scientific reports

OPEN



Integrating graph and reinforcement learning for vaccination strategies in complex networks

Zhihao Dong¹, Yuanzhu Chen^{1⊠}, Cheng Li², Terrence S. Tricco³ & Ting Hu¹

Pandemics like COVID-19 have a huge impact on human society and the global economy. Vaccines are effective in the fight against these pandemics but often in limited supplies, particularly in the early stages. Thus, it is imperative to distribute such crucial public goods efficiently. Identifying and vaccinating key spreaders (i.e., influential nodes) is an effective approach to break down the virus transmission network, thereby inhibiting the spread of the virus. Previous methods for identifying influential nodes in networks lack consistency in terms of effectiveness and precision. Their applicability also depends on the unique characteristics of each network. Furthermore, most of them rank nodes by their individual influence in the network without considering mutual effects among them. However, in many practical settings like vaccine distribution, the challenge is how to select a group of influential nodes. This task is more complex due to the interactions and collective influence of these nodes together. This paper introduces a new framework integrating Graph Neural Network (GNN) and Deep Reinforcement Learning (DRL) for vaccination distribution. This approach combines network structural learning with strategic decision-making. It aims to efficiently disrupt the network structure and stop disease spread through targeting and removing influential nodes. This method is particularly effective in complex environments, where traditional strategies might not be efficient or scalable. Its effectiveness is tested across various network types including both synthetic and realworld datasets, demonstrting a potential for real-world applications in fields like epidemiology and cybersecurity. This interdisciplinary approach shows the capabilities of deep learning in understanding and manipulating complex network systems.

Pandemics, such as COVID-19, can impose harm and loss on human life and the social economy ^{1,2}. Vaccines are among the most important medical resources in the fight against a pandemic, but they can face shortages in many countries ³, especially at the early stage of vaccine distribution. Therefore, it is imperative to distribute such crucial public goods effectively. The utilization of Social Network Analysis (SNA) in public health, particularly in the field of immunization, represents a significant intersection of epidemiology and network theory⁴. This approach is predicated on the understanding that human interactions and social structures play a critical role in the spread of infectious diseases. Identifying and targeting those influential individuals within these networks can be an effective strategy for disease control and prevention ⁵. As we know, not every node plays the same role in the network ^{6,7}. Some individuals can have greater influence than others ⁸.

In the last few years, researchers have proposed various centrality-based measures, like degree centrality (number of direct connections), betweenness centrality ⁹ (role in connecting different parts of the network), and closeness centrality ¹⁰ (proximity to all other nodes), for identifying influential individuals in the networks. One of the challenges in using centrality measures is the dynamic nature of real-world networks. Networks evolve over time, and so do node centralities. Additionally, these measures, while powerful, can sometimes oversimplify the complexities of real-world networks. To incorporate more information of networks, some new centrality measures have been proposed, such as PageRank ¹¹, Coreness ¹², LocalRank ¹³, VoteRank ¹⁴, ClusterRank ¹⁵, LeaderRank ¹⁶, and TwitterRank ¹⁷. Each of these measures provides a unique lens through which to view and analyze network influencers. Their effectiveness and applicability can vary significantly based on the unique

¹School of Computing, Queen's University, Kingston, Canada. ²School of Engineering Science, Simon Fraser University, Burnaby, Canada. ³Department of Computer Science, Memorial University of Newfoundland, St. John's, Canada. [⊠]email: yuanzhu.chen@queensu.ca

characteristics of each network being analyzed, such as its size, density, the nature of connections, and the context in which the network operates (e.g., social, biological or technological).

Another research direction is to split the network into various levels. In these studies, higher levels represent the network's core and lower levels refer to its periphery. This approach ranks nodes by assuming those at the center of the network wield greater influence. The *k*-core decomposition ¹⁸ is one representative of such a technique. Drawing inspiration from this, several methodologies $^{19-22}$ have been developed that adapt the *k*-core concept. These methods modify the basic *k*-core approach for determining influential nodes. Their findings indicated that the identified nodes surpass traditional measures in terms of spreading efficiency. Nonetheless, a significant limitation of these models is their relatively blunt precision. Depending on the network's structure, numerous nodes might share the same *k*-core value, yet their actual influence may vary.

Apart from the centrality-based methods and *k*-core based algorithms, researchers also proposed heuristics to identify influential nodes efficiently. Chen et al. ²³ developed an algorithm based on degree discounting that outpaces the typical greedy algorithm by over a million times ²⁴, while achieving a similar level of accuracy. For networks featuring community structures, He et al. ²⁵ have proposed a method centered on communities, employing algorithms for detecting communities ²⁶ to pinpoint essential nodes within different groups. Additionally, Morone and Makse ²⁷ have redefined the task of vital node identification as an optimal percolation problem, focusing on identifying the smallest group of pivotal nodes.

Recently, some studies proposed methods grounded in data-driven machine learning to identify influential nodes in complex networks. Zhao et al. ²⁸ approached the issue of vital node identification through a classification model trained on a large subset of nodes from the original network. In a different study, Yu et al. ²⁹ employed network embedding techniques in temporal networks to pick out influential nodes. Similarly, Khajehnejad et al. ³⁰ applied adversarial graph embedding in networks with distinct communities to find influential nodes. Hao et al. ³¹ adopted network representation learning techniques for identifying influential nodes in networks with overlapping communities. Nevertheless, most of the methods above focus on ranking nodes based on finding individual influential nodes. In numerous practical scenarios, the task is often to locate a subset of influential nodes that are key to maintaining network connectivity and facilitating the dissemination of information or pathogen. Finding a group of influential nodes is more difficult than finding individual ones because the selected nodes can affect each other ³².

We propose a graph learning and reinforcement learning based vaccination strategy to identify a group of critical nodes in the network. It combines Graph Neural Networks (GNN) with Deep Reinforcement Learning (DRL), which brings together GNN's ability to efficiently process and learn from graph-structured data with DRL's prowess in learning optimal decision-making strategies in complex environments. In this hybrid approach, a GNN is used to represent the state of the environment in DRL. The graph-structured data is fed to the GNN, which effectively captures the relationships and interactions between different nodes in the network and generates a detailed representation of the network's structure³³. This enriched state representation is then piped to the DRL system. DRL, with its capabilities in learning optimal policies over time, can make use of the structured data provided by GNN to make informed decisions about which nodes in a network are influential.

Model

Framework

Figure 1 illustrates the proposed framework. It employs a model-free RL approach in conjunction with a graph representation learning module. There are two key components – the encoder and the decoder. The encoder's role is to produce representations for both the states and actions within this context. Subsequently, the decoder utilizes these representations to construct a score function, guiding the selection of the appropriate action for a given state.

The algorithm follows a greedy approach, where it progressively builds a viable solution by adding nodes based on the underlying graph structure. This solution is continually updated to fulfill the stated objective. Greedy algorithms are commonly employed as a strategy for creating approximation and heuristic algorithms for various graph-related problems.

The encoder can either be a graph embedding model or a graph neural network. Specifically, this study uses Graph Convolutional Networks (GCN) ³⁴, which are a type of GNN. Networks from the training set are fed into the encoder. It maps the structural information of the network into a low-dimensional latent space. Following multiple recursive iterations, all nodes acquires their own embedding vectors that encompass their structural position information within the network with the extensive interplay among node features. A virtual node that represents the whole graph is added in order to extract further information about the network. It is connected to all real nodes. The representation of the whole graph then can be obtained by the summation of those of all nodes. The decoder is implemented as a multi-layer perceptron (MLP). This component utilizes the representations derived from the encoder for both states and actions to compute a score that serves as a quantitative assessment of potential actions.

During the training phase, a graph is chosen at random from a set of synthetic graphs. This graph is subsequently processed through the proposed framework, where the agent engages in a game-like scenario to identify influential nodes. In this context, the state refers to the remaining network after node removals, the action is the elimination of an influential node, and the reward stands for the reduction in the number of infected nodes resulting from the action. To ascertain the appropriate action for a given state, GCN is employed to derive an embedding vector for each node (represented by orange bars in Figure 1). The embedding vectors are then utilized to compute scalar *Q* values (depicted as green bars with heights indicative of the values in Figure 1) for all nodes, forecasting the potential long-term benefits of the action.

In practical scenarios, the finely-tuned agent can be deployed on actual networks once the training is complete. The procedure is as follows. First, the network under consideration would be transformed into





compact vectors. Then, the *Q* values for each node are estimated using these vectors. The node with the highest *Q* value is selected at every iteration, and the network is modified accordingly. This procedure is repeated until the network arrives at a predetermined terminal condition (such as maximum number of nodes). The series of nodes that are sequentially removed during this process form the ideal group of influential nodes.

The principal strength of this framework lies in its capacity to effectively handle deferred rewards, which can in turn be used by a simple greedy algorithm to achieve incremental improvement of the objective function. Updates to the network embeddings are made based on the partial solution at each stage of the greedy algorithm, enabling the integration of fresh insights regarding the contribution of every node to the ultimate solution.

Q-learning

The node identification process can be considered as a Markov Decision Process (MDP) using a quintuple $(\mathscr{S}, \mathscr{A}, \mathscr{T}, \mathscr{R}, \gamma)$ in line with standard RL. In this formulation, \mathscr{S} represents the collection of states, \mathscr{A} corresponds to the collection of available actions, $\mathscr{T} : \mathscr{S} \times \mathscr{A} \times \mathscr{S} \to [0, 1]$ denotes the transition function, $\mathscr{R} : \mathscr{S} \to \mathbb{R}$ represents the reward function, and γ serves as the discount-rate parameter. Within this framework, a policy denoted as $\pi : \mathscr{A} \times \mathscr{S} \to [0, 1]$ indicates the probability of taking an action when the agent is in a given state. Here, with a given policy π , a value-function is formulated to quantify the long-term accumulated rewards of states over the future in an episode. The value function represents the expected discounted return, defined as:

$$V^{\pi}(s) = E \left[R(s_0) + \gamma R(s_1) + \gamma^2 R(s_1) + \dots \mid s_0 = s, \pi(s) \right].$$
(1)

The policy of optimizing the expected return can be determined by:

$$J(\pi) = \max_{\pi} E\left[V^{\pi}(s)\right]$$

$$\pi^{*}(s) = \arg\max_{\pi} J(\pi)$$
(2)

Then, we can establish a policy π consisting of a sequence of selected nodes, strategically devised to maximize the expected discounted return, spanning from the initial state to the terminal state.

Q-learning ³⁵ stands as one of the frequently employed algorithms for tackling the MDP problem. Its main task is to estimate value functions. The action-value function under a policy π is defined as,

$$Q^{\pi}(s,a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^{t} \times \mathscr{R}(s_{t}) \mid s_{0} = s, a_{0} = a, \pi\right],$$
(3)

where s_0 and a_0 represent the state and the action taken in the beginning, respectively. The Q-learning algorithm operates on the foundation of the Bellman equation:

$$Q_{t+1}(s,a) \leftarrow Q_t(s,a) + \alpha \times \left[r + \gamma \times \arg\max_{a'} Q_t\left(s',a'\right) - Q_t(s,a) \right].$$
(4)

The Temporal Difference (TD) incremental learning ³⁶ is applied to recursively update the Q-table. An episode in the Q-learning algorithm consists of numerous steps, spanning from the initial state to the terminal state. During each step, the agent operating under a policy chooses an action based on the observation of the environment. Subsequently, it receives an associated reward and transitions to the next state. This iterative process persists until either the action-value function converges or a predetermined number of episodes is completed.

For discrete states, it is relatively straightforward to represent the action-value function as a table. However, this representation becomes increasingly challenging as the number of states grows and is often infeasible when dealing with continuous states. In these scenarios, we can use a parameterized function to represent the action-value function. The task will focus on determining the optimal values for the parameter θ . With this approach, it becomes manageable to utilize a Deep Neural Network (DNN) to approximate the action-value function. Upon completion of training the DNN, it is ready to derive an approximation of the optimal policy by adopting a greedy strategy based on the predicted values.

Reinforcement learning formulation

States

In this context, a state denoted as *S* is defined as a series of actions, which correspond to nodes within a graph denoted as *G*. With the nodes in the graph already being represented by their embeddings, this state can be thought of as a vector in a *p*-dimensional space, mathematically expressed as the summation over all nodes in *V*, i.e., $\sum_{v \in V} \mu_v$. This embedding-based state representation offers the advantage of versatility, as it can be applied across a range of distinct graphs. Moreover, it is important to note that the determination of the terminal state \hat{S} is contingent upon the specific number of nodes requiring vaccination.

Actions

An action denoted as v corresponds to the removal of a node from the current state S within the graph G. Furthermore, we will depict actions by employing their corresponding p-dimensional node embeddings, represented as μ_v . Here, we underscore that this definition remains applicable across graphs of diverse sizes.

Rewards

The primary aim of the exploratory agent is to identify the optimal solution, characterized by the highest resistance to the spreading, at any point within an episode. In a formal context, the reward function, denoted as r(S, v), is represented as the alteration in the influence of the spreading that occurs subsequent to the execution of action v, leading to a transition to the next state S', which can be expressed as:

$$r(S,v) = \frac{I(h(S), G) - I(h(S'), G)}{|V|},$$
(5)

where I(h(S), G) is the number of infected nodes using SIR simulation and $I(h(\phi), G) = |V|$. To address potential variations in reward scales stemming from different graph sizes, the reward is standardized by dividing it by the total number of vertices, denoted |V|. In this work, we chose a discount factor $\gamma = 0.95$, which is a commonly used value in reinforcement learning literature to balance immediate and future rewards effectively³⁷.

Policy

In reinforcement learning, a policy represents the agent's behavioral strategy, which guides the agent in selecting subsequent actions. The policy established is deterministic greedy, which is defined as $\pi(v \mid S) = \operatorname{argmax}_{v' \in \bar{S}} \widehat{Q}(h(S), v')$. Under this policy, taking an action means removing a node from the graph, consequently resulting in the accrual of a reward r(S, v).

Results

To demonstrate the effectiveness of the proposed model, it is compared with Graph Dismantling with Machine learning (GDM) ³⁸, Generalized Network Dismantling (GND) ³⁹, and Explosive Immunization (EI)⁴⁰. The GDM method utilizes machine learning, specifically graph convolutional-style layers, to efficiently dismantle complex networks by identifying and removing critical nodes. The GND method optimizes the process of fragmenting a network by removing nodes with varying costs, utilizing a spectral bisection approach combined with a weighted vertex cover problem to break the network into isolated components. The EI method combines the explosive percolation paradigm with a strategy that incrementally selects nodes to immunize by evaluating their potential to maintain network fragmentation.

The experiment was conducted on an Ubuntu 20.04.4 LTS system with Lenovo ThinkStation, Xeon 5118, 256 GB RAM and NVIDIA RTX 2080 Ti with 12GB memory size. The proposed model was trained on 10 small synthetic random networks, each consisting of 40 nodes with an average degree of 6. The training process took approximately 6 hours.

Comparative analysis of the degradation in network structures

The network connectivity serves as the conduit through which pathogens can propagate to a greater scope. The topological structure of the network determines its suitability for the dissemination of pathogens or information. This section assesses the proposed framework's ability to disrupt network structure ⁴¹ considering five key aspects: edge quantity, connectivity, epidemic threshold, largest component size, and the count of

network components. These dimensions collectively provide a comprehensive evaluation of the framework's impact on network integrity and dynamics. Each dimension is associated with a specific network structural property, and measurements are conducted on the residual network following the removal of a specific number of nodes. The epidemic threshold is a pivotal measure indicating a network's defense against pathogen spread. For a pathogen to successfully propagate, its propagation rate must exceed the network's epidemic threshold ⁴². Hence, networks with higher epidemic thresholds are more challenging for pathogens to infiltrate and propagate within. The largest component in a network refers to the largest set of nodes within a network where any node is reachable from every other node. This component is often of particular interest because it represents the most significant and interconnected part of the network and plays a crucial role in determining the overall structure and connectivity of the network. The component number indicates how many separate groups of nodes exist in a network, each of which is internally connected but not connected to nodes outside their respective group. It is an important topological property of a network and can provide insights into its structure and connectivity.

Graph Dismantling with Machine learning (GDM) ³⁸, Generalized Network Dismantling (GND) ³⁹, and Explosive Immunization (EI)⁴⁰ strategies are used as comparison baselines. To demonstrate the broad versatility of our proposed approach, we carried our experiments on five distinct synthetic network types and four real-world networks. Synthetic networks include Erdős-Rényi (ER) random networks, ER random networks with communities, scale-free networks, scale-free networks with communities, and small-world networks. The presence of communities within a network can significantly impact the spreading phenomenon, as nodes within the same community often exhibit more connections with one another compared to nodes in separate communities. In our experiments, communities are introduced into ER random networks and scale-free networks to assess the adaptability of the proposed method to networks with varying characteristics.

Figures 2 and 3 show that all five metrics of interest are improved (e.g., networks are compromised) after the removal of 20% selected nodes. The proposed strategy consistently outperformed the GDM, GND and EI methods for both the synthetic and real-world networks. The three plots in the top row of Figure 2 illustrate that small-world networks exhibit the highest resilience against structural disruption, while scale-free networks are the most susceptible to immunization. This distinction is primarily due to the presence of prominent highdegree nodes, often referred to as hubs, within scale-free networks, which play a pivotal role in the spread of information or pathogens. Detecting and immunizing these hubs can have a substantial effect on the network's propagation potential. Nevertheless, small-world networks tend to exhibit greater homogeneity and lack conspicuous hubs. Each node can rapidly reach out to others in a small number of steps, making it intricate to disrupt the network structure by immunizing only a limited number of nodes. In contrast, random networks do not display a bias towards hubs or short network distances, falling somewhere in between. It is also worth noting that the proposed approach demonstrates consistent performance across networks with or without community structures, underscoring its resilience to community variations. The performance improvement of the proposed method over the three baseline methods varies across the different real-world datasets illustrated in Figure 3,



Fig. 2. The degradation of network structures resulting from the removal of 20% nodes by different models on synthetic datasets with various properties. Each dimension has been scaled to fall within a range of 0 to 1 to enhance visualization. To ensure consistency in the result display, we use the inverse of component number and epidemic threshold as the measurement. This means a lower reciprocal value signals a stronger ability of the network to prevent the spread of pathogens. The proposed method proves to be the most effective in dismantling the network structure, leaving the smallest area remaining in each dataset.

.....



Fig. 3. The degradation of network structures resulting from the removal of 20% nodes by different models on real-world datasets. Consistent with the findings in synthetic networks, the proposed method demonstrates superior effectiveness in breaking down the network structure, resulting in the smallest area remaining across each dataset.



Fig. 4. The infection scale on five synthetic networks with 20% of nodes removed by various models, where $\beta = 0.1$, $\gamma = 0.01$ and $\rho = 0.1$. Each network consists of 500 nodes. The proposed approach outperforms the baseline methods across all datasets. This is primarily because it more effectively reduces the network's conductivity and leads to smaller peaks in infection scale compared to the baselines. For instance, in the ER random network, immunizing 20% of the nodes identified by the proposed method results in a decrease of the infection scale peak from 86.8% to 75.6%, compared to using the GDM method. These results represent the average of 100 independent runs.

likely due to the distinct characteristics of each dataset. Nevertheless, the proposed method consistently outperforms the baselines in all datasets, highlighting its adaptability and effectiveness.

The spreading simulation on the immunized networks

The SIR (Susceptible-Infected-Recovered) model, as implemented in the EoN (EpidemicsOnNetworks) framework ^{43,44}, is employed to simulate network spreading events across five different synthetic network types and four real-world networks. Simulations are performed with a transmission rate per edge (β) of 0.1, a recovery rate per node (γ) of 0.01, and an initial infection ratio (ρ) of 0.1, representing the proportion of initially infected nodes to the total network nodes.

Figures 4 and 5 present the infection scale outcomes of SIR simulation for the GDM, GND, EI, and the proposed strategies, applied to synthetic and real-world datasets, respectively. The *y*-axis represents the scale of infection over time. All curves follow a similar pattern: they show a swift increase in infection rates at the initial stages, reach a peak, and then gradually decrease to nearly zero. This pattern occurs because there is an abundance of susceptible nodes initially, allowing for a high potential for infection. As the number of susceptible nodes diminishes, the recovery rate surpasses the infection rate, leading to a decline in the infection levels.



Fig. 5. The infection scale on four real-world networks with 20% of nodes removed by various models, where $\beta = 0.1$, $\gamma = 0.01$ and $\rho = 0.1$. The presented approach surpasses the baseline methods in every dataset tested. For example, in the weaver network, targeting 20% of the nodes as identified by this new method leads to a reduction in the peak of infection scale from 83.2% to 35.8%, a notable improvement over the GDM method. These results represent the average of 100 independent runs.

There are distinctions between the timing and magnitude of peak infection for the different networks and strategies. Both the three baseline approaches and the proposed strategy can reduce the peak of infection scale. However, the proposed method outperforms the all the baselines, resulting in a consistently lower peak of infection scale across all different types of networks (i.e., both synthetic and real-world datasets). For example, in the ER random network, immunizing 20% of the nodes identified through the proposed method leads to a reduction in the peak of infection scale from 86.8% to 75.6%, in contrast to employing the GDM method. In the random network with community, the performance of the four methods is comparable to that observed in the ER random network. Here, the infection scale peak for the proposed method is 63.4%, while the GDM method exhibits a peak infection scale of 82.5%. This indicates that the proposed method has strong resilience to community variations.

Comparing the performance of our method in random, scale-free, and small-world networks, the lowest infection peak (43.8%) occurs within the scale-free network, the highest (75.6%) in the random network, and an intermediate peak (61.3%) is observed in the small-world network. This occurs due to the scale-free network's heterogeneous degree distribution, which features more hubs. Our method is capable of efficiently identifying these hubs and preventing the virus spreading in the scale-free network. On the other hand, the small-world network is conducive to rapid virus spreading due to their unique combination of high clustering and short path lengths. It is challenging to stop the spread by immunizing only a subset of nodes. This is mainly because the small-world network contains highly interconnected clusters that enable rapid local spreading, while the presence of shortcuts between distant nodes ensures that the virus can quickly bridge otherwise distant parts of the network.

Figures 6 and 7 show the cumulative infection scale results of the synthetic and real-world datasets, respectively. The presented approach surpasses the baseline methods in every dataset tested, which underscores the versatility and effectiveness of the proposed method. For some networks, the cumulative infection scale is reduced by over 10 percentage points. Table 1 provides a summary of infection scale peak and final infection scale of all the methods.

Another notable observation is that, in random, scale-free and small-world networks, the proposed approach can delay the occurrence of the infection scale peak when compared to the other three strategies. This indicates that the proposed strategy surpasses the other strategies in targeting influential nodes to hinder the spread of disease.

Discussion

Managing the spread of phenomena in complex networks is crucial in fields such as disease spread and viral marketing. An essential element in this context is the challenge of pinpointing key nodes with strong dissemination abilities that are capable of projecting information broadly across the network. Practical observations reveal that common measures of node importance, such as degree and betweenness, fall short in identifying nodes with effective dissemination capabilities. For instance, a node might be connected to many others, but if it is on the network's fringe, its influence is diminished.

Influence maximization and epidemic control in networks are well-known examples of combinatorial optimization problems. IMM ⁴⁵, a traditional method, utilizes a martingale-based statistical approach to estimate influence spread. While traditional methods are typically highly efficient and come with strong theoretical guarantees, they often involve complex dependencies in their analysis and rely on approximations. In recent years, rapid developments in machine learning have paved the way for a new, learning-based approach to solving combinatorial optimization problems. For instance, GLIE ⁴⁶ employs a GNN to address influence maximization with high scalability by predicting the influence of nodes and ranking them to optimize seed set selection. A key practical advantage of neural network approaches is their ability to easily integrate additional node information by incorporating the corresponding embeddings into the input. Another notable work, FINDER ⁴⁷, utilizes a deep Q-learning architecture, where node representations are derived through three GraphSage layers, to solve the network dismantling problem. In this approach, the reward is based on the size of the giant connected component; each node chosen aims to dismantle the network as effectively as possible. FINDER further demonstrates the potential of learning-based methods in tackling combinatorial optimization challenges in complex networks. The proposed approach in this paper combines a graph representation learning model and



Fig. 6. The cumulative infection scale on five synthetic networks with 20% of nodes removed by various models, where $\beta = 0.1$, $\gamma = 0.01$ and $\rho = 0.1$. Each network consists of 500 nodes. The proposed approach outperforms the baseline methods across all datasets. For instance, in the scale-free network, immunizing 20% of the nodes identified by the proposed method results in a decrease of the final infection scale from 95.3% to 69.1%, compared to using the GND method. These results represent the average of 100 independent runs.



Fig. 7. The cumulative infection scale on four real-world networks with 20% of nodes removed by various models, where $\beta = 0.1$, $\gamma = 0.01$ and $\rho = 0.1$. The presented approach surpasses the baseline methods in every dataset tested. For example, in the weaver network, targeting 20% of the nodes as identified by this new method leads to a reduction in the final infection scale from 98.8% to 47.9%, compared to using the GDM method. These results represent the average of 100 independent runs.

DRL to process and learn from graph-structured data and make informed decisions in complex environments. The framework proposed employs a model-free RL approach working in the environment extracted from the network structure by a GNN module, aiming to disrupt network structure and hinder disease spread effectively. It is accomplished by a one-time offline training on small synthetic graphs for a specific application scenario without the need for specialized domain knowledge.

The adaptability and effectiveness of the proposed approach was investigated by conducting experiments on both synthetic and real-world networks. Removing nodes identified by the proposed method enhances the uniformity of the residual network and is more successful in breaking down the network structure than the baseline methods. SIR simulations on networks immunized by this method reveal its higher efficacy in targeting influential nodes to impede disease transmission, significantly lowering the peak of the infection scale, e.g., from 86.8% to 75.6%, compared to using the GDM method in the ER random network. Our experimental results reveal that halting the spread of a virus within a random network presents a more significant challenge than scale-free networks and small-world networks. The final infection scale in the small-world network is 93.4%, in the scale-free network it is 69.1%, and in the random network it is 94.6%. The results also demonstrate the model's effectiveness across various network types and scenarios, emphasizing its potential in real-world applications. This shows an encouraging potential of applying deep learning methods to grasping the fundamental principles governing complex networked systems. This advancement enables the design of networks that are more robust and capable of resisting attacks and breakdowns.

	Peak Infe	ction Scale			Final Infection Scale			
Network	GDM	GND	EI Method	Our Method	GDM	GND	EI Method	Our Method
SYN-SF-100	83.5±0.9	70.4±1.1	69.4±1.3	54.4±2.5	99.2±0.1	90.9±1.1	92.4 <u>±</u> 0.7	80.6±1.5
SYN-SF-500	66.3±1.0	76.4±0.5	67.4 <u>±</u> 0.7	43.8±1.7	89.2 <u>±</u> 0.6	95.3±0.3	90.8±0.5	69.1±1.4
SYN-SF-1000	64.6±0.6	76.0±0.3	67.8 <u>±</u> 0.5	50.9±0.9	86.4 <u>±</u> 0.5	94.8±0.2	90.5 <u>±</u> 0.4	76.0±0.9
SYN-ER-100	87.8±0.6	86.3±0.7	83.6±0.9	82.0±0.8	99.6±0.1	99.5±0.1	99.2 <u>±</u> 0.1	99.1±0.2
SYN-ER-500	86.8±0.3	82.1±0.4	84.3±0.4	75.6±0.5	99.4±0.1	98.3±0.1	99.0 <u>±</u> 0.1	94.6±0.6
SYN-ER-1000	86.9±0.2	82.1±0.3	84.6±0.3	76.0±0.4	99.5±0.0	98.3±0.1	99.0 <u>±</u> 0.1	95.2±0.3
SYN-SW-100	78.4±1.2	72.6±3.3	78.2±1.2	56.4±6.2	99.4±0.2	98.1±0.6	99.5 <u>±</u> 0.2	87.7±6.3
SYN-SW-500	75.8±1.5	74.1±1.4	78.1±0.8	61.3±1.8	99.2±0.2	97.7±1.0	99.6 <u>±</u> 0.1	93.4±0.9
SYN-SW-1000	78.3±0.4	71.5±1.2	78.9 <u>±</u> 0.5	60.2±1.2	99.2±0.1	96.5 ± 1.0	99.7 <u>±</u> 0.0	92.6±0.7
SYN-SF-C-100	77.0±1.1	69.1±2.7	80.2 <u>±</u> 0.9	47.5±3.7	94.5±0.6	90.1±1.1	98.0 <u>±</u> 0.3	69.4±2.6
SYN-SF-C-500	83.2±0.3	67.7±0.7	78.8 <u>±</u> 0.4	43.7±1.0	98.5±0.1	88.2±0.6	97.4 <u>±</u> 0.1	65.2±1.3
SYN-SF-C-1000	82.6±0.3	72.5±0.3	80.5±0.3	52.7±0.5	97.7±0.1	90.1±0.3	97.4 <u>±</u> 0.1	72.4±0.6
SYN-ER-C-100	83.8±1.0	79.2±1.3	80.5±1.5	69.3±2.0	99.4±0.1	97.7±0.4	98.8±0.2	92.8±0.9
SYN-ER-C-500	82.5±0.3	76.4±0.5	80.7±0.3	63.4±1.0	98.9±0.1	97.0±0.3	98.3±0.2	89.5±0.7
SYN- ER-C-1000	82.5±0.2	76.4±0.2	79.8±0.3	63.5±0.7	99.2±0.0	97.5±0.1	98.5±0.1	87.8±0.6
N-netscience	74.2±1.9	59.7±3.3	73.6±2.1	20.6±1.8	91.4±2.1	74.9±4.1	93.8±2.2	26.0±2.4
N-weaver	83.2±1.5	60.1±4.7	77.9±1.7	35.8±6.2	98.8±0.2	76.8±5.1	97.4 <u>±</u> 0.5	47.9±8.2
N-mammalia	70.1±5.0	55.5±4.4	55.9±5.1	24.9±3.5	88.5±5.3	73.8±5.0	76.8 <u>±</u> 6.5	32.6±4.7
N-tortoise	45.2±4.1	28.2±2.7	31.2±2.8	18.0±1.7	60.2 ± 5.8	38.1±3.4	39.8±3.5	23.4±2.2

Table 1. Infection scales for different networks and methods

.....

This research is not without certain limitations. Learning the embeddings can be time consuming when the network is very large, as embeddings need to be learned before each action. The group of individuals whose vaccination can generate the most benefit to the whole society depends upon demographic properties, such as age and profession, which is not included in our framework. This makes it hard to achieve social utility and equity simultaneously. Nonetheless, this study uncovers a more effective strategy for identifying influential nodes in a deep learning manner.

The introduced framework is versatile and can be applied in various scenarios, including halting the spread of infectious diseases or misinformation, preventing cascading failures in cyber-physical systems, and identifying high-risk accounts in financial networks. On the other hand, reinforcing important nodes in large networks can effectively increase the systems' robustness against malfunctions, attacks, and deterioration in the food security ⁴⁸, human gene regulatory network ⁴⁹, and supply chains ⁵⁰.

Finally, this method paves the way for using artificial intelligence to understand the spreading phenomena in complex networked systems, which enables us to design dependable networks from the ground up. For the future research, we could consider utilizing other metrics as rewards, such as IMM ⁴⁵, to enhance the efficiency of the framework. We could also enhance node characteristics by including supplementary details like the node's age, vaccination costs, and vaccine effectiveness. This method will shift the objective from simply identifying influential nodes to incorporating probabilistic features of nodes. As a result, the issue our study tackles will be more practical and closely aligned with real-world scenarios.

Methods Datasets

To assess the versatility and effectiveness of our method, experiments were conducted using both synthetic network and real-world network datasets. The statistical characteristics of these networks are presented in Table 2.

Synthetic networks used include random, scale-free, and small-world. Each has distinctive features. Community structures were also considered in random and scale-free networks, as community structures influence network spreading. Note that the small-world network did not have an accompanying community variant by definition. Random networks were generated via the Erdős-Rényi (ER) model ⁵¹, scale-free networks through the Barabási-Albert (BA) model ⁵², and small-world networks using the Watts-Strogatz (WS) model ⁵³. All the network generators above are available in NetworkX ⁵⁴. Additionally, a random modular network generator ⁵⁵ was employed for creating networks with community features. It was run with a desired value of modularity, *Q*, of 0.5 and number of community, *m*, of 3.

Four real-world network datasets were used in the experiments. Netscience ⁵⁶ is a collaboration network of scientists working on network theory and experiment. In this network, nodes represent scientists, and an edge between two scientists is included if they have co-authored a research paper. Weaver ⁵⁷ describes a real-world animal interaction network, specifically focusing on the social interactions among weaver birds. Each node in the network represents an individual weaver bird from a specific colony observed within a particular year. An edge is drawn between two individual birds if they used the same nest chambers. Mammalia ⁵⁸ is a vole

Network	Туре	Nodes	Edges	Average degree	Max-degree	Clustering	Diameter
SYN-SF-100	Synthetic	100	291	5.8	29	0.14	5
SYN-SF-500	Synthetic	500	1491	6.0	85	0.07	5
SYN-SF-1000	Synthetic	1000	2991	6.0	111	0.04	6
SYN-ER-100	Synthetic	100	309	6.2	14	0.04	5
SYN-ER-500	Synthetic	500	1504	6.0	14	0.01	8
SYN-ER-1000	Synthetic	1000	2990	6.0	14	0.01	8
SYN-SW-100	Synthetic	100	200	4.0	6	0.40	10
SYN-SW-500	Synthetic	500	1000	4.0	7	0.37	16
SYN-SW-1000	Synthetic	1000	2000	4.0	7	0.37	17
SYN-SF-Community-100	Synthetic	100	273	5.5	17	0.26	7
SYN-SF-Community-500	Synthetic	500	1218	4.9	24	0.03	11
SYN-SF-Community-1000	Synthetic	1000	2924	5.8	38	0.02	10
SYN-ER-Community-100	Synthetic	100	238	4.8	12	0.06	6
SYN-ER-Community-500	Synthetic	500	1077	4.3	11	0.01	9
SYN-ER-Community-1000	Synthetic	1000	2149	4.3	11	0.00	10
N-netscience	Real-world	379	914	4.8	34	0.74	17
N-weaver	Real-world	64	177	5.5	21	0.60	6
N-mammalia	Real-world	171	363	4.2	12	0.74	23
N-tortoise	Real-world	283	418	3.0	11	0.43	8

 Table 2.
 Properties of network datasets

interaction network dataset. Each node in the network represents an individual vole. An edge is added between two nodes if those individuals were caught in the same trap at any point during the primary trapping sessions. Tortoise ⁵⁹ is structured to map the interactions among tortoises based on their shared use of burrows. Each node in the network represents an individual tortoise. An edge is established between two tortoise nodes if they share the same burrow.

Metrics of interest

The aim of the proposed approach is to maximize network disruption with a limited number of node removals. To evaluate its effectiveness, five key performance indicators are used to assess the extent of network destruction following node removal by both the proposed method and other baseline approaches.

Edge quantity: In the context of complex networks, the edge quantity refers to the total number of edges (or connections) in a network. Each edge represents a connection or link between two nodes in the network. This metric is crucial in understanding the structure and connectivity of the network, as it indicates how many pairs of nodes are directly connected to each other. A lower edge count in the remaining networks indicates that an approach is more effective in disrupting the network, thereby making it more challenging for a virus to spread.

Count of network components: In a complex network, the count of network components refers to the number of distinct sub-networks or clusters within the entire network that are not connected to each other. Each component is a subset of nodes and edges, where every node in a component is connected directly or indirectly to every other node in the same component, but there are no connections between nodes in different components. This concept is key to understanding the network's structure, as it reveals how the network is partitioned or fragmented. A larger number of components in the residual networks suggests that a method is more efficient in fragmenting the network, consequently enhancing the network's ability to impede the spread of a virus. To ensure consistency in the result display, we use the reciprocal of the number of components as the measurement, such that a smaller reciprocal indicates a greater capability of the network in hindering pathogen propagation.

Size of the largest component: The size of the largest component refers to the number of nodes in the largest sub-network within the entire network. This largest component, also known as the giant component, is the biggest group of nodes that are interconnected either directly or indirectly through a series of edges. The size of this component is a key metric in network analysis, as it reflects the extent of connectivity and can indicate the robustness or vulnerability of the network, especially in the context of processes like information dissemination or disease spread. A smaller size of the largest component in the remaining networks implies that a approach is more effective in breaking down the network.

Epidemic threshold ⁶⁰: Two key factors influencing a pathogen's spread through a network are its spreading rate and the network's epidemic threshold. The spreading rate is influenced by the pathogen's biological traits, while the epidemic threshold represents the network's ability to withstand such a spreading. An epidemic occurs when the pathogen's spreading rate surpasses the network's epidemic threshold; otherwise, the pathogen's spread diminishes. We assess the impact on the network's epidemic threshold following node removal using the proposed approach and other baseline methods. Prior research ⁶⁰ suggested that in an SIS (Susceptible-Infected-Susceptible) model, a network's epidemic threshold, denoted as τ , is determined by the equation

$$\tau = \frac{1}{\lambda_{\max}},\tag{6}$$

where λ_{\max} represents the largest eigenvalue of the network's adjacency matrix. This equation is used to compute the epidemic thresholds for networks.

Average node connectivity ⁶¹: The definition of average node connectivity in a complex network is the average, over all pairs of vertices, of the maximum number of internally disjoint paths connecting these vertices ⁶¹. It can be calculated as

$$\bar{\kappa}(G) = \frac{\sum_{u,v} \kappa_G(u,v)}{\binom{n}{2}},\tag{7}$$

where $\kappa_G(u, v)$ is defined to be the maximum value of k for which node u and node v are k-connected, and n is the size of graph G. Average node connectivity in a network refers to the expected number of nodes that need to be removed to disconnect a pair of non-adjacent nodes. This concept is crucial for assessing the network's resilience. It also focuses on the maximum number of distinct paths that can be formed between any two nodes, offering a deep insight into the network's overall interconnectedness and robustness.

SIR simulation

The Susceptible-Infected-Recovered (SIR) epidemic model is utilized to evaluate the influence of particular nodes on network spread. This model categorizes a population of *N* individuals into three distinct stages:

- 1. Susceptible (S): These individuals have not yet contracted the infection and are vulnerable to catching it.
- 2. Infected (I): Individuals in this category have caught the disease and are capable of transmitting it to the susceptible ones.
- 3. Recovered (R): Once individuals have gone through the infection phase, they are deemed removed from the cycle of the disease, meaning they can neither contract nor spread the infection again. Initially, all nodes are classified as susceptible (S), except for the specific node being scrutinized for its spreading efficiency, which is classified as infected (I). During each time step *t*, each node in the infected state (I) has a probability β (infection rate) of infecting adjacent susceptible nodes. Subsequently, these infected nodes have a probability γ (recovery rate) of moving to the recovered state (R).

This approach allows for a detailed understanding of how specific nodes affect the spread within a network, underlining the dynamics of disease or information propagation in a structured population. The parameters β and γ are crucial in determining the speed and extent of spread within the network. The SIR model is extensively employed in various fields for its effectiveness in simulating the spread of infectious diseases and behavioral patterns in social and information networks.

Data availability

Datasets of Netscience, Weaver, Mammalia, and Tortoise ⁶² are available in https://github.com/zhihaod/GNNR L/tree/main/realdataset.

Code availability

The code accompanying this work is publicly available on the Github repository https://github.com/zhihaod/G NNRL.

Received: 3 April 2024; Accepted: 4 November 2024 Published online: 02 December 2024

References

- 1. Who coronavirus (covid-19) dashboard. (Accessed July 14, 2023). https://covid19.who.int/.
- Kang, Q. et al. Machine learning-aided causal inference framework for environmental data analysis: a covid-19 case study. Environmental Science & Technology 55, 13400–13410 (2021).
- 3. Padma, T. Covid vaccines to reach poorest countries in 2023 despite recent pledges. Nature 595, 342-343 (2021).
- 4. Dong, Z., Chen, Y., Tricco, T. S., Li, C. & Hu, T. Practical strategy of acquaintance immunization without contact tracing. In Proceedings of the 13th IEEE International Conference on Social Computing and Networking (SocialCom), 845–851 (2020).
- 5. Dong, Z., Chen, Y., Tricco, T. S., Li, C. & Hu, T. Hunting for vital nodes in complex networks using local information. *Scientific Reports* 11, 9190 (2021).
- 6. Lü, L. et al. Vital nodes identification in complex networks. Physics Reports 650, 1-63 (2016).
- 7. Lalou, M., Tahraoui, M. A. & Kheddouci, H. The critical node detection problem in networks: A survey. *Computer Science Review* 28, 92–117 (2018).
- 8. Kempe, D., Kleinberg, J. & Tardos, É. Influential nodes in a diffusion model for social networks. In Automata, Languages and Programming: 32nd International Colloquium (ICALP), 1127–1138 (2005).
- 9. Freeman, L. C. Centrality in social networks conceptual clarification. Social Networks 1, 215-239 (1978).
- 10. Sabidussi, G. The centrality index of a graph. Psychometrika 31, 581-603 (1966).
- 11. Brin, S. & Page, L. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems* **30**, 107–117 (1998).
- 12. Kitsak, M. et al. Identification of influential spreaders in complex networks. Nature Physics 6, 888-893 (2010).

- Chen, D., L
 ü, L., Shang, M.-S., Zhang, Y.-C. & Zhou, T. Identifying influential nodes in complex networks. *Physica A: Statistical Mechanics and its Applications* 391, 1777–1787 (2012).
- 14. Zhang, J.-X., Chen, D.-B., Dong, Q. & Zhao, Z.-D. Identifying a set of influential spreaders in complex networks. *Scientific Reports* 6, 27823 (2016).
- Chen, D.-B., Gao, H., Lü, L. & Zhou, T. Identifying influential nodes in large-scale directed networks: the role of clustering. *PloS One* 8, e77455 (2013).
- 16. Lü, L., Zhang, Y.-C., Yeung, C. H. & Zhou, T. Leaders in social networks, the delicious case. PloS One 6, e21202 (2011).
- Weng, J., Lim, E.-P., Jiang, J. & He, Q. Twitterrank: finding topic-sensitive influential twitterers. In Proceedings of the 3rd ACM International Conference on Web Search and Data Mining, 261–270 (2010).
- 18. Kitsak, M. et al. Identification of influential spreaders in complex networks. *Nature Physics* 6, 888–893 (2010).
- Wang, Z., Zhao, Y., Xi, J. & Du, C. Fast ranking influential nodes in complex networks using a k-shell iteration factor. *Physica A: Statistical Mechanics and its Applications* 461, 171–181 (2016).
- Wan, Y.-P., Wang, J., Zhang, D.-G., Dong, H.-Y. & Ren, Q.-H. Ranking the spreading capability of nodes in complex networks based on link significance. *Physica A: Statistical Mechanics and its Applications* 503, 929–937 (2018).
- 21. Yang, F. et al. Identifying the most influential spreaders in complex networks by an extended local k-shell sum. *International Journal of Modern Physics C* 28, 1750014 (2017).
- 22. Ma, L.-L., Ma, C., Zhang, H.-F. & Wang, B.-H. Identifying influential spreaders in complex networks based on gravity formula. *Physica A: Statistical Mechanics and its Applications* **451**, 205–212 (2016).
- 23. Chen, W., Wang, Y. & Yang, S. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 199–208 (2009).
- 24. Domingos, P. & Richardson, M. Mining the network value of customers. In Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 57–66 (2001).
- He, J.-L., Fu, Y. & Chen, D.-B. A novel top-k strategy for influence maximization in complex networks with community structure. *PloS One* 10, e0145283 (2015).
- 26. Newman, M. E. & Girvan, M. Finding and evaluating community structure in networks. Physical Review E 69, 026113 (2004).
- 27. Morone, F. & Makse, H. A. Influence maximization in complex networks through optimal percolation. Nature 524, 65-68 (2015).
- 28. Zhao, G., Jia, P., Huang, C., Zhou, A. & Fang, Y. A machine learning based framework for identifying influential nodes in complex networks. *IEEE Access* 8, 65462–65471 (2020).
- 29. Yu, E.-Y., Fu, Y., Chen, X., Xie, M. & Chen, D.-B. Identifying critical nodes in temporal networks by network embedding. *Scientific Reports* 10, 12494 (2020).
- 30. Khajehnejad, M. *et al.* Adversarial graph embeddings for fair influence maximization over social networks. arXiv:2005.04074 (2020).
- Wei, H. et al. Identifying influential nodes based on network representation learning in complex networks. *PloS one* 13, e0200091 (2018).
- 32. Khalil, E., Dai, H., Zhang, Y., Dilkina, B. & Song, L. Learning combinatorial optimization algorithms over graphs. Advances in Neural Information Processing Systems **30** (2017).
- Dong, Z., Chen, Y., Tricco, T. S., Li, C. & Hu, T. Ego-aware graph neural network. *IEEE Transactions on Network Science and Engineering* 11, 1756–1770 (2024).
- 34. Kipf, T. N. & Welling, M. Semi-supervised classification with graph convolutional networks. In *Proceedings of the International Conference on Learning Representations (ICLR)* (2016).
- 35. Watkins, C. J. & Dayan, P. Q-learning. Machine Learning 8, 279–292 (1992).
- 36. Sutton, R. S. Learning to predict by the methods of temporal differences. Machine Learning 3, 9-44 (1988).
- 37. Sutton, R. S. Reinforcement learning: an introduction. A Bradford Book (2018).
- Grassia, M., De Domenico, M. & Mangioni, G. Machine learning dismantling and early-warning signals of disintegration in complex systems. *Nature Communications* 12, 5190 (2021).
- Ren, X.-L., Gleinig, N., Helbing, D. & Antulov-Fantulin, N. Generalized network dismantling. Proceedings of the National Academy of Sciences 116, 6554–6559 (2019).
- 40. Člusella, P., Grassberger, P., Pérez-Reche, F. J. & Politi, A. Immunization and targeted destruction of networks using explosive percolation. *Physical Review Letters* 117, 208301 (2016).
- Chen, Y. & Dong, Z. Disnet: A general framework for dissolving networks. In Proceedings of International Wireless Communications and Mobile Computing (IWCMC), 1890–1895 (2021).
- 42. Barabási, A.-L. Network Science (Cambridge University Press, 2016).
- 43. Kiss, I. Z., Miller, J. C., Simon, P. L. et al. Mathematics of epidemics on networks. Cham: Springer 598 (2017).
- Miller, J. C. & Ting, T. EoN (Epidemics on Networks): a fast, flexible python package for simulation, analytic approximation, and analysis of epidemics on networks. arXiv:2001.02436 (2020).
- Tang, Y., Shi, Y. & Xiao, X. Influence maximization in near-linear time: A martingale approach. In Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, 1539–1554 (2015).
- Panagopoulos, G., Tziortziotis, N., Vazirgiannis, M. & Malliaros, F. Maximizing influence with graph neural networks. In Proceedings of the International Conference on Advances in Social Networks Analysis and Mining, 237–244 (2023).
- Fan, C., Zeng, L., Sun, Y. & Liu, Y.-Y. Finding key players in complex networks through deep reinforcement learning. *Nature Machine Intelligence* 2, 317–324 (2020).
- Liu, S. et al. Integrating dijkstra's algorithm into deep inverse reinforcement learning for food delivery route planning. Transportation Research Part E: Logistics and Transportation Review 142, 102070 (2020).
- Sha, Z., Chen, Y. & Hu, T. Nspa: characterizing the disease association of multiple genetic interactions at single-subject resolution. Bioinformatics Advances 3, vbad010 (2023).
- 50. Rolf, B. et al. A review on reinforcement learning algorithms and applications in supply chain management. *International Journal of Production Research* 61, 7151–7179 (2023).
- 51. ERDdS, P. & R &wi, A. On random graphs I. Publ. Math. Debrecen 6, 18 (1959).
- 52. Barabási, A.-L. & Albert, R. Emergence of scaling in random networks. *Science* **286**, 509–512 (1999).
- 53. Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world' networks. Nature 393, 440-442 (1998).
- Hagberg, A. A., Schult, D. A. & Swart, P. J. Exploring network structure, dynamics, and function using networkx. In Proceedings of the 7th Python in Science Conference, 11–15 (2008).
- Sah, P., Singh, L. O., Clauset, A. & Bansal, S. Exploring community structure in biological networks with random graphs. BMC Bioinformatics 15, 1–14 (2014).
- 56. Newman, M. E. Finding community structure in networks using the eigenvectors of matrices. Physical Review E 74, 036104 (2006).
- 57. Van Dijk, R. E. et al. Cooperative investment in public goods is kin directed in communal nests of social birds. *Ecology Letters* **17**, 1141–1148 (2014).
- 58. Davis, S., Abbasi, B., Shah, S., Telfer, S. & Begon, M. Spatial analyses of wildlife contact networks. *Journal of the Royal Society Interface* 12, 20141004 (2015).
- Sah, P. et al. Inferring social structure and its drivers from refuge use in the desert tortoise, a relatively solitary species. *Behavioral Ecology and Sociobiology* 70, 1277–1289 (2016).

- Chakrabarti, D., Wang, Y., Wang, C., Leskovec, J. & Faloutsos, C. Epidemic thresholds in real networks. ACM Transactions on Information and System Security 10, 1–26 (2008).
- 61. Beineke, L. W., Oellermann, O. R. & Pippert, R. E. The average connectivity of a graph. Discrete Mathematics 252, 31–45 (2002).
- 62. Rossi, R. & Ahmed, N. The network data repository with interactive graph analytics and visualization. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 29 (2015).

Author contributions

Zhihao Dong and Yuanzhu Chen devised the research project. Zhihao Dong, Yuanzhu Chen, Terrence S. Tricco, Cheng Li and Ting Hu performed the research. Zhihao Dong, Yuanzhu Chen, and Terrence S. Tricco analyzed data and wrote the article.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Y.C.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2024