IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 August 2022, accepted 30 August 2022, date of publication 5 September 2022, date of current version 14 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3204041

RESEARCH ARTICLE

Predicting the Robustness of Real-World Complex Networks

RUIZI WU, JIE HUANG, ZHUORAN YU, AND JUNLI LI
College of Computer Science, Sichuan Normal University, Chengdu 610066, China

Corresponding author: Junli Li (lijunli@sicnu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 62002249.

ABSTRACT Many real-world natural and social systems can be modeled as complex networks. As random failures and malicious attacks can seriously destroy the structure of complex networks, it is critical to ensure their robustness and maintain the functions. Generally, connectivity and controllability robustness are adopted to evaluate the performance of networked systems against external attacks and/or failures. A sequence of values is measured to dynamically indicate the network robustness with iterative node- or edge-removal. Calculating the robustness of large-scale real-world networks is usually time consuming, whereas deep-learning provides an efficient methodology to estimate network robustness performance. In this paper, a multi-convolutional neural network (CNN) method called Real-RP is designed to predict the robustness of real-world complex networks. Unknown real-world networks are first classified into known network categories, and their robustness performance is then predicted based on the knowledge of the specific network category trained using a substantial number of synthetic networks. Experimental results show that: 1) real-world complex networks can be classified by a CNN with high precision, and 2) the robustness performance of real-world networks can be predicted with lower average errors compared to existing methods.

INDEX TERMS Complex network, convolutional neural network, robustness, prediction.

I. INTRODUCTION

Real-world networked systems can be abstracted and studied as complex networks. Thus, the interactions among internal components are well described. Complex networks are ubiquitous, e.g., Internet, transportation networks, neural networks and so on [1], [2], [3], [4]. Due to the fact that complex networks include different structures and they are widely used in many different areas, various examples can be found in nature and real world. For example, the nervous system of the worm Caenorhabditis elegans can be regarded as a network formed by a large number of nerve cells interconnected by nerve fibers [5]. Computer network can be regarded as a network formed by self-working computers interconnected through communication media such as optical cable, twisted pair, coaxial cable, etc. [5]. Similarly, there are power grid [6], social relationship network [7], [8], transportation network [9] and so on.

Real-world networked systems often suffer from various external attacks and random failures, for example, the blackout in North America in 2003 [10]. A straightforward case of networks in normal life is the transportation system. Partial breakdown of the transportation system has a global effect on traffic. It is because a partial failure can be shifted to other parts through the traffic lines. When the local failures are relatively small, the system can easily bear them, and the failures are invisibly resolved. When the failed parts transfer too much extra loads to their neighbors, their neighbors will also fail and the loads will be transferred to their neighbors accordingly. All of a sudden, the cascade failure happens [1].

Connectivity and controllability play important parts in analyzing real-world systems such as power grid [11] and transportation networks [9]. Connectivity plays an important role for a network, which states the extensive process that the various parts connect to each other as a whole. Controllability

The associate editor coordinating the review of this manuscript and approving it for publication was Dost Muhammad Khan.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

refers to the ability for a networked system can be steered from any initial states to any target states under a certain control inputs, within a finite duration of time. Both connectivity and controllability are necessary for network to perform its fundamental tasks.

In a complex network, random failures and malicious attacks are always inevitable, which has an influence on the structure and basic functionality. Strengthening the network against such destructive failures and attacks has become an important issue receiving wide attention [12], [13], [14], [15], [16], [17]. Robustness is the capacity for networks to maintain its structure and basic functionalities against malicious attacks. Great progress has been made in the study of robustness. Especially for scale-free networks, scaling exponent and assortativity are important parameters, depicting the degree distribution and similar-degree connection propensity. Under malicious node attacks, the increasing of these two parameters contributes to the robustness markedly and provides another way to enhance the robustness of scale-free networks [18]. Nowadays great achievements have been made in robustness optimization based on a single measure. However, it is still a challenge to multiple attack scenarios. Based on the unique features of complex networks, a new parallel fitness evaluation method guided by a network property parameter is designed and embedded in a reference vector-guided multi-objective evolutionary algorithm. A computationally efficient multi-objective optimization algorithm is designed to solve the robustness optimization under multiple attack scenarios [19]. However, frequent calculation of robustness still remains a pain point.

In this paper, network robustness refers to both connectivity robustness and controllability robustness. The connectivity robustness and controllability robustness of a network are evaluated by the values of network connectivity and controllability. When a series of node- or edge-removal attacks is applied to the network, the values change respectively. The percolation theory [20] implies that the largest connected component(LCC) plays an important role in maintaining the network structure. So the connectivity robustness measure can be calculated based on the changes of the proportion of LCC [14]. As for controllability robustness, the changes in the proportion of driver nodes are recorded as the measure.

Machine learning methods show great performance in data mining. Due to its data-driven nature, deep neural networks are capable to learn complex data features without human intervention. Deep learning such as convolutional neural network (CNN) brings tremendous development to image processing [21], which also provides a useful tool for network robustness prediction.

An adjacency matrix of network can be converted into an image with one channel, and synthetic images are generated by applying different existing synthetic network generation models. Available existing network data set resources can be found in [22]. Afterwards, the processed image data can be processed by CNNs using an image processing manner [23], [24], [25].

The structure of network data make it hard to extracting effective node features for CNN. That is, the structure of a network is generally very irregular and the data has very high dimensions, so that it does not maintain the main properties, such as translation invariance. To fit these characters, some research try to learn feature represents for nodes. Specifically, lower-dimensional representations are generated from compacting higher-dimensional raw graph data, and then downstream classification or regression tasks are performed by processing the lower-dimensional representation data. Patchy-SAN [26] is a typical algorithm.

Although the CNN-based prediction approach performs well on synthetic network robustness prediction, it requires a full-knowledge of complex network. The experiments indicate that the robustness of networks is dependent on the topologies and structures, especially the real-world networks are so different in these two aspects above.

The current single CNN based predictors have got a certain degree of high accuracy, however, experiments on real-world networks show that the level of error values do not meet expectations. In this paper, we propose a new method, named Real-RP, to predict the robustness of real-world networks. We firstly classify the topology of real-world networks into nine refined network topology categories, so that the downstream task can obtain more topology knowledge of the network. Then a corresponding regressor is applied in predicting the robustness of the input network. Compared with human-synthesized networks, real-world networks have less prior knowledge and indistinct features. Thus, our contributions also include proposing efficient classification criterion and accurately estimating the unknown networks. With fundamental of the right estimation, the robustness of real-world networks are accurately predicted.

The following content is organized as follows. Section II reviews the measures of network connectivity and controllability robustness against destructive node-removal attacks. Section III introduces the details of the method Real-RP. Section IV presents experimental results with analysis and comparison. Section V summarizes the investigation.

II. ROBUSTNESS OF NETWORK

This section describes the mathematical model of connectivity robustness and controllability robustness. Connectivity robustness records the changes of connectivity under a series of malicious attacks, while controllability robustness records the changes of controllability. Malicious attacks include nodes or edges removal. In this paper, we set node removal as default. In the process of nodes removal, we select each node randomly to be delete till there is only one node left.

A. CONTROLLABILITY ROBUSTNESS

Given a time-invariant networked system, which is described as $\dot{x} = Ax + Bu$. A and B are constant matrix with tunable dimensions. x is system state vector and u is the outer control input. The system is state controllable if and only if matrix $[B\ AB\ A^2B\ \ldots\ A^{N-1}B]$ has full rank, where N is the dimension of A, as well as the size of the networked system. If state vector x can be driven from any initial state to any desired state in the state space by a suitable control input u within finite time, we call the system state controllable. The concept of structural controllability is generalized from controllability, which is estimated by two parameterized matrices A and B. In a network system, if there are specific parameter values of A and B that can ensure the parametric system be state controllable, then the system is structurally controllable.

We applied the fraction of driver nodes, n_D as the measure of controllability for a networked system, which is defined by the following equation:

$$n_D = \frac{N_D}{N} \tag{1}$$

where N_D represents the number of driver nodes, N represents the size of the networked system. N_D is related to the maximum matching of a directed network [27]. As a result, the minimum number of driver nodes N_D can be precisely calculated by equation:

$$N_D = max(1, N - |E^*|) \tag{2}$$

where $|E^*|$ is the number of edges in the maximum matching E^*. However, in a undirected network, edge matching nodes could not be recognized. Consequently, maximum matching theory cannot be used on undirected networks. Later, exact controllability theory expends the calculation method of controllability to undirected networks. Exact controllability theory [28] shows the minimum number of needed driver nodes in an undirected network can be computed by:

$$N_D = max(1, N - rank(Adj)) \tag{3}$$

In which, Adj is the adjacency matrix of the network, $rank(A)$ represents calculating the rank of the matrix A.

Then, we calculate the controllability robustness by the equation:

$$R_{ctrl}(i) = \frac{N_D(i)}{N - i}, \quad i = 1, 2, 3 \ldots N - 1 \tag{4}$$

where i is the attacking times, N is the size of the networked system. $N_D(i)$ is the number of driver nodes while having removed i nodes. In every iteration, after removing a node, a normalized controllability value is recorded. Thus, a curve of controllability value is recorded after the whole attacking process.

B. CONNECTIVITY ROBUSTNESS

Similar to R_{ctrl}, the measure of connectivity robustness in this paper is defined by:

$$R_{lcc}(i) = \frac{N_{LCC}(i)}{N - i}, \quad i = 1, 2, 3 \ldots N - 1 \tag{5}$$

where $N_{LCC}(i)$ is the size of largest connected component while removing i nodes.

While comparing the prediction precision, we need a scalar error value. Here, the values at the corresponding positions of the two curves are subtracted, and we can get the error curve. Then we average the curve to get the error between the attacking simulation and the different methods. Processing details are as the equation:

$$\xi = \sum_{i=1}^{N^1} |R_{pred}(i) - R_{sim}(i)| \tag{6}$$

where ξ represent the error value, N^1 means the size of sampled robustness curve, R_{pred} and R_{sim} represents the predicted robustness and the attacking simulation robustness, respectively.

III. ROBUSTNESS PREDICTOR

In this section, we briefly reviews the predictors PCR and Patchy-SAN for network robustness prediction.

In PCR [24], a VGG-based CNN structure with several convonlutional layers and full connected layers is designed to process network adjacency matrices, which can be converted to gray-scale images and then used directly as the input to CNN. And the last fully-connected layer map the features into a vector as output of the whole model. PCR updates parameters in the model through supervised learning, which has high generalization in domain tasks. However, PCR just uses very little knowledge of the input network. If more network knowledge can be made better use of, such as network typologies, logical boundaries and so on, the prediction can be more precise. On this basis, we consider the effect of prior knowledge and add it to robustness prediction of real-world networks.

For complex network data have distinguished continuous and discrete attributes that are different from general image data. Patchy-SAN [26] here is proposed to learn a low dimensional feature representations from high dimensional network. That is to transform macroscopical structural and other node level information into low dimensional data, which can be easily proposed by downstream CNN models.

In Patchy-SAN, *SAN* represents three main procedures: sample, assembly, normalization.

First, the nodes with greater priority are utilized for the following processing steps instead of all nodes. All nodes are arranged in descending order according to an importance labeling procedure. Then a part of all nodes are selected to be processed. Each of selected nodes is assembled to a local sub-network with the same size. For each node, a similar breadth-first traversal method is used to find the nodes with higher labeling values as the neighborhood in sub-network. Later, the normalization procedure is applied on every sub-network, after which every node in sub-network is assigned a label, so that all nodes can be ranked according to their significance. Last but not least, the normalized sub-networks with same data structure can be efficiently processed by downstream tasks.

Straightforward implementations of PCR and Patchy-SAN for feature extractions are not sufficient in real-world network predictions. Real-RP is designed for predicting robustness of real-world networks, which is hard for the two algorithms above. Given some prior knowledge, real-world networks can be firstly classified into several categories, and then the connectivity robustness can be predicted using the corresponding predictor. Given several network types that are commonly used, the users are able to prepare a specific CNN predictor for each type, other than treat these networks in some general forms. For a classify task, the larger the distances between categories are, the easier it is to classify them. However, sometimes one network may be similar with more than one typologies. The big distance between different typologies will have a bad impact on finding the most similar category. In Real-RP, we apply nine different network typologies as different categories to guarantee our classifier can classify networks into the most similar categories. Here, Real-RP are shown in the pseudo-code are as Alg.1.

```
Algorithm 1 Predict Real-World Network Robustness
Input: $A$: adjacency matrix; $C_0$: well trained classifier;
    $Pred_{1...N_c}$: well trained predictors for $N_c = 9$ classes;
    $Pred_{all}$: well trained predictors used for networks difficult to classify;
Output: robustness curve $R$;
initial $category = 0$ and $score = 0$;
compute classification probability $probabilities = classifier(A)$;
$category = ArgMax(probabilities)$;
$score = Max(probabilities)$;
if $score < 0.8$ then
    $R = Pred_{all}(A)$;
else
    $R = Pred_{category}(A)$;
end if
return R
```

The classifier and predictors are all implemented by CNN models and their structures are similar. The CNN structure is like Fig.1, where classifier and predictors have common convolution operators. But the last layers are different. For a classifier, a SoftMax function is arranged as the task worker. And for a predictor, the last layer is a fully connected layer, which reshapes the feature information to a desired vector. More details are shown in the Table.1.

There are seven convolution blocks and each block contains a convolutional layer followed by a ReLU($F(x) = max(0, x)$) as the activation function. Note that in the last convolution block there is one more convolutional layer. After every convolutional layer is a max pooling layer. The pooling layers reduce the dimensions and retain the important feature from the input to the next layer. Since network adjacency matrix contains sparse values between 0 and 1 and only have one channel, max pooling is used, which works well especially when the image background is dark. Following the 7 convolution blocks, two fully-connected layers are reconfigured to process the output. In the supervised training, the mean-squared error between the predicted robustness and the simulation value is employed as the loss function as the following equation.

TABLE 1. CNN Structure Details.

Blocks	Layer	Kernel Size	Stride	Output Channel
Block 1	Conv2D-1	7x7	1	64
	MaxPooling-1	2x2	2	64
Block 2	Conv2D-2	5x5	1	64
	MaxPooling-2	2x2	2	64
Block 3	Conv2D-3	3x3	1	128
	MaxPooling-3	2x2	2	128
Block 4	Conv2D-4	3x3	1	128
	MaxPooling-4	2x2	2	128
Block 5	Conv2D-5	3x3	1	256
	MaxPooling-5	2x2	2	256
Block 6	Conv2D-6	3x3	1	256
	MaxPooling-6	2x2	2	256
Block 7	Conv2D-7	3x3	1	512
	Conv2D-8	3x3	1	512
	MaxPooling-7	2x2	2	512

$$\mathcal{L} = \frac{1}{N}\sum_{i=0}^{N-1} ||\hat{R(i)} - R(i)|| \tag{7}$$

where N represents the size of network, $\hat{R(i)}$ is the predicted robustness while $R(i)$ is the simulated robustness. $||\cdot||$ represents the Euclidean norm.

IV. EXPERIMENTAL STUDIES

We designed a series of experiments to verify the improvement in predicting the connectivity robustness and controllability robustness of real-world networks. The model are trained on synthetic networks and then tested on both synthetic networks and real-world networks, respectively. We applied nine types of synthetic network models, including the Erdos-Renyi (ER)random-graph [29], Barabasi, Albert-Laszlo (BA) scale-free [30], [31], generic scale-free (SF) [32], onion-like generic scale-free(OS) [14], Newman-Watts small-world(SW-NW) [33], Watts-Strogatz small-world(SW-WS) [5], q-snapback(QS) [34], random triangle(RT) [35] and random hexagon (RH) [35] networks.

Specifically, the nine types of synthetic networks are generated by corresponding mathematical models. ER network is generated based on ER random model, whose basic idea to connect each pair of N nodes with probability P until the network has enough edges as settings. BA and SF network are both follow power law distribution. Differently, a BA network is generated according to the preferential attachment scheme [30], while a SF network is generated according to a series of predefined weights for every node. $w_i = (1 + \mu)^{-\sigma}$,where $i = 1, 2, \ldots, N, \sigma \in [0, 1)$ and $\mu \ll N$. Here, N represents the number of network to be generated. Then, pick two nodes i, j with a probability proportional to their weights as the source and target nodes for each edge.

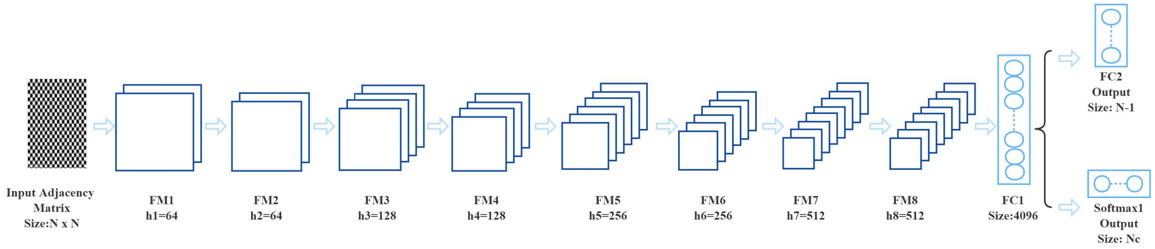


FIGURE 1. CNN structures in Real-RP.Input is adjacency matrix of a network with N nodes, after which are total 8 feature maps(FM) processed by convolutional layers. After FMs is a special layer for downstream task. For classifier, the last layer is a softmax layer, otherwise, it is a fully connected layer. Here $N_c = 9$, represents the number of network classes.

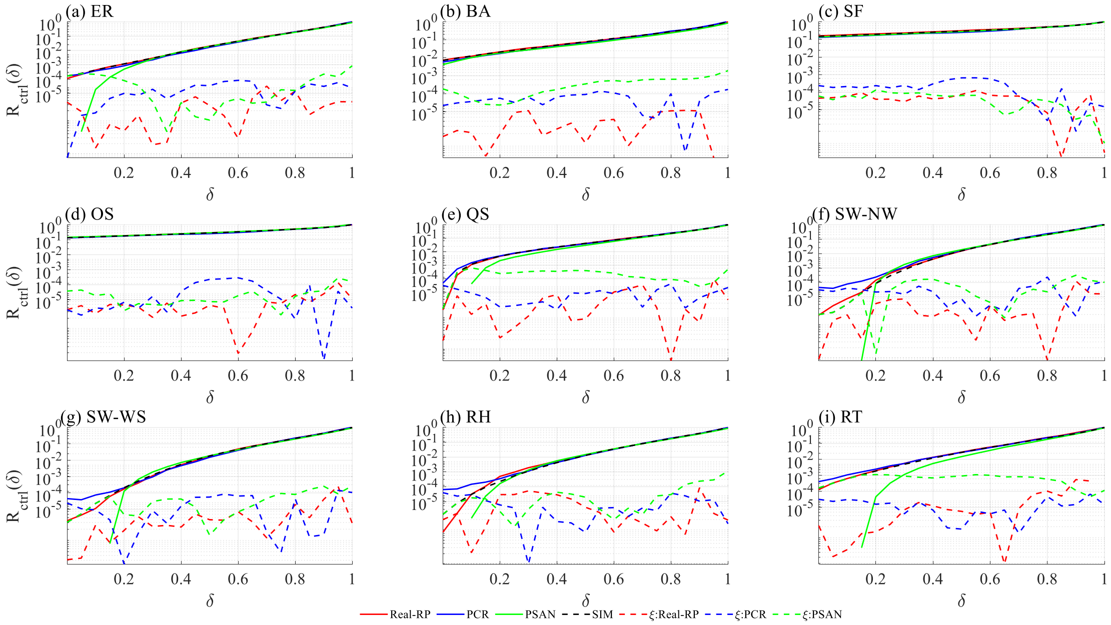


FIGURE 2. Comparison of prediction results using Real-RP, PCR, and PATCHY-SAN, for controllability robustness of directed synthetic networks(N = 500) under RA.

The resulting SF network follows the power-law distribution k^{-r}, where k represents the average degree of the aiming network, $\gamma = 1 + (1/\sigma)$, which is independent of μ. In this article, σ is set to 0.999 such that the power-law distribution has a reasonable scaling exponent $\gamma = 2.001$. As for OS network, which is generated based on an SF and carried out rewiring operations for $2{*}N$ times towards assortativity maximization. Since the rewiring operations will never change the degree of nodes, OS network also follows the power law. There are two small-world models to generate networks with small-world network feature, SW-NW and SW-WS. The two models both start producing an N-node loop, which has two connected nearest-neighbors, that is, a node i will connect to nodes $i-1, i+1, i-2, i+2$ via edges $e_{i-1,i}, e_{i,i+1}, e_{i-2,i}, e_{i,i+2}$. In the next step, SW-NW adds other edges without removing existing edges [33], while SW-WS will remove existing edges and then add new edges as well as rewiring operations [5]. In QS network, there is a main backbone chain with multiple snap-back edges [34]. Here, there is only one layer, that is $r = 1$. The out-degree of the ith node $d_{out}(i), i = 1, 2, \dots, N$ is calculated by Eqa.8:

$$d_{out}(i) = \begin{cases} 1, & i = 1, 2, \dots r \\ 1 + \lfloor \frac{i-1}{r} \rfloor \cdot q, & i = r+1, \dots, N-1 \\ \lfloor \frac{i-1}{r} \rfloor \cdot q, & i = N \end{cases} \quad (8)$$

RT and RH are both generated according to Henneberg increasing mechanism. What is different is that RT is made of randomly generated triangles, while RH is made of hexagons.

In the following experiments, while training the classifier, we generated 1000 samples for every type of network, which is total 9000 samples. For each sample, network adjacency matrix is data X and their topology categories is the label Y. In this paper, N_c is set as 9, indicating that the classification result is a 9-dimensional vector, which represents the probability of being classified into each category. A SoftMax operator is applied as the last layer of the classifier, whose output are continuous values from 0 to 1. Here we set a threshold $\theta = 0.8$. Networks with a maximum classification

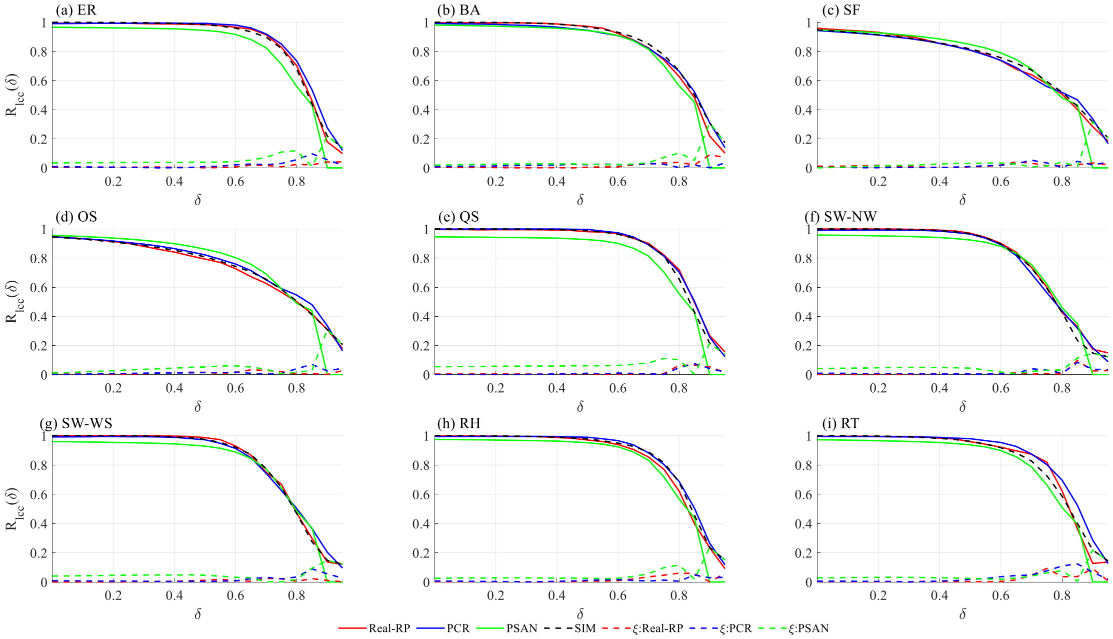


FIGURE 3. Comparison of prediction results using Real-RP, PCR, and PATCHY-SAN, for connectivity robustness of directed synthetic networks(N = 500) under RA.

TABLE 2. Comparison of average prediction errors on directed networks among Real-RP, PCR and Patchy-SAN, where N=500. And the signs under Real-RP data are results of Kruskal-Wallis H-test results. '+' denotes that Real-RP significantly outperforms the other two algorithms; '≈' denotes that no significant difference between two tested algorithms; '−' denotes that Real-RP performs significantly worse than other two algorithms.

Average Prediction Error ξ		ER	BA	SF	OS	QS	SW-NW	SW-WS	RH	RT
Controllability Robustness of Directed Synthetic Networks under RA	Real-RP	**0.0192**	**0.0238**	**0.0270**	**0.0268**	**0.0168**	**0.0172**	**0.0175**	**0.0202**	**0.0226**
		+ +	+ +	+ +	+ +	+ +	+ +	+ +	+ +	≈ +
	PCR	0.0229	0.0289	0.0343	0.0295	0.0191	0.0211	0.0202	0.0223	0.0240
	PATCHY-SAN	0.0292	0.0439	0.0494	0.0515	0.0322	0.0241	0.0231	0.0260	0.0498
Connectivity Robustness of Directed Synthetic Networks under RA	Real-RP	**0.0376**	**0.0436**	**0.0644**	**0.0616**	**0.0276**	**0.0312**	**0.0258**	0.0419	**0.0469**
		≈ +	≈ +	≈ +	≈ +	+ +	+ +	+ +	+ +	≈ +
	PCR	0.0378	0.0463	0.0662	0.0635	0.0329	0.0433	0.0375	**0.0355**	0.0472
	PATCHY-SAN	0.0724	0.0667	0.0833	0.0833	0.0878	0.0830	0.0769	0.0644	0.0659

probability greater than θ are considered classifiable and is processed by the regressor of the corresponding category. Otherwise, the network will be processed by a common regressor.

For each regressor, we prepare 6000 samples as training data. In contrast to the training process of classifier, the label Y is network robustness curve. While computing the robustness curves, we adopt random criterion to attack every sample, which means that for each iteration, we randomly choose one node to remove from the network, then calculate the connectivity robustness and controllability robustness. For each network sample with N nodes, we attack $N-1$ times so that there is only one isolated node eventually. Finally, we obtain the robustness curve as labels.

For both classifier and regressor, all networks in training data are with 500 nodes and average degree is between 3.5 and 5.

All experiments are performed on a PC Intel (R) Core i7-8750H CPU @ 2.20GHz, which has memory (RAM) 16 GB with running Windows 10 Home 64-bit Operating System. And for all CNN models, the programs are deployed on computing platform with a GPU Tesla V100-16G.

A. PREDICTING ON SYNTHETICS

We predict both connectivity robustness and controllability robustness of networks including directed and undirected networks by PCR [24], Patchy-SAN [26] and Real-RP. Fig.2 and Fig.3 show the connectivity robustness and controllability robustness prediction results on directed networks. Fig.4 and Fig.5 show the prediction performance on undirected networks.

In all figures, nine types of networks mentioned above are tested, δ represents the proportion of attacked nodes to total N nodes.$R_{lcc}(\delta)$ and $R_{ctrl}(\delta)$ means predicted connectivity robustness and controllability robustness while removing nodes with ratio δ, respectively.

In Fig.2, the above algorithms all fit out the trend of the curve well. Very low levels of errors are generally observed in nine different networks in Real-RP's results. PCR performs relatively poor in SF and OS, which have more or

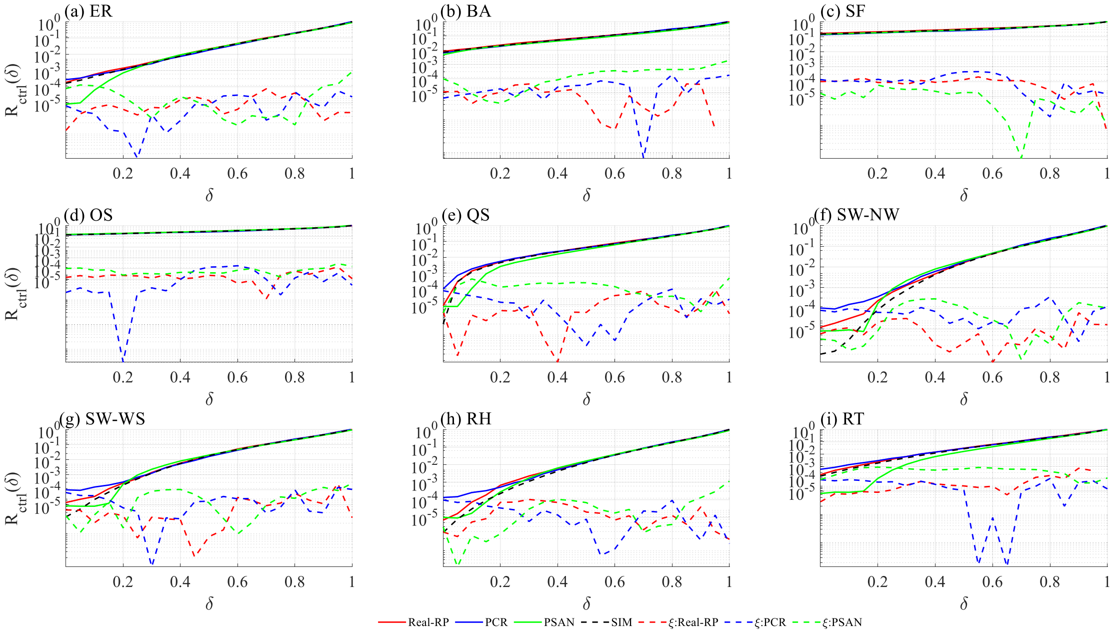


FIGURE 4. Comparison of prediction results using Real-RP, PCR, and PATCHY-SAN, for controllability robustness of undirected synthetic networks(N = 500) under RA.

TABLE 3. Comparison of average prediction error on undirected synthetic networks among Real-RP, PCR and Patchy-SAN, where *N*=500.

Average Prediction Error ξ		ER	BA	SF	OS	QS	SW-NW	SW-WS	RH	RT
Controllability Robustness of Undirected Synthetic Networks under RA	Real-RP	**0.0201**	**0.0246**	**0.0290**	**0.0276**	**0.0178**	**0.0188**	**0.0184**	**0.0218**	**0.0243**
		+ +	+ +	+ +	+ +	+ +	+ +	+ +	+ +	≈ +
	PCR	0.0228	0.0278	0.0318	0.0287	0.0206	0.0240	0.0209	0.0242	0.0258
	PATCHY-SAN	0.0279	0.0417	0.0481	0.0523	0.0285	0.0255	0.0231	0.0264	0.0465
Connectivity Robustness of Undirected Synthetic Networks under RA	Real-RP	**0.0371**	**0.0441**	**0.0638**	**0.0615**	**0.0269**	**0.0337**	**0.0287**	0.0421	**0.0476**
		+ +	+ +	≈ +	+ +	+ +	+ +	+ +	+ +	≈ +
	PCR	0.0396	0.0442	0.0654	0.0640	0.0360	0.0423	0.0368	**0.0375**	0.0495
	PATCHY-SAN	0.0595	0.0510	0.0720	0.0743	0.0758	0.0739	0.0691	0.0506	0.0527

less heterogeneity characteristics. As for Patchy-SAN, it is of slightly less accurate in predicting networks BA, QS and RT, on which the error curve of Patchy-SAN goes higher than other two algorithms. Connectivity robustness and controllability robustness are both robustness measures from different angles. Consequently, the two have different going trends. For controllability robustness, it changes on a mall variation via different attacking approaches, while connectivity robustness has more dramatic magnitude change. Fig.3 shows the prediction results on connectivity robustness. The $R_{lcc}(\delta)$ curve has an downward trend overall. In the middle and later stages of the curve, the slope gets higher, making it harder to predict. As a result, the errors of all three algorithms has improved more or less. Nevertheless, the error curve of Real-RP still goes lower than PCR and Patchy-SAN.

Table.2 shows the average predicting errors. Besides, to verify the reliability of data in a statistical sense, Kruskal-Wallis H-test is also carried out on the predicting results [36]. In predicting controllability robustness, Real-RP significantly outperforms the other two algorithms except RT, and the error values on all nine types of networks keep in a very low level. For connectivity robustness, Real-RP has excellent performance in QS, SW-NW, SW-WS, RT. In the other five types, the error values of Real-RP are still lower, but Kruskal-Wallis H-test shows that there is no significant difference between Real-RP and PCR. Besides, not only in predicting controllability robustness, but in predicting connectivity robustness, Real-RP performs significantly better than Patchy-SAN in all types.

TABLE 4. Details of Real-world Networks.

ID	N	<K>
Real-1	503	3.01
Real-2	508	2.63
Real-3	500	2.46
Real-4	502	2.49
Real-5	502	2.25
Real-6	496	2.24
Real-7	495	2.44
Real-8	501	2.26
Real-9	495	2.44

The same experiments are carried out on undirected networks, Fig.4 and Fig.5 show the prediction performance on undirected networks of those nine types. And Table.3presents

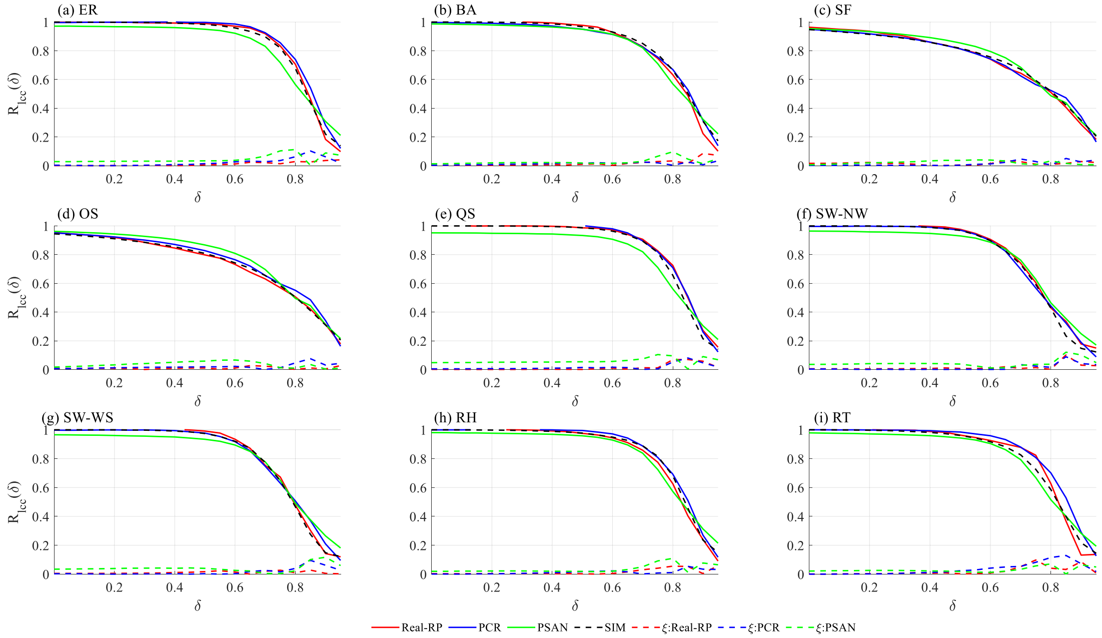


FIGURE 5. Comparison of prediction results using Real-RP, PCR, and PATCHY-SAN, for connectivity robustness of undirected synthetic networks(N = 500) under RA.

TABLE 5. Comparison of Prediction Errors among Real-RP, PCR, Patchy-SAN.

Average Prediction Error ξ on Real-world networks		Real-1	Real-2	Real-3	Real-4	Real-5	Real-6	Real-7	Real-8	Real-9
Controllability Robustness under RA	Real-RP	**0.0277**	**0.0304**	**0.0207**	**0.0310**	**0.0284**	**0.0317**	**0.0157**	**0.0263**	**0.0295**
	PCR	0.1912	0.1806	0.1513	0.1817	0.1667	0.1841	0.1637	0.1920	0.1938
	PATCHY-SAN	0.2357	0.2988	0.2515	0.2978	0.3103	0.3355	0.2972	0.3363	0.3082
Connectivity Robustness under RA	Real-RP	**0.0505**	**0.0635**	**0.0750**	**0.0780**	**0.0861**	**0.0899**	**0.0932**	**0.0990**	**0.1014**
	PCR	0.1068	0.1298	0.1116	0.1047	0.1301	0.1161	0.1005	0.1529	0.1073
	PATCHY-SAN	0.4536	0.4827	0.4323	0.4197	0.3819	0.4452	0.4114	0.4763	0.3828

TABLE 6. Feature values of synthetic graphs.

category	ho	avg_bet	avg_cc	avg_path
er	1.23	1624.08	0.01	4.39
ba	2.19	1291.34	0.04	3.77
sf	7.49	634.63	0.29	2.90
so	7.47	637.15	0.27	2.94
qs	1.38	8261.30	0.01	86.40
swnw	1.12	2289.53	0.10	5.59
swws	1.14	2115.13	0.07	5.25
rh	1.17	1531.71	0.01	4.07
rt	1.36	1610.18	0.15	4.23
avg	2.73	2243.89	0.09	11.95

TABLE 7. Features of four real-world network. Here to varify the class result, each feature is compared with the average value of synthetic graph features. '+' means the feature value is bigger than the average value of synthetic graphs, '–' means the feature value is smaller than that.

id	category	ho	avg_bet	avg_cc	avg_path
1	sf	88.86	6.22	0.01	2.50
		+	-	-	-
2	sf	44.14	13.07	0.01	2.85
		+	-	-	-
3	sw-ws	2.40	32.98	0.08	3.78
		+	-	+	-
4	sw-nw	236.64	2.06	0.01	1.80
		+	-	-	-

the prediction errors between prediction results and true values. In most cases, Real-RP could obtain the lowest level of error and outperforms significantly other two algorithms for comparison.

The above experiments show that Real-RP has advantages in robustness prediction not only for directed networks but also for undirected networks.

B. PREDICTION ON REAL-WORLD NETWORKS

After having trained models on synthetics, we tested predicting accuracy on real-world networks. Here, we randomly collected nine real-world networks from Reddit-multi datasets [22]. These real-world networks are of about 500 nodes, which have completely different typologies. Table.4 shows the details of selected 9 real-world networks, including network scales and network average degrees. Notice that all real-world networks have different scales. In the process, we randomly remove or add several nodes to adjust these networks' scale to $N = 500$ exactly. Table.5 shows the error values of real-world networks of Real-RP, PCR, Patchy-SAN. On the 9 real-world networks, we can

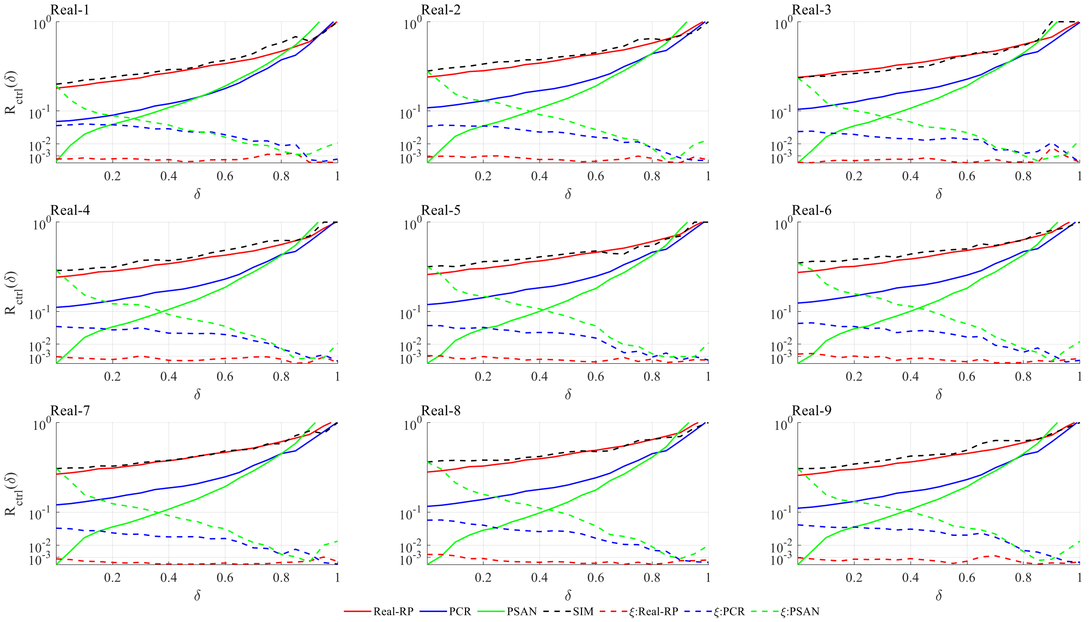


FIGURE 6. Comparison of prediction results using Real-RP, PCR, and PATCHY-SAN, for controllability robustness of real-world networks(N = 500) under RA.

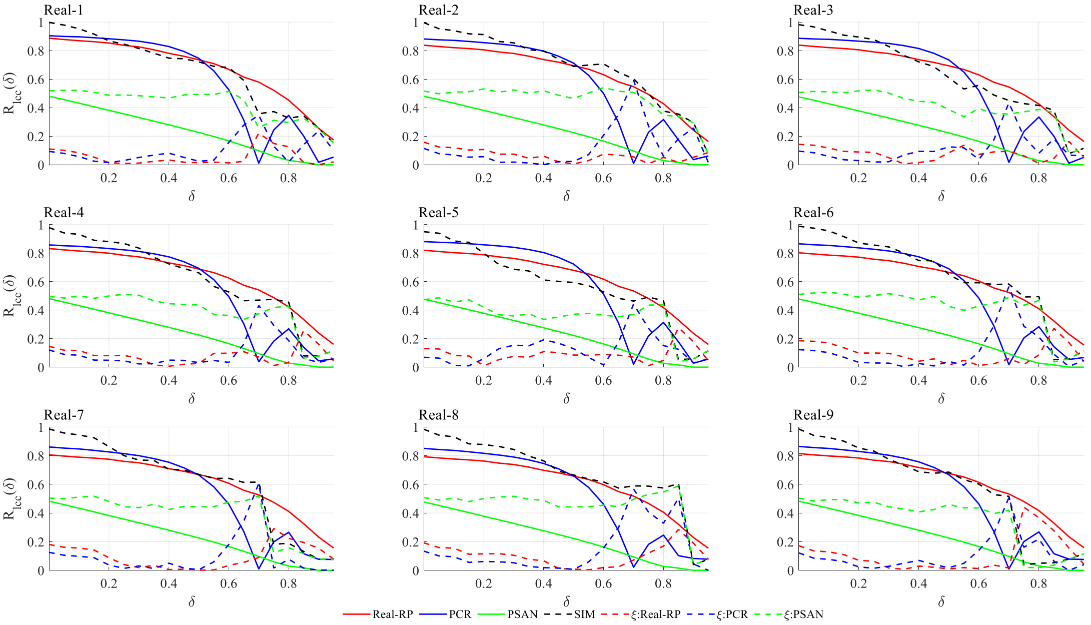


FIGURE 7. Comparison of prediction results using Real-RP, PCR, and PATCHY-SAN, for connectivity robustness of real-world networks(N = 500) under RA.

see that Real-RP can stably predict controllability robustness with very low errors. In predicting connectivity robustness, the error is relatively high, as is the case with the synthetic graphs.

Compared to the traditional attack simulations, Real-RP significantly reduce computing time. The average run time of attack simulations is 11.36s on experimental real networks, while Real-RP could finish the computing process in 0.24s.

C. VERIFY CLASSIFICATION

Adding more network categories bring higher precision predicting the robustness of real world networks. In this paper, the work aims to improve the capability to classify real-world networks. On the basis of this, we verify the classify results by comparing the feature models, which include four features, heterogeneity(*ho*), average clustering coefficient (*avg_cc*), average betweeness(*avg_bet*) and average path length(*avg_path*).

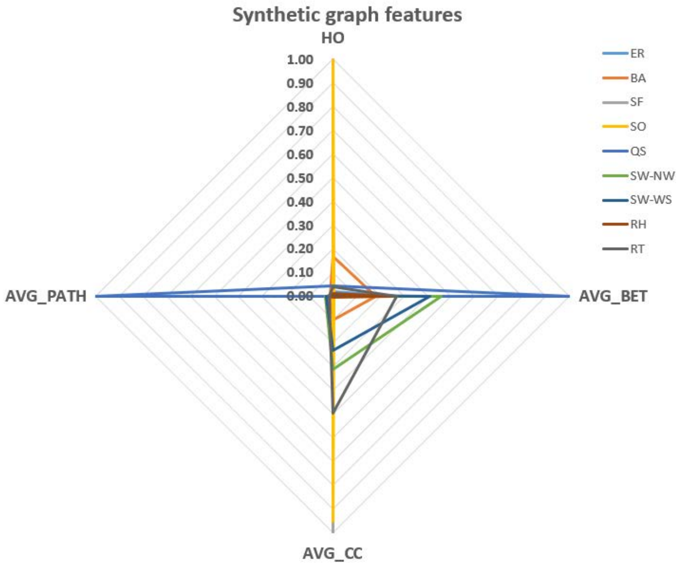


FIGURE 8. Radar graph for synthetic networks. The figure describes the numerical distribution of each network over four kinds of features.

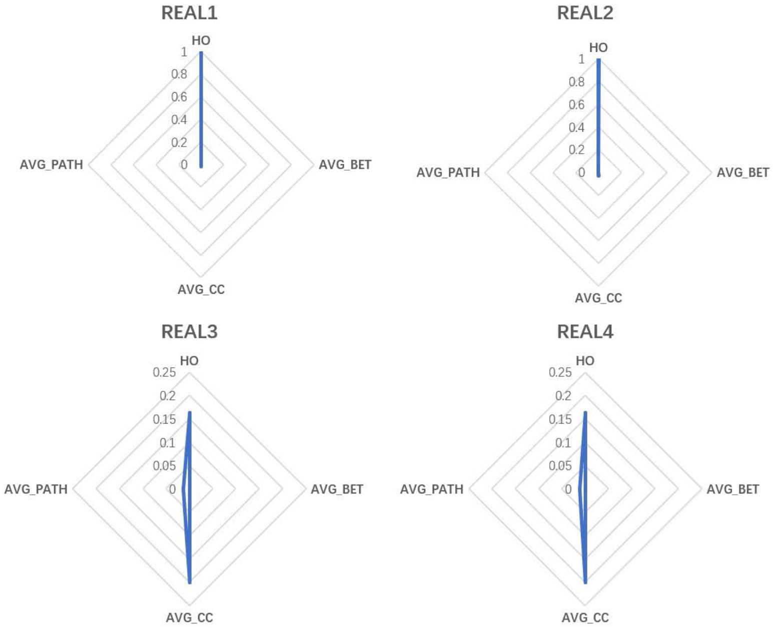


FIGURE 9. Radar graph for four randomly selected real networks. Each dimension of the graph represents a normalized feature. ***REAL*1** and ***REAL*2** are both classified into SF. ***REAL*3** is classified into SW-WS. ***REAL*4** is classified into SW-NW.

We posteriorly calculate the features of all kinds of synthetic graphs generated by mathmatical models to conclude the common feature models. 1350 samples for each kind of graphs are calculated. The normalized posterior feature model of synthetic graph are as the following Fig.8. And the feature values are presented as Table.6.

Here SF and BA are of obviously big HO, while SW should have little average path length. However, for a SF network, the calculated feature model is the sufficient condition of being classified into SF by our classifier. That is, we can calculate the features of a given real-world network to confirm that this network is classified rightly. But a SF generated by mathematical model will have more features. For example, SF or BA network both have power law distribution with high heterogeneity, and a large part of real-world networks also show this property and can be classified into SF or BA.

Then we select randomly four real-world networks, $REAL1$ and $REAL2$ classified into SF, $REAL3$ is classified into SW-WS, and *REAL*4 is classified into SW-WS. Note that the four networks are different from the 9 real-world networks above. Here we posteriorly calculate their features after classification to compare them with the average value of synthetic networks. The radar graph of the features of the four network is as Fig.9. We divide the feature values into two parts, bigger than the average value of all synthetic networks and less than that. If a feature both of real-world network and synthetic networks of corresponding category performs bigger than the average value, then we consider the classification result to be positive. As the following Fig.7, *REAL*1 and *REAL*2 both have much higher *ho* than other categories, which is consistent with existing knowledge [30], [31], [32]. Average path lengths of *REAL*3 and *REAL*4 performs much lower than the average value of synthetic networks, which means the two networks represent relatively obvious small-world characteristics.

V. CONCLUSION

Many deep learning methods have been used in predicting numerical characteristics and classification. Such approaches have achieved great process on decreasing time consuming. Recently predicting robustness of networks under attackings like nodes or edges removal has made some progress. However, existing mathematical models can partially simulates the features of real-world networks. Some works that make good use of deep learning method performs well on synthetic networks, which can not get satisfactory results on real-world networks. This paper focuses on predicting robustness of real-world networks and proposes a new method, Real-RP. More network categories were set and more corresponding predictors are added into the predicting framework. Such improvement reduces the prediction error on real-world networks. Since the prediction is based on network topology knowledge, which is obtained by classification. Classier on the framework was tested posteriorly. And experiment results show that the real-world networks were classified in a more suitable category and performs better at predicting results.

REFERENCES

[1] A.-L. Barabási, *Network Science*. Cambridge, U.K.: Cambridge Univ. Press, 2016.

[2] M. E. Newman, *Networks: An Introduction*. London, U.K.: Oxford Univ. Press, 2010.

[3] G. Chen, X. Wang, and X. Li, *Fundamentals of Complex Networks: Models, Structures and Dynamics*, 2nd ed. Hoboken, NJ, USA: Wiley, 2014.

[4] G. Chen and Y. Lou, *Naming Game: Models, Simulations and Analysis*. Cham, Switzerland: Springer, 2019.

[5] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[6] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," in *The Structure and Dynamics of Networks*. Princeton, NJ, USA: Princeton Univ. Press, 2011, pp. 195–206.

[7] J. M. Hofman, A. Sharma, and D. J. Watts, "Prediction and explanation in social systems," *Science*, vol. 355, no. 6324, pp. 486–488, 2017.

[8] H. Ebel, L.-I. Mielsch, and S. Bornholdt, "Scale-free topology of e-mail networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 66, no. 3, Sep. 2002, Art. no. 035103.

[9] S. Wandelt, X. Shi, and X. Sun, "Estimation and improvement of transportation network robustness by exploiting communities," *Rel. Eng. Syst. Saf.*, vol. 206, Feb. 2021, Art. no. 107307.

[10] Y. Guo, "To focus on improving power system reliability—A pondering over the east north-America major blackout," *Autom. Electr. Power Syst.*, vol. 27, no. 19, pp. 1–5, 2003.

[11] L. Cuadra, S. Salcedo-Sanz, J. D. Ser, S. Jiménez-Fernández, and Z. W. Geem, "A critical review of robustness in power grids using complex networks concepts," *Energies*, vol. 8, no. 9, pp. 9211–9265, 2015.

[12] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, "Attack vulnerability of complex networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 65, no. 5, May 2002, Art. no. 056109.

[13] B. Shargel, H. Sayama, I. R. Epstein, and Y. Bar-Yam, "Optimization of robustness and connectivity in complex networks," *Phys. Rev. Lett.*, vol. 90, no. 6, Feb. 2003, Art. no. 068701.

[14] C. M. Schneider, A. A. Moreira, J. S. Andrade, Jr., S. Havlin, and H. J. Herrmann, "Mitigation of malicious attacks on networks," *Proc. Nat. Acad. Sci. USA*, vol. 108, no. 10, pp. 3838–3841, 2011.

[15] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Control centrality and hierarchical structure in complex networks," *PLoS ONE*, vol. 7, no. 9, Sep. 2012, Art. no. e44459.

[16] A. Bashan, Y. Berezin, S. V. Buldyrev, and S. Havlin, "The extreme vulnerability of interdependent spatially embedded networks," *Nature Phys.*, vol. 9, pp. 667–672, Aug. 2013.

[17] Y.-D. Xiao, S.-Y. Lao, L.-L. Hou, and L. Bai, "Optimization of robustness of network controllability against malicious attacks," *Chin. Phys. B*, vol. 23, no. 11, Nov. 2014, Art. no. 118902.

[18] S. Wang and J. Liu, "Robustness of single and interdependent scale-free interaction networks with various parameters," *Phys. A, Stat. Mech. Appl.*, vol. 460, pp. 139–151, Oct. 2016.

[19] S. Wang, J. Liu, and Y. Jin, "A computationally efficient evolutionary algorithm for multiobjective network robustness optimization," *IEEE Trans. Evol. Comput.*, vol. 25, no. 3, pp. 419–432, Jun. 2021, doi: 10.1109/TEVC.2020.3048174.

[20] M. Li, R.-R. Liu, L. Lü, M.-B. Hu, S. Xu, and Y.-C. Zhang, "Percolation on complex networks: Theory and application," *Phys. Rep.*, vol. 907, pp. 1–68, Apr. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0370157320304269

[21] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Oct. 2014.

[22] K. Kersting, N. M. Kriege, C. Morris, P. Mutzel, and M. Neumann. (2016). *Benchmark Data Sets for Graph Kernels*. [Online]. Available: http://graphkernels.cs.tu-dortmund.de

[23] Y. Lou, Y. He, L. Wang, K. F. Tsang, and G. Chen, "Predicting the robustness of undirected network controllability," in *Proc. 39th Chin. Control Conf. (CCC)*, Jul. 2020, pp. 1–6.

[24] Y. Lou, Y. He, L. Wang, and G. Chen, "Predicting network controllability robustness: A convolutional neural network approach," *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 4052–4063, May 2022, doi: 10.1109/TCYB.2020.3013251.

[25] Y. Lou, R. Wu, J. Li, L. Wang, and G. Chen, "A convolutional neural network approach to predicting network connectedness robustness," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 4, pp. 3209–3219, Oct. 2021, doi: 10.1109/TNSE.2021.3107186.

[26] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 2014–2023.

[27] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, pp. 167–173, May 2011.

[28] Z. Yuan, C. Zhao, Z. Di, W.-X. Wang, and Y.-C. Lai, "Exact controllability of complex networks," *Nature Commun.*, vol. 4, no. 1, p. 2447, Dec. 2013.

[29] P. Erdős and A. Rényi, "On the strength of connectedness of a random graph," *Acta Math. Acad. Sci. Hungarice*, vol. 12, nos. 1–2, pp. 261–267, 1961.

[30] A. L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, pp. 509–512, Sep. 1999.

[31] A.-L. Barabási, "Scale-free networks: A decade and beyond," *Science*, vol. 325, no. 5939, pp. 412–413, Jul. 2009.

[32] K.-I. Goh, B. Kahng, and D. Kim, "Universal behavior of load distribution in scale-free networks," *Phys. Rev. Lett.*, vol. 87, no. 27, Dec. 2001, Art. no. 278701.

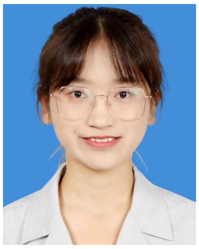
[33] M. E. J. Newman and D. J. Watts, "Renormalization group analysis of the small-world network model," *Phys. Lett. A*, vol. 263, nos. 4–6, pp. 341–346, Dec. 1999.

[34] Y. Lou, L. Wang, and G. Chen, "Toward stronger robustness of network controllability: A snapback network model," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 9, pp. 2983–2991, Sep. 2018.

[35] G. Chen, Y. Lou, and L. Wang, "A comparative study on controllability robustness of complex networks," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 66, no. 5, pp. 828–832, May 2019.

[36] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *J. Amer. Stat. Assoc.*, vol. 47, no. 260, pp. 583–621, 1952.

RUIZI WU received the B.E. degree from Southwest Jiaotong University, in 2020. He is currently pursuing the graduate degree with the School of Computer Science, Sichuan Normal University. His research interests include complex networks, evolutionary computation, and machine learning.



JIE HUANG received the B.E. degree from Shenyang Jianzhu University, in 2020. She is currently pursuing the graduate degree with the School of Computer Science, Sichuan Normal University. Her research interests include complex networks, evolutionary computation, and machine learning.

ZHUORAN YU received the B.E. degree from Sichuan Normal University, in 2020, where he is currently pursuing the graduate degree with the School of Computer Science. His research interests include complex networks and evolutionary computation.

JUNLI LI received the Ph.D. degree from Zhejiang University, in 2002. He is currently a Professor with the College of Computer Science, Sichuan Normal University, Chengdu, China. His research interests include complex networks, evolutionary computation, and machine learning.