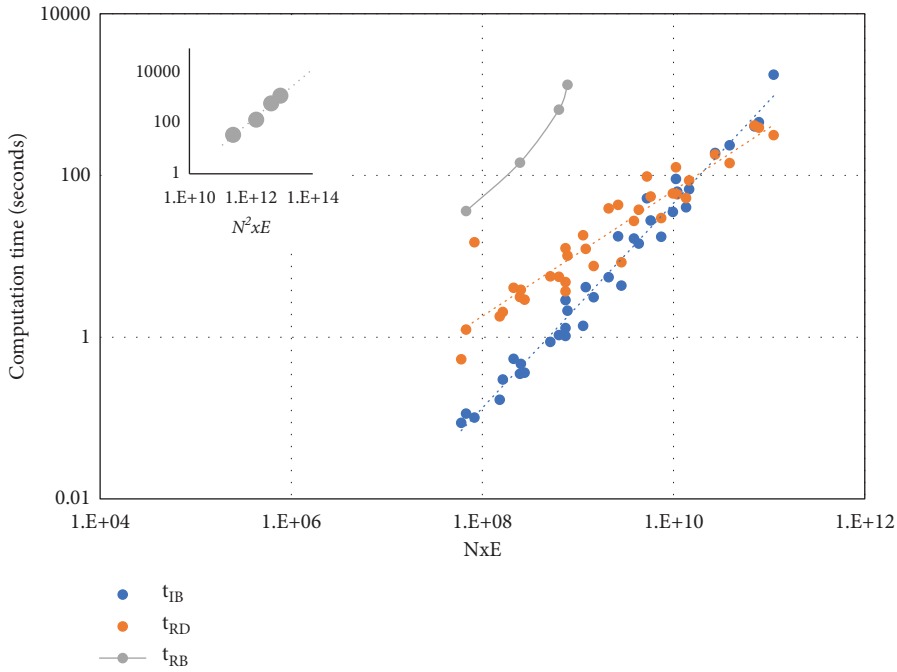

Figure 1: Computation time of a complete Monte Carlo network node attack simulation for all studied real-world social networks (using initial betweenness (IB) and recalculated degree (RD) attack strategy) and for 4 networks (using recalculated betweenness strategy (RB)) as a function of the product $N \times E$ (node number (N) edge number (E). We found that tIB and tRD scale approximately linearly with respect to the product $N \times E$, while tRB scales linearly with respect to the product $N^2 \times E$ (insert graph). From this result, we can estimate that the RB simulation time for the largest networks in our dataset will take more than 50 days using the same hardware.

such as ML can be an efficient way to study the network's complexity.

Our work comprises three steps: (1) collect a real-world network dataset and compute NSIs; (2) run Monte Carlo node attack simulations to estimate network robustness; (3) build and evaluate a model that predicts network robustness from their NSIs. The paper is organized as follows: in Section 2, we describe our dataset of 48 real-world social networks. In Section 3, we describe the network robustness Monte Carlo simulation method and three ML models for predicting the network robustness, i.e., simple and multiple linear regression (SLR and MLR, respectively) and random forest (RF) model. Section 4 presents the main results, and finally, we discuss and conclude in Section 5.

2. Real-World Social Network Datasets and Robustness Estimation

Real-world social networks are downloaded from two sources: the Stanford Large Network Dataset Collection (https://snap.stanford.edu/data/) and the Network Repository social networks (https://networkrepository.com/soc.php). We select 48 social networks with a node number (N) ranging over five orders of magnitude. The smallest network is the "Twitch user-user network of gamers who stream in Portugal" having $N = 1{,}914$, and the largest network is the "e-mail network from an EU research institution" with $N = 265{,}216$. However, the network with the largest number of edges (E) is the "BlogCatalog social blog" with $E = 4{,}186{,}390$. The social networks used in this study are unweighted (i.e., we do not take into account edge weights) and undirected (we do not consider edge directionality).

Table 1 summarizes 48 real-world social networks and their NSIs. Besides N and E, we also compute the following NSIs:

(i) Network density <k> is the average node degree, i.e., the average number of edges per node.

(ii) Fitted scaled-free exponent (α): we assume that all social network degree distributions follow a power law of $P(k) \sim k{-}\alpha$ where k is the node degree. The power exponent value α is fitted using the ordinary least squared method. From this fitting, we also extract the fitting variance of α, denoted by α^2.

(iii) Assortativity (a): the assortativity coefficient is a Pearson correlation coefficient of the degree between pairs of linked nodes [40], which varies between −1 and 1. A positive value of a indicates a preferential connection between nodes of a similar degree, while negative values indicate that nodes of different degree have more change to connect.

(iv) Modularity (Q): The modularity indicator Q calculates how a network can be partitioned into subnetworks (modules or communities):

$$Q = \frac{1}{2E}\sum_{i,j}\left(a_{ij} - \frac{k_i k_j}{2E}\right)\delta\left(c_i, c_j\right), \quad (1)$$

Table 1: Structural statistics of real-world social networks: node (N), edge (E), average node degree $<k>$, fitted power-law exponent α, the fitting variance of the power law exponent α^2, assortativity coefficient a, modularity Q, global clustering coefficient C, and average node closeness Cl.

Nb	Network description	Shortname	N	E	$<k>$	α	α^2	a	Q	C	Cl
1	Blue verified Facebook page networks of artist category	Artist	50,515	819,306	32.4	1.937	5.437	0.002	0.177	0.053	0.275
2	Blue verified Facebook page networks of athlete category	Athlete	13,868	86,859	12.5	2.130	4.778	0.005	0.547	0.129	0.237
3	Blue verified Facebook page networks of company category	Company	14,115	52,311	7.4	1.995	4.191	0.022	0.632	0.153	0.193
4	Blue verified Facebook page networks of government category	Government	7,059	89,456	25.3	1.829	4.401	0.004	0.478	0.224	0.270
5	Blue verified Facebook page networks of new site category	new_sites	27,919	206,260	14.8	2.097	5.082	0.009	0.509	0.114	0.233
6	Blue verified Facebook page networks of politician category	Politician	5,910	41,730	14.1	2.058	4.474	0.005	0.660	0.301	0.219
7	Blue verified Facebook page networks of public figure category	public_figure	11,567	67,115	11.6	1.841	4.212	0.009	0.441	0.167	0.221
8	Blue verified Facebook page networks of tV show category	Tvshow	3,894	17,263	8.9	1.622	3.279	0.037	0.770	0.591	0.166
9	Citation NW of arXiv High Energy Physics (phenomenology) paper	Cit-HepPh	34,548	421,579	24.4	2.528	6.310	0.003	0.472	0.146	0.237
10	Citation NW of arXiv High Energy Physics (theory) paper	Cit-HepTh	27,772	352,808	25.4	1.916	5.006	0.006	0.424	0.120	0.000
11	Collaboration network of arXiv astro physics	CA-AstroPh	18,774	396,161	42.2	2.174	6.021	0.050	0.412	0.316	0.000
12	Collaboration network of arxiv condensed matter	CA-CondMat	23,135	186,937	16.2	2.584	6.235	0.074	0.649	0.258	0.245
13	Collaboration network of arxiv general relativity	CA-GrQc	5,244	28,981	11.1	2.290	4.895	0.192	0.781	0.611	0.000
14	Collaboration network of arxiv high energy physics	CA-HepPh	12,010	237,011	39.5	1.407	4.013	0.103	0.383	0.657	0.000
15	Collaboration network of arxiv high energy physics theory	CA-HepTh	9,879	51,972	10.5	3.306	6.907	0.080	0.708	0.272	0.000
16	Deezer's users friendship networks from Croatia	deezer_HR	54,575	498,203	18.3	3.461	7.954	0.005	0.525	0.115	0.224
17	Deezer's users friendship networks from Hungary	deezer_HU	47,540	222,888	9.4	4.435	8.525	0.008	0.580	0.093	0.189
18	Deezer's users friendship networks from Romania	deezer_RO	41,775	125,827	6.0	3.392	6.402	0.008	0.682	0.075	0.160
19	E-mail communication network from enron	Email-enron	36,694	367,663	20.0	1.446	4.213	0.036	0.333	0.085	0.307
20	E-mail network from a EU research institution	Email-EuAll	265,216	420,046	3.2	0.646	1.901	−0.039	0.047	0.007	0.000
21	Follower relationships network of European users from deezer	deezer_Europe	28,283	92,753	6.6	2.981	5.972	0.011	0.603	0.096	0.159
22	Network of trusting consumers from the review site Epinions.com	Soc-Epinions1	75,881	508,838	13.4	1.512	4.289	0.001	0.247	0.082	0.237
23	Page-page network of verified facebook sites	musae_facebook	22,472	171,003	15.2	2.029	4.945	0.011	0.630	0.232	0.206
24	Slashdot social network from February 2009	Slashdot0902	82,170	948,465	23.1	1.617	4.728	0.080	0.202	0.026	0.250
25	Slashdot social network from November 2008	Slashdot0811	77,362	905,469	23.4	1.603	4.685	0.083	0.207	0.026	0.252
26	Social network of github developers.	musae_git	37,702	289,004	15.3	1.267	3.482	−0.001	0.152	0.012	0.314
27	Social network of LastFM users from asia.	lastfm_Asia	7,626	27,807	7.3	1.807	3.730	0.006	0.679	0.179	0.195
28	Twitch user-user networks of gamers who stream in English	musae_ENGB	7,128	35,325	9.9	1.204	2.770	0.001	0.267	0.042	0.277
29	Twitch user-user networks of gamers who stream in French	musae_FR	6,551	112,667	34.4	1.303	3.392	−0.001	0.084	0.054	0.378
30	Twitch user-user networks of gamers who stream in German	musae_DE	9,500	153,139	32.2	1.305	3.476	−0.001	0.062	0.046	0.374
31	Twitch user-user networks of gamers who stream in Portugal	musae_PTBR	1,914	31,300	32.7	1.127	2.680	−0.003	0.081	0.131	0.402

Table 1: Continued.

Nb	Network description	Shortname	N	E	$<k>$	α	α^2	a	Q	C	Cl
32	Twitch user-user networks of gamers who stream in Russian	musae_RU	4,387	37,305	17.0	1.054	2.522	−0.003	0.101	0.049	0.337
33	Twitch user-user networks of gamers who stream in Spain	musae_ES	4,650	59,383	25.5	1.327	3.233	−0.001	0.109	0.084	0.352
34	Wikipedia page-to-page networks on chameleon topic	musae_chameleon	2,279	36,102	31.7	0.974	2.381	0.006	0.203	0.445	0.291
35	Wikipedia page-to-page networks on crocodile topic	musae_crocodile	11,633	180,021	31.0	0.892	2.483	−0.010	0.181	0.039	0.316
36	Wikipedia page-to-page networks on squirrel topic	musae_squirrel	5,203	217,074	83.4	0.748	2.245	−0.001	0.072	0.451	0.335
37	Wikipedia who-votes-on-whom network	Wiki-vote	7,117	103,690	29.1	1.412	3.644	−0.001	0.025	0.136	0.318
38	BlogCatalog social blog	BlogCatalog1	88,784	4,186,390	94.3	2.265	9.866	−0.001	0.018	0.060	0.331
39	BlogCatalog social blog version 2	BlogCatalog2	97,884	2,043,701	41.8	2.141	9.330	−0.001	0.006	0.057	0.355
40	BlogCatalog social blog version 3	BlogCatalog3	10,312	333,983	64.8	1.929	6.939	−0.001	0.026	0.091	0.424
41	Douban online social network	Douban	154,908	654,188	8.4	3.946	10.755	−0.048	0.093	0.010	0.195
42	Gowalla location-based social networking	Gowalla	196,591	950,327	9.7	1.386	5.337	0.006	0.501	0.023	0.221
43	TheMarker cafe online social network	TheMarker	69,413	1,644,849	47.4	2.547	9.799	0.000	0.022	0.046	0.332
44	Brightkite location-based online social network	Brightkite	58,228	214,078	7.4	2.330	6.802	0.005	0.539	0.111	0.224
45	The friendships network between users of the website http://www.hamsterster.com	Hamsterster	2,426	16,630	13.7	2.599	6.051	0.067	0.394	0.231	0.404
46	A google-plus subgraph	Soc-gplus	23,628	39,242	3.3	1.188	4.021	−0.068	−0.027	0.004	0.251
47	Anybeat online social network	Anybeat	12,645	67,053	10.6	0.922	3.294	−s0.009	0.133	0.018	0.323
48	Advogato online social network	Advogato	6,551	51,332	15.7	2.064	5.785	0.078	0.312	0.111	0.000

where E is the number of edges, a_{ij} is the element of the adjacency matrix A in the row i and column j, k_i is the degree of i, k_j is the degree of j, c_i is the module (or community) of i, c_j that of j, the sum goes over all i and j pairs of nodes, and $\delta(x, y)$ is 1 if $x = y$ and 0 otherwise [13].

(v) Global clustering coefficient (C): the global clustering coefficient (C) is based on triplets of nodes. A triplet is three nodes that are connected by either two (open triplet) or three (closed triplet) undirected edges. The global clustering coefficient is the number of closed triplets (or $3x$ triangles because a triangle comprises 3 overlapping triplets, each centered at one of the three nodes) over the total number of triplets (both open and closed). The formula is as follows:

$$C = \frac{\lambda_{\text{closed}}}{\lambda_{\text{total}}}, \tag{2}$$

where λ_{closed} is the number of closed triplets and λtotal is the total number of triplets in the network. The global clustering coefficient represents the overall probability for the network to have adjacent nodes interconnected, thus making more tightly connected modules [41].

(vi) Average closeness (Cl) is the average of all network nodes' closeness, where the closeness (or closeness centrality) of a node is calculated as the reciprocal of the sum of the length of the shortest paths between the node and all other nodes in the graph [42, 43]:

$$Cl = \overline{Cl}_i = \frac{1}{N}\sum_{i=1}^{N} Cl_i, \text{with} Cl_i = \frac{1}{\sum_{j\neq i} d(i,j)}, \tag{3}$$

where N is the number of nodes and $d(i, j)$ is the length of the shortest path between nodes i and j.

2.1. Network Robustness Monte Carlo Simulation. For each network, we run two node removal processes using Monte-Carlo simulations. Nodes are removed consecutively following the ranking of initial betweenness (IB) and the ranking of the recalculated degree (RD). In the case of ties, e.g., nodes with an equal betweenness or degree score, we removed one of them at random. After each node removal, we compute the network robustness measure and the relative size of the largest connected component LCC, together with the accumulated proportion of nodes removed q. Finally, we obtain two curves LCC (q) corresponding to two node removal processes, IB and RD. The whole simulation is repeated 10 times, and the final curves LCC (q) are the average results.

In addition, we compute a single value defined as the network robustness (R), as performed by Bellingeri et al. [44], and the area below the normalized LCC curve during the removal process, $R = \overline{\text{LCC}(q)}$. R therefore can be between two theoretical extremes, $R \approx 0$ (absolute fragile network) and $R \approx 0.5$ (absolute robust network). We denote

RRD and RIB as the network robustness against RD and IB node attack strategies, respectively.

In summary, we collect 48 real-world social networks, and then, we compute 9 NSIs for each network as inputs. In parallel, we run Monte Carlo simulations and obtain the robustness represented by two metrics, RRD and RIB. The higher they are, the more robust the network is. Those two metrics are the output of each network and will be predicted using ML models.

3. Machine Learning Approach

This section presents the details of SLR, MLR, and RF models.

3.1. Simple Linear Regression Model (SLR). Linear regression is the simplest model for prediction. The SLR model between the network robustness R and an NSI x is expressed by the linear equation:

$$R = a_0 + a_1 x, \tag{4}$$

where a_0 is the intercept and a_1 is the slope. In (4), an ordinary least square (OLS) is applied for estimating coefficients by minimizing an appropriate loss function [45, 46]. Once the OLS process, which is also called the fitting process, is performed, we can use (1) to predict the robustness R of a new network for a given indicator x. In addition, we derive a statistics t-test from the OLS process with the null hypothesis H0: $a_1 = 0$. A rejection of H0 means that there is a significant linear relationship between R and the NSI x.

We run the SLR model fit for all NSIs listed in Table 1 excluding E because it can be expressed in terms of two other NSIs : $E = N < k > /2$.

3.2. Multiple Linear Regression Model. Multiple linear regression (MLR) is an extension of SLR for multidimension variables $x = (x_1, x_2, \ldots, x_n)$, where $x_1, x_2, \ldots, x_n$ are NSIs. The linear equation between network robustness R and NSIs is as follows:

$$R = a_0 + a_1 x_1 + a_2 x_2 + \cdots + a_n x_n, \tag{5}$$

where a_i are coefficients obtained from the OLS method.

3.3. Random Forest Model. The random forest (RF) belongs to the ensemble class of ML models, indicating that it aggregates the prediction from an ensemble of ML base models, here, decision tree regression (DTR) models. We briefly describe the DTR in the following section.

A DTR starts with the root of the tree containing all samples (48 networks in our case). It then splits into two different nodes by selecting samples whose value of a certain variable is higher or lower than a certain threshold value. Figure 2(a) represents a basic decision tree diagram for our dataset. The root node containing 48 networks splits into two other nodes by considering whether the variable (NSI in our case) scale-free exponent α is higher or lower than 2.5.

The DTR selects the variable, and its splitting value is based on information theory, in concrete considering the entropy concept. Entropy is a metric of uncertainty of a node. The DTR splits a node by maximizing the information gain, which is the weighted difference between the total entropy of two resulting nodes and the entropy of the initial node. The DTR successively splits until a stopping condition is reached, for example if the size of the current node is smaller than 20. The final node is also called a leaf node. In Figure 2(a), after the first split of the root, the left child node becomes a leaf node, while the right child node continues to split into two leaf nodes.

Once the final DTR is obtained, it can be used to predict the value of a new sample as follows. The new sample will be classified into one of the leaves, and its prediction value will be the average value of all the samples that are classified into the same leaf.

Finally, the RF model creates multiple decision trees randomly drawn from the data, usually several hundred, and averaging the results from all trees to output a new result often leads to strong predictions [47, 48].

The decision tree can fit nonlinear datasets because it can split the same NSI multiple times. However, decision tree is easy to be overfitting, i.e., it is too sensitive to the training data while failing to predict new coming (testing) data. In order to address this problem, a random forest (RF) model is obtained by creating multiple randomly drawn decision trees from data, usually several hundred. The final regression prediction will be the average prediction of all the decision trees [47–49] (in this work, we implement an RF with 300 DTRs). Using an RF, "feature importance" measurement can be derived to rank the NSI [50].

3.4. Data Preparation, Validation, and Performance Evaluation. All NSIs can be computed from the network's data, and thus, our dataset did not contain missing values. We also exclude E because of redundancy as mentioned above. The other 8 NSIs are normalized to avoid large differences in the indicators' range:

$$x'_{i,j} = \left(x_{i,j} - \overline{x}_i\right)\sigma\left(x_i\right), \tag{6}$$

where $x_{i,j}$ is the value of the NSI i for observation (network) j and $\overline{x}_i$ and $\sigma(x_i)$ are the mean and the standard deviation of the NSI i, respectively.

In the first step, we use the whole dataset to build ML models and compare the results between models and two target variables. However, due to overfitting problems in many ML models, the model's performance for new data is not always coherent as that in the training step, and we need to validate models in the second step. We choose the leave-one-out validation [51]. In this way, we train each of the above models 48 times: each time the whole dataset excluding one observation is used to train the model, and then, the model is used to predict the target value of the remaining (hold-out) observations and repeats for each of 48 hold-out observations. The overall evaluation result is the average across all 48 regressions.

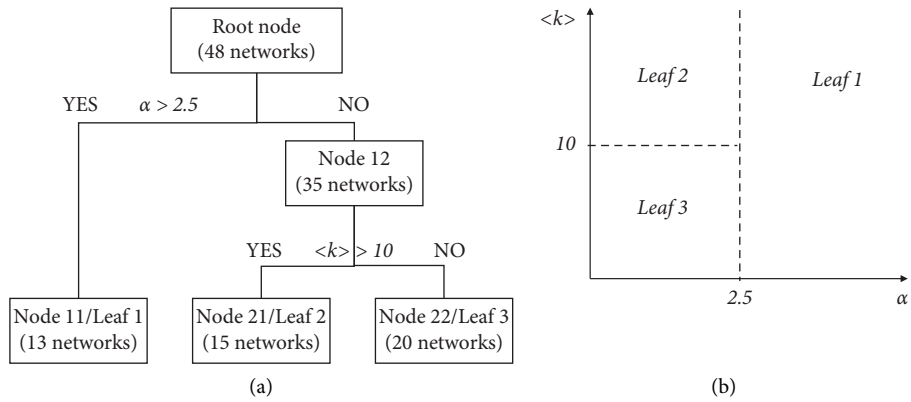

Figure 2: (a) An example of a decision tree: the root node containing 48 networks splits into two other nodes, Node 11 and Node 12, with 13 and 35 networks, respectively, according to the value of the scale-free exponent α. Then, Node 12 splits into two other nodes, Node 21 and Node 22, with 15 and 20 networks, respectively, according to the value of the network density $<k>$. We assume that at Node 11, Node 21, and Node 22, no split is possible because of a certain stopping rule, and thus, they become final leaves. In general, any NSI can be used to divide networks at any split, and the decision tree can be arbitrarily complex depending on the stopping rule. (b) An illustration of the same decision tree in the 2-dimension (α and $<k>$) space with final leaves.

It is noted that for the SLR model, we only consider regression coefficients in order to analyze the dependence of robustness metrics with respect to each NSI. However, for MLR and RF models, we analyze the prediction of robustness metrics using four common evaluation metrics for regression problems, the root mean square error (RMSE) and the coefficient of determination (also named the explained variance ratio, R^2) as analytical metrics and the frequency distribution and the Q-Q plot of residual errors as graphical metrics.

RMSE is the square root of the summation of the squared difference between observed and predicted data points. The RMSE has the same unit as the target feature and is generally considered the model error. A lower RMSE value represents superior prediction results. The formula of the RMSE is provided by

$$RMSE = \sqrt{\frac{\sum_{j=1}^{n}\left(R_j - R_{\text{predicted},j}\right)^2}{n}}, \tag{7}$$

where n is the number of observations, R_j denotes the empirical (simulated) network robustness, and $R_{\text{predicted},\,j}$ is the predicted value of robustness for the observation j.

R^2 is used to represent the general prediction performance of regression models. R^2 is one minus the ratio of the remaining variance and the original variance. The formula of R^2 is provided by

$$R^2 = 1 - \frac{\sum_{j=1}^{n}\left(R_j - R_{\text{predicted},j}\right)^2}{\sum_{j=1}^{n}\left(R_j - \overline{R}_j\right)^2}, \tag{8}$$

where n is the number of observations, R_j is the simulation robustness, $R_{\text{predicted},\,j}$ denotes the predicted value for observation j, and $\overline{R}_j$ is the average of all the simulation robustness. R^2 varies between 0 (the model has no prediction ability) and 1 (the model correctly predicts all values).

The residual error, $\varepsilon = R_{\text{empirical}} - R_{\text{predicted}}$, is simply an error between the empirical (simulated) network robustness and the predicted value of robustness. The distribution histogram of ε is expected to be close to the origin. Furthermore, the most important assumption of a linear regression model is that residual errors are independent, and consequently, these errors are expected to be normally distributed.

The network is analyzed using the "graph-tool" library in *Python*. All data preparation, model building, and evaluation are written using the *Python* code. The hardware for numerical simulations is a PC with an i9-10850 Intel processor and 32 GB RAM.

4. Results

4.1. Network Robustness as a Function of the NSIs and SLR T-Test. The simulation robustness of each network RIB and RRD is represented in Table 2. Overall, we found that RRD is slightly smaller than RIB for most networks (43 out of 48 networks), with an average of 0.148 vs. 0.173, respectively. It suggests that the RD strategy has more efficacy than IB for attacking real-world social networks. The largest and sparsest network, Email-EuAll ($N = 265{,}216$ and $<k \geq 1.58$), has the smallest robustness with an equal RIB and RRD of 0.001. In contrast, the gemsec_deezer_HR network, with $N = 54{,}575$ and $<k \geq 9.12$, has the strongest robustness with an RIB and RRD of 0.375 and 0.338, respectively.

In Figure 3, we plot RRD and RIB as a function of 8 independent NSIs, and we found that RRD and RIB behave similarly in all cases. The SLR unveils some significant relationships between R and NSIs (Figure 3 and Table 3). For example, in Figure 3(a), we can see that both RRD and RIB slightly decrease with the network size N. This linear dependence between robustness RRD and RIB and N is tested by using the SLR model, and we found that it is statistically significant, with a confidence level of 95% (p value <0.05, Table 3).

Table 2: Simulation result by IB and RD node attack strategies, represented by the network robustness metrics R_{IB} and R_{RD}, for all 48 real-world social networks (sort by networks' size from smallest to largest).

Short names	N	E	R_{IB}	R_{RD}
musae_PTBR	1,914	31,300	0.257	0.214
musae_chameleon	2,279	36,102	0.153	0.143
Hamsterster	2,426	16,630	0.134	0.133
Tvshow	3,894	17,263	0.139	0.153
musae_RU	4,387	37,305	0.209	0.149
musae_ES	4,650	59,383	0.248	0.202
musae_squirrel	5,203	217,074	0.298	0.184
CA-GrQc	5,244	28,981	0.057	0.069
Politician	5,910	41,730	0.198	0.195
musae_FR	6,551	112,667	0.288	0.240
Advogato	6,551	51,332	0.100	0.090
Government	7,059	89,456	0.311	0.283
Wiki-vote	7,117	103,690	0.136	0.144
musae_ENGB	7,128	35,325	0.180	0.132
lastfm_asia	7,626	27,807	0.171	0.137
musae_DE	9,500	153,139	0.282	0.229
CA-HepTh	9,879	51,972	0.097	0.091
BlogCatalog3	10,312	333,983	0.194	0.181
public_figure	11,567	67,115	0.204	0.168
musae_crocodile	11,633	180,021	0.124	0.071
CA-HepPh	12,010	237,011	0.138	0.162
Anybeat	12,645	67,053	0.039	0.028
Athletes	13,868	86,859	0.234	0.199
Company	14,115	52,311	0.175	0.150
CA-AstroPh	18,774	396,161	0.193	0.212
musae_facebook	22,472	171,003	0.228	0.206
CA-CondMat	23,135	186,937	0.113	0.113
Soc-gplus	23,628	39,242	0.002	0.001
Cit-HepTh	27,772	352,808	0.342	0.307
new_sites	27,919	206,260	0.264	0.228
deezer_Europe	28,283	92,753	0.186	0.153
Cit-HepPh	34,548	421,579	0.350	0.307
Email-Enron	36,694	367,663	0.048	0.039
musae_git	37,702	289,004	0.168	0.122
deezer_RO	41,775	125,827	0.261	0.200
deezer_HU	47,540	222,888	0.343	0.287
Artists	50,515	819,306	0.299	0.265
deezer_HR	54,575	498,203	0.375	0.338
Brightkite	58,228	214,078	0.107	0.083
TheMarker	69,413	1,644,849	0.113	0.100
Soc-Epinions1	75,881	508,838	0.066	0.054
Slashdot0811	77,362	905,469	0.093	0.073
Slashdot0902	82,170	948,465	0.103	0.077
BlogCatalog1	88,784	4,186,390	0.072	0.063
BlogCatalog2	97,884	2,043,701	0.016	0.014
Douban	154,908	654,188	0.026	0.024
Gowalla	196,591	950,327	0.160	0.115
Email-EuAll	265,216	420,046	0.001	0.001
Average	38,026	391,698	0.173	0.148
Std	51,490	687,601	0.098	0.085

Interestingly, RRD and RIB do not statistically linearly depend on the network density $<k>$ as found previously in [4, 52] (Figure 3(b) and Table 3). This contrasting observation would suggest that network robustness also depends on other NSIs and that the network density alone cannot predict the whole network's robustness as previously seen.

Besides N, the only other NSI that shows a significant linear relationship is the modularity Q (Figure 3(f)) in the case of RRD.

However, in Figure 3, we still observe some nonlinear dependencies. For example, in Figure 3(e), we show that network robustness decreases with the assortativity coefficient a when $a > 0$. However, it decreases faster when a is close to 0 and increases with a when $a < 0$.

Similarly, in Figure 3(g), we found that the relationship between RRD and RIB and the global clustering coefficient C follows an inverted u-shaped pattern. We ran a two-line statistical test [53] and found that two-line (or broken line) regression is significantly better than a single-line test. The breakpoint was found to be $C = 0.115$. Both RRD and RIB linearly increase with C (with a significance level of 95%) up to the breakpoint and linearly decrease with C (with a significance level of 95%). One possible explanation is that if the network is sparse, more triplets help increase the network's connectivity and thus increase its robustness. However, above a certain value (when $C = 0.115$), more triplets may denote the presence of hubs or central nodes, which are likely to be the target of intentional node removal strategies such as RD and IB, consequently lowering network robustness.

4.2. Machine Learning Prediction of Network Robustness. The results of the previous section suggest that the social network's robustness depends on multiple NSIs in a highly complex, multidimensional, and nonlinear manner. To improve the model prediction, in this section, we use two multiple variable ML models, MLR and RF, to predict network robustness.

The results of multiple linear regression MLR are shown in Table 4. We found that both R_{IB} and R_{RD} have a positive overall linear regression coefficient with respect to α, Q, Cl, and $<k>$ and a negative overall linear regression coefficient with respect to α^2, a, C, and N. Moreover, the MLR result indicate that α, α^2, and $<k>$ are the most significant coefficients. A positive linear regression coefficient for the average node degree $<k>$ suggests that networks are more robust when k is higher, while all other NSIs are fixed. This result agrees with previous outcomes demonstrating that denser networks may be more resistant to the attack [4, 52]. However, the different results between the MLR and SLR would suggest that there is a strong correlation between $<k>$ and other NSIs. In addition, the MLR model predicts R_{IB} better than R_{RD}, with an R^2 coefficient of 58.04% compared to 51.76%. Nevertheless, the RMSE was smaller for R_{RD}, with a value of 0.0657, compared to 0.0709 for RIB (this is because the standard deviation of RIB is higher than that of R_{RD}, as shown in Table 2 (bottom row)).

Because of the nonlinearity found in the previous section, we expect that the regression result using the RF model will be improved. Table 5 represents the regression result of the RF model. We found that R^2 increases to 92.24% and 91.88% for R_{IB} and R_{RD} regressions, respectively. Interestingly, the RF model predicts R_{RD} roughly as well as R_{RD}, while MLR predicts R_{IB} better than R_{RD}, suggesting that R_{RD}

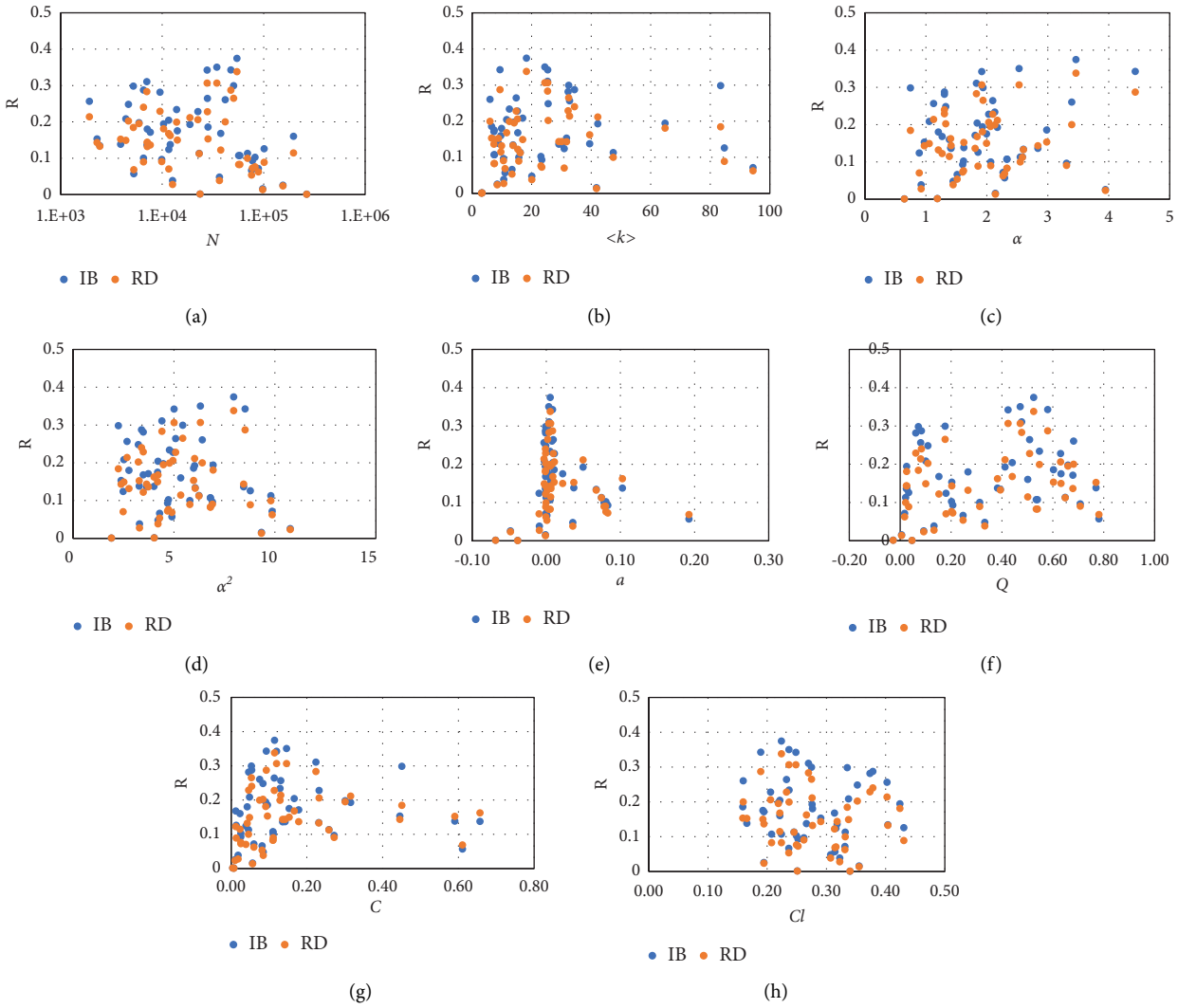

Figure 3: Simulation result by IB and RD node attack strategies, represented by the network robustness metrics R_{IB} and R_{RD}, for all the 48 real-world social networks as a function of 8 NSIs.

Table 3: The SLR results for 8 NSIs. The last two columns show the slope, and in parenthesis, the R^2 values of the SLR between the NSI and RIB or RRD. The bold character with an asterisk indicates a significant relationship, with a confidence level of 95%.

Nb	NSI	R_{IB}	R_{RD}
1	N	$\mathbf{-6.920{\cdot}10^{-7}\ (0.132)}^{*}$	$\mathbf{-6.377{\cdot}10^{-7}\ (0.150)}^{*}$
2	<k>	0.0006 (0.015)	0.0004 (0.010)
3	α	0.0215 (0.031)	0.0247 (0.056)
4	α^2	−0.0039 (0.007)	−0.0011 (0.000)
5	α	−0.3282 (0.019)	−0.1184 (0.003)
6	Q	0.0917 (0052)	$\mathbf{0.1022\ (0.087)}^{*}$
7	C	0.0274 (0.001)	0.0781 (0.021)
8	Cl	−0.1515 (0.010)	−0.1607 (0.016)

may follow a stronger nonlinear relationship with NSIs than RIB. Additionally, the RMSE improved both for R_{IB} and R_{RD}, with a value of 0.0272 and 0.0241, respectively. Interestingly, the feature importance ranking in Table 5 shows that with an RF model, the assortativity a, the global closeness C, and the node number N are the most important NSIs. This result agrees with the exploratory observations shown in Figure 3 as discussed above.

In Figure 4, we compare network robustness R_{IB} and R_{RD} with the prediction value given by MLR and RF using a scatter plot. The scatter plots indicate that RF fit data significantly better than MLR, where the predicted actual data points are closer to the diagonal line $y = x$. Meanwhile, for MLR regression, we still found nonlinear dependency between the actual and predicted values. As a matter of fact, the MLR model was not able to capture the inherent nonlinearity dependency in the actual data. We also analyzed the residual errors of the above regression using the frequency histogram and QQ-plot and found that they follow a normal distribution relatively well (Figures 5–8).

Finally, we run leave-one-out regression for both models MLR and RF in order to avoid overfitting. The result is summarized in Table 6, and the scatter plots are shown in Figure 9. We found that the prediction result is less accurate than the above "in-sample" training with lower RMSEs in both MLR and RF models. We obtained an RMSE of 0.0812

Table 4: Fit coefficients and the evaluation result given by the MLR. R_{IB} or R_{RD} columns show the slope coefficient, and in parenthesis, the standard error values for the NSI. The bold character with an asterisk indicates a significant relationship between the NSI and the robustness R, with a confidence level of 95%.

Nb	NSI	Regression coefficients	
		R_{IB}	R_{RD}
1	α	**0.113 (0.028)***	**0.088 (0.026)***
2	α^2	**−0.118 (0.028)***	**−0.083 (0.026)***
3	α	**−0.029 (0.013)***	−0.025 (0.012)
4	Q	**0.047 (0.02)***	0.038 (0.019)
5	C	**−0.034 (0.016)***	−0.016 (0.015)
6	Cl	0.005 (0.014)	0.007 (0.013)
7	N	−0.013 (0.013)	−0.014 (0.012)
8	$<k>$	**0.077 (0.017)***	**0.056 (0.016)***
9	Intercept	**0.173 (0.01)***	**0.148 (0.009)***
MLR results			
	RMSE	0.0709	0.0657
	R^2	58.04%	51.76%

Table 5: Feature importance of the NSI and the evaluation result given by RF.

NSI	Feature importance	
	R_{IB}	R_{RD}
α	0.0622	0.0654
α^2	0.0535	0.0463
a	0.2765	0.1912
Q	0.0823	0.0658
C	0.1114	0.1834
Cl	0.0581	0.0584
N	0.2683	0.2759
$<k>$	0.0873	0.1133
RMSE	0.0272	0.0241
R^2	92.24%	91.88%

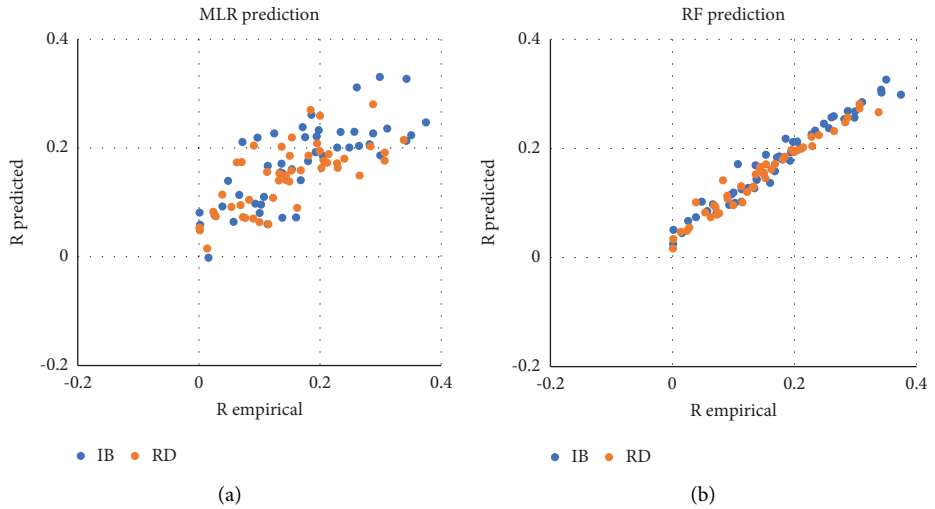

Figure 4: Scatter plots between the predicted value of robustness (R predicted) and simulated (R empirical) for MLR (a) and RF model (b). The model is trained using the whole dataset, and the predicted values are of the same dataset.

and 0.0760 for R_{IB} and R_{RD} predictions using MLR, respectively, and an RMSE of 0.0733 and 0.0636 for R_{IB} and R_{RD} predictions using RF, respectively. Even though the regression results are less effective because we predict the single sample which is independent of the remaining samples used for training (building the ML model), residual errors still fit well to a normal distribution as shown in the histogram and QQ-plots (Figures 10–13).

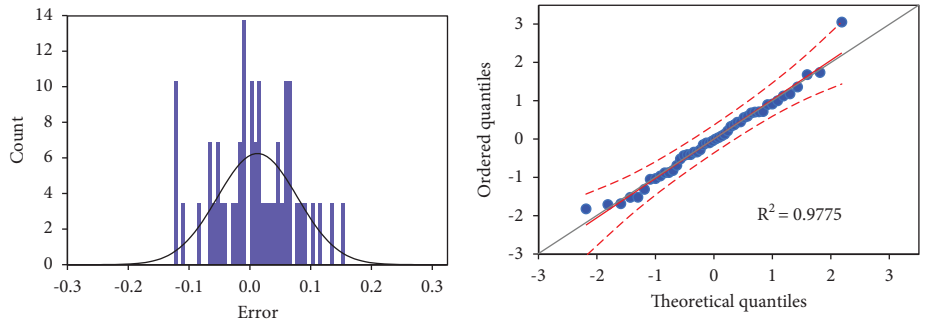

Figure 5: Histogram of residual errors for MLR prediction of the IB strategy for the whole dataset (a) and its QQ-plot (b).

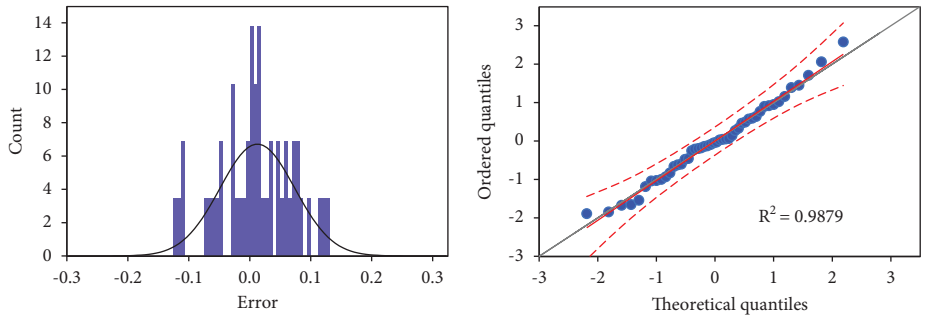

Figure 6: Histogram of residual errors for MLR prediction of the RD strategy for the whole dataset (a) and its QQ-plot (b).

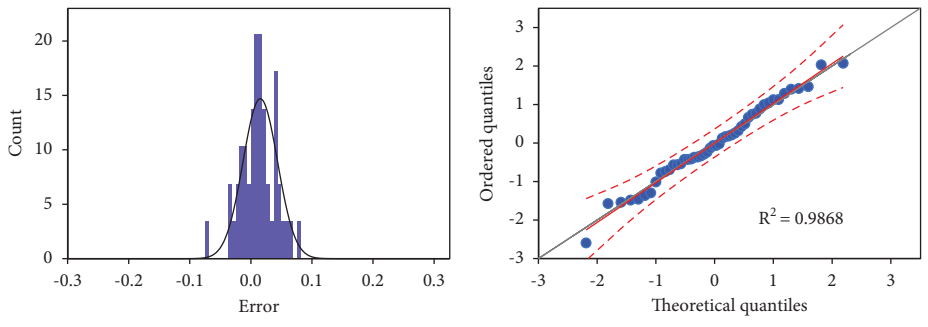

Figure 7: Histogram of residual errors for RF prediction of the IB strategy for the whole dataset (a) and its QQ-plot (b).

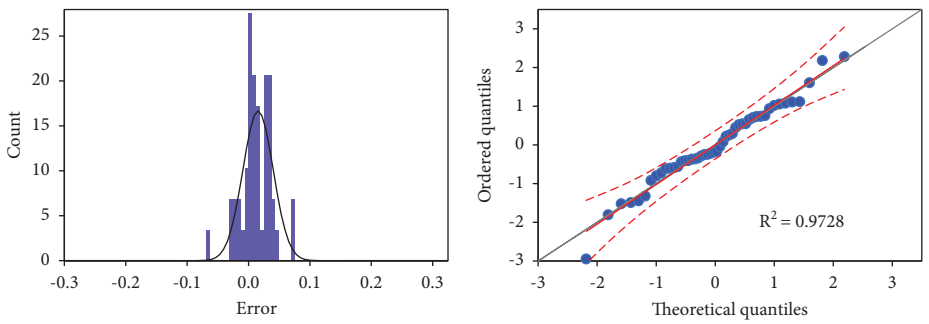

Figure 8: Histogram of residual errors for RF prediction of the RD strategy for the whole dataset (a) and its QQ-plot (b).

Table 6: MLR and RF evaluation results using the leave-one-out method.

	MLR	RF
R_{IB}		
RMSE	0.0812	0.0733
R^2	31.30%	43.87%
R_{RD}		
RMSE	0.0760	0.0636
R^2	19.30%	43.47%

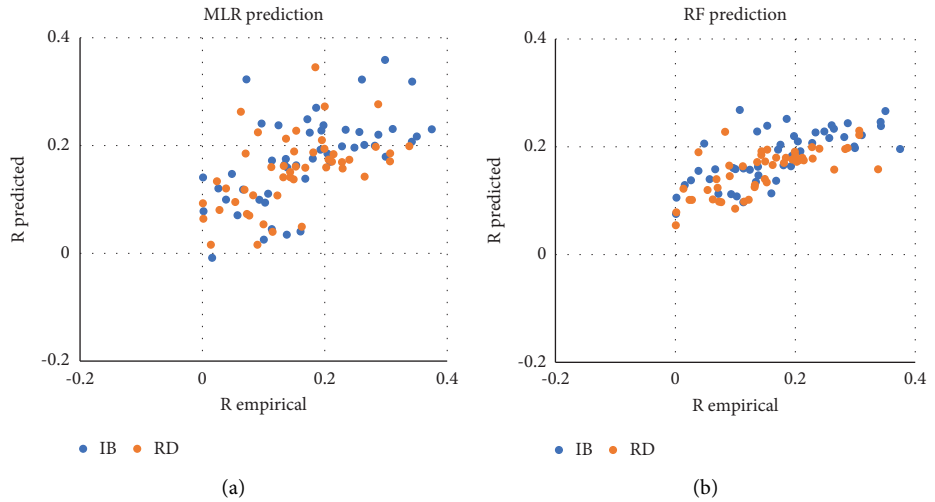

Figure 9: Scatter plots between the predicted value of robustness (R predicted) and simulated (R empirical) of the hold-out observation for MLR (a) and RF model (b). The model is trained using the whole dataset excluding one observation (hold-out observation) and is used to predict the outcome of the hold-out observation.

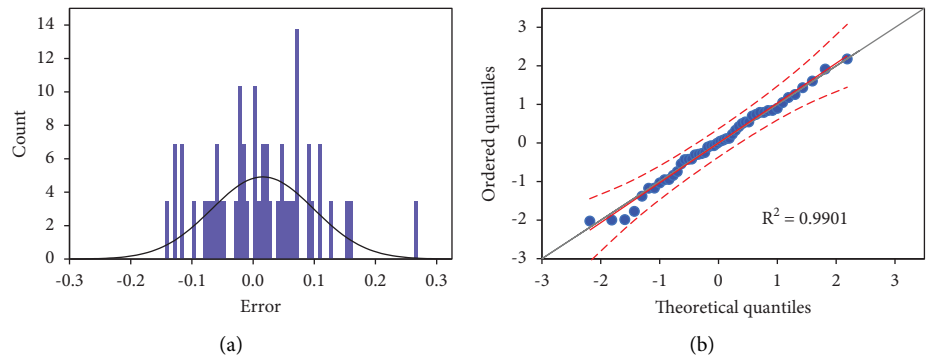

Figure 10: Histogram of residual errors for MLR prediction of the IB strategy for the leave-one-out sample (a) and its QQ-plot (b).

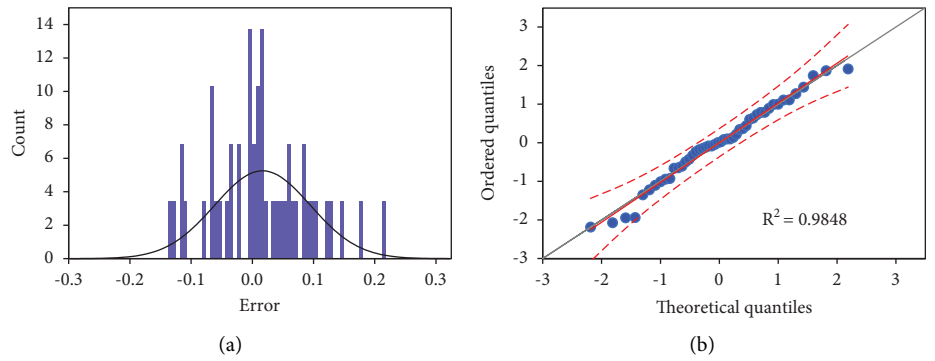

Figure 11: Histogram of residual errors for MLR prediction of the RD strategy for the leave-one-out sample (a) and its QQ-plot (b).

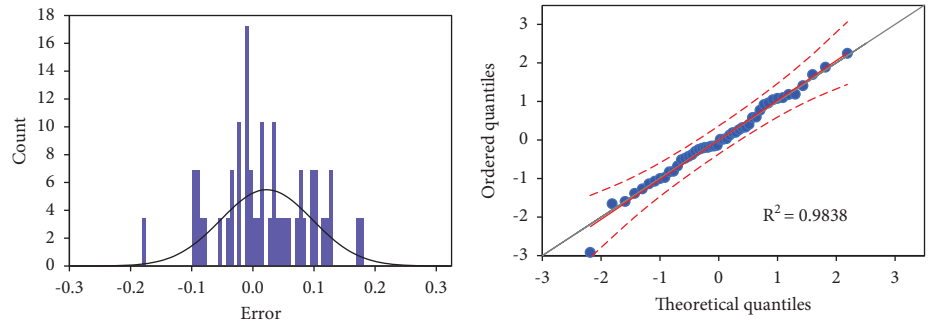

Figure 12: Histogram of residual errors for RF prediction of the IB strategy for the leave-one-out sample (a) and its QQ-plot (b).

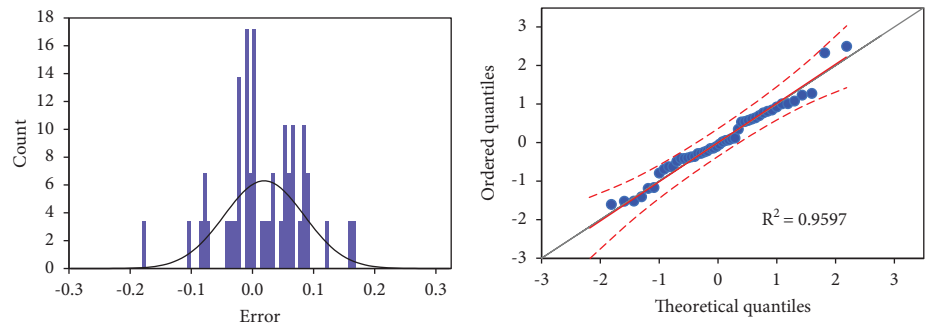

Figure 13: Histogram of residual errors for RF prediction of the RD strategy for the leave-one-out sample (a) and its QQ-plot (b).

5. Discussion and Conclusion

In this work, we have analyzed the robustness of 48 real-world social networks with the node number ranging over five orders of magnitude, from 1,914 to 265,216. Using Monte Carlo simulations, we have run two commonly used node attack strategies, IB and RD strategies, whose computation time is within our hardware capability. We found that their corresponding simulation time, t_{IB} and t_{RD}, scales linearly with the product of the network's node number and edge number, i.e., $N \times E$. We also found that the two attack strategies IB and RD present similar efficacy when evaluated by the unique robustness metric R, with RD slightly better than IB (average R_{RD} is slightly smaller than average R_{IB}). It suggests that for the social networks used in this study, the RD strategy is the most efficient strategy to dismantle (breakdown) networks, both in terms of computational cost and breakdown efficiency.

To understand how the structure of a social network determines its robustness, we investigate the relationship between the metric R and a set of network structural indicators (NSIs) from the literature. The simple linear regression (SLR) between R and NSIs shows low goodness of fitting, and it is overall not able to produce significant prediction models. The low goodness of SLR would indicate that network robustness depends on NSIs in a nonlinear manner.

To improve fitting, we have developed two machine learning models to predict two robustness metrics R_{RD} and R_{RD} from the combination of 8 NSIs, multiple linear regression (MLR), and random forest (RF) model. The latter one is chosen as it can handle nonlinear data well and is built on a collection of base models, decision tree classifiers. We found clearly that the random forest model can predict network robustness better than the multiple linear regression model. In concrete, the RF model predicts network robustness with an RMSE of 0.0272 and 0.0241 for R_{IB} and R_{RD}, respectively. This result is encouraging to predict real-world social network robustness, although the error is about 16% (for R_{IB}, the RMSE is 0.0272 compared to an average R_{IB} of 0.173, and for R_{RD}, the RMSE is of 0.0241 compared to an average R_{RD} of 0.148). Meanwhile, when the leave-one-out evaluation is applied, the RMSE increases to 0.0733 and 0.0636 for R_{IB} and R_{RD}, respectively, which is about one-third of the average value.

Finally, MLR indicates that the most important factors to predict RIB are the exponent α and the average node degree $<k>$, for both R_{IB} and R_{RD}. In particular, a higher value of α is correlated with higher R_{IB} and R_{RD}. Higher absolute values of the exponent α denote a network with fewer hub nodes (highly connected nodes) [35]. In consequence, the RD and IB attack strategies cannot find large hub nodes whose removal may disintegrate the network faster, resulting in higher values of R_{RD} and R_{IB}. Additionally, MLR indicates that $<k>$ is positively related to lower R_{IB} and R_{IB}. This last outcome agrees with previous results, demonstrating that networks with higher edge density may be more resistant to the attack [4, 52]. On the

other hand, it confirms that SLR, which focuses on a single NSI, may not be able to predict the robustness of real-world social networks.

Our work demonstrates that the ML model can be used to predict network robustness with acceptable results. Therefore, it alleviates the need to run a full Monte Carlo simulation on a network when only approximate robustness is needed. Meanwhile, more network datasets are expected to improve the accuracy of ML models. This work also contributes to the understanding of the relationship between real-world social network robustness and its structural indicators. Finally, we have proved that using a data-driven approach to predict the outcome of the nonlinear and complex dynamic process, such as network robustness, is an appropriate approach [54–60].

Abbreviations

RD: Recalculated degree node attack strategy
IB: Initial betweenness node attack strategy
RB: Recalculated betweenness node attack strategy
t_{IB}: Total simulation time for the attack strategy IB
t_{RD}: Total simulation time for the attack strategy RD
t_{RB}: Total simulation time for the attack strategy RB
SLR: Simple linear regression model
MLR: Multiple linear regression model
RF: The random forest model
DTR: Decision tree regression model
NSI: Network structural indicator
RMSE: Mean squared error
R^2: Coefficient of determination (also named the explained variance ratio)
a_0: Intercept coefficient of SLR
a_1: Slope coefficient of SLR
OLS: Ordinary least square method
ε: Error between the empirical (simulated) network robustness and the predicted value of robustness
α: Fitted scale-free exponent
k: Node degree
$<k>$: Average node degree
a: Degree assortativity
Cl: Global closeness
C: Global clustering coefficient
LCC: Largest connected component
N: Number of nodes
E: Number of edges
Q: Modularity indicator
α^2: Fitting variance of α
q: Accumulated proportion of nodes removed
R: Network robustness
R_{RD}: Network robustness against RD node attack strategies
R_{IB}: Network robustness against IB node attack strategies.

Appendix

The histogram and the QQ-plot of residual errors of all regressions are given in Figures 5–8 and Figures 10–13.

Data Availability

All the 48 real-world social networks are downloaded from the Stanford Large Network Dataset Collection (https://snap.stanford.edu/data/) and the Network Repository social networks (https://networkrepository.com/soc.php).

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

QN conceived analyses. NKKN, HHP, TTL, and TMN performed simulations. QN, NKKN, FS, RA, MB, and DC wrote the paper.

Acknowledgments

This work was supported by Vietnam's Ministry of Science and Technology (MOST) under the Vietnam-Italy Scientific and Technological Cooperation Program for the period of 2021–2023 and by the Vietnam National University Ho Chi Minh City (VNU-HCM), Ho Chi Minh city, Vietnam under grant nos. B2017-42-01. This research was funded by a grant from the Italian Ministry of Foreign Affairs and International Cooperation. This project received funding from the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme (grant agreement No. (816313)). The authors are greatly thankful to Van Lang University, Vietnam, for providing the budget for this study.

References

[1] S. Lehmann and Y.-Y. Ahn, "Complex spreading phenomena in social Systems," *Computational Social Sciences)*, Springer International Publishing, Berlin, Germany, 2018.
[2] M. Bellingeri, D. Cassi, and S. Vincenzi, "Efficiency of attack strategies on complex model and real-world networks," *Physica A: Statistical Mechanics and its Applications*, vol. 414, pp. 174–180, 2014.
[3] M. Bellingeri, D. Bevacqua, F. Scotognella et al., "Link and node removal in real social networks: a Review," *Frontiers in Physiology*, vol. 8, p. 228, 2020a.
[4] S. Iyer, T. Killingback, B. Sundaram, and Z. Wang, "Attack robustness and centrality of complex networks," *PLoS One*, vol. 8, no. 4, Article ID e59613, 2013.
[5] F. Morone and H. A. Makse, "Influence maximization in complex networks through optimal percolation," *Nature*, vol. 524, no. 7563, pp. 65–68, 2015.
[6] K. Nguyen and Q. Nguyen, "Resilience of stock cross-correlation network to random breakdown and intentional attack," *Studies in Computational Intelligence*, vol. 760, pp. 553–561, 2018.
[7] S. Wandelt, X. Sun, D. Feng, M. Zanin, and S. Havlin, "A comparative analysis of approaches to network-dismantling," *Scientific Reports*, vol. 8, no. 1, Article ID 13513, 2018.
[8] R. Pastor-Satorras, C. Castellano, P. Van Mieghem, and A. Vespignani, "Epidemic processes in complex networks," *Reviews of Modern Physics*, vol. 87, no. 3, pp. 925–979, 2015.

[9] C. Stegehuis, R. van der Hofstad, and J. S. H. van Leeuwaarden, "Epidemic spreading on complex networks with community structures," *Scientific Reports*, vol. 6, no. 1, Article ID 29748, 2016.

[10] C. Li, L. Wang, S. Sun, and C. Xia, "Identification of influential spreaders based on classified neighbors in real-world complex networks," *Applied Mathematics and Computation*, vol. 320pp. 512–523, C, 2018.

[11] J. Wang, C. Li, and C. Xia, "Improved centrality indicators to characterize the nodal spreading capability in complex networks," *Applied Mathematics and Computation*, vol. 334, pp. 388–400, 2018.

[12] L. Weng, F. Menczer, and Y.-Y. Ahn, "Virality prediction and community structure in social networks," *Scientific Reports*, vol. 3, no. 1, p. 2522, 2013.

[13] P. Holme, "Efficient Local strategies for vaccination and network attack," *Europhysics Letters*, vol. 68, no. 6, pp. 908–914, 2004.

[14] L. K. Gallos, F. Liljeros, P. Argyrakis, A. Bunde, and S. Havlin, "Improving immunization strategies," *Physical Review E*, vol. 75 pp. 1–4, 2007.

[15] J. Hadidjojo and S. A. Cheong, "Equal graph partitioning on estimated infection network as an effective epidemic mitigation measure," *PLoS One*, vol. 6, no. 7, Article ID e22124, 2011.

[16] M. A. Amaral, M. Md Oliveira, and M. A. Javarone, "An spidemiological model with Voluntary Quarantine strategies Governed by evolutionary vame dynamics," *Chaos, Solitons & Fractals*, vol. 143, Article ID 110616, 2021.

[17] H. Amini and A. Minca, "Epidemic spreading and qquilibrium social distancing in geterogeneous networks," *Dyn Games Appl*, vol. 12, no. 1, pp. 258–287, 2022.

[18] M. Bellingeri, D. Bevacqua, F. Scotognella, R. Alfieri, and D. Cassi, "A comparative analysis of link removal strategies in real complex weighted networks," *Scientific Reports*, vol. 10, pp. 3911–3915, 2020b.

[19] M. Bellingeri, M. Turchetto, D. Bevacqua et al., "Modeling the consequences of social distancing over epidemics spreading in complex social networks: from link removal analysis to SARS-CoV- 2 Prevention," *Frontiers in Physiology*, vol. 9, Article ID 681343, 2021.

[20] D. Achlioptas, R. M. D'souza, and J. Spencer, "Explosive percolation in random networks," *Science*, vol. 323, no. 5920, pp. 1453–1455, 2009.

[21] G. Dong, J. Fan, L. M. Shekhtman et al., "Resilience of networks with community structure behaves as if under an external field," *Proceedings of the National Academy of Sciences*, vol. 115, no. 27, pp. 6911–6915, 2018.

[22] O. Riordan and L. Warnke, "Explosive percolation is continuous," *Science*, vol. 333, no. 6040, pp. 322–324, 2011.

[23] Y. Sun, C. Liu, C.-X. Zhang, and Z.-K. Zhang, "Epidemic spreading on weighted complex networks," *Physics Letters A*, vol. 378, no. 7-8, pp. 635–640, 2014.

[24] A. Majdandzic, B. Podobnik, S. V. Buldyrev, D. Y. Kenett, S. Havlin, and H. Eugene Stanley, "Spontaneous recovery in dynamical networks," *Nature Physics*, vol. 10, no. 1, pp. 34–38, 2014.

[25] S. Wandelt, X. Shi, X. Sun, and M. Zanin, "Community detection boosts network dismantling on real-world networks," *IEEE Access*, vol. 8, pp. 111954–111965, 2020.

[26] R. Albert, H. Jeong, and A.-L. Barabasi, "Diameter of the world-wide web," *Nature*, vol. 401, pp. 130-131, 1999.

[27] R. Cohen, K. Erez, D. ben Avraham, and S. Havlin, "Resilience of the internet to random breakdowns," *Physical Review Letters*, vol. 85, no. 21, pp. 4626–4628, 2000.

[28] Q. Nguyen, T. V. Vu, H. D. Dinh et al., "Modularity affects the robustness of scale-free model and real-world social networks under betweenness and degree-based node attack," *Appl Netw Sci*, vol. 6, no. 1, p. 82, 2021b.

[29] X. Sun, V. Gollnick, and S. Wandelt, "Robustness analysis metrics for worldwide airport network: a comprehensive study," *Chinese Journal of Aeronautics*, vol. 30, no. 2, pp. 500–512, 2017.

[30] Q. Nguyen, N. K. K. Nguyen, D. Cassi, and M. Bellingeri, "New betweenness centrality node attack strategies for real-world complex weighted networks," *Complexity*, vol. 2021, pp. 1–17, Article ID 1677445, 2021a.

[31] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, vol. 40, no. 1, p. 35, 1977.

[32] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press, Cambridge UK, 1994.

[33] U. Brandes, "A faster algorithm for betweenness centrality," *Journal of Mathematical Sociology*, vol. 25, no. 2, pp. 163–177, 2001.

[34] Q. Nguyen and T. Trang Le, "Structure and robustness of Facebook's pages networks," in *Proceedings of the 2019 the 10th Conference on Network Modeling and Analysis (Marami 2019)*, Dijon, France, November 2019.

[35] M. E. J. Newman, "Power laws, Pareto distributions and Zipf's law," *Contemporary Physics*, vol. 46, no. 5, pp. 323–351, 2005.

[36] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*, Cambridge University Press, Cambridge UK, 2014.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, NV, USA, June 2016.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Computer Vision – ECCV 2016*, J. M. ., N. S. Leibe and M. Welling, Eds., vol. 9908B, pp. 630–645, Springer International Publishing, Cham, 2016.

[39] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[40] M. E. J. Newman, "Spread of epidemic Disease on networks," *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, vol. 66, no. 1, Article ID 016128, 2002.

[41] M. E. J. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.

[42] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, "Network robustness and Fragility: percolation on random graphs," *Physical Review Letters*, vol. 85, no. 25, pp. 5468–5471, 2000.

[43] S. Wasserman and K. Faust, *Social Network Analysis: A Handbook*, Sage, CA, USA, 2000.

[44] M. Bellingeri, D. Bevacqua, F. Scotognella, and D. Cassi, "The heterogeneity in link weights may decrease the robustness of real-world complex weighted networks," *Scientific Reports*, vol. 9, no. 1, Article ID 10692, 2019.

[45] A. S. Goldberger, "Classical linear regression," *Econometric Theory*, p. 158, John Wiley & Sons, NY, USA, 1964.

[46] F. Hayashi, *Econometrics*, p. 15, Princeton University Press, NJ, USA, 2000.

[47] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and Regression Trees*, CRC Press, Boca Raton, Florida, 1984.
[48] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, Morgan kaufmann, Burlington, Massachusetts, USA, 2006.
[49] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001a.
[50] R. Genuer, J. M. Poggi, and C. Tuleau-Malot, "Variable selection using random forests," *Pattern Recognition Letters*, vol. 31, no. 14, pp. 2225–2236, 2010.
[51] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: With Applications in R*, Springer, NY, USA, 2013.
[52] R. Albert and A.-L. Baraba´si, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47–97, 2002.
[53] R. H. Jones and B. A. Molitoris, "A statistical method for determining the breakpoint of two lines," *Analytical Biochemistry*, vol. 141, no. 1, pp. 287–290, 1984 Aug 15.
[54] R. Albert, H. Jeong, and A.-L. Baraba´si, "Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–382, 2000.
[55] A. Bavelas, "Communication gatterns in task-oriented groups," *Journal of the Acoustical Society of America*, vol. 22, no. 6, pp. 725–730, 1950.
[56] M. Bellingeri and D. Cassi, "Robustness of weighted networks," *Physica A: Statistical Mechanics and Its Applications*, vol. 489, pp. 47–55, 2018.
[57] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, "Attack vulnerability of complex networks," *Physical Review A*, vol. 65, no. 5, Article ID 056109, 2002.
[58] M. E. J. Newman, "Assortative mixing in networks," *Physical Review Letters*, vol. 89, no. 20, Article ID 208701, 28 October 2002.
[59] G. Sabidussi, "The centrality index of a graph," *Psychometrika*, vol. 31, no. 4, pp. 581–603, 1966.
[60] Y. Sun, C. Liu, C.-X. Zhang, and Z.-K. Zhang, "Epidemic spreading on weighted complex networks," *Physics Letters A*, vol. 378, no. 7-8, pp. 635–640, 2014.