A Convolutional Neural Network Approach to Predicting Network Connectedness Robustness

Yang Lou, *Member, IEEE*, Ruizi Wu, Junli Li, Lin Wang, *Senior Member, IEEE*, and Guanrong Chen, *Life Fellow, IEEE*

***Abstract*—To quantitatively measure the connectedness robustness of a complex network, a sequence of values that record the remaining connectedness of the network after a sequence of node- or edge-removal attacks can be used. However, it is computationally time-consuming to measure the network connectedness robustness by attack simulations for large-scale networked systems. In the present paper, an efficient method based on convolutional neural network (CNN) is proposed to train for estimating the network connectedness robustness. The new approach is motivated by the facts that 1) the adjacency matrix of a network can be converted to a gray-scale image and CNN is very powerful for image processing, and 2) CNN has proved very effective in predicting the controllability robustness of complex networks. Extensive experimental studies on directed and undirected, as well as synthetic and real-world networks suggest that: 1) the proposed CNN-based methodology performs excellently in the prediction of the connectedness robustness of complex networks as a process; 2) it performs fairly well as the indicator for the connectedness robustness, compared to other predictive measures.**

***Index Terms*—Complex network, convolutional neural network, connectedness, robustness, prediction.**

I. Introduction

MANY real-world systems can be modeled as complex networks. The study of various complex networks is currently pervading all kinds of sciences, reaching out to engineering and technology in interdisciplinary fields. This subject attracts increasing interest and attention from research communities in computer science, statistical physics, systems engineering, applied mathematics, and biological as well as social sciences [1]–[4].

For a complex network, its connectedness is guaranteed by a sufficient number of edges that properly connect the nodes. The connectedness is necessary for the network to perform its fundamental tasks such as controllability and synchronizability, although the specific measures of these functions are mostly different. Since random failures and malicious attacks are indispensable in real-world applications, which typically destroy the connectedness of the network, it has becoming a major concerned issue to strengthen the network connectedness against such destructive failures and attacks [5]–[10]. Typically, destructive failures and attacks take place in the form of node- or edge-removals, which cause significant consequences to the network functioning or even lead to complete network crashing. In these scenarios, the ability of a network to maintain its connectedness against failures or attacks is referred to as the *connectedness robustness*, or simply the *robustness*, in this paper.

Network attacks can be classified as *random* and *targeted* attacks, which can be modeled and analyzed in computer simulations. Targeted attacks aim at removing some intentionally selected objects (e.g., a node that has the largest degree), while random attacks do such removals at random. Here, for targeted attacks it is presumed that a targeted node or edge is more crucial than other nodes or edges in maintaining the connectedness. However, evaluating the importance of nodes or edges is computationally intensive, and often practically intractable especially for large-scale networks. Conceptually, this requires to quantify the importance by some centrality measures, such as degree, betweenness, closeness, and eigenvector [11]. A selected centrality measure is used as the indicator of (nodal or edge) importance in an attack or defence strategy. Among the existing centrality measures, degree and betweenness are the most frequently used two [12], [13]. Besides centrality, other commonly-used measures of importance include neighborhood similarity [14], branch weighting [15], and structural holes [16].

From an attacker's point of view, the module-based attack strategy [17], [18] is particularly effective, which selectively attacks the inter-community nodes and edges, which are demonstrated important to maintain the connectedness among communities. Also, the damage-based attack strategy [19]

Manuscript received May 7, 2021; revised June 30, 2021; accepted August 19, 2021. Date of publication August 27, 2021; date of current version December 9, 2021. This work was supported in part by the National Natural Science Foundation of China under Grants 62002249 and 61873167, in part by the Hong Kong Research Grants Council under the GRF Grant CityU11206320, in part by the Open Project Program of the State Key Lab of CAD&CG (A2112), Zhejiang University, and in part by the Foundation of Key Laboratory of System Control and Information Processing, Ministry of Education, P. R. China under Grant Scip202103. Recommended for acceptance by Dr. Gang Yan. *(Yang Lou and Ruizi Wu contributed equally to this work.) (Corresponding authors: Guanrong Chen and Junli Li.)*

Yang Lou is with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China, and also with the Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai 200240, China (e-mail: felix.lou@my.cityu.edu.hk).

Ruizi Wu and Junli Li are with the College of Computer Science, Sichuan Normal University, Chengdu 610066, China (e-mail: vridge@foxmail.com; li.junli@vip.163.com).

Lin Wang is with the Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China, and also with the Key Laboratory of System Control and Information Processing, Ministry of Education, Shanghai 200240, China (e-mail: wanglin@sjtu.edu.cn).

Guanrong Chen is with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China (e-mail: eegchen@cityu.edu.hk).

Digital Object Identifier 10.1109/TNSE.2021.3107186

uses a measure of *damage* to describe the destruction level of an attack, where the damage is defined as the change of the largest connected component (LCC) size before and after the attack. Along this line, the (normalized) size of LCC is widely used as a measure for connectedness robustness [7]. Furthermore, it is observed that the attack-and-defend iteration process can enhance the network robustness in an evolution manner [20]. It is commonly known that onion-like structured heterogeneous networks are robust against attacks [7], [21]–[23]. In this research direction, there are extensive studies on various issues regarding network robustness, including the robustness of other types of networks such as a network of networks [24], [25] and multiplex networks [26], which found some encouraging real-world applications in e.g. power grids [27], [28].

Given fixed numbers of nodes and edges, the network robustness against various attacks can be improved by rewiring [21], [27], [29]–[33]. If there is no restriction on the number of edges, quite intuitively adding extra edges properly can enhance the robustness [34]. Spectral measures offer easy-to-access indicators for detecting the network robustness, with which meta-heuristic algorithms can be applied to optimizing the robustness [33], [35]–[39].

Regarding robustness optimization, deep neural networks provides a useful tool, which has shown powerful capability in image processing. Successful applications of deep learning techniques include network controllability robustness prediction [40]–[42] and critical node identification [43]. As a kind of effective deep neural networks [44], convolutional neural network (CNN) is able to automatically analyze inner features of a dataset and output desirable results with respect to classification or regression, without human interference.

Traditionally, the network robustness is evaluated by attack simulations, which however are extremely computationally time-consuming, especially for large-scale complex networks. The major computational cost includes: 1) searching for the node to attack, e.g., the node with maximum betweenness; 2) calculating the connectedness measure, e.g., the LCC. Both have to be calculated iteratively therefore consuming a large amount of computing resources and time. To deal with such technical problems so as to improve the computational efficiency, in this paper a CNN-based robustness predictor (CNN-RP) is proposed. The CNN-RP is used to predict the network robustness through the entire process of attacks, by computing and visualizing the size curve of (normalized) LCC against node-removal attacks, However, edge-removal attacks are very different in nature therefore will be studied elsewhere.

The design of CNN-RP is motivated by the following observations: 1) although some features and indicators (e.g., spectral measures) are reliable to describe the overall robustness, they cannot reflect the sequential details throughout the entire attack process; 2) the detailed robustness information about the process against sequential attacks may be obtained via attack simulations, which however are very time-consuming and even infeasible; 3) complex networks can be equivalently converted to gray-scale images, and CNN techniques have proved efficient in processing such images. Here, the designed CNN-RP follows the same CNN structure used in the controllability robustness predictor [40], [42], but with different objectives and functions. Compared to the controllability robustness prediction, it is more challenging to predict the connectedness robustness, since the variation of the connectedness could be higher than that of the controllability. Thus, an additional filter will be designed and used, as detailed in Subsection III-B. Extensive experimental studies demonstrate that 1) the designed CNN-RP can well predict the evolving size curves of LCC against sequential node-removals for both directed and undirected, synthetic and real-world networks, with a good generalization ability; 2) the CNN-RP not only approximates the entire attack process, but also provides a good (or even better) predictive measure compared with the classical spectral measures.

The reminder of this paper is organized as follows: Section II reviews the measure of network connectedness robustness against destructive attacks. Section III introduces the new CNN-RP. In Section IV, experimental results are presented with analysis and comparison. Finally, Section V concludes the investigation.

II. Network Robustness

In this paper, the network connectedness robustness is measured by the normalized LCC [7]. The LCC of a directed network is the largest weakly connected subnetwork, where a directed graph is *weakly connected* if it remains to be connected after all the directed edges are changed to be undirected. Two LCC-based robustness measures are used, one for the attacking process and the other for the resultant network. The former is represented by a real vector (a normalized LCC curve) while the later is represented by a real value.

Specifically, the measure of the network robustness in terms of a normalized LCC curve (NLC) is calculated by

$$s(i) = \frac{N_{\mathrm{LCC}}(i)}{N-i}\,, \quad i = 0, 1, \ldots, N-1\,, \tag{1}$$

where $N_{\mathrm{LCC}}(i)$ represents the number of nodes in the LCC, and $s(i)$ is its normalized value (NLC) obtained after a total number of i nodes have been removed from the network; N is the original number of nodes in the network before being attacked.

The overall measure of the network robustness is then calculated by

$$\bar{s} = \frac{1}{N}\sum_{i=0}^{N-1} s(i)\,. \tag{2}$$

With the above measure, for two given complex networks under the same sequential attacks, the one with a larger $\bar{s}$ value is considered having better connectedness robustness.

Now, given two NLCs, $\mathbf{s}_1 = [s_1(0), s_1(1), \ldots, s_1(N-1)]$ and $\mathbf{s}_2 = [s_2(0), s_2(1), \ldots, s_2(N-1)]$, the difference between the two curves is calculated by

$$\boldsymbol{\xi} = |\mathbf{s}_1 - \mathbf{s}_2|\,, \tag{3}$$

where $\boldsymbol{\xi} = [\xi(0), \xi(1), \ldots, \xi(N-1)]$ represents the sequential differences (or errors) between the two curves, where $\boldsymbol{\xi}(i) = |s_1(i) - s_2(i)|$.

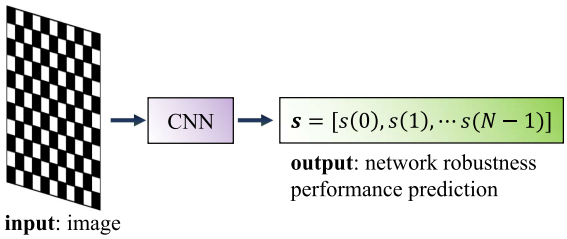

Fig. 1. The framework using CNN to predict network robustness. The input is an adjacency matrix converted image; the output is the predicted NLC curve.

Finally, the average error $\bar{\xi}$ is calculated by

$$\bar{\xi} = \frac{1}{N}\sum_{i=0}^{N-1} \xi(i)\,. \tag{4}$$

Thus, the vector $\boldsymbol{\xi}$ can be used to measure the errors of the NLC predictions throughout the attack process; while the scalar $\bar{\xi}$ measures the overall error of the NLC prediction.

III. Network Robustness Predictor

Different from the predictors for the network controllability robustness against destructive attacks [40], [42], in this paper CNN is used to predict the connectedness robustness, which turned out to have a greater variation than the controllability robustness. An illustrative example of the connectedness robustness and controllability robustness will be given later in Subsection IV-D. To deal with the large variation in the prediction, a filter is useful, which is installed in CNN-RP following the CNN output, as detailed below.

A. Convolutional Neural Network

The general framework of the proposed CNN-RP is shown in Fig. 1, where a CNN is trained for network robustness prediction. As can be seen from Fig. 2, the framework of this CNN-RP is relatively simple, which consists of several groups of convolutional layer, rectified linear units (ReLUs) and max pooling layer, where ReLU is the activation function.

The structure of the CNN-RP is shown in Fig. 2. The detailed parameter settings are given in Table I. The VGG architecture [45] is employed, which incorporates a greater network depth and a smaller kernel size. The 7 feature map (FM) processing layers are denoted as FM 1 to FM 7 respectively.

In simulations, for input of size around 1000×1000 as in the experiments reported below, the number of FM groups is set to 7, which should be set to be greater for input of larger sizes. Each FM consists of a convolutional layer, a ReLU, and a max-pooling layer. Convolutional layers are adopted here because of their efficiency in dealing with large-sized images. ReLU (with $f(x) = \max\{0, x\}$) is a widely-used activation function for $2D$ data [46]. The pooling layers reduce the dimensions from the input to the next layer. Since the interest of images in this work is only in the lighter pixels, max pooling is used, which works well especially when the image background is dark. Following the 7 FMs, two fully-connected layers are configurated to process the output.

The mean-squared error between the predicted NLC and the true NLC is employed as the loss function, as follows:

$$\mathcal{L} = \frac{1}{N}\sum_{i=0}^{N-1} ||\hat{s}(i) - s(i)||\,, \tag{5}$$

where $\hat{s}(i)$ is the i-th value of the predicted NLC, and $s(i)$ is the i-th value of the true NLC by simulation; $||\cdot||$ represents the Euclidean norm. The training process for CNN-RP aims to minimize Eq. (5).

B. Filter for LCC-Curves

Due to the nature of data-driven algorithms, it is possible that CNN outputs some logically unreasonable data. For instance, the number of nodes of LCC in a network under attacks must be monotonically non-increasing, but the output of CNN-RP may violate this principle. To regulate the output of CNN-RP, a filter is used, which is designed based on existing prior knowledge. In configuration, the upper and lower bounds of the LCC size are imposed onto the output of CNN-RP, and logically unreasonable data are replaced by interpolated values. The filter consists of two parts, the first part limits the upper and lower bounds while the second regulates the monotonic non-increase feature, as formulated by Eqs. (6) and (7), respectively.

After each attack, the number of nodes in LCC of the reminder network will be greater than or equal to 1, but less than or equal to the current (temporal) network size. Thus, each LCC value must be constrained by the following conditions:

$$N_{LCC}(i) = \begin{cases} N-i, & \text{if } N_{LCC}(i) > N-i, \\ 1, & \text{if } N_{LCC}(i) < 1, \\ N_{LCC}(i), & \text{otherwise}, \end{cases} \tag{6}$$

where $N_{\mathrm{LCC}}(i)$ represents the number of nodes in LCC, as in Eq. (1).

Regarding the local increase in the size of LCC, if there is any position in the LCC curve, returned by CNN-RP, where the value is greater than its preceding value (local increase), then an interpolation formulated by Eq. (7) is applied. Specifically, suppose that it is detected as $N_{LCC}(k) > N_{LCC}(i)$ $(k \geq i+1)$, which violates the monotonically non-increasing condition. In this situation, the algorithm will continue to search along $j = k+1, k+2, \ldots$, until $N_{LCC}(j) < N_{LCC}(i)$ is detected. To that end, an interpolation is applied as follows:

$$N_{LCC}(k) = N_{LCC}(i) + \frac{k-i}{i-j}\cdot\left(N_{LCC}(i) - N_{LCC}(j)\right), \tag{7}$$

where the integers i, j, and k satisfy $k \geq i+1$, $k \leq j-1$, and $i \leq j-2$. An example of interpolation is shown in Fig. 3.

Note that 1) the filter does not check the correctness of the predicted data, but only deals with the logically unreasonable data; for example, it does not check whether $N_{LCC}(k)$ in Fig. 3 is overestimated or underestimated, since the true values are unknown to the filter. 2) Only the size of LCC is monotonically non-increasing during sequential attacks, but the normalized LCC curve, as shown in Eq. (1), is not so.

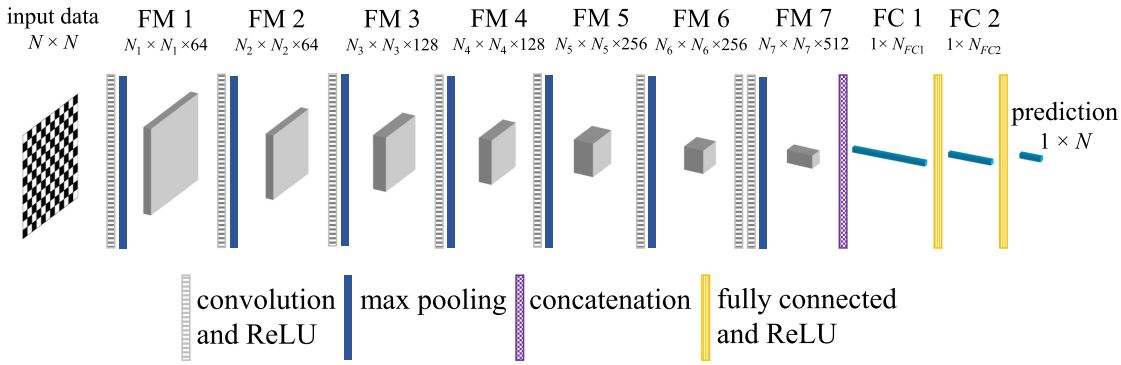

Fig. 2. The structure of the CNN-RP. FM represents for *feature map* and FC for *fully connected*. The input is the adjacency matrix converted image; the output is a $1 \times N$ vector that represents the predicted LCC curve. The data size $N_i = \lceil N/(i+1) \rceil$, $i = 1, 2, \ldots, 7$. The concatenation layer reshapes the matrix to a vector, from FM 7 to FC 1, i.e., $N_{FC1} = N_7 \times N_7 \times 512$. N_{FC2} is a hyperparameter and $N_{FC2} \in (N_{FC1}, N)$. Always set $N_{FC2} = 4096$ for the networks of sizes $N = 1000$ in this paper.

TABLE I
PARAMETERS IN SEVEN GROUPS OF CONVOLUTIONAL LAYERS

Group	Layer	Kernel size	Stride	Output channel
Group 1	Conv7-64	7x7	1	64
	Max2	2x2	2	64
Group 2	Conv5-64	5x5	1	64
	Max2	2x2	2	64
Group 3	Conv3-128	3x3	1	128
	Max2	2x2	2	128
Group 4	Conv3-128	3x3	1	128
	Max2	2x2	2	128
Group 5	Conv3-256	3x3	1	256
	Max2	2x2	2	256
Group 6	Conv3-256	3x3	1	256
	Max2	2x2	2	256
Group 7	Conv3-512	3x3	1	512
	Max2	2x2	2	512

IV. EXPERIMENTAL STUDIES

The performance of CNN-RP is demonstrated by extensive numerical experiments.

Four representative synthetic (directed and undirected) network models are simulated: the Erdös–Rényi (ER) random-graph [47], generic scale-free (SF) [48]–[50], q-snapback (QS) [51], and Newman–Watts (SW) small-world [52] networks. The detailed generation methods for these network models can be found in [40] and [42], respectively. CNN-RP is trained for predicting the network robustness using the data collected form these synthetic networks, and then tested on the same or different distributed synthetic network data, as well as on 12 real-world networks.

Specifically, for directed networks, the following four cases are studied: 1) both training and testing data are drawn from the same dataset. 2) The testing data are the training samples (with different average degrees) from a different dataset. 3) The CNN-RP trained by synthetic network data is tested on 12 real-world network data, for which the study of the first case is also extended to undirected networks. 4) CNN-RP is compared to the spectral measure in predicting the overall network robustness under same attacks.

In experiments, the network size is set to 1000 for synthetic networks, while it is real data size for any real-world network.

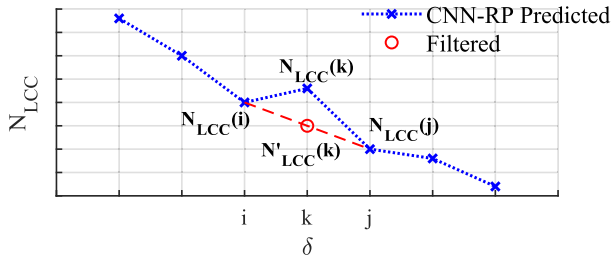

Fig. 3. An example of interpolation. Since CNN-RP returned value $N_{LCC}(k) > N_{LCC}(i)$, it is replaced by the interpolated value $N'_{LCC}(k)$ (red circle) obtained from Eq. (7).

The training data are drawn from a set of randomly-generated network instances, where the average degree $\langle k \rangle$ is set to 5, 8, and 10, respectively. The total number of training samples is $9600 = 4 \times 3 \times 800$, which contain 4 topologies, 3 degrees, and 800 random instances for each configuration. There is another set of instances, where the average degree $\langle k \rangle$ is set to 4, 7, and 9 respectively, which are used for the case that the training and testing data are taken from different distributions, respectively.

For the real-world networks used, their basic information is summarized in Table II.

Three node-removal attack strategies are simulated, namely the random attack (RA), and targeted betweenness-based (TB) and targeted degree-based (TD) attacks. RA removes randomly-selected nodes, while TB and TD remove nodes with maximum betweenness and maximum degree, respectively. For TB and TD, if two or more nodes have the same maximum value (either betweenness or degree), one of them is randomly selected to remove by the attack.

The experiments are performed using a PC Intel (R) Core i7-8750H CPU @ 2.20 GHz, with memory (RAM) 16 GB, running Windows 10 Home 64-bit Operating System.

A. Directed Synthetic Networks

1) Training and Testing Data are Both From the Same Dataset: Figs. 4–7 show the results when the average degree $\langle k \rangle$ is set to 5, 8, and 10, respectively, for both training and testing data. In each figure, *pv* represents the CNN-RP

TABLE II
BASIC INFORMATION OF 12 REAL-WORLD NETWORKS [53]

id	network name	N	M
rwn1	ba_1k_6k	1000	5964
rwn2	bcsstk08	1074	7017
rwn3	bcsstk09	1083	9760
rwn4	er_graph_1k_6k	1000	6000
rwn5	G43	1000	9990
rwn6	G51	1000	5909
rwn7	geo1k_10k	1000	10000
rwn8	ia-email-univ	1133	5451
rwn9	lp_fffff800	1028	6401
rwn10	photogrammetry	1388	11816
rwn11	Roget	1022	5075
rwn12	SmaGri	1059	4919

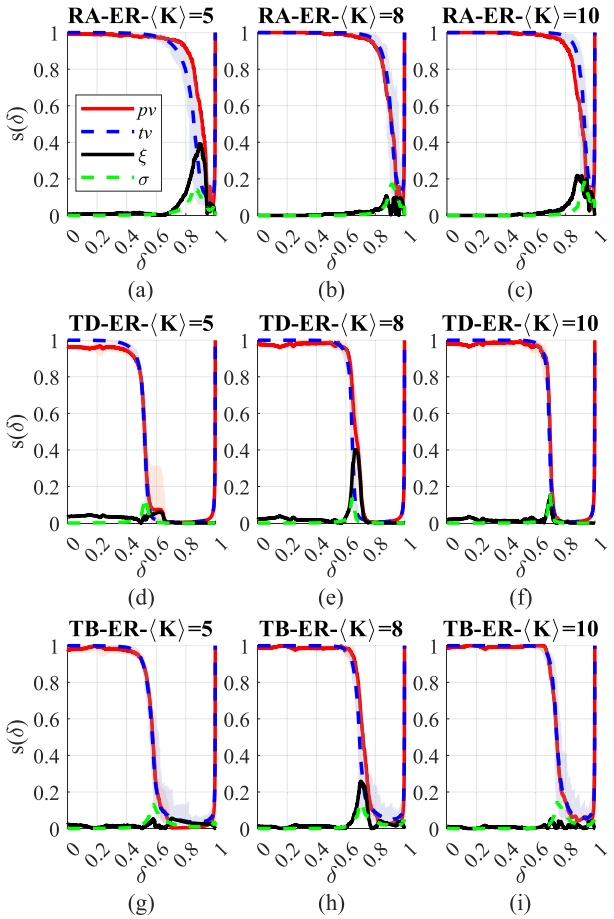

Fig. 4. [color online] Results of CNN-RP NLC prediction for ER networks under RA, TD, and TB, respectively. δ represents the proportion of removed nodes; $s(\delta)$ represents the ratio of LCC versus the current network size, as shown in Eq. (1).

predicted curve; *tv* represents the true curve obtained by attack simulations; ξ represents the prediction error that can be calculated by Eq. (3); and σ represents the standard deviation of the testing data that are randomly collected. The shadow in the same color represents the range of standard deviation. These figures show that CNN-RP can predict NLCs well for ER, SF, SW, and QS networks, not only in the general shapes but also in details such as the curve turning points. The prediction error is small, but slightly higher than

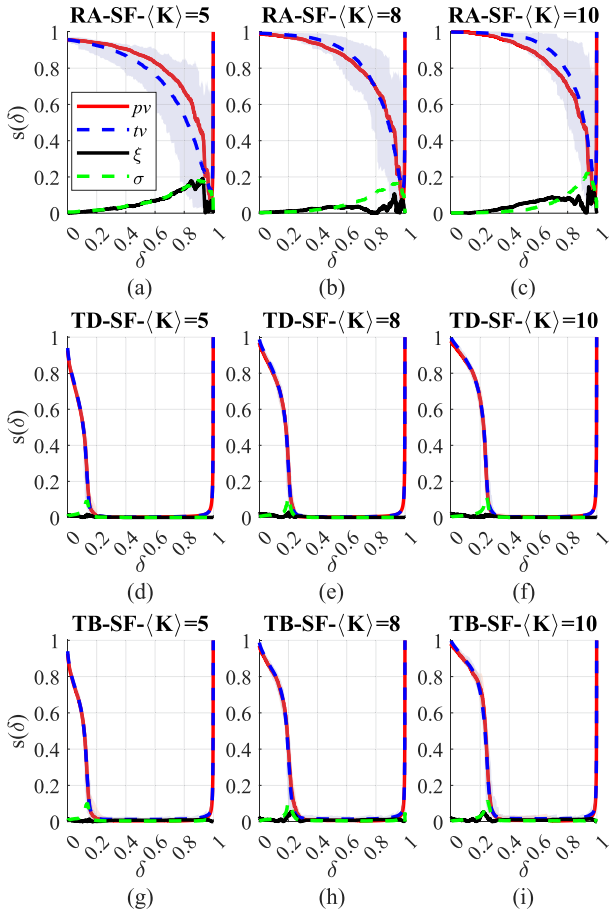

Fig. 5. [color online] Results of CNN-RP NLC prediction for SF networks under RA, TD, and TB, respectively. δ represents the proportion of removed nodes; $s(\delta)$ represents the ratio of LCC versus the current network size, as shown in Eq. (1).

the standard deviation of the testing data. In addition, compared to attack simulations, CNN-RP can return the network connectedness robustness performance within a run of significantly shorter run time. For example, for ER networks with $N = 1000$ and $\langle k \rangle = 5$ under random attacks, the average run time for attack simulation is 11.65 seconds, while it is only 0.12 second by CNN-RP.

Compared to Figs. 4, 6 and 7, which show that ER, SW, and QS networks can maintain good robustness against random and targeted attacks, Fig. 5 shows that SF networks are more fragile than the other three, when the network sizes are the same. Nevertheless, in all the cases, CNN-RP can well predict the NLCs. The overall prediction error is small, but relatively large in the period when the network become drastically disconnected (the curve drops abruptly).

2) Training and Testing Data are From Different Distributions: Fig. 9 shows the results of CNN-RP predicting the NLCs of the networks with average degree $\langle k \rangle = 4$, 7, and 9, respectively, under random attacks. Table III shows the prediction error $\bar{\xi}$ and standard deviation $\bar{\sigma}$ of the testing data. Together with Fig. 8, the overall errors and standard deviation values are mostly of the same order in magnitude of about 10^{-2}. For SF networks, the obtained prediction errors are

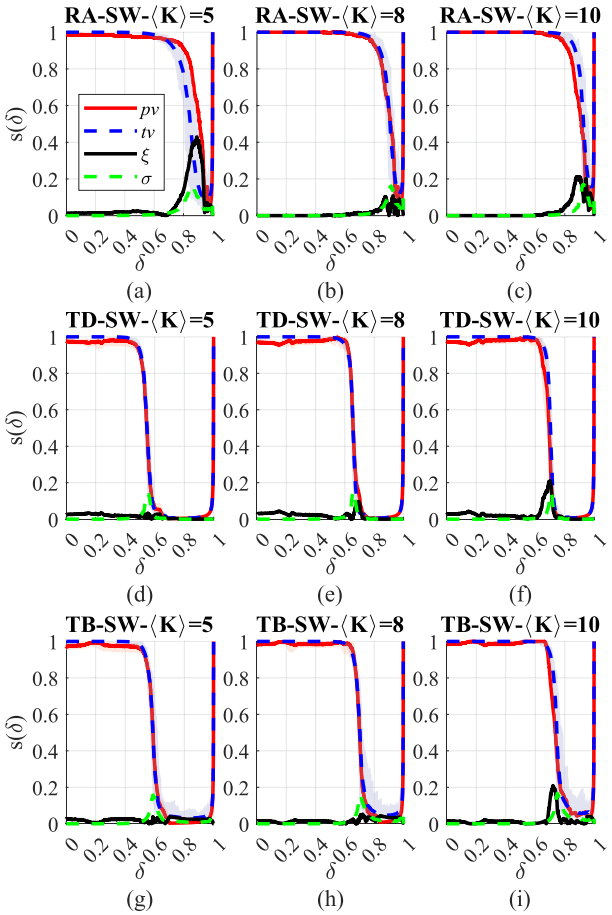

Fig. 6. [color online] Results of CNN-RP NLC prediction for SW networks under RA, TD, and TB, respectively. δ represents the proportion of removed nodes; $s(\delta)$ represents the ratio of LCC versus the current network size, as shown in Eq. (1).

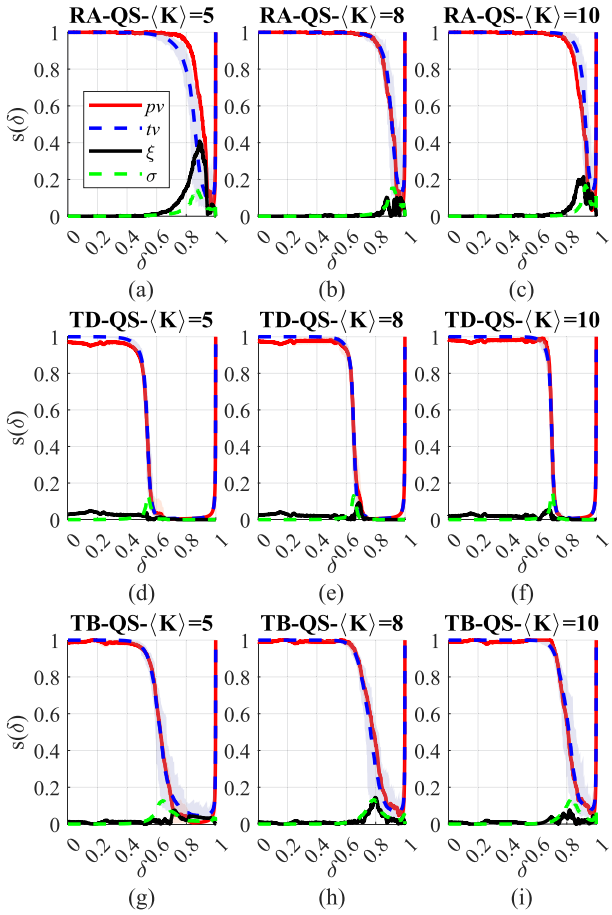

Fig. 7. [color online] Results of CNN-RP NLC prediction for QS networks under RA, TD, and TB, respectively. δ represents the proportion of removed nodes; $s(\delta)$ represents the ratio of LCC versus the current network size, as shown in Eq. (1).

slightly lower than the standard deviations, while for ER, SW, and QS networks, the prediction errors are slightly higher than the standard deviations.

B. Undirected and Real-World Networks

Fig. 10 shows the results of CNN-RP predicting the robustness performance for 12 undirected networks under RA. Again, CNN-RP shows a competitive performance with a low error level. Here, the CNN-RP is newly trained using a set of undirected networks as the training data.

Fig. 11 shows the results of CNN-RP predicting the robustness performance for 12 real-world networks under RA. The CNN-RP trained using the synthetic networks as shown in Subsection IV-A. Since the sizes of some real-world networks are slightly larger than 1000, as shown in Table II, resizing is performed on the graph-converted images, i.e., a pair of rows and columns is randomly picked and removed until it reaches $N = 1000$. For each network, the random resizing is repeated 20 times, and the prediction results and errors are then averaged.

It shows that CNN-RP can predict the rough contour, while the details of the NLCs are not well revealed. This implies that there is a lack of real-world data in the training data. However, choosing the representative real-world data for the training data is also a non-trivial problem.

C. Compared to Spectral Measures

Spectral measures are commonly used to predict or quantify the network robustness regarding connectedness. Here, 6 typical spectral measures are compared in predicting the network robustness. They are spectral radius (SR), spectral gap (SG), natural connectivity (NC), algebraic connectivity (AC), effective resistance (ERe), and spanning tree count (STC). Details (definitions and calculations) for these spectral measures can be found in, e.g., [32]. In the above-discussed comparisons, CNN-RP is used to predict the entire NLC, which can be converted to a scalar by taking the mean value using Eq. (4).

In this work, the above prediction measures (namely, SR, SG, NC, AC, ERe, STC, and CNN-RP) are used to predict the ordinal ranks of network robustness. As mentioned, there are 4 network types (namely, ER, SF, SW, and QS). For each type of network, there are 5 average degrees (namely, $\langle k \rangle = 5, 7, 8, 9, 10$). For each network type and each average degree, there are 100 randomly-generated instances. Thus, there are totally $4 \times 5 \times 100 = 2000$ networks. The predicted ranks of network

TABLE III
THE MEAN PREDICTION ERROR VERSUS THE STANDARD DEVIATION OF THE TESTING DATA. THE AVERAGE DEGREE FOR TRAINING DATA IS SET TO $\langle k\rangle = 5$, 8, AND 10, RESPECTIVELY; WHILE FOR TESTING DATA IS SET TO $\langle k\rangle = 4$, 7, AND 9, RESPECTIVELY

			$\langle k\rangle = 4$	$\langle k\rangle = 7$	$\langle k\rangle = 9$	overall
RA	ER	$\bar{\xi}$	0.0897	0.0140	0.0213	0.0417
		$\bar{\sigma}$	0.0237	0.0170	0.0151	0.0186
	SF	$\bar{\xi}$	0.1064	0.0173	0.0386	0.0541
		$\bar{\sigma}$	0.0705	0.0603	0.0550	0.0619
	SW	$\bar{\xi}$	0.1098	0.0137	0.0193	0.0476
		$\bar{\sigma}$	0.0215	0.0174	0.0153	0.0181
	QS	$\bar{\xi}$	0.0907	0.0119	0.0186	0.0404
		$\bar{\sigma}$	0.0218	0.0179	0.0157	0.0185

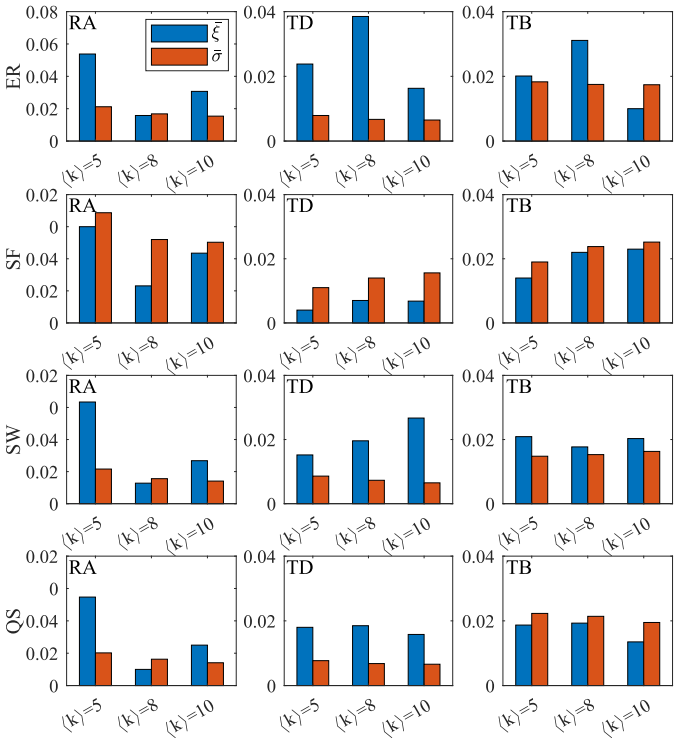

Fig. 8. [color online] Comparison of the mean prediction error ($\bar{\xi}$) versus the standard deviation ($\bar{\sigma}$) of the testing data. The average degree for both training and testing data is set to $\langle k\rangle = 5$, 8, and 10, respectively.

robustness are obtained using the above-mentioned prediction measures, where each measure returns a predicted rank-list of 2000 values. As a benchmark, the true ranks are obtained from simulations. Then, the 7 predicted rank-lists are compared to the true rank-list. The rank error σ_r is calculated by

$$\sigma_r = |\hat{rl} - rl|\,, \tag{8}$$

where $\hat{rl}$ represents the predicted rank-list (by either a spectral measure or CNN-RP), and rl represents the true rank-list obtained from simulations.

The resultant rank error information is summarized in Table IV. For example, given two predicted rank-lists, $\hat{rl}_1 = [1, 4, 5, 3, 2]$ and $\hat{rl}_2 = [5, 1, 2, 4, 3]$, and a true rank-list, $rl_t = [2, 1, 5, 4, 3]$, the rank errors are obtained as $\sigma_{r1} = [1, 3, 0, 1, 1]$ and $\sigma_{r2} = [3, 0, 3, 0, 0]$, respectively. The numbers of '0' in

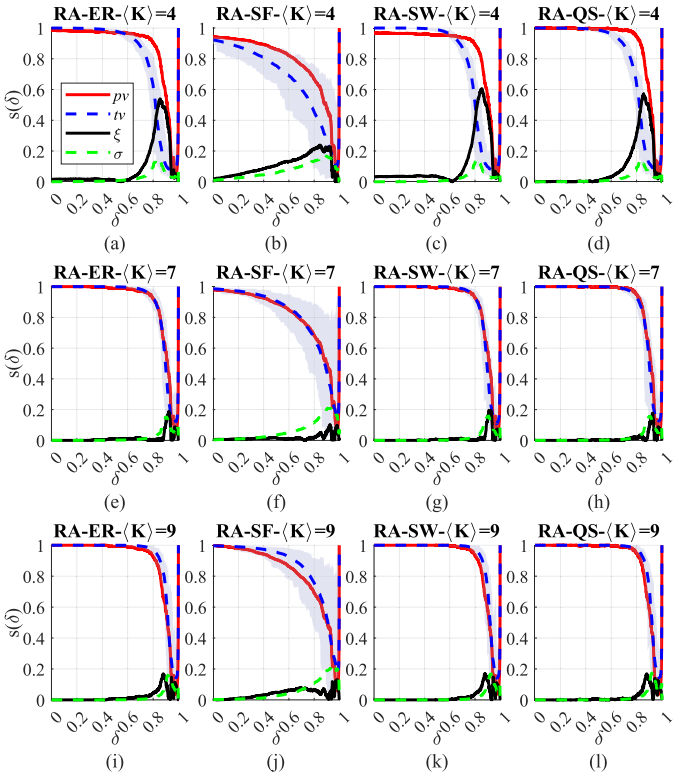

Fig. 9. [color online] Results of CNN-RP NLC prediction for synthetic networks under RA, where the testing data ($\langle k\rangle = 4, 7, 9$) are different from the training data ($\langle k\rangle = 5, 8, 10$). δ represents the proportion of removed nodes; $s(\delta)$ represents the ratio of LCC versus the current network size, as shown in Eq. (1).

σ_{r_1} and σ_{r_2} are counted as the 'correct rank' in the table. The 'average rank error,' 'max rank error,' and 'min rank error' are calculated accordingly. Moreover, the number of network instances, which are predicted to be within top 10% (ordinal ranks in terms of connectedness robustness) and also confirmed to be within top 10% by simulation, is counted and included in the 'top 10%' column. The numbers in the 'bottom 10%' column are similarly calculated.

As shown in Table IV, AC receives the minimum 'average rank error' 190.72, followed by CNN-RP with an average rank error 272.44. AC obtains the smallest 'max rank error,' followed by CNN-RP. Only AC, ERe and CNN-RP receive a 'min rank error' 0, implying that these measures predict at least once that is exactly the same as the true rank. CNN-RP predicts 3 ranks correctly. STC, AC and CNN-RP predict a number of correct top 10% and bottom 10% networks, whose robustness values are truly top 10% and bottom 10% according to the simulation results. The test dataset contains 2000 networks, giving 200 networks ranked as top 10% and bottom 10%, respectively.

The predictive measures AC and STC, as well as the proposed CNN-RP, return good prediction results, better than other spectral measures. More importantly, CNN-RP returns not only the predictive results, but also predictive values throughout the entire LCC changing process; while the spectral measures return only a single quantitative value. However, CNN-RP requires a substantial amount of training data, while the spectral measures do not.

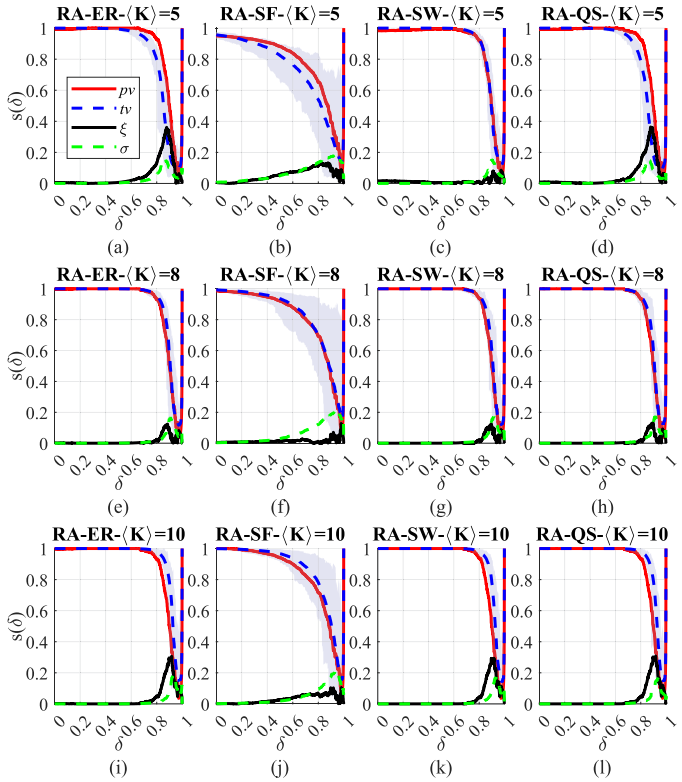

Fig. 10. [color online] Results of CNN-RP NLC prediction for synthetic *undirected* networks under RA, where the average degrees are set to $\langle k\rangle = 5$, 8, and 10, respectively. δ represents the proportion of removed nodes; $s(\delta)$ represents the ratio of LCC versus the current network size, as shown in Eq. (1).

D. Compared to Predictor for Controllability Robustness

CNN-RP uses the same CNN structure as that in the predictor for controllability robustness (PCR) [40]. The computational complexity of CNN-RP is similar to PCR. Both CNN-RP and PCR use a single CNN to perform the regression tasks for all the networks, but the task for CNN-RP is more difficult than that for PCR, since the variation of LCC is greater.

Fig. 12 shows an example of the comparison between the connectedness robustness and the controllability robustness. In Fig. 12(a), it requires a proportion $4/6$ of driver nodes and there is a proportion $6/6$ in the LCC; but in Fig. 12(b), it requires a proportion $5/5$ of driver nodes and there is a proportion of $1/5$ in the LCC. The change of "controllability" is from 0.667 to 1, not as drastic as the change of "connectedness" from 1 to 0.2. Removing a node will increase the number of driver nodes at most by 1 regarding the controllability, but it may reduce the number of nodes by a number as high as N regarding the LCC. The installed filter helps relieve the variation burden in the connectedness robustness prediction. Note that PCR obtains an average error rate clearly lower than the standard deviation of the testing data, while CNN-RP obtains an average error rate that is slightly higher than the standard deviation on the testing dataset.

The conventional spectral measures have been developed to predict the connectedness robustness for a long time, while there is no evidence that these spectra are suitable for predicting the controllability robustness. On the other hand, CNNs

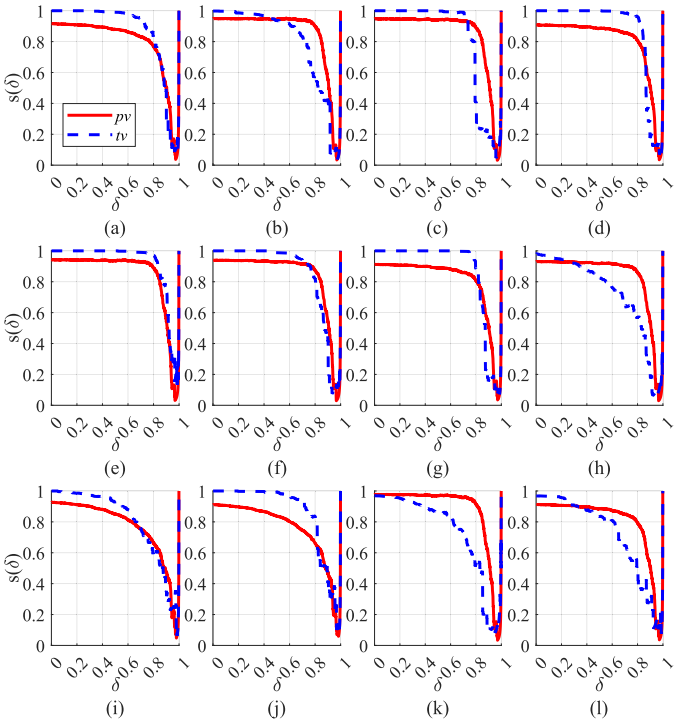

Fig. 11. [color online] Results of CNN-RP NLC prediction for *real-world* networks under RA, where δ represents the proportion of removed nodes; $s(\delta)$ represents the ratio of LCC versus the current network size, as shown in Eq. (1). Basic information of these networks are presented in Table II.

TABLE IV
COMPARISON OF THE PREDICTION ERROR INFORMATION FOR THE 7 PREDICTIVE MEASURES. BOLD NUMBERS ARE RESULTS FROM THE BEST PERFORMING PREDICTION MEASURES

	average rank error	max rank error	min rank error	correct rank	top 10%	bottom 10%
SR	736.41	1745	1	0	0	0
SG	736.26	1750	6	0	0	0
NC	736.40	1745	1	0	0	0
AC	**190.72**	**955**	**0**	**4**	95	37
ERe	1033.15	1988	1	0	0	0
STC	294.94	1124	**0**	**4**	**112**	100
CNN-RP	272.44	1080	**0**	3	50	**161**

are effective and efficient in predicting many general features and performances of networked systems that have no analytical solutions. As a matter of fact, in the comparison discussed in [42], the CNN methods outperform the spectral measures in predicting the controllability robustness. However, in the present work, CNN-RP receives the overall rank-2 performance, following the algebraic connectivity, yet nevertheless it performs better than other measures including spectral measures. Therefore, the results obtained in Subsection IV-C are truly satisfactory and indeed quite encouraging.

E. Utilities of the Filter

The utility of the installed filter is to filter out the unreasonable data predicted by CNN. Fig. 13(a) shows the LCC predictions with and without a filter, respectively. It is clear

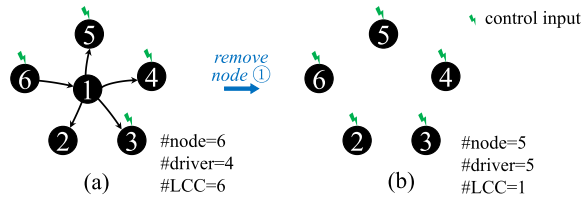

Fig. 12. An example of the difference between connectedness robustness and controllability robustness: (a) given a weakly connected network that has 6 nodes and requires 4 driver nodes; (b) after the hub node is removed, it becomes a network with 5 isolated nodes that requires 5 driver nodes.

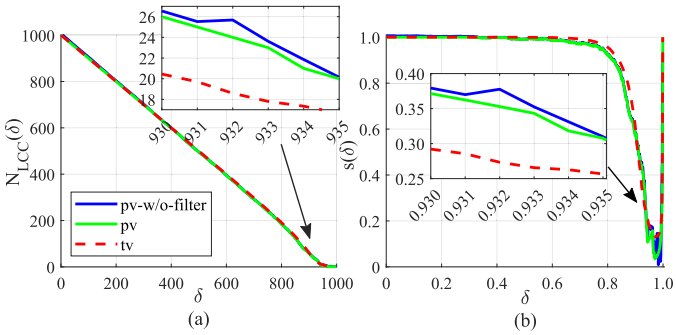

Fig. 13. Comparison of the predictions with and without the filter: (a) LCC prediction and (b) NLC prediction. It is an ER with $\langle k\rangle = 8$, under random attacks.

TABLE V
COMPARISON OF THE AVERAGE ERRORS WITH AND WITHOUT THE FILTER, FOR ER WITH $\langle k\rangle = 8$, UNDER RANDOM ATTACKS

	$\bar{\xi}$ of *pv*	$\bar{\xi}$ of *pv* without filter	$\Delta\bar{\xi}$
ER	0.0214	0.0222	0.0008
SF	0.0475	0.0489	0.0014
SW	0.0183	0.0223	0.0039
QS	0.0169	0.0179	0.0010

that without the filter, the blue curve violates the nature that the number of nodes of LCC in a network under attacks must be monotonically non-increasing. In contrast, the green curve filters out these unreasonable data, becoming closer to the true curve. It is worthy mentioning that although the number of nodes of LCC is monotonically non-increasing, the NLC curve is not, as illustrated in Fig. 13(b), as δ is approaching 1.

Although precision check is not the utility of the filter, it is observed that the prediction precision can be improved after installing the filter. Table V shows a comparison of the CNN-RP prediction, where $\bar{\xi}$ (see Eq. (4)) represents the average error of the prediction and $\Delta\bar{\xi}$ represents the average error reduction by the filter. A consistent error reduction can be observed when the filter is installed.

F. Shuffling on the Converted Images

As shown in Fig. 14, the generation mechanism of synthetic networks may impose some visible features to the adjacency

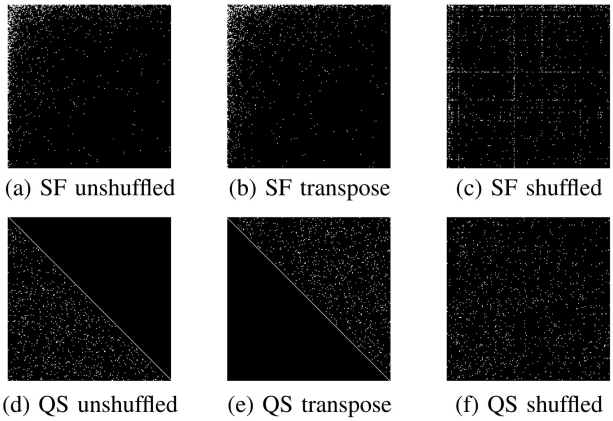

Fig. 14. Example of SF and QS networks: (a), (d) unshuffled; (b), (e) transpose; and (c), (f) shuffled with $n_{\rm sh} = 500$, where network size $N = 200$ and average degree $\langle k\rangle = 5.14$.

TABLE VI
AVERAGE ERROR ($\bar{\xi}$) OF THE CNN-RP PREDICTION, AS THE NUMBER OF RANDOM SHUFFLES $n_{\rm sh}$ CHANGES. THE NETWORKS HAVE AVERAGE DEGREE $\langle k\rangle = 8$, UNDER RANDOM ATTACKS

	ER	SF	SW	QS
unshuffled	0.0214	0.0475	0.0183	0.0169
transpose	0.0214	0.0476	0.0174	**0.0671**
$n_{\rm sh} = 50$	0.0214	0.0457	0.0184	0.0197
$n_{\rm sh} = 100$	0.0214	0.0449	0.0184	0.0247
$n_{\rm sh} = 150$	0.0213	0.0434	0.0187	0.0274
$n_{\rm sh} = 200$	0.0212	0.0459	0.0189	0.0280
$n_{\rm sh} = 500$	0.0214	**0.0860**	0.0212	0.0216

matrix converted images. For example, for SF network, due to the preferential attachment mechanism, the 'old' nodes (with smaller node indices) have higher degrees, and thus there is a spark in the upper-left corner as shown in Fig. 14(a). These features can be filtered out by performing random shuffling as shown in Fig. 14(b)., which means to randomly exchange the rows and columns of the adjacency matrices. The simulation results in [42] show that the existence of these visible features does not affect the CNN performance in both network classification and controllability robustness prediction. Note that exchanging the rows and columns of an adjacency matrix will only affect the image, but not the network topology.

In the following experiment, the CNN-RP performance is investigated when the training data are unshuffled, while the testing data are shuffled. Let $n_{\rm sh}$ be the number of random shuffles; and $n_{\rm sh} = 1$ means that there is a pair of randomly selected nodes exchanging their indices (namely, exchanging their rows and columns in the adjacency matrix).

Table VI shows that the average error of the prediction, which can be calculated by Eq. (4), is generally not sensitive to the shuffling of adjacency matrices. Specifically, for SF networks, the prediction error becomes larger only when $n_{\rm sh} = 500$; as for QS networks, the prediction result is degraded when the input is the transpose of the original image. As can be seen from Fig. 14(e), the QS transpose image is significantly different from the QS unshuffled image (although the network topology remains the same). In contrast, the SF transpose image is

not significantly different from the SF unshuffled image. The degraded performance is likely caused by this significant image difference. However, although the images are clearly different after shuffling, CNN-RP can still perform well on processing these shuffled images, while the number of shuffles generally does not affect the prediction error.

V. Conclusions

This paper proposes a fast and effective approach to predicting the connectedness robustness of complex networks against node-removal attacks. Conventionally, the network robustness is determined by attack simulations, from which a sequence of measure values are collected to record the connectedness of the remaining network after a sequence of attacks, which is computationally very time-consuming if the network size is large. In this paper, CNN-RP is proposed to predict the connectedness robustness of various complex networks, based on the successful applications of CNNs for image processing and network controllability robustness prediction. Extensive numerical experiments on directed and undirected, synthetic and real-world networks have been performed, demonstrating the effectiveness of CNN-RP in prediction performances: 1) CNN-RP can predict the network connectedness robustness with a low average error, which is in the same order in magnitude as the standard deviation of the testing dataset. 2) The CNN-based predictor provides a good and even better predictive measure than the traditional powerful spectral measures. This paper demonstrates once again that the CNN-based prediction technique has a good potential for generalization with a wide range of applications to complex networks.

References

[1] A.-L. Barabási, Netw. 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, Inc., 2014.

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

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

[6] 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, 2003, Art. no. 068701.

[7] C. M. Schneider, A. A. Moreira, J. S. Andrade, 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.

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

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

[10] 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, 2014, Art. no. 118902.

[11] S. Iyer, T. Killingback, B. Sundaram, and Z. Wang, "Attack robustness and centrality of complex networks," *PLoS One*, vol. 8, no. 4, 2013, Art. no. e59613.

[12] Q. Nguyen, H. Pham, D. Cassi, and M. Bellingeri, "Conditional attack strategy for real-world complex networks," *Phys. A: Statist. Mech. Appl.*, vol. 530, 2019, Art. no. 121561.

[13] T. Nie, Z. Guo, K. Zhao, and Z.-M. Lu, "New attack strategies for complex networks," *Phys. A: Statist. Mech. Appl.*, vol. 424, pp. 248–253, 2015.

[14] Y.-R. Ruan, S.-Y. Lao, J.-D. Wang, L. Bai, and L.-D. Chen, "Node importance measurement based on neighborhood similarity in complex network," *Acta Phys. Sinica*, vol. 66, no. 3, 2017, Art. no. 038902.

[15] M. Šimon, I. Dirgová Luptáková, L. Huraj, M. Host'oveckỳ, and J. Pospíchal, "Combined heuristic attack strategy on complex networks," *Math. Problems Eng.*, vol. 2017, 2017, Art. no. 6108563.

[16] H. Yang and S. An, "Critical nodes identification in complex networks," *Symmetry*, vol. 12, no. 1, p. 123, 2020, Art. no. 6108563.

[17] B. R. da Cunha, J. C. Gonzalez-Avella, and S. Goncalves, "Fast fragmentation of networks using module-based attacks," *PLoS One*, vol. 10, no. 11, 2015, Art. no. e0142824.

[18] S. Shai, D. Y. Kenett, Y. N. Kenett, M. Faust, S. Dobson, and S. Havlin, "Critical tipping point distinguishing two types of transitions in modular network structures," *Phys. Rev. E*, vol. 92, no. 6, 2015, Art. no. 062805.

[19] H. Wang, J. Huang, X. Xu, and Y. Xiao, "Damage attack on complex networks," *Phys. A: Statist. Mech. Appl.*, vol. 408, pp. 134–148, 2014.

[20] L. Ma, J. Liu, and B. Duan, "Evolution of network robustness under continuous topological changes," *Phys. A: Statist. Mech. Appl.*, vol. 451, pp. 623–631, 2016.

[21] Z.-X. Wu and P. Holme, "Onion structure and network robustness," *Phys. Rev. E*, vol. 84, no. 2, 2011, Art. no. 026106.

[22] T. Tanizawa, S. Havlin, and H. E. Stanley, "Robustness of onionlike correlated networks against targeted attacks," *Phys. Rev. E*, vol. 85, no. 4, 2012, Art. no. 046109.

[23] Y. Hayashi and N. Uchiyama, "Onion-like networks are both robust and resilient," *Sci. Rep.*, vol. 8, no. 1, pp. 1–13, 2018.

[24] J. Gao, S. V. Buldyrev, S. Havlin, and H. E. Stanley, "Robustness of a network of networks," *Phys. Rev. Lett.*, vol. 107, no. 19, 2011, Art. no. 195701.

[25] G. Dong, J. Gao, R. Du, L. Tian, H. E. Stanley, and S. Havlin, "Robustness of network of networks under targeted attack," *Phys. Rev. E*, vol. 87, no. 5, 2013, Art. no. 052804.

[26] B. Min, S. Do Yi, K.-M. Lee, and K.-I. Goh, "Network robustness of multiplex networks with interlayer degree correlations," *Phys. Rev. E*, vol. 89, no. 4, 2014, Art. no. 042811.

[27] C. M. Schneider, N. Yazdani, N. A. Araújo, S. Havlin, and H. J. Herrmann, "Towards designing robust coupled networks," *Sci. Rep.*, vol. 3, no. 1, pp. 1–7, 2013.

[28] L. Cuadra, S. Salcedo-Sanz, J. Del 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.

[29] A. Zeng and W. Liu, "Enhancing network robustness against malicious attacks," *Phys. Rev. E*, vol. 85, no. 6, 2012, Art. no. 066130.

[30] V. H. Louzada, F. Daolio, H. J. Herrmann, and M. Tomassini, "Smart rewiring for network robustness," *J. Complex Netw.*, vol. 1, no. 2, pp. 150–159, 2013.

[31] L. Bai, Y.-D. Xiao, L.-L. Hou, and S.-Y. Lao, "Smart rewiring: Improving network robustness faster," *Chin. Phys. Lett.*, vol. 32, no. 7, 2015, Art. no. 078901.

[32] H. Chan and L. Akoglu, "Optimizing network robustness by edge rewiring: A general framework," *Data Mining Knowl. Discov.*, vol. 30, no. 5, pp. 1395–1425, 2016.

[33] Y. Lou, S. Xie, and G. Chen, "Searching better rewiring strategies and objective functions for stronger controllability robustness," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 68, no. 6, pp. 2112–2116, Jun. 2021.

[34] X.-B. Cao, C. Hong, W.-B. Du, and J. Zhang, "Improving the network robustness against cascading failures by adding links," *Chaos, Solitons Fractals*, vol. 57, pp. 35–40, 2013.

[35] L. Hou, S. Lao, B. Jiang, and L. Bai, "Enhancing complex network controllability by rewiring links," in *Proc. Int. Conf. Intell. System Des. Eng. Appl.*, 2013, pp. 709–711.

[36] J. Xu, J. Wang, H. Zhao, and S. Jia, "Improving controllability of complex networks by rewiring links regularly," in *Proc. Chin. Control Decis. Conf.*, 2014, pp. 642–645.

[37] R. C. Gunasekara, C. K. Mohan, and K. Mehrotra, "Multi-objective optimization to improve robustness in networks," in *Multi-Objective Optimization*. Berlin, Germany: Springer, 2018, pp. 115–139.

[38] J. Liu, H. A. Abbass, and K. C. Tan, "Evolving robust networks using evolutionary algorithms," in *Evolutionary Computation Complex Networks*. Berlin, Germany: Springer, 2019, pp. 117–140.

[39] S. Wang and J. Liu, "Designing comprehensively robust networks against intentional attacks and cascading failures," *Inf. Sci.*, vol. 478, pp. 125–140, 2019.
[40] Y. Lou, Y. He, L. Wang, and G. Chen, "Predicting network controllability robustness: A convolutional neural network approach," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2020.3013251.
[41] A. Dhiman, P. Sun, and R. Kooij, "Using machine learning to quantify the robustness of network controllability," in *Proc. Int. Conf. Mach. Learn. Netw.*, 2021, pp. 19–39.
[42] Y. Lou, Y. He, L. Wang, K. F. Tsang, and G. Chen, "Knowledge-based prediction of network controllability robustness," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2021.3071367.
[43] C. Fan, L. Zeng, Y. Sun, and Y.-Y. Liu, "Finding key players in complex networks through deep reinforcement learning," *Nat. Mach. Intell.*, vol. 2, pp. 317–324, 2020.
[44] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, 2015.
[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[46] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.
[47] P. Erdös and A. Rényi, "On the strength of connectedness of a random graph," *Acta Mathematica Hungarica*, vol. 12, no. 1-2, pp. 261–267, 1964.
[48] C.-L. Pu, W.-J. Pei, and A. Michaelson, "Robustness analysis of network controllability," *Phys. A: Statist. Mech. Appl.*, vol. 391, no. 18, pp. 4420–4425, 2012.
[49] K.-I. Goh, B. Kahng, and D. Kim, "Universal behavior of load distribution in scale-free networks," *Phys. Rev. Lett.*, vol. 87, no. 27, 2001, Art. no. 278701.
[50] F. Sorrentino, "Effects of the network structural properties on its controllability," *Chaos: An Interdiscipl. J. Nonlinear Sci.*, vol. 17, no. 3, 2007, Art. no. 033101.
[51] Y. Lou, L. Wang, and G. Chen, "Toward stronger robustness of network controllability: A snapback network model," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 65, no. 9, pp. 2983–2991, Sep. 2018.
[52] M. E. Newman and D. J. Watts, "Renormalization group analysis of the small-world network model," *Phys. Lett. A*, vol. 263, no. 4-6, pp. 341–346, 1999.
[53] R. A. Rossi and N. K. Ahmed, "An interactive data repository with visual analytics," *SIGKDD Explor.*, vol. 17, no. 2, pp. 37–41, 2016. [Online]. Available: http://networkrepository.com

Yang Lou (Member, IEEE) received the B.E. degree from Xidian University, Xi'an, China in 2008, the M. S. degree from Ningbo University, Ningbo, China in 2012, and the Ph.D. degree from the City University of Hong Kong, Hong Kong in 2017. He is currently a Postdoctoral Fellow with the *Centre for Chaos and Complex Networks*, City University of Hong Kong. His research interests include complex networks, evolutionary computation, and machine learning.

Ruizi Wu received the B.E. degree from Southwest Jiaotong University, Chengdu, China, in 2020. He is currently working toward the Graduation degree with the School of Computer Science, Sichuan Normal University, Chengdu, China. His research interests include complex networks, evolutionary computation, and machine learning.

Junli Li received the Ph.D. degree from Zhejiang University, Hangzhou, China, in 2002. He is currently a Professor with the College of Computer Science, Sichuan Normal University, Chengdu, China. His research interests include image processing, target tracking, and computational intelligence.

Lin Wang (Senior Member, IEEE) received the B.S. and M.S. degrees from the School of Mathematical Sciences, Shandong Normal University, Jinan, China, in 2003 and 2006, respectively, and the Ph.D. degree in operations research and control theory from the Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China, in 2009. She is currently a Professor with the Department of Automation, Shanghai Jiao Tong University, Shanghai, China. Her current research interests include multiagent systems, adaptive complex networks, and coordination of multiple manipulators.

Guanrong Chen (Life Fellow, IEEE) received the M.Sc. degree in computer science from Sun Yat-sen University, Guangzhou, China, in 1981 and the Ph.D. degree in applied mathematics from Texas A&M University, College Station, TX, USA, in 1987. Since 2000, he has been a Chair Professor and the Founding Director of the Centre for Chaos and Complex Networks, City University of Hong Kong, Hong Kong, prior to that he was a tenured Full Professor with the University of Houston, Houston, TX, USA. He was the recipient of the State Natural Science Award of China in 2008, 2012 and 2016, respectively. He was awarded the 2011 Euler Gold Medal, Russia, and conferred Honorary Doctorates by the Saint Petersburg State University, Russia in 2011 and by the University of Le Havre, Normandy, France in 2014. He is a Member of the Academy of Europe and a Fellow of The World Academy of Sciences, and has been a Highly Cited Researcher in Engineering according to Clarivate Web of Science since 2009.