applied sciences

*Article*

# A New BAT and PageRank Algorithm for Propagation Probability in Social Networks

**Wei-Chang Yeh [1], Wenbo Zhu [2,*], Chia-Ling Huang [3], Tzu-Yun Hsu [1], Zhenyao Liu [1] and Shi-Yi Tan [1]**

[1] Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu 300044, Taiwan; yeh@ieee.org (W.-C.Y.); vm6y3mp6123@gapp.nthu.edu.tw (T.-Y.H.); liuzhenyao49@gmail.com (Z.L.); s108034871@m108.nthu.edu.tw (S.-Y.T.)
[2] School of Mechatronical Engineering and Automation, Foshan University, Foshan 528000, China
[3] Department of International Logistics and Transportation Management, Kainan University, Taoyuan 33857, Taiwan; clhuang@mail.knu.edu.tw
* Correspondence: zhuwenbo@fosu.edu.cn

**Abstract:** Social networks have increasingly become important and popular in modern times. Moreover, the influence of social networks plays a vital role in various organizations, including government organizations, academic research organizations and corporate organizations. Therefore, strategizing the optimal propagation strategy in social networks has also become more important. Increasing the precision of evaluating the propagation probability of social networks can indirectly influence the investment of cost, manpower and time for information propagation to achieve the best return. This study proposes a new algorithm, which includes a scale-free network, Barabási–Albert model, binary-addition tree (BAT) algorithm, PageRank algorithm, Personalized PageRank algorithm and a new BAT algorithm to calculate the propagation probability of social networks. The results obtained after implementing the simulation experiment of social network models show that the studied model and the proposed algorithm provide an effective method to increase the efficiency of information propagation in social networks. In this way, the maximum propagation efficiency is achieved with the minimum investment.

**Keywords:** propagation probability; social networks; Barabási–Albert model; binary-addition tree (BAT) algorithm; PageRank algorithm; Personalized PageRank algorithm

check for updates

**Citation:** Yeh, W.-C.; Zhu, W.; Huang, C.-L.; Hsu, T.-Y.; Liu, Z.; Tan, S.-Y. A New BAT and PageRank Algorithm for Propagation Probability in Social Networks. *Appl. Sci.* **2022**, *12*, 6858. https://doi.org/10.3390/app12146858

Academic Editor: Gianluca Lax

Received: 29 May 2022
Accepted: 5 July 2022
Published: 6 July 2022



## 1. Introduction

With the commercial effect of viral marketing and word-of-mouth marketing, social networks are regarded as a medium for propagating information, ideas and influence between nodes [1–4]. The propagated influence of information propagation in social networks, which use the behavior of user information propagation in social networks to evaluate propagated influence and find the key nodes in order to achieve the greatest influence on social networks during the propagation of these nodes [1,5], is a hot research area [1,5–9]. In other words, investors focus on the high connectivity and high propagation of social networks and accurately invest information in the source nodes of propagation rather than invest many resources in many nodes. The use of social network propagation achieves the purpose of information propagation, which is also an advantage of social networks [10].

In fact, people's lives and sources of information are largely affected by social networks [11], so understanding how information propagates within social networks is crucial for designing effective promotion strategies and preventing the propagation of malicious information. It is found social networks are scale-free networks in the real world [12,13], and the degree distribution of scale-free networks follows the power law distribution [14]. Many practical and important networks—such as the World Wide Web, power grids, etc.—are scale-free networks, which share information between nodes in the network through network connections [14,15].

In the era of social networking, such networks have both flaws and virtues [16]. The benefits include the rapid transmission of information [16] and the development of e-commerce platforms [17], interpersonal relationships [17], etc., while there are also many disadvantages, such as internet viruses [18], rumors [19], etc. The effective promotion of policies, placement of commercial advertisements and prevention of the spread of malicious information in social networks are important topics in social network analysis. For example, if a node with a high probability of spreading is identified when the expected spread is large, better spread can be achieved by placing advertisements on this node. On the contrary, when misinformation spreads among social networks, identifying nodes with a higher probability of spreading in the early stage of dissemination can effectively prevent the dissemination of information in a timely and effective manner. By predicting the source node with the best communication ability in the social network, providing them with information and asking them to use their own influence to publicize that information, a series of communication effects can be triggered.

Therefore, the motivation of this research is to propose a new method for the prediction and modeling of a social network's propagation probability so as to quickly determine the nodes with greater propagation influence in social networks and their propagation probabilities. In addition, the personalization vector in the Personalized PageRank is used to adjust the distribution of node propagation probability in the network, and the influence of user preference on the propagation probability is analyzed.

Much research has been devoted to modeling the propagation process in social networks [20–24]. Among the different models of scale-free networks, the Barabási–Albert model proposed by Barabási and Albert is more common and has many applications [15]. Therefore, this study adopts the Barabási–Albert model to simulate propagation in social networks and generate test data to verify the performance of the proposed algorithm.

The method of measuring the probability of propagation in social networks will vary depending on the focus of research. The most intuitive method is to use the number of node connections with a high probability of page appearance when the number of node connections is large, which is called in-degree [25,26]. However, using this method will lead to a large number of spam links intentionally generated by people in order to increase the probability of the page. The TwitterRank algorithm assigns different propagation weights for Twitter users to track the similarity of different specific topics and web page structures, while the probability will vary with the posting rate of the post [27]. The PageRank algorithm is based on web page links; a page's PageRank value represents random clicks on the page as the probability distribution of reaching any particular node, and the source of its value is based on the PageRank values of all other pages linked to it [28]. This study adopts the PageRank algorithm originally used by Google to analyze the relevance and importance of web pages and uses the PageRank value to measure the importance of nodes to convert it into an appropriate propagation probability. However, studies in [29–31] pointed out that users of social networks have different topic preferences, resulting in different topics with different propagation effects. Therefore, in order to analyze the influence of different user preferences on propagation probability, this study also uses Personalized PageRank to measure the different node weights due to different node themes [32–34].

There are many methods which can be used to search for the propagation path of social networks. Clauset et al. proposed that the hierarchical structure can be inferred from the network and applied to solve the path prediction problem in social networks [35]. Backstrom and Leskovec proposed a prediction algorithm based on a supervised random propagation path; the random path is more convincing because the transfer probability between nodes is different. However, this has not been applied in practice because the model requires significant computing time [36]. Whether based on random paths or fully supervised, these methods are generally complex and difficult-to-understand and are more suitable for problems with a fixed number of sink nodes. However, the propagation process in the social networks studied in this work does not have a fixed sink node, so the propagation problem in social networks to be predicted in this study requires an easy-to-understand new method to predict its propagation path and propagation probability.

This study adopts a binary-addition tree algorithm (BAT) to predict propagation in social networks [37]. BAT algorithms are similar to depth-first search (DFS) [38] and breadth-first search (BFS) [39] algorithms. However, BAT algorithms–in addition to being easy to understand and program, have the main advantage of flexibly in searching all propagation states in the binary network, which makes path exhaustion simpler and easier to change for different applications [37].

This study aims to propose a new method to calculate the probability of propagation areas in social networks and to quickly identify nodes in the network that require close attention. In addition to adopting the traditional PageRank algorithm and the Personalized PageRank algorithm, this study also proposes a novel state concept and a new method based on the combination of a BAT algorithm and the PageRank algorithm to predict the area and probability of possible impacts of propagation in the modeled social network and perform social network analysis (SNA).

Hence, the manuscript's value is emphasized in following list:

1. Firstly, we prove that the social network is a scale-free network and use the Barabási–Albert model to construct a prediction model.
2. After enumerating all possible propagation states through the BAT algorithm, the PageRank algorithm is used to calculate the occurrence probability and propagation probability of each social network node, and the Personalized PageRank algorithm is used to meet the user's preference.
3. Finally, a new BAT algorithm is used to calculate the probability prediction values of all nodes to different propagation ranges.
4. Furthermore, we demonstrate that the maximum propagation state PageRank value proposed in this study can be used as an effective tool for identifying propagation probability and illustrate the influence of user preference topicality on propagation probability.
5. Overall, the value of this prediction model is in providing a new method for predicting social network dissemination, in helping social networks create effective information dissemination effects and in achieving maximum dissemination efficiency with the smallest delivery cost in order to increase the influence of network nodes.

The structure of the remainder of this study is as follows. Section 2 introduces the social network, its propagation model, the scale-free network, the Barabási–Albert model, the PageRank algorithm, the Personalized PageRank algorithm and the BAT algorithm. Section 3 shows the research method and proposes the propagation status in social networks, demonstrates how to numerically present the propagation path and the calculation of the propagation probability, and combines the BAT algorithm with the PageRank algorithm to propose a new BAT algorithm for this study. Propagation in social networks is simulated, and the experimental results are presented to assist the analysis in Section 4. Finally, we summarize the research contributions and identify future work in Section 5.

## 2. Preliminaries

This study uses the Barabási–Albert model to model the scale-free problem of propagation probability in social networks and adopts the modified BAT algorithm combined with the PageRank algorithm to make predictions. Before proceeding to research methods, this section introduces the background of social networks, scale-free networks, the Barabási–Albert model, the PageRank algorithm and the BAT algorithm.

### 2.1. Social Networks

The term "social network" was proposed by Barnes in 1954 [24]; it is used to describe a community structure of the relationship between members in a complex social system and can be presented graphically. Theoretically, a social network should contain two elements: nodes and edges (connection relationships) [23]; the former can represent organizations, communities, individuals, etc., and the latter can represent relationships such as friendships, business partners, etc. In this study, all web pages on the Internet, such as social platforms or fan pages, are regarded as nodes, and hyperlinks are regarded as edges [40].

Due to the availability of large datasets and multidomain applications, social networks have been rapidly developed and applied in many fields in the past decade [41–46]. The customer data of social networks are used to discover social network groups and develop a comprehensive algorithm to enhance the model of the advertising system [41]. Travelers use online social networks to maintain personal information, and backend systems analyze and record the traveler's location to provide the best itinerary according to the route and traveler's preferences [42]. By analyzing the social behavior of social structure nodes, the social attributes of the social network are mined to provide humanized services [43]. Considering the social network in the driving environment, the structure of the video sensor node (VSN) and communication architecture is described to simulate the similar goals of vehicles, passengers and drivers on the road in order to to contribute to future intelligent traffic control [44].

A large number of studies have found that the real social network is a network which appears to be random but actually follows the degree distribution of nodes, which presents a power distribution [47–49]. In research [11], a propagation tree of 227 actual blog page nodes is presented. The propagation tree takes the source node as the center point. Nodes with fewer than four propagation layers have an average branch count greater than 1, while the number of branches for nodes with higher propagation layers is almost 1, as shown in Figure 1.

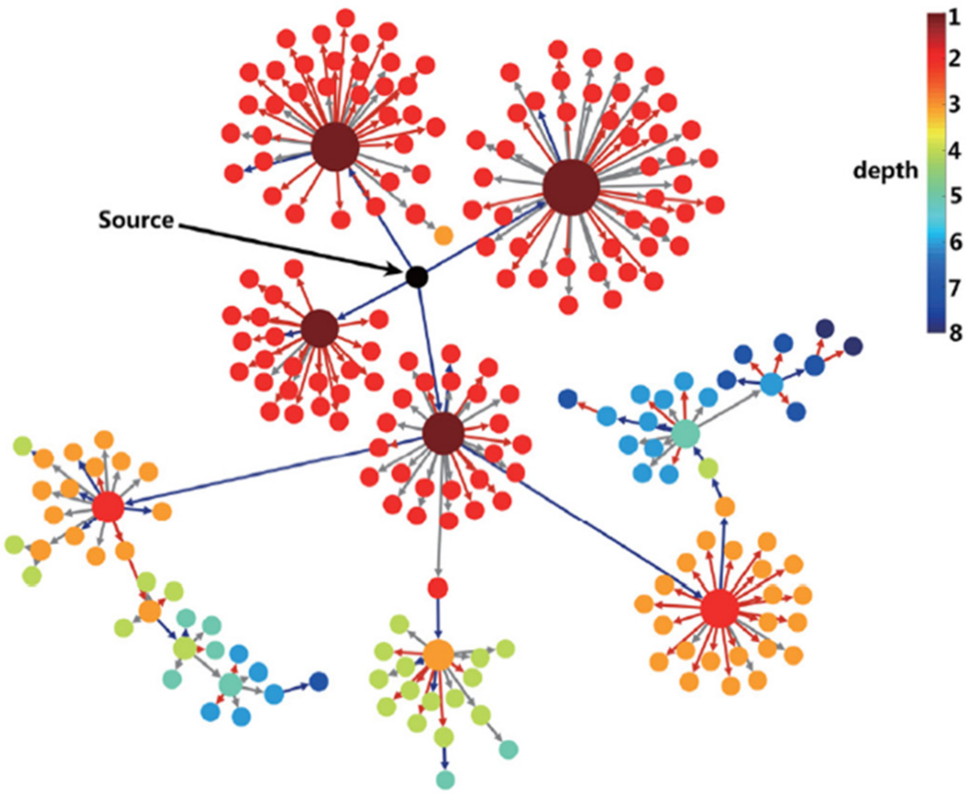



**Figure 1.** Propagation tree and propagation layers of 227 blog page nodes.

Coincidentally, in order to present the backend computing system of its own platform and prove that it has 2.2 billion monthly active users, Facebook presents all Facebook social platform users with blue dots on the black layer and connects the relationship between users with blue lines, which is drawn in Figure 2. An interesting note is that is that the black layer gradually takes on the shape of the world map. Through Figure 2, we can know the actual communication characteristics of the social network.


**Figure 2.** Facebook depicts all connected community users.

In the branch structure of nodes of the propagation tree in Figures 1 and 2, it is found that the propagation path of the social network is centered on the source node. Most nodes in the network are only connected to a few key nodes. Relatively speaking, there are very few key nodes connected to more nodes in the network. A network with this feature is a scale-free network, proposed by Barabási in 1999 [15], and a few key nodes are called a hub.

Many large and important networks are scale-free networks, such as the World Wide Web (WWW) [15], the Internet [50], and the metabolic network [51], for which the degree distribution of the network follows the power distribution. The social network is one of the applications of the WWW [52], and many studies have proved that the social network is a scale-free network [15,53–55].

The degree distribution of a scale-free network is a power distribution [15], and most nodes are only connected to the hub node so that the hub node can master the overall propagation of the social network [56]. Moreno and Vazquez, in their research on whether the proportion of immune nodes on scale-free network can eradicate epidemic viruses, believe that this approach would be better than randomly picking nodes across the network to make them immune if the hub nodes on a scale-free network have immune effects against epidemics [48]. Therefore, mastering hub nodes is also very important in the prediction of propagation in social networks.

### 2.2. Scale-Free Networks and the Barabási–Albert Model

As mentioned in Section 2.1, many scholars have proved that social networks are scale-free networks. In a scale-free network, when a node has $k$ connections, their degree distribution follows a power distribution [15], as in Equation (1).

$$P(k) \sim k^{-\eta} \quad (1)$$

The power distribution has a higher incidence when the value of $k$ is small, and occurrences are very rare when its value grows [13,55]. In the power distribution function, even if the value of $k$ only increases by a small amount, the effect on $P(k)$ can be affected by the power series; when the power is negative, that is $k^{-\eta}$ as shown in Figure 3.

The scale-free web following a power distribution was discovered by Barabási and Albert in their collaborative study of the WWW in 1998. The WWW consists of several highly connected pages. Most of these pages have no more than 4 hyperlinks, but very few pages have thousands of links [15]. Social networks in the real world will grow in size by adding new nodes, which start from a small source node. By adding new nodes, the number of nodes increases regularly throughout the network [13–15,57,58]. For example, the WWW grows exponentially with the addition of new pages, and research citations grow with the publication of new papers, which Barabási and Albert call the scale-free network [15].

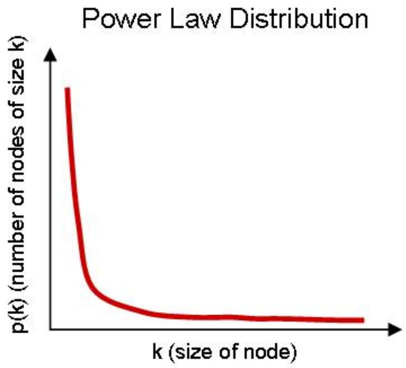



**Figure 3.** Power distribution plot.

In networks in the real world, newly added nodes are generally connected to the nodes with more connections in the original network; therefore, this phenomenon is called "rich-get-richer" [58–61]. For example, new webpages tend to link to already-popular webpages such as the homepage of the website because such webpages are easy to find and widely known. The generation of new nodes in the network and the preferential connection of nodes with more connections in the original network prompted Barabási and Albert to propose the Barabási–Albert model in 1999 [15].

The algorithm of the Barabási–Albert model is based on two modes [59,61], as follows:

1. Growth mode: The network in the real world continues to expand and grow, such as by adding new friends on social platforms or creating new pages on the Internet.
2. Preferential connection mode: New nodes tend to connect to nodes with more connections in the original network when they join. When selecting a new node to connect to, it is assumed that there are $n$ nodes, where $n = 0, 1, 2, \ldots, (n-1)$; the probability of the new node connecting to node $i$, namely $P_i$, depends on the degree of node $i$, namely $k_i$, as in Equation (2).

$$P_i = \frac{k_i}{\sum_{j=0}^{n-1} k_j} \quad (2)$$

To understand how scale-free networks propagate, Barabási and Albert presented a simple process for generating scale-free networks in Figure 4 [61]. Starting from three connected nodes ($t = 1$), the original nodes in the layer are represented by solid lines, and the newly added nodes are represented by hollow circles.

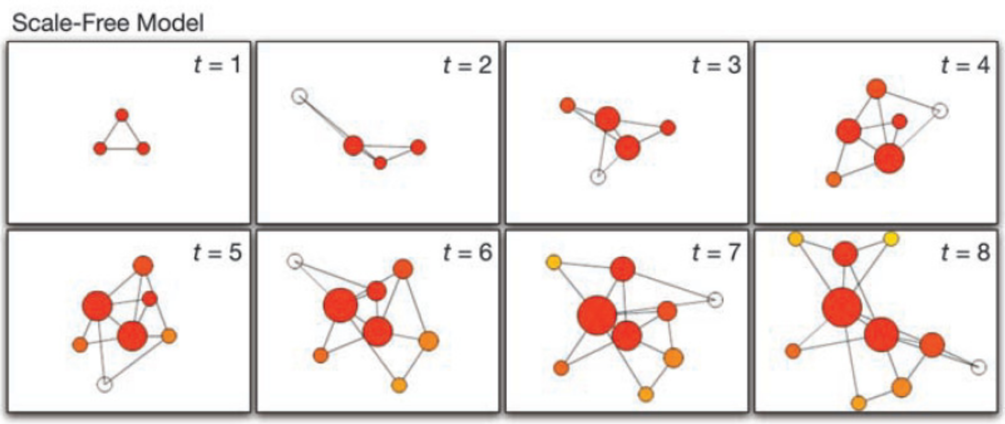



**Figure 4.** Scale-free network generation process.

Due to the growth mode and priority connection mode, the newly added nodes prefer to connect to the original nodes with more connections, leading to the natural output of several highly connected nodes, which means that the probability of selecting a new node is proportional to the number of node connections.

A typical feature of scale-free networks is that the node degree obeys a power distribution. Most nodes in the network are only connected to a few nodes, and very few nodes are connected to most nodes; such key nodes are called hubs. The existence of such nodes makes the scale-free network more resistant to sudden failures, but it is also vulnerable to collective attacks [15], as shown in Figure 5. From Equation (2), it can be seen that most nodes are non-hub nodes and that the degree of connection $k_i$ is very small in the scale-free network. Based on the preferential connection mode, some highly numbered non-hub nodes can be gradually transformed into hubs, as new nodes are more likely to be connected to it [15,57–61]. In other words, all nodes approximately follow a power distribution [14,54].

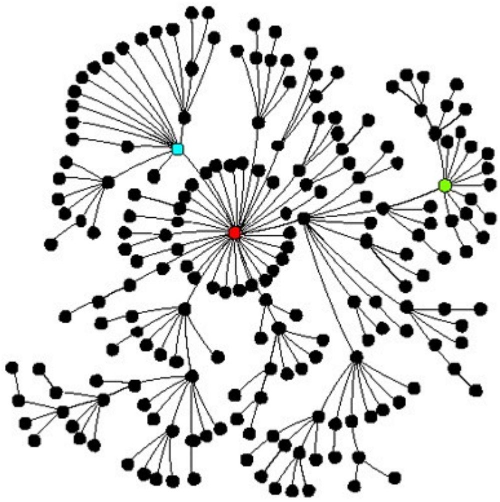


**Figure 5.** Scale-free network.

Among different models of scale-free networks, the Barabási–Albert model proposed by Barabási and Albert is the most common and has many applications for scale-free networks [15]. Therefore, this study uses the Barabási–Albert model to construct a scale-free network to predict the propagation probability of social networks.

### 2.3. PageRank Algorithm

The PageRank algorithm is an algorithm previously invented by Google Inc. to rank the importance of pages in search results by their company's search engine [62,63]. The PageRank value refers to the possibility of a web page being seen. Each web page has an individual PageRank value, which depends on the link relationship between web pages. A higher PageRank value means that a web page is more popular [26,28,62–64].

#### 2.3.1. PageRank

The PageRank algorithm considers the connection structure of a network to measure probability, and its complete definition is as Equation (3) [28]. Equation (3) implements the idea of the PageRank value; that is, the PageRank value of each web page is positively correlated with the PageRank value of web pages linked to the web page [28,62]. Hence, the more pages that are linked to a page, the higher its PageRank value can be. A website with high PageRank value can more easily obtain an even higher PageRank value. The calculation process is shown in Table 1 [65].

$$PR(u) = (1-d) + d \sum_{v \in B(u)} \frac{PR(v)}{N_v} \tag{3}$$

where $u$: web page. $B(u)$: a set of pages connecting $u$. $PR(u)$: the PageRank value of page $u$. $PR(v)$: the PageRank value of page $v$. $N_v$: the number of outgoing links from page $v$. $d$: a damping factor between 0 and 1, with a default value of 0.85. When $d$ = 0.85, there is an 85% probability that Internet users will continue to the next page after visiting a particular page.

**Table 1.** PageRank algorithm.

| | |
|---|---|
| Input: | $G(V, E)$, where $V = \{0, 1, 2, \ldots, u-1\}$. |
| Output: | PageRank value of nodes. |
| Step P0: | Let $t$ = 0, set the initial PageRank value of all nodes to $1/u$. |
| Step P1: | According to Equation (3), update the PageRank values of all nodes. |
| Step P2: | For nodes with $N_v$ = 0, evenly distribute the PageRank values of other nodes. |
| Step P3: | If the PageRank value of all nodes does not change, stop; otherwise, let $t = t + 1$ and go to P0. |

The PageRank algorithm iteratively updates a node's PageRank value by accumulating the PageRank value of each node connected to it, then dividing by the number of the referencing node. First, initialize all PageRank values with the same weight, then update the PageRank values of all nodes with Equation (3). The $N_v$ = 0 in step P2 means that the number of outgoing links from page $v$ is 0, such as going to the page directly through the URL. At this time, the PageRank value of other nodes is evenly distributed. The update is repeated until the PageRank value converges to a unique constant.

### 2.3.2. Personalized PageRank

The PageRank algorithm introduces the concept of random viewers into the calculation process, which gives equal weight to all pages of incoming links. In other words, the PageRank algorithm only uses the link structure of network and cannot judge the similarity in the content of web pages. In addition, the PageRank algorithm distributes the weights evenly according to the number of outgoing links so that the pages that are not related to the theme receive the same attention as the pages that are related to the theme, resulting in a theme shift [66]. No matter how high the PageRank value is, if the page is off-topic, the PageRank value returned to the user has little value [34,66,67].

In order to solve the problem of topic shift in PageRank algorithm, Page et al. proposed a Personalized PageRank method to measure the different importance of nodes [68]. Personalized PageRank affects the assigned weight of each node in the random walk and uses the personalized vector, namely $r_u$, to bias the random walk process to a specific node [32–34,67,69–73], as shown in Equation (4).

$$PR(u) = (1-d)r_u + d \sum_{v \in B(u)} \frac{PR(v)}{N_v} \quad (4)$$

where $u$: web page. $B(u)$: a set of pages connecting $u$. $PR(u)$: the PageRank value of page $u$. $PR(v)$: the PageRank value of page $v$. $N_v$: the number of outgoing links from page $v$. $d$: a damping factor between 0 and 1. $i$: the preferred target page, when $u = i$, $r_u$ = 1; otherwise, $r_u$ = 0.

From Equations (3) and (4), it can be found that the difference between Personalized PageRank and PageRank is that the personalized vector in PageRank moves to any random node with the same probability, while Personalized PageRank adjusts the personalized vector so that random viewers always move to a node or set of nodes that they are interested in rather than from all nodes in the network to any node. If we want random viewers to move to a specific node, such as page $i$, the personalized vector is 1—that is, $r_u$ = 1—and the personalized vectors of the remaining nodes are assigned 0 [32–34].

Personalized PageRank can send random viewers to nodes related to preferences according to user-specified preferences, change node weights [33], and improve the problem of PageRank and topic bias. This study uses Personalized PageRank as a way to adjust the topicality of nodes and explore the impact of user preferences on the probability of propagation.

### 2.4. BAT Algorithm

A binary state network, which indicates that the element state in the network is only two-state (success or failure), is the most basic type of network and is widely used in the path architecture of various fields, including transmission [74,75], network [76–78], data analysis [79] and others [80–84]. Therefore, in recent years, research on binary state networks has been carried out and applied to all the above systems. With the growing scale of networks, a simpler and more efficient algorithm is needed to calculate the reliability of binary state networks to evaluate the performance and stability of networks.

The BAT algorithm was developed by Yeh in 2021 [37], which is an exhaustive method to find all possible state vectors and proposed a path-based hierarchical search algorithm to filter all connection vectors. This algorithm has been used in network reliability [37,85], network resilience [86], wildfire propagation path prediction [87], and others [88–96]. Compared with other path finding algorithms, such as Minimal Path algorithm (MP) or Minimal Cut algorithm (MC), the time complexity of BAT algorithm is the most efficient in reliability calculation of binary state network [37]. In other exhaustive algorithms, the experimental results confirm that the computer memory and efficiency required by the BAT algorithm are also better than the common depth-first search algorithm and breadth-first search algorithm [37].

The BAT algorithm uses simple binary addition to generate all possible state vectors. Since it is a binary state, the value of any coordinate in the state vector is 0 or 1. In terms of social network propagation, when the status is 1, it means that the linked page has been propagated; when the status is 0, it means that the linked page has not been propagated.

Suppose a binary state network graph $G(V, E)$ has $m$ coordinates, and each edge has only two states, i.e., 0 or 1. $V = \{0, 1, 2, \ldots, n-1\}$, $X_k$ is the $k$th obtained state vector, and the $i$th coordinate $X(a_i)$ is the state of one of the edges $a_i$, where $a_i \in E$. The complete BAT algorithm is shown in Table 2 [37].

**Table 2.** BAT algorithm.

| | |
|---|---|
| Input: | $G(V, E)$ |
| Output: | All possible non-repetitive state vectors. |
| Step B0: | Let SUM = 0, $k = 1$, $X_1 = X$, and $X$ is a zero vector with $m$ coordinates. |
| Step B1: | Assume $i = m$. |
| Step B2: | If $X(a_i) = 0$, then $X(a_i) = 1$, $k = k + 1$, $X_k = X$, SUM = SUM + 1, proceed to Step B4. |
| Step B3: | If $i > 1$, let $X(a_i) = 0$, SUM = SUM − 1, $i = i - 1$, and go to Step B2. |
| Step B4: | If SUM = $m$, stop. At this time, $X_1, X_2, \ldots, X_k$ are all possible state vectors. Otherwise, proceed to Step B1. |

As shown in Step B0, all coordinates start with a zero vector. The binary number of $X_k$ is updated by adding 1, as listed in Steps B1 to B3. The exhaustive process can be stopped by repeating the above steps until all coordinates are equal to 1, i.e., SUM = $m$.

The BAT algorithm proposes a more efficient, easy-to-understand, easy-to-program and easy-to-modify path search algorithm for binary state networks. Therefore, we apply the BAT algorithm to the problem formulated in this study.

## 3. Research Methods

The main purpose of this study is to predict the propagation probability of social networks in different propagation ranges and identify the key nodes in the network, which enables the attainment of the maximum propagation benefits in a social network through small investment costs. In this study, the spread of propagation paths in social networks is considered, and its final destination and number of destinations are unknown. Thus, it is difficult to estimate what might occur compared to regular network problems. This section focuses on the state presentation of network propagation and the application of algorithms and establishes a series of research methods.

### 3.1. Mathematics Notations

| | |
|---|---|
| $G(V, E)$ | Graph of binary state network; |
| $V$ | The total number of nodes included in the network graph, $V = \{0, 1, 2, \ldots, n-1\}$; |
| $E$ | The total number of edges included in the network graph; |
| $i$ | Node, $i \in V$; |
| $V(i)$ | The set of adjacent points of node $i$; |
| $Deg(i)$ | The total number of neighbors of node $i$; |
| $PR(i)$ | PageRank value of node $i$; |
| $S_k(i)$ | The kth state of node $i$, $k = 0, 1, \ldots, 2^{\lvert Deg(i) \rvert} - 1$; |
| $X$ | State vector of propagation path; |
| $X(i)$ | The state of node $i$; |
| $PR(S_k(i))$ | PageRank value of state $S_k(i)$; |
| $Pr(S_k(i))$ | The probability of occurrence of state $S_k(i)$; |
| $C(i)$ | Total number of state combinations for node $i$, $C(i) = 2^{\lvert Deg(i) \rvert}$; |
| $PR_{max}(i)$ | The PageRank value of maximum propagation state for node $i$; |
| $N_{page}$ | At least the number of nodes to be propagated including the propagation source node; |
| $Pr(s, N_{page}) = R$ | Probability of propagating to $N_{page}$ nodes starting from $s$; |
| $j$ | Newly propagated node; |
| $s$ | The source node of the propagation; |
| $T_l$ | The set of nodes that have propagated at state stage $l$; |
| $T^*$ | The set of nodes to be propagated at state stage $l$; |
| $P_l$ | Propagation probability for nodes that have already propagated ($T_l$); |
| $R_l$ | The newly added path probability at state stage $l$. |

### 3.2. Propagation Status

The node status of a social network is divided into two types: propagated and not propagated. This section uses a bridge network as an example, as shown in Figure 6. Each node $i$ belongs to the node set $V$ of the network graph, i.e., $i \in V$, and each node can be regarded as a user or page that is propagated in the social network.

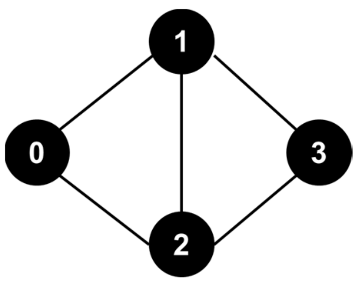



**Figure 6.** Network $G(V, E)$.

Node 0 connects node 1 and node 2, i.e., $Deg(i) = 2$ and $V(i) = \{2, 1\}$. The propagation path of node 0 includes propagation to Ø, {1}, {2} and {1,2}, respectively, representing that information is propagated from node 0 but not propagated; information is propagated from node 0 and is not propagated to other nodes after being propagated to node 1; information is propagated from node 0 and is not propagated to other nodes after being propagated to node 2; information is propagated from node 0 and propagates to node 1 and node 2 and then stops propagating.

The PageRank value of node $i$: $PR(i)$ is obtained using the PageRank algorithm in Section 2.3. Because the PageRank value represents the probability of the node appearing in the network, all PageRank values in the network add up to 1. Table 3 lists the total number of neighbors, the set of neighbors and the PageRank value of node $i$ in the network in Figure 6.

**Table 3.** Basic information of network in Figure 6.

| $i$ | $Deg(i)$ | $V(i)$ | $PR(i)$ |
|---|---|---|---|
| 0 | 2 | {2, 1} | 0.204787 |
| 1 | 3 | {3, 2, 0} | 0.295213 |
| 2 | 3 | {3, 1, 0} | 0.295213 |
| 3 | 2 | {2, 1} | 0.204787 |

### 3.2.1. State Labels and State Vectors

Let $S_k$ ($i$) be the $k$th state of node $i$, which represents the different propagation states of nodes and also known as state labels. Because BAT algorithm representations are all binary—and in order to successfully enumerate all node states without repetition—the binary state labels are converted to node state labels according to the rules in Table 4 [77].

**Table 4.** Transition rules of binary state label.

| | |
|---|---|
| Rule 1: | In binary status labels, the number of bits ($m$) of the state value is equal to $Deg(i)$. |
| Rule 2: | In $V(i)$, node $i$ is in descending node order according to its label number. |
| Rule 3: | In the binary state label, the $m$th digit is equal to 0 or 1, 0 means that the $m$th node is not included in the state, and 1 means that the $m$th node is included in the state. |

Taking the network $G(V, E)$ in Figure 6 as an example, enumerate all state vectors of node 0 by BAT algorithm. According to rule 1, the number of bits of the state value of node 0 is 2. According to rule 2, the $V(0)$ equals {2, 1} by descending arrangement. According to rule 3, the state labels exhaustively enumerated by the BAT algorithm can be represented by binary state labels as 00, 01, 10 and 11, respectively. That is, $S_0(0)$ = Ø, $S_1(0)$ = {1}, $S_2$ (0) = {2}, $S_3(0)$ = {1, 2}. Hence, it can be seen that node 0 is propagated to Ø, {1}, {2} and {1, 2}. Table 5 lists the node state labels and binary state labels of the example network.

**Table 5.** State label of network in Figure 6.

| **State \ Node** | **0** | | **1** | | **2** | | **3** | |
|---|---|---|---|---|---|---|---|---|
| $V(i)$ | **{2, 1}** | | **{3, 2, 0}** | | **{3, 1, 0}** | | **{2, 1}** | |
| $S_k$ ($i$) | **Node State Label** | **Binary State Label** | **Node State Label** | **Binary State Label** | **Node State Label** | **Binary State Label** | **Node State Label** | **Binary State Label** |
| $S_0$ ($i$) | Ø | [0 0] | Ø | [0 0 0] | Ø | [0 0 0] | Ø | [0 0] |
| $S_1$ ($i$) | {1} | [0 1] | {0} | [0 0 1] | {0} | [0 0 1] | {1} | [0 1] |
| $S_2$ ($i$) | {2} | [1 0] | {2} | [0 1 0] | {1} | [0 1 0] | {2} | [1 0] |
| $S_3$ ($i$) | {2, 1} | [1 1] | {2, 0} | [0 1 1] | {1, 0} | [0 1 1] | {2, 1} | [1 1] |
| $S_4$ ($i$) | | | {3} | [1 0 0] | {3} | [1 0 0] | | |
| $S_5$ ($i$) | | | {3, 0} | [1 0 1] | {3, 0} | [1 0 1] | | |
| $S_6$ ($i$) | | | {3, 2} | [1 1 0] | {3, 1} | [1 1 0] | | |
| $S_7$ ($i$) | | | {3, 2, 0} | [1 1 1] | {3, 1, 0} | [1 1 1] | | |

In order to record the propagated path of a social network, this study uses $X$ as the state vector to record the node state label of propagation. The state vector is presented with a diagonal line to identify the propagation state, the right side of diagonal line is the propagation node, and the left side of the diagonal line is the current state of propagation node. Taking $X$ = (2/1) of the network $G(V, E)$ in Figure 6 as an example, it represents state 2 of node 1, which is transferred to node 2, as can be found from Table 5. If $X$ = (2/0, 4/2, 0/3), it means that the propagation starts from node 0, it propagates to node 2 after state 2 of node 0 occurs, then state 4 of node 2 occurs after passing to node 2. and the propagation is passed to node 3—that is, the sink node—and node 3 is in state 0 at this time, indicating that propagation has stopped.

In addition to recording the propagated path in social networks, the source node and sink node can also be clearly determined from the state vector. Exhausting the propagation paths in the form of state vectors can reduce the repetition of the propagation paths on one hand and have a comprehensive understanding of the propagation paths is on the other hand.

### 3.2.2. State Probability and Propagation Probability

In Section 2, it is mentioned that the social network is a scale-free network; that is, it conforms to the Barabási–Albert model with a growth mode and a preferential connection mode. In preferential connection mode, the probability that a new node can be found to connect to node $i$ is proportional to the node's connectivity $k_i$, so the connection probability of a web page is particularly important. Using the PageRank value introduced in Section 2.3 as probability, the $PR(S_k\ (i))$ is defined as the probability of $S_k\ (i)$, called state probability. When this node occurs in state $S_k\ (i)$, define $Pr(S_k\ (i))$ as the probability of occurrence of state $S_k\ (i)$. In other words, $Pr(S_k\ (i))$ can be regarded as the probability of node propagation. $PR(S_k\ (i))$ is to calculate the connection probability obtained by a node according to the PageRank values of all nodes in the state corresponding to node $i$ [82], as shown in Equation (5).

$$PR(S_k(i)) = \sum_{|S_j(i)|=1 and S_j(i)\subseteq S_k(i)} PR(S_j(i)) \quad (5)$$

Because all possible state probabilities must sum to 1, the PageRank value is normalized as in Equation (6).

$$Pr(S_k(i)) = \frac{PR(S_k(i))}{\sum_{k=0}^{2|Deg(i)|-1} PR(S_k(i))} \quad (6)$$

In all nodes, the probability of state 0 is non-transitive. From the preferential connection mode in the Barabási–Albert model, a newly added node can freely connect to any node in the network, and the probability of connecting to two nodes is twice as high as connecting to one node [82]. Hence, the probability of state 0 is defined as Equation (7).

$$PR(S_0(u)) = 0.5 \times \text{Min}\{PR(i), i \in V(u)\} \quad (7)$$

Taking node 1 of the network in Figure 6 as an example, the state probability and propagation probability are shown in Table 6.

**Table 6.** State probabilities and propagation probabilities for node 1 of network in Figure 6.

| State | Propagation Node | $PR(S_k(i))$ | $Pr(S_k(i))$ |
|---|---|---|---|
| $S_0(1)$ | Ø | $0.5 \times \text{Min}\{PR(0), PR(2), PR(3)\}$ $= 0.5 \times 0.204787 = 0.102394$ | 0.102394/2.921542 = 0.0350478 |
| $S_1(1)$ | {0} | $PR(0) = 0.204787$ | 0.204787/2.921542 = 0.0700956 |
| $S_2(1)$ | {2} | $PR(2) = 0.295213$ | 0.295213/2.921542 = 0.101047 |
| $S_3(1)$ | {0, 2} | $PR(0) + PR(2) = 0.5$ | 0.5/2.921542 = 0.171142 |
| $S_4(1)$ | {3} | $PR(3) = 0.204787$ | 0.204787/2.921542 = 0.0700956 |
| $S_5(1)$ | {0, 3} | $PR(0) + PR(3) = 0.409574$ | 0.409574/2.921542 = 0.140191 |
| $S_6(1)$ | {2, 3} | $PR(2) + PR(3) = 0.5$ | 0.5/2.921542 = 0.171142 |
| $S_7(1)$ | {0, 2, 3} | $PR(0) + PR(2) + PR(3) = 0.704787$ | 0.704787/2.921542 = 0.241238 |
| Sum | | 2.921542 | 1 |

According to the above method, the state probability and the propagation probability of the network in Figure 6 are obtained as shown in Tables 7 and 8.

### 3.2.3. PageRank Value of Maximum Propagation State

When a node reaches the maximum propagation state, it means that the node propagates to the subset of all possible nodes of $V(i)$ at the same time. Because the propagation state of each node is obtained by the BAT algorithm, propagation stops when all state vector coordinates are 1 during the exhaustive process so that the maximum propagation state of each node is the last state vector in the process of BAT algorithm. For example, the maximum state of node 0 is to propagate to node 1 and node 2 at the same time; that is, the binary state label is [1, 1], and $C(i)$ is used to present the total number of state combinations. Because it is a binary network model, $C(i) = 2^{|Deg(i)|}$, and the value of $k$ in the state label $S_k(i)$ is encoded from 1 so that $k = 2^{|Deg(i)|} - 1$ for the maximum propagation state.

**Table 7.** State probability for network in Figure 6.

| $i$ | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| $V(i)$ | {2 1} | {3 2 0} | {3 1 0} | {2 1} |
| $PR(S_0(i))$ | 0.147606 | 0.102394 | 0.102394 | 0.147606 |
| $PR(S_1(i))$ | 0.295213 | 0.204787 | 0.204787 | 0.295213 |
| $PR(S_2(i))$ | 0.295213 | 0.295213 | 0.295213 | 0.295213 |
| $PR(S_3(i))$ | 0.590426 | 0.5 | 0.5 | 0.590426 |
| $PR(S_4(i))$ | | 0.204787 | 0.204787 | |
| $PR(S_5(i))$ | | 0.409574 | 0.409574 | |
| $PR(S_6(i))$ | | 0.5 | 0.5 | |
| $PR(S_7(i))$ | | 0.704787 | 0.704787 | |

**Table 8.** Propagation probability for network in Figure 6.

| $i$ | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| $V(i)$ | {2 1} | {3 2 0} | {3 1 0} | {2 1} |
| $PR(S_0(i))$ | 0.111111 | 0.035048 | 0.035048 | 0.111111 |
| $PR(S_1(i))$ | 0.222222 | 0.070096 | 0.070096 | 0.222222 |
| $PR(S_2(i))$ | 0.222222 | 0.101047 | 0.101047 | 0.222222 |
| $PR(S_3(i))$ | 0.444444 | 0.171142 | 0.171142 | 0.444444 |
| $PR(S_4(i))$ | | 0.070096 | 0.070096 | |
| $PR(S_5(i))$ | | 0.140191 | 0.140191 | |
| $PR(S_6(i))$ | | 0.171142 | 0.171142 | |
| $PR(S_7(i))$ | | 0.241238 | 0.241238 | |

The PageRank value of each node represents the importance of the corresponding page, and a page linked to by many pages will have a higher PageRank value. When the node propagation state reaches the maximum state, it propagates to all adjacent nodes at the same time. At this time, let $PR_{max}(i)$ be the PageRank value of the maximum propagation state. Each node has a maximum propagation state. Taking the network in Figure 6 as an example, the maximum propagation state of node 1 is to propagate to node 0, node 2 and node 3 at the same time. At this time, the PageRank value of the maximum propagation state of node 1 is the sum of the PageRank values of node 0, node 2 and node 3 [73]; that is, $PR_{max}(1) = 0.204787 + 0.295213 + 0.204787 = 0.704787$. Table 9 shows the set of adjacent points, the total number of state combinations, the PageRank value and the PageRank value of maximum propagation state for each node of the network in Figure 6.

**Table 9.** Values related to the propagation state of network in Figure 6.

| $i$ | $V(i)$ | $C(i)$ | $PR(i)$ | $PR_{max}(i)$ |
|---|---|---|---|---|
| 0 | {2, 1} | 4 | 0.204787 | 0.295213 + 0.295213 = 0.590426 |
| 1 | {3, 2, 0} | 8 | 0.295213 | 0.204787 + 0.295213 + 0.204787 = 0.704787 |
| 2 | {3, 1, 0} | 8 | 0.295213 | 0.204787 + 0.295213 + 0.204787 = 0.704787 |
| 3 | {2, 1} | 4 | 0.204787 | 0.295213 + 0.295213 = 0.590426 |

### *3.3. A New BAT Algorithm*

The BAT algorithm can satisfy most of the networks with known conditions, such as the path exhaustion of a network with one source node and one sink node, one source node with multiple sink nodes, or multiple source nodes to multiple sink nodes [80]. However, the propagation of social networks involves a source node and an unknown number of sink nodes. In addition to the unknown number of sink nodes, the node propagation state is also unknown. In order to predict the actual propagation situation of a social network and realize a path search by combining different states, the traditional BAT algorithm is modified to increase the suitability of the BAT algorithm to the practical problems discussed in this study. Assuming that information is propagated from node $s$, the purpose is to find all possible situations and the corresponding probabilities that the propagation of social network propagates to at least $N_{page}$ pages. Table 10 shows the process of the new BAT algorithm.

**Table 10.** The new BAT algorithm.

| | |
|---|---|
| Input: | $G(V, E)$, $V(i)$, $S_k(i)$, $s$, $N_{page}$ |
| Output: | $Pr\ (s, N_{page}) = R$ |
| Step A0: | Let $i = s$, $l = 0$, $R = 0$, $T_l = \{ i \}$, $P_l = 1$, $X(i) = 1$, go to Step A1. |
| Step A1: | If $X(i) = 2^{\lvert Deg(i) \rvert}$, then node $i$ has exceeded the maximum propagation state, go to Step A8. |
| Step A2: | Let $T^* = \{j \mid j \in [V(i) - T_l]\}$. |
| Step A3: | Let $T_{l+1} = T_l \cup T^*$. If $T^* = \emptyset$, then go to Step A6. |
| Step A4: | When the number of nodes in $T^* \cup T_l$ is greater than or equal to $N_{page}$, then $R_l = P_l \times Pr(S_{X(i)}(i))$, $R = R + R_l$, and let $l = l + 1$, $T_l = T_{l-1}$ and go to Step A7. |
| Step A5: | $X(j) = 0$ for all $j \in T^*$, $P_{l\,=}\ P_{l-1} \times Pr(S_{X(i)}(i))$. |
| Step A6: | Let $l = l + 1$, if $i$ has the next node in $T_l$, update $i$ to be the next node and go to Step A1. |
| Step A7: | Let $X(i) = X(i) + 1$, go to Step A1. |
| Step A8: | If $i$ has a previous node in $T_l$, update $i$ to be the previous node and go to Step A7; if $i$ does not have a previous node in $T_l$, stop searching, and $R$ is the final propagation probability at this time. |

As shown in Step A0, the initial state vector includes the first node to start propagation, thus, its state starts from $X(i) = 1$. The other nodes that are not the initial propagation node must start from state 1, as shown in Step A5. At this time, Step A1 determines whether the node has reached the maximum state. If the node reaches its maximum state, the algorithm must replace the node or stop searching, as shown in Step A8. If the node does not reach its maximum state, then go to Step A2 and select the unconnected node in the adjacent node set as the node set to be propagated. Step A3 determines whether the node set to be propagated is an empty set. If it is an empty set, go to Step A6 to update the node or go to Step A7 to add the node state. If Step A3 judges that the node set to be propagated has newly added nodes, then go to Step A4 to judge whether it matches the number of $N_{page}$. If it matches the number of $N_{page}$, add its probability $R_l$ to the required $R$, then go to Step A7 to increase the state and continue to search for other propagation states. If the node has reached the maximum state and no other nodes can be updated, stop searching, as shown in Step A8.

For example, the propagation probability of propagating at least 2 pages from node 0 is $Pr\ (s = 0, N_{page} = 2)$. Based on step A2, node 0 has propagated to adjacent node 1, i.e., $\lvert T^* \cup T_l \rvert \geq 2$, which means that this state is guaranteed to be propagated to at least 2 nodes, and we can add the propagation probability of this step to $Pr\ (s = 0, N_{page} = 2)$, as shown in step A4. Then, increase the original current state of $X(i) = 1$ by 1 and continue to search for other possible propagation nodes, as shown in step A7. At this time, if $N_{page} = 3$, the node should be updated to the next node in $T_l$, namely node 1, as shown in step A6. At this point, the state of node 1 will start from $X(i) = 0$ and continue to search for other possible propagation nodes, as shown in step A5. The above analysis is based on steps A4 and A7, which are similar to the steps of the traditional BAT.

Tables 11 and 12 show the calculation process that starts with node 1 and propagates to $N_{page} = 2$ and $N_{page} = 3$, respectively.

**Table 11.** Process of finding $Pr$ ($s$ = 0, $N_{page}$ = 2) for the example network.

| $l$ | $i$ | $X(i)$ | $T_l$ | $T^*$ | $P_l$ | $Pr(S_{X(i)}(i))$ | $R$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | {0} | {1} | 1 | $Pr(S_1(0))$ | $R_0$ |
| 1 | 0 | 2 | {0} | {2} | 1 | $Pr(S_2(0))$ | $R_0 + R_1$ |
| 2 | 0 | 3 | {0} | {1, 2} | 1 | $Pr(S_3(0))$ | $R_0 + R_1 + R_2$ |

**Table 12.** Process of finding $Pr$ ($s$ = 0, $N_{page}$ = 3) for the example network.

| $l$ | $i$ | $X(i)$ | $T_l$ | $T^*$ | $P_l$ | $Pr(S_{X(i)}(i))$ | $R$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | {0} | {1} | 1 | $Pr(S_1(0))$ | 0 |
| 1 | 1 | 0 | {0, 1} | Ø | $Pr(S_1(0))$ | $Pr(S_0(1))$ | 0 |
| 2 | 1 | 1 | {0, 1} | Ø | $Pr(S_1(0))$ | $Pr(S_1(1))$ | 0 |
| 3 | 1 | 2 | {0, 1} | {2} | $Pr(S_1(0))$ | $Pr(S_2(1))$ | $R_3$ |
| 4 | 1 | 3 | {0, 1} | {2} | $Pr(S_1(0))$ | $Pr(S_3(1))$ | $R_3 + R_4$ |
| 5 | 1 | 4 | {0, 1} | {3} | $Pr(S_1(0))$ | $Pr(S_4(1))$ | $R_3 + R_4 + R_5$ |
| 6 | 1 | 5 | {0, 1} | {3} | $Pr(S_1(0))$ | $Pr(S_5(1))$ | $R_3 + R_4 + R_5 + R_6$ |
| 7 | 1 | 6 | {0, 1} | {2, 3} | $Pr(S_1(0))$ | $Pr(S_6(1))$ | $R_3 + R_4 + R_5 + R_6 + R_7$ |
| 8 | 1 | 7 | {0, 1} | {2, 3} | $Pr(S_1(0))$ | $Pr(S_7(1))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8$ |
| 9 | 0 | 2 | {0} | {2} | 1 | $Pr(S_2(0))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8$ |
| 10 | 2 | 0 | {0, 2} | Ø | $Pr(S_2(0))$ | $Pr(S_0(2))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8$ |
| 11 | 2 | 1 | {0, 2} | Ø | $Pr(S_2(0))$ | $Pr(S_1(2))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8$ |
| 12 | 2 | 2 | {0, 2} | {1} | $Pr(S_2(0))$ | $Pr(S_2(2))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8 + R_{12}$ |
| 13 | 2 | 3 | {0, 2} | {1} | $Pr(S_2(0))$ | $Pr(S_3(2))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8 + R_{12} + R_{13}$ |
| 14 | 2 | 4 | {0, 2} | {3} | $Pr(S_2(0))$ | $Pr(S_4(2))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8 + R_{12} + R_{13} + R_{14}$ |
| 15 | 2 | 5 | {0, 2} | {3} | $Pr(S_2(0))$ | $Pr(S_5(2))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8 + R_{12} + R_{13} + R_{14} + R_{15}$ |
| 16 | 2 | 6 | {0, 2} | {1, 3} | $Pr(S_2(0))$ | $Pr(S_6(2))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8 + R_{12} + R_{13} + R_{14} + R_{15} + R_{16}$ |
| 17 | 2 | 7 | {0, 2} | {1, 3} | $Pr(S_2(0))$ | $Pr(S_7(2))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8 + R_{12} + R_{13} + R_{14} + R_{15} + R_{16} + R_{17}$ |
| 18 | 0 | 3 | {0} | {1, 2} | 1 | $Pr(S_3(0))$ | $R_3 + R_4 + R_5 + R_6 + R_7 + R_8 + R_{12} + R_{13} + R_{14} + R_{15} + R_{16} + R_{17} + R_{18}$ |

In Table 11, node 0 is used as the source node for propagation. To calculate $N_{page}$ = 2, it is only necessary to to propagate to 1 node. First, node 0 starts to search for a path from state 1; that is, $S_1(0)$. At this time, according to step A4, it is judged that the propagation of node 0 to node 1 has satisfied $|T^* \cup T_l| \geq 2$, so the probability of this path ($R_0$) can be added to $Pr$ ($s$ = 0, $N_{page}$ = 2), and the state of node 1 is added by 1 according to step A7; that is, $S_2(0)$. The propagation of node 0 to node 2 also satisfies $|T^* \cup T_l| \geq 2$ so the probability of this path ($R_1$) can be added to $Pr$ ($s$ = 0, $N_{page}$ = 2) and the state of node 1 is added by 1 and $S_3(0)$ can be calculated in the same way. When node 0 reaches state 4, it has exceeded $X(i) = 2^{|Deg(i)|} - 1$; that is, the maximum propagation state. Jump to step A8 according to step A2. If there are no other nodes in $T_l$ that can be searched, stop searching. At this time, $R = Pr$ ($s$ = 0, $N_{page}$ = 2) = $R_0 + R_1 + R_2$ is the final propagation probability.

In Table 12, when $N_{page}$ = 3 is calculated with node 0 as the source node for propagation—except for state $S_3(0)$—the remaining two states, namely $S_1(0)$ and $S_2(0)$, must be reconnected to other nodes to reach $N_{page}$ = 3. When $l$ = 0, node 0 starts from state 1 as step A0; that is, it propagates to node 1. At this time, $|T^* \cup T_l|$ is not greater than or equal to $N_{page}$ = 3, so the probability $R_0$ at this time cannot be added to the probability $R$ and enter node 1 to find other connection nodes as shown in step A6. When $l$ = 1, node 1 starts from state $S_0(1)$, $S_0(1)$ is a new empty set and does not satisfy $|T^* \cup T_l| \geq 3$, so probability $R_1$ cannot be added to probability $R$. Similarly, $l$ = 2. When $l$ = 3, node 1 reaches state $S_2(1)$; that is, it propagates to node 2. At this time, $T^*$ adds a new node 2, which satisfies $|T^* \cup T_l| \geqq 3$, so the probability $R_3$ is added to the probability $R = Pr$ ($s$ = 0, $N_{page}$ = 3). The state of node 1 is increased by 1—that is, $S_3(1)$—and continues to find a feasible path, as shown in steps A4 and A7. When node 1 reaches state 7, that is, $S_7(1)$, $X(i) = 2^{|Deg(i)|} - 1$ means node $i$ has reached its maximum state at this time. After calculating $R_8$, find the previous node in $T_l$ to update $i$; that is, node 0. This means that the path under the premise of $S_1(0)$ has been searched as shown in step A8. After updating node $i$ to be node 0, add 1 to its state; that is, $S_2(0)$ = {2}. The steps of subsequent state increase and screening probability are the same as $S_1(0)$. Until there is no previous node in $T_l$ that can be updated and $i$ also reaches the maximum state, stop finding the point. At this time, $R = Pr\ (s = 0, N_{page} = 3)$ is the final propagation probability.

## 4. Simulation Experiment

This study simulates the propagation in a social network and presents the research results in Section 4. The proposed algorithms in this study are all coded in Python with version 3.6, and the simulation experiments are all executed on Spyder using a Windows 10 laptop with an Intel Core i5-8250U CPU (1.60 GHz, 8 GB RAM).

### *4.1. Model Introduction*

NetworkX is a Python package that can randomly generate different types of networks and can also be used to analyze network structures, build network models or draw networks. Using NetworkX to generate Barabási–Albert models with different numbers of nodes can ensure that the network model conforms to the power distribution and preferential connection mode [97], as shown in Figures 7 and 8.

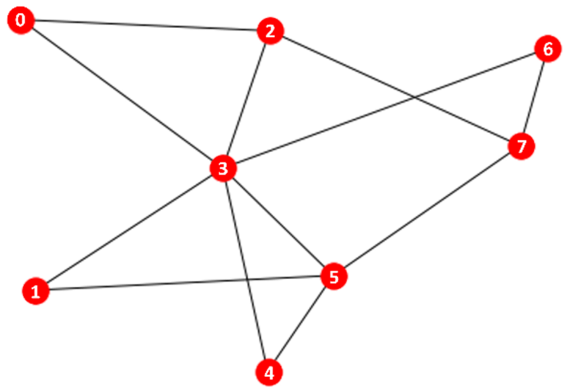



**Figure 7.** Network of Barabási–Albert model with 8 nodes and 12 edges.

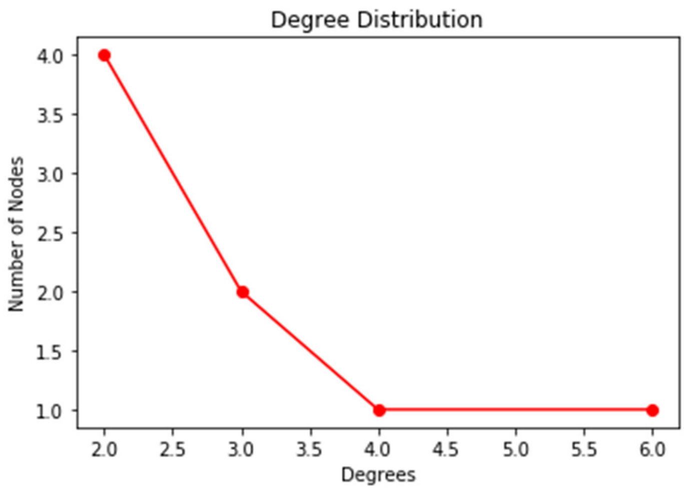



**Figure 8.** Degree distribution of network in Figure 7.

A Barabási–Albert model with 8 nodes and 12 edges is randomly generated by NetworkX in Python as shown in Figure 7. Its degree distribution conforms to the power distribution as shown in Figure 8.

Figure 7 shows the network graph used to simulate the scale-free network in this study, which contains 8 nodes and 12 edges where $i = 0, 1, \ldots, 7$ and $i \in V$. In the experiment, we assume that the source node of propagation starts from $i = 0, 1, \ldots, 7$, respectively, and consider all the propagation probabilities to be $N_{page} = 1, 2, \ldots, 8$. Therefore, this experiment contains a total of 64 experimental results.

Due to the influence of user preference on the propagation probability [29–31], the assigned weight of each node in the random walk varies according to user preference on the web page, as shown in Section 2.3.2. For example, an advertiser plans to advertise a food. One of the nodes in the network of Figure 7 is a food-related website. At this time, the personalization vector of this node is $r_u = 1$, and the personalization vectors of the other nodes are $r_u = 0$. Thus, the model performs an experiment of assigning weights to three different nodes as follows.

Case 1: All nodes are assigned the same weight; that is, the traditional PageRank is used to calculate the page probability, and the propagation probability is $Pr(s, N_{page}) = R_1$.

Case 2: Set node 3, for which the total number of original adjacent nodes is the highest, as the preference node and use Personalized PageRank to calculate the probability of the page; that is, $r_3 = 1$, $r_0 = r_1 = r_2 = r_4 = r_5 = r_6 = r_7 = 0$. At this time, the propagation probability $Pr(s, N_{page})$ equals $R_2$.

Case 3: Set node 1, for which the original PageRank value is the lowest, as the preference node and use the personalized PageRank to calculate the probability of the page; that is, $r_1 = 1$, $r_0 = r_2 = r_3 = r_4 = r_5 = r_6 = r_7 = 0$. At this time, the propagation probability $Pr(s, N_{page})$ equals $R_3$.

In addition to analyzing and predicting the propagation probability of a social network, this section also changes theme of the node in disguise through the personalized vector. If the node is more in line with the user's preference, the allocation weight of the node is increased to interfere with the PageRank value. Table 13 presents the total number of adjacent nodes for node $i$, the set of adjacent nodes, the total number of state combinations, the PageRank value, the PageRank value of maximum propagation state, the Personalized PageRank value and the Personalized PageRank value of the maximum propagation state, which are denoted as $Deg(i)$, $V(i)$, $C(i)$, $PR(i)$, $PR_{max}(i)$, $PPR(i)$ and $PPR_{max}(i)$, respectively.

**Table 13.** Basic information of the network graph in Figure 7.

| Case | | | | Case 1 | | Case 2 ($r_3 = 1$) | | Case 3 ($r_1 = 1$) | |
|---|---|---|---|---|---|---|---|---|---|
| $i$ | $Deg(i)$ | $V(i)$ | $C(i)$ | $PR(i)$ | $PR_{max}(i)$ | $PPR(i)$ | $PPR_{max}(i)$ | $PPR(i)$ | $PPR_{max}(i)$ |
| 0 | 2 | {3 2} | 4 | 0.0879 | 0.3626 | 0.0795 | 0.4526 | 0.0569 | 0.3191 |
| 1 | 2 | {5 3} | 4 | 0.0867 | 0.3990 | 0.0789 | 0.4859 | 0.2226 | 0.4203 |
| 2 | 3 | {7 3 0} | 8 | 0.1254 | 0.4512 | 0.1088 | 0.5162 | 0.0824 | 0.3807 |
| 3 | 6 | {6 5 4 2 1 0} | 64 | 0.2372 | 0.6367 | 0.3438 | 0.5633 | 0.2367 | 0.6762 |
| 4 | 2 | {5 3} | 4 | 0.0867 | 0.3990 | 0.0789 | 0.4859 | 0.0726 | 0.4203 |
| 5 | 4 | {7 4 3 1} | 16 | 0.1618 | 0.5368 | 0.1421 | 0.5945 | 0.1836 | 0.6189 |
| 6 | 2 | {7 3} | 4 | 0.0881 | 0.3633 | 0.0750 | 0.4367 | 0.0582 | 0.3238 |
| 7 | 3 | {6 5 2} | 8 | 0.1261 | 0.3753 | 0.0929 | 0.3260 | 0.0871 | 0.3242 |

*4.2. Results and Discussion*

Table 14 shows the probability of propagation to $N_{page}$ nodes and the running time of Case 1 with node $i$ as the propagation source node.

From Table 14, it can be found that the propagation probability is 100% when $N_{page}$ is 1 because the propagation source node in a social network starts to propagate itself in each of Case 1, Case 2 and Case 3. That is $Pr(i, N_{page} = 1) = 1$. When the propagation range increases—that is, $N_{page}$ increases—the propagation probability of each node decreases, which means that the propagation probability of reaching more nodes is lower and that the propagation probability is negatively correlated with the propagation range.

**Table 14.** Propagation probability results for the network in Figure 7.

| $i$ | $N_{page}$ | $R_1$ Runtime | $R_1$ | $R_2$ | $R_2 - R_1$ | $R_3$ | $R_3 - R_1$ |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0.0000 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 0.0000 |
| | 2 | 0.0020 | 0.9204 | 0.9433 | 0.0229 | 0.9394 | 0.0190 |
| | 3 | 0.0030 | 0.9071 | 0.9345 | 0.0274 | 0.9313 | 0.0242 |
| | 4 | 0.0040 | 0.8923 | 0.9187 | 0.0265 | 0.9197 | 0.0274 |
| | 5 | 0.0120 | 0.8681 | 0.8890 | 0.0209 | 0.8954 | 0.0273 |
| | 6 | 0.0788 | 0.8163 | 0.8261 | 0.0098 | 0.8395 | 0.0233 |
| | 7 | 0.6682 | 0.6992 | 0.6940 | −0.0052 | 0.7164 | 0.0172 |
| | 8 | 4.7575 | 0.4056 | 0.3947 | −0.0109 | 0.4214 | 0.0159 |
| 1 | 1 | 0.0000 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 0.0000 |
| | 2 | 0.0020 | 0.9079 | 0.9319 | 0.0239 | 0.9015 | −0.0064 |
| | 3 | 0.0030 | 0.9007 | 0.9264 | 0.0257 | 0.8884 | −0.0123 |
| | 4 | 0.0040 | 0.8878 | 0.9136 | 0.0259 | 0.8703 | −0.0174 |
| | 5 | 0.0120 | 0.8536 | 0.8734 | 0.0198 | 0.8261 | −0.0274 |
| | 6 | 0.0838 | 0.7997 | 0.8081 | 0.0085 | 0.7610 | −0.0386 |
| | 7 | 0.7416 | 0.6704 | 0.6660 | −0.0044 | 0.6198 | −0.0506 |
| | 8 | 4.6496 | 0.3942 | 0.3836 | −0.0106 | 0.3484 | −0.0458 |
| 2 | 1 | 0.0000 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 0.0000 |
| | 2 | 0.0020 | 0.9762 | 0.9811 | 0.0049 | 0.9817 | 0.0054 |
| | 3 | 0.0030 | 0.9563 | 0.9685 | 0.0122 | 0.9695 | 0.0132 |
| | 4 | 0.0050 | 0.9401 | 0.9542 | 0.0141 | 0.9588 | 0.0187 |
| | 5 | 0.0140 | 0.9171 | 0.9275 | 0.0104 | 0.9380 | 0.0209 |
| | 6 | 0.1087 | 0.8644 | 0.8658 | 0.0013 | 0.8865 | 0.0221 |
| | 7 | 0.8953 | 0.7145 | 0.7020 | −0.0125 | 0.7301 | 0.0156 |
| | 8 | 5.5236 | 0.3756 | 0.3608 | −0.0148 | 0.3849 | 0.0093 |
| 3 | 1 | 0.0000 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 0.0000 |
| | 2 | 0.0030 | 0.9979 | 0.9979 | 0.0000 | 0.9987 | 0.0008 |
| | 3 | 0.0030 | 0.9901 | 0.9891 | −0.0009 | 0.9906 | 0.0005 |
| | 4 | 0.0050 | 0.9674 | 0.9626 | −0.0048 | 0.9674 | 0.0000 |
| | 5 | 0.0199 | 0.9196 | 0.9073 | −0.0123 | 0.9111 | −0.0085 |
| | 6 | 0.1437 | 0.8383 | 0.8169 | −0.0214 | 0.8253 | −0.0131 |
| | 7 | 1.0767 | 0.6683 | 0.6404 | −0.0279 | 0.6514 | −0.0169 |
| | 8 | 6.1967 | 0.3471 | 0.3259 | −0.0213 | 0.3403 | −0.0068 |
| 4 | 1 | 0.0000 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 0.0000 |
| | 2 | 0.0020 | 0.9079 | 0.9319 | 0.0239 | 0.9015 | −0.0064 |
| | 3 | 0.0030 | 0.9007 | 0.9264 | 0.0257 | 0.8961 | −0.0047 |
| | 4 | 0.0030 | 0.8878 | 0.9136 | 0.0259 | 0.8814 | −0.0064 |
| | 5 | 0.0130 | 0.8536 | 0.8734 | 0.0198 | 0.8388 | −0.0148 |
| | 6 | 0.0911 | 0.7997 | 0.8081 | 0.0085 | 0.7786 | −0.0210 |
| | 7 | 0.7126 | 0.6704 | 0.6660 | −0.0044 | 0.6458 | −0.0246 |
| | 8 | 4.5909 | 0.3942 | 0.3836 | −0.0106 | 0.3881 | −0.0061 |
| 5 | 1 | 0.0000 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 0.0000 |
| | 2 | 0.0020 | 0.9900 | 0.9918 | 0.0018 | 0.9927 | 0.0027 |
| | 3 | 0.0030 | 0.9745 | 0.9817 | 0.0072 | 0.9720 | −0.0026 |
| | 4 | 0.0060 | 0.9582 | 0.9683 | 0.0102 | 0.9518 | −0.0063 |
| | 5 | 0.0209 | 0.9233 | 0.9298 | 0.0064 | 0.9068 | −0.0166 |
| | 6 | 0.1337 | 0.8561 | 0.8520 | −0.0041 | 0.8304 | −0.0257 |
| | 7 | 0.9995 | 0.6894 | 0.6736 | −0.0158 | 0.6599 | −0.0294 |
| | 8 | 5.9696 | 0.3574 | 0.3413 | −0.0161 | 0.3447 | −0.0127 |
| 6 | 1 | 0.0000 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 0.0000 |
| | 2 | 0.0030 | 0.9201 | 0.9495 | 0.0293 | 0.9370 | 0.0168 |
| | 3 | 0.0030 | 0.9046 | 0.9387 | 0.0341 | 0.9273 | 0.0228 |
| | 4 | 0.0030 | 0.8940 | 0.9288 | 0.0348 | 0.9186 | 0.0246 |
| | 5 | 0.0100 | 0.8739 | 0.9049 | 0.0310 | 0.8991 | 0.0252 |
| | 6 | 0.0760 | 0.8211 | 0.8423 | 0.0211 | 0.8427 | 0.0215 |
| | 7 | 0.6580 | 0.7012 | 0.7086 | 0.0074 | 0.7161 | 0.0149 |
| | 8 | 4.4182 | 0.4036 | 0.3967 | −0.0069 | 0.4186 | 0.0150 |

**Table 14.** *Cont.*

| $i$ | $N_{page}$ | $R_1$ Runtime | $R_1$ | $R_2$ | $R_2 - R_1$ | $R_3$ | $R_3 - R_1$ |
|---|---|---|---|---|---|---|---|
| 7 | 1 | 0.0000 | 1.0000 | 1.0000 | 0.0000 | 1.0000 | 0.0000 |
| | 2 | 0.0020 | 0.9715 | 0.9720 | 0.0005 | 0.9781 | 0.0065 |
| | 3 | 0.0030 | 0.9463 | 0.9555 | 0.0092 | 0.9617 | 0.0154 |
| | 4 | 0.0049 | 0.9333 | 0.9486 | 0.0153 | 0.9508 | 0.0176 |
| | 5 | 0.0180 | 0.9207 | 0.9382 | 0.0176 | 0.9396 | 0.0189 |
| | 6 | 0.1271 | 0.8725 | 0.8871 | 0.0146 | 0.8930 | 0.0205 |
| | 7 | 0.8932 | 0.7200 | 0.7245 | 0.0045 | 0.7335 | 0.0135 |
| | 8 | 5.7668 | 0.3773 | 0.3721 | −0.0052 | 0.3857 | 0.0084 |

#### 4.2.1. Neighbors and Propagation Probability of Node *i*

From the results of $R_1$ and $R_2$ in Table 14, if the adjacent nodes of two different nodes are the same, the probability of propagating to $N_{page}$ nodes is the same; that is, $Pr(i, N_{page}) = Pr(j, N_{page})$ if $V(i) = V(j)$. For example, in Figure 7, the adjacent nodes of node 1 and node 4 are node 5 and node 3, that is $V(1) = V(4)$. At this time, the probability of propagating to $N_{page} = 1, 2, \ldots, 8$ is all the same that are 1.0000, 0.9079, 0.9007, 0.8878, 0.8536, 0.7997, 0.6704 and 0.3942, respectively. However, in $R_3$, because the personalized vector changes the assigned weight of node 1, the PageRank value of node 1 is increased resulting in different propagation probabilities even if the adjacent nodes of node 1 and node 4 are the same. Therefore, it can be judged that if the adjacent connected nodes are the same and the PageRank values of the nodes are the same, the probability of the final propagation to $N_{page}$ nodes is the same although the propagation paths are different. And the propagation probability is different if the PageRank values are different due to the intervention of the personalized vector.

The total number of adjacent nodes also affects the propagation probability. It can be seen from Table 14 that when the number of adjacent nodes is propagated to a smaller number of $N_{page}$, the node with more total adjacent nodes has a higher propagation probability; on the contrary, when it propagates to a larger number of $N_{page}$, the node with fewer total adjacent nodes has a higher propagation probability. For example, the total number of adjacent nodes of node 3 and node 5 are the first and the second, respectively, which are the nodes with the highest total number of adjacent nodes in Figure 7. Whether in Case 1, Case 2 or Case 3, when $N_{page} = 0, 1, \ldots, 4$, node 3 and node 5 have higher propagation probabilities than other nodes—especially node 3, which has the largest number of adjacent nodes. However, when $N_{page} = 8$, node 3 has the lowest propagation probability compared to other nodes. On the contrary, the total number of adjacent nodes of node 0, node 1, node 4 and node 6 is 2, which is the lowest total number of adjacent nodes in Figure 7. When $N_{page} = 0, 1, \ldots, 7$, node 0 has the lowest propagation probability compared to other nodes. However, when $N_{page}$ =8, the propagation probability of node 0 compared to other nodes: $R_1 = 0.4056$ and $R_3 = 0.4214$ ranks first, and $R_2 = 0.3947$ ranks second. Hence, the node with a larger total number of adjacent nodes can be selected as the source node when a small range must be propagated. However, in the final stage of propagation, nodes with fewer adjacent nodes have a higher probability of propagating to more pages.

#### 4.2.2. PageRank Value and Propagation Probability of Maximum Propagation State

The PageRank value achieving the maximum propagation state means that it will be propagated to all adjacent nodes at the same time if the node is the source node; that is, if the maximum propagation state is reached. The PageRank value reaching the maximum propagation state is also related to the propagation probability. Table 15 shows the ranking of PageRank value of maximum propagation state nodes relative to other nodes in the three cases. Combining the information analysis of Tables 14 and 15, the following results are obtained.

**Table 15.** Ranking for PageRank value of maximum propagation state nodes.

| $i$ | Case 1 $PR_{max}(i)$ | Ranking | Case 2 $PPR_{max}(i)$ | Ranking | Case 3 $PPR_{max}(i)$ | Ranking |
|---|---|---|---|---|---|---|
| 0 | 0.3626 | 7 | 0.4526 | 6 | 0.3191 | 8 |
| 1 | 0.3990 | 4 | 0.4859 | 4 | 0.4203 | 3 |
| 2 | 0.4512 | 3 | 0.5162 | 3 | 0.3807 | 5 |
| 3 | 0.6367 | 1 | 0.5633 | 2 | 0.6762 | 1 |
| 4 | 0.3990 | 4 | 0.4859 | 4 | 0.4203 | 3 |
| 5 | 0.5368 | 2 | 0.5945 | 1 | 0.6189 | 2 |
| 6 | 0.3633 | 6 | 0.4367 | 7 | 0.3238 | 7 |
| 7 | 0.3753 | 8 | 0.3260 | 8 | 0.3242 | 6 |

In terms of a smaller propagation range, the node with a larger PageRank value of maximum propagation state has a higher propagation probability. For the larger propagation range, the node with a smaller PageRank value of maximum propagation state has a higher propagation probability. For example, the PageRank values of the maximum propagation states of node 3 and node 5 are ranked higher than those of other nodes, and the propagation probabilities corresponding to $N_{page} = 1, 2, \ldots, 4$ are all among the top. However, the propagation probabilities of node 3, including $R_1$, $R_2$ and $R_3$, are all the lowest when $N_{page} = 8$. On the other hand, nodes 0, 6 and 7—with lower PageRank values of maximum propagation state—have higher propagation probabilities when the propagation range is large.

It can be seen from the above results that to prevent the propagation of malicious messages, it is recommended to prioritize the protection of nodes with a higher PageRank value of maximum propagation state at the beginning of the propagation of messages, which can effectively and quickly prevent the propagation of messages.

In order to verify that the conclusions in Sections 4.2.1 and 4.2.2 are not mode coincidences, several scale-free networks with different numbers of nodes and edges are built with NetworkX. The models and propagation probabilities are attached in Appendix A Table A1. Judging from the propagation probability of other models, if the set of adjacent nodes of a node is the same, the probability of propagating to $N_{page}$ nodes is the same; when the propagation range is small, nodes with a higher total number of adjacent nodes and a larger PageRank value of maximum propagation state have a higher propagation probability, which is consistent with the above conclusion.

#### 4.2.3. Personalization Vectors and Propagation Probability

In the propagation process of social network, the authority, professionalism, trust and influence of propagation nodes are all related to user preferences, but most influence studies of social networks largely ignore this [31]. In this study, the personalized vector in the Personalized PageRank algorithm is used to intervene in the random viewers in the PageRank algorithm to prefer a specific node in the random walk process so as to increase the distribution weight of the node to represent the user's preference node or relevance of information propagation.

In Case 2, in which node 3 was originally the node with the highest number of adjacent nodes and the highest PageRank value, we set random viewers to prefer a specific node 3 in the random walk process. From Table 11, it can be found that when the personalized vector $r_3 = 1$ of node 3, $PPR(3) = 0.3438$ is higher than the original $PR(3) = 0.2372$. However, in the $R_2$–$R_1$ column in Table 14, it can be found that most of the propagation probabilities increase, while the propagation probability of node 3 decreases. The reason for this is that node 3 connects most of the nodes, resulting in it affecting the PageRank value of maximum propagation state of other nodes, causing them to also increase when the PageRank value of node 3 increases, as shown in Table 16.

**Table 16.** In Case 2, the PageRank value of maximum propagation state increases and decreases.

| $i$ | $PR_{max}(i)$ | $PPR_{max}(i)$ | Increase/Decrease | Rank Change |
|---|---|---|---|---|
| 0 | 0.3626 | 0.4526 | 0.0900 | 7→6 |
| 1 | 0.3990 | 0.4859 | 0.0869 | 4→4 |
| 2 | 0.4512 | 0.5162 | 0.0650 | 3→3 |
| 3 | 0.6367 | 0.5633 | −0.0734 | 1→2 |
| 4 | 0.3990 | 0.4859 | 0.0869 | 4→4 |
| 5 | 0.5368 | 0.5945 | 0.0577 | 2→1 |
| 6 | 0.3633 | 0.4367 | 0.0734 | 6→7 |
| 7 | 0.3753 | 0.3260 | −0.0494 | 8→8 |

In Case 3, in which node 1 was originally the node with the lowest number of adjacent nodes and the lowest PageRank value, we set random viewers to prefer a specific node 1 in the random walk process. When the PageRank value of node 1 increases, it affects the PageRank value of maximum propagation state of other nodes. However, because there are not many nodes connected to node 1, the PageRank value of maximum propagation state does not increase much. As shown in Table 17, the number of nodes with increased propagation probability is small. From the $R_3$–$R_1$ column in Table 14, it can be found that most of the propagation probabilities decrease, which means that increasing the personalized vector of nodes with fewer connections has no obvious effect on the propagation probability of nodes.

**Table 17.** In Case 3, the PageRank value of maximum propagation state increases and decreases.

| $i$ | $PR_{max}(i)$ | $PPR_{max}(i)$ | Increase/Decrease | Rank Change |
|---|---|---|---|---|
| 0 | 0.3626 | 0.3191 | −0.0436 | 7→8 |
| 1 | 0.3990 | 0.4203 | 0.0213 | 4→3 |
| 2 | 0.4512 | 0.3807 | −0.0705 | 3→5 |
| 3 | 0.6367 | 0.6762 | 0.0395 | 1→1 |
| 4 | 0.3990 | 0.4203 | 0.0213 | 4→3 |
| 5 | 0.5368 | 0.6189 | 0.0821 | 2→2 |
| 6 | 0.3633 | 0.3238 | −0.0395 | 6→7 |
| 7 | 0.3753 | 0.3242 | −0.0511 | 8→6 |

When we bias the random viewers to the nodes with more adjacent nodes in the random walk process, the propagation probability of more nodes can be increased. However, increasing the distribution weight of nodes with fewer adjacent nodes does not have much influence. Therefore, it is assumed that an advertiser plans to place an advertisement and that there are two nodes with the same theme. If a node with a higher number of adjacent nodes is selected, it relatively increases the probability of propagating more pages.

## 5. Conclusions

The contribution of this research is to propose a new method, which is novel, simple and easy to apply, to calculate the propagation probability of social network propagation to other web pages. After confirming that the social network is a scale-free network, the Barabási–Albert model is used to build a model. Then, the PageRank algorithm and the Personalized PageRank algorithm are adopted to calculate the occurrence probability and propagation probability of each node. After enumerating all possible propagation states of each node using the BAT algorithm, the new BAT algorithm is used to calculate the propagation probability under different propagation ranges.

The new BAT algorithm can calculate the predicted value of $Pr\ (s, N_{page})$ for all nodes $i$ and $N_{page}$; that is, the propagation probability of a social network, as shown in Table 12. In addition, the personalized vector in the Personalized PageRank algorithm is used to intervene the preferences of random viewers in the random walk process so that the prediction model is more in line with the actual user situation.

The Barabási–Albert model is used to construct a scale-free network to simulate the spread of social networks. However, the amount of computation will increase exponentially according to the network scale and spread because the degree distribution of the scale-free network follows a power distribution. The limitation of this model is that running time increases with the expansion of the network scale. If it must be applied to a large scale-free network, the GPU requirement must be increased to overcome the computing speed, or it can be applied to a small network such as a company intranet, etc.

The method proposed in this study could be applied in many areas, such as to cybersecurity problems. The proposed scale-free network model, new BAT algorithm, PageRank value of maximum propagation state and personalized vector concept can be extended to other network problems in the future.

**Author Contributions:** Conceptualization, W.-C.Y., W.Z., C.-L.H., T.-Y.H., Z.L. and S.-Y.T.; methodology, W.-C.Y., W.Z., C.-L.H. and T.-Y.H.; software, W.-C.Y., W.Z., C.-L.H. and T.-Y.H.; validation, W.-C.Y., W.Z., C.-L.H. and T.-Y.H.; formal analysis, W.-C.Y., W.Z., C.-L.H. and T.-Y.H.; investigation, W.-C.Y., W.Z. and C.-L.H.; resources, W.-C.Y., W.Z., C.-L.H. and T.-Y.H.; data curation, W.-C.Y., W.Z., C.-L.H. and T.-Y.H.; writing—original draft preparation, W.-C.Y., W.Z., C.-L.H. and T.-Y.H.; writing—review and editing, W.-C.Y., W.Z. and C.-L.H.; visualization, W.-C.Y. and W.Z.; supervision, W.-C.Y. and W.Z.; project administration, W.-C.Y. and W.Z.; funding acquisition, W.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported in part by National Natural Science Foundation of China (Grant No. 621060482), Research and Development Projects in Key Areas of Guangdong Province (Grant No. 2021B0101410002) and Ministry of Science and Technology, R.O.C (MOST 107-2221-E-007-072-MY3, MOST 110-2221-E-007-107-MY3, MOST 109-2221-E-424-002 and MOST 110-2511-H-130-002).

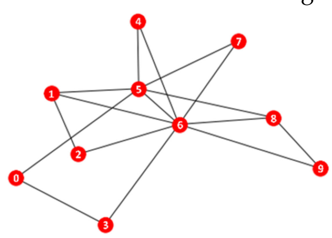
**Acknowledgments:** We wish to thank the anonymous editor and the referees for their constructive comments and recommendations, which significantly improved this paper. This article was once submitted to arXiv as a temporary submission that was only for reference and did not provide the copyright.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

**Table A1.** Propagation probability results for different numbers of nodes and edges.

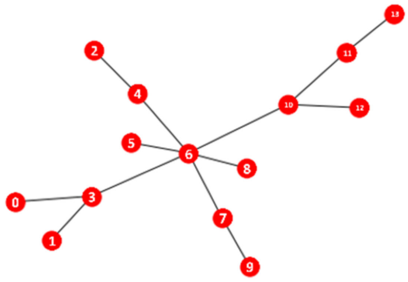
Model 1: 10 nodes and 16 edges.




| Npage | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Node 0 | 1 | 0.9343 | 0.9112 | 0.9078 | 0.9045 | 0.8976 | 0.8692 | 0.7730 | 0.5548 | 0.2411 |
| Node 1 | 1 | 0.9830 | 0.9754 | 0.9725 | 0.9660 | 0.9486 | 0.8938 | 0.7505 | 0.4958 | 0.2067 |
| Node 2 | 1 | 0.9335 | 0.9264 | 0.9239 | 0.9177 | 0.9035 | 0.8610 | 0.7411 | 0.5065 | 0.2279 |
| Node 3 | 1 | 0.9458 | 0.9240 | 0.9219 | 0.9168 | 0.9066 | 0.8751 | 0.7766 | 0.5564 | 0.2408 |
| Node 4 | 1 | 0.9027 | 0.9014 | 0.8983 | 0.8925 | 0.8769 | 0.8259 | 0.7003 | 0.4657 | 0.2137 |
| Node 5 | 1 | 0.9984 | 0.9940 | 0.9889 | 0.9823 | 0.9604 | 0.8893 | 0.7256 | 0.4599 | 0.1879 |
| Node 6 | 1 | 0.9996 | 0.9979 | 0.9924 | 0.9812 | 0.9540 | 0.8807 | 0.7184 | 0.4560 | 0.1867 |
| Node 7 | 1 | 0.9027 | 0.9014 | 0.8983 | 0.8925 | 0.8769 | 0.8259 | 0.7003 | 0.4657 | 0.2137 |
| Node 8 | 1 | 0.9830 | 0.9754 | 0.9725 | 0.9660 | 0.9486 | 0.8938 | 0.7505 | 0.4958 | 0.2067 |
| Node 9 | 1 | 0.9335 | 0.9264 | 0.9239 | 0.9177 | 0.9035 | 0.8610 | 0.7411 | 0.5065 | 0.2279 |

**Table A1.** *Cont.*

Model 2: 14 nodes and 13 edges.

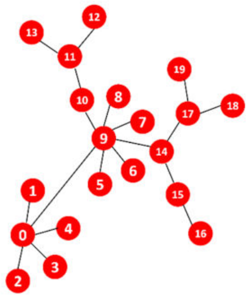



| Node \ Npage | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Node 0 | 1 | 0.6667 | 0.6309 | 0.5570 | 0.5424 | 0.5007 | 0.4254 | 0.3264 | 0.2233 | 0.1329 |
| Node 1 | 1 | 0.6667 | 0.6309 | 0.5570 | 0.5424 | 0.5007 | 0.4254 | 0.3264 | 0.2233 | 0.1329 |
| Node 2 | 1 | 0.6667 | 0.5836 | 0.5797 | 0.5588 | 0.5089 | 0.4273 | 0.3247 | 0.2206 | 0.1307 |
| Node 3 | 1 | 0.9821 | 0.9090 | 0.8256 | 0.7852 | 0.6995 | 0.5702 | 0.4188 | 0.2729 | 0.1536 |
| Node 4 | 1 | 0.9585 | 0.8727 | 0.8553 | 0.8043 | 0.7080 | 0.5713 | 0.4164 | 0.2699 | 0.1524 |
| Node 5 | 1 | 0.6667 | 0.6639 | 0.6464 | 0.6002 | 0.5191 | 0.4127 | 0.2950 | 0.1886 | 0.1050 |
| Node 6 | 1 | 0.9986 | 0.9858 | 0.9400 | 0.8463 | 0.7061 | 0.5376 | 0.3681 | 0.2240 | 0.1183 |
| Node 7 | 1 | 0.9585 | 0.8727 | 0.8553 | 0.8043 | 0.7080 | 0.5713 | 0.4164 | 0.2699 | 0.1524 |
| Node 8 | 1 | 0.6667 | 0.6639 | 0.6464 | 0.6002 | 0.5191 | 0.4127 | 0.2950 | 0.1886 | 0.1050 |
| Node 9 | 1 | 0.6667 | 0.5836 | 0.5797 | 0.5588 | 0.5089 | 0.4273 | 0.3247 | 0.2206 | 0.1307 |
| Node 10 | 1 | 0.9844 | 0.9280 | 0.8463 | 0.7596 | 0.6886 | 0.5729 | 0.4260 | 0.2762 | 0.1521 |
| Node 11 | 1 | 0.9325 | 0.7726 | 0.6960 | 0.6306 | 0.6004 | 0.5329 | 0.4259 | 0.2974 | 0.1762 |
| Node 12 | 1 | 0.6667 | 0.6354 | 0.5934 | 0.5215 | 0.4884 | 0.4243 | 0.3321 | 0.2278 | 0.1332 |
| Node 13 | 1 | 0.6667 | 0.5317 | 0.4917 | 0.4250 | 0.4139 | 0.3811 | 0.3190 | 0.2346 | 0.1472 |

| Node \ Npage | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Node 0 | 0.0662 | 0.0260 | 0.0071 | 0.0011 | | | | | | |
| Node 1 | 0.0662 | 0.0260 | 0.0071 | 0.0011 | | | | | | |
| Node 2 | 0.0664 | 0.0272 | 0.0080 | 0.0012 | | | | | | |
| Node 3 | 0.0716 | 0.0260 | 0.0065 | 0.0009 | | | | | | |
| Node 4 | 0.0730 | 0.0278 | 0.0074 | 0.0010 | | | | | | |
| Node 5 | 0.0497 | 0.0189 | 0.0052 | 0.0008 | | | | | | |
| Node 6 | 0.0527 | 0.0186 | 0.0047 | 0.0006 | | | | | | |
| Node 7 | 0.0730 | 0.0278 | 0.0074 | 0.0010 | | | | | | |
| Node 8 | 0.0497 | 0.0189 | 0.0052 | 0.0008 | | | | | | |
| Node 9 | 0.0664 | 0.0272 | 0.0080 | 0.0012 | | | | | | |
| Node 10 | 0.0688 | 0.0242 | 0.0061 | 0.0008 | | | | | | |
| Node 11 | 0.0852 | 0.0318 | 0.0083 | 0.0011 | | | | | | |
| Node 12 | 0.0645 | 0.0244 | 0.0067 | 0.0010 | | | | | | |
| Node 13 | 0.0757 | 0.0302 | 0.0086 | 0.0013 | | | | | | |

**Table A1.** *Cont.*

Model 3: 20 nodes and 19 edges.




| Npage | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Node 0 | 1 | 0.9968 | 0.9708 | 0.8913 | 0.8025 | 0.7439 | 0.6819 | 0.5957 | 0.4970 | 0.3929 |
| Node 1 | 1 | 0.6667 | 0.6602 | 0.6213 | 0.5539 | 0.5108 | 0.4788 | 0.4269 | 0.3633 | 0.2949 |
| Node 2 | 1 | 0.6667 | 0.6602 | 0.6213 | 0.5539 | 0.5108 | 0.4788 | 0.4269 | 0.3633 | 0.2949 |
| Node 3 | 1 | 0.6667 | 0.6602 | 0.6213 | 0.5539 | 0.5108 | 0.4788 | 0.4269 | 0.3633 | 0.2949 |
| Node 4 | 1 | 0.6667 | 0.6602 | 0.6213 | 0.5539 | 0.5108 | 0.4788 | 0.4269 | 0.3633 | 0.2949 |
| Node 5 | 1 | 0.6667 | 0.6655 | 0.6557 | 0.6300 | 0.5915 | 0.5347 | 0.4554 | 0.3687 | 0.2867 |
| Node 6 | 1 | 0.6667 | 0.6655 | 0.6557 | 0.6300 | 0.5915 | 0.5347 | 0.4554 | 0.3687 | 0.2867 |
| Node 7 | 1 | 0.6667 | 0.6655 | 0.6557 | 0.6300 | 0.5915 | 0.5347 | 0.4554 | 0.3687 | 0.2867 |
| Node 8 | 1 | 0.6667 | 0.6655 | 0.6557 | 0.6300 | 0.5915 | 0.5347 | 0.4554 | 0.3687 | 0.2867 |
| Node 9 | 1 | 0.9994 | 0.9931 | 0.9675 | 0.9194 | 0.8486 | 0.7475 | 0.6228 | 0.4955 | 0.3760 |
| Node 10 | 1 | 0.9241 | 0.9016 | 0.8218 | 0.7460 | 0.7123 | 0.6589 | 0.5815 | 0.4880 | 0.3881 |
| Node 11 | 1 | 0.9670 | 0.8099 | 0.5975 | 0.5400 | 0.5293 | 0.5038 | 0.4627 | 0.4077 | 0.3379 |
| Node 12 | 1 | 0.6667 | 0.6006 | 0.4251 | 0.3614 | 0.3578 | 0.3454 | 0.3217 | 0.2890 | 0.2464 |
| Node 13 | 1 | 0.6667 | 0.6006 | 0.4251 | 0.3614 | 0.3578 | 0.3454 | 0.3217 | 0.2890 | 0.2464 |
| Node 14 | 1 | 0.9778 | 0.9492 | 0.8799 | 0.8124 | 0.7379 | 0.6683 | 0.5995 | 0.5108 | 0.4093 |
| Node 15 | 1 | 0.9313 | 0.7719 | 0.7313 | 0.6871 | 0.6155 | 0.5613 | 0.5201 | 0.4603 | 0.3834 |
| Node 16 | 1 | 0.6667 | 0.5292 | 0.4939 | 0.4784 | 0.4291 | 0.3836 | 0.3609 | 0.3266 | 0.2788 |
| Node 17 | 1 | 0.9726 | 0.8513 | 0.7005 | 0.6422 | 0.5848 | 0.5423 | 0.5056 | 0.4512 | 0.3805 |
| Node 18 | 1 | 0.6667 | 0.6119 | 0.4835 | 0.4451 | 0.4052 | 0.3698 | 0.3503 | 0.3191 | 0.2761 |
| Node 19 | 1 | 0.6667 | 0.6119 | 0.4835 | 0.4451 | 0.4052 | 0.3698 | 0.3503 | 0.3191 | 0.2761 |

| Npage | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Node 0 | 0.2897 | 0.1976 | 0.1248 | 0.0726 | 0.0380 | 0.0172 | 0.0064 | 0.0018 | 0.0003 | $2.77 \times 10^{-5}$ |
| Node 1 | 0.2244 | 0.1576 | 0.1023 | 0.0615 | 0.0336 | 0.0160 | 0.0063 | 0.0019 | 0.0004 | $3.51 \times 10^{-5}$ |
| Node 2 | 0.2244 | 0.1576 | 0.1023 | 0.0615 | 0.0336 | 0.0160 | 0.0063 | 0.0019 | 0.0004 | $3.51 \times 10^{-5}$ |
| Node 3 | 0.2244 | 0.1576 | 0.1023 | 0.0615 | 0.0336 | 0.0160 | 0.0063 | 0.0019 | 0.0004 | $3.51 \times 10^{-5}$ |
| Node 4 | 0.2244 | 0.1576 | 0.1023 | 0.0615 | 0.0336 | 0.0160 | 0.0063 | 0.0019 | 0.0004 | $3.51 \times 10^{-5}$ |
| Node 5 | 0.2101 | 0.1425 | 0.0895 | 0.0522 | 0.0279 | 0.0133 | 0.0053 | 0.0016 | 0.0003 | $3.05 \times 10^{-5}$ |
| Node 6 | 0.2101 | 0.1425 | 0.0895 | 0.0522 | 0.0279 | 0.0133 | 0.0053 | 0.0016 | 0.0003 | $3.05 \times 10^{-5}$ |
| Node 7 | 0.2101 | 0.1425 | 0.0895 | 0.0522 | 0.0279 | 0.0133 | 0.0053 | 0.0016 | 0.0003 | $3.05 \times 10^{-5}$ |
| Node 8 | 0.2101 | 0.1425 | 0.0895 | 0.0522 | 0.0279 | 0.0133 | 0.0053 | 0.0016 | 0.0003 | $3.05 \times 10^{-5}$ |
| Node 9 | 0.2672 | 0.1760 | 0.1076 | 0.0609 | 0.0314 | 0.0142 | 0.0053 | 0.0015 | 0.0003 | $2.38 \times 10^{-5}$ |
| Node 10 | 0.2893 | 0.2014 | 0.1313 | 0.0790 | 0.0426 | 0.0197 | 0.0074 | 0.0021 | 0.0004 | $3.33 \times 10^{-5}$ |
| Node 11 | 0.2596 | 0.1866 | 0.1268 | 0.0802 | 0.0453 | 0.0217 | 0.0082 | 0.0023 | 0.0004 | $3.59 \times 10^{-5}$ |

**Table A1.** *Cont.*

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Node 12 | 0.1943 | 0.1422 | 0.0985 | 0.0643 | 0.0379 | 0.0191 | 0.0077 | 0.0023 | 0.0004 | $4.14 \times 10^{-5}$ |
| Node 13 | 0.1943 | 0.1422 | 0.0985 | 0.0643 | 0.0379 | 0.0191 | 0.0077 | 0.0023 | 0.0004 | $4.14 \times 10^{-5}$ |
| Node 14 | 0.3056 | 0.2110 | 0.1336 | 0.0767 | 0.0395 | 0.0178 | 0.0066 | 0.0018 | 0.0003 | $2.95 \times 10^{-5}$ |
| Node 15 | 0.2989 | 0.2164 | 0.1431 | 0.0857 | 0.0464 | 0.0222 | 0.0087 | 0.0025 | 0.0005 | $4.14 \times 10^{-5}$ |
| Node 16 | 0.2227 | 0.1660 | 0.1134 | 0.0699 | 0.0390 | 0.0195 | 0.0081 | 0.0025 | 0.0005 | $4.7 \times 10^{-5}$ |
| Node 17 | 0.2987 | 0.2157 | 0.1422 | 0.0854 | 0.0460 | 0.0215 | 0.0081 | 0.0023 | 0.0004 | $3.52 \times 10^{-5}$ |
| Node 18 | 0.2235 | 0.1666 | 0.1133 | 0.0702 | 0.0393 | 0.0193 | 0.0077 | 0.0023 | 0.0004 | $4.17 \times 10^{-5}$ |
| Node 19 | 0.2235 | 0.1666 | 0.1133 | 0.0702 | 0.0393 | 0.0193 | 0.0077 | 0.0023 | 0.0004 | $4.17 \times 10^{-5}$ |

## References


1. Ebrahimi, P.; Basirat, M.; Yousefi, A.; Nekmahmud, M.; Gholampour, A.; Fekete-Farkas, M. Social Networks Marketing and Consumer Purchase Behavior: The Combination of SEM and Unsupervised Machine Learning Approaches. *Big Data Cogn. Comput.* **2022**, *6*, 35. [CrossRef]
2. Mallipeddi, R.R.; Kumar, S.; Sriskandarajah, C.; Zhu, Y. A Framework for Analyzing Influencer Marketing in Social Networks: Selection and Scheduling of Influencers. *Manag. Sci.* **2022**, *68*, 75–104. [CrossRef]
3. Popova, O.I.; Gagarina, N.M.; Minina, T.B.; Holodilov, A.A. Digital Marketing of Social Networks as a Factor for Sustainable Business Development During the COVID-19 Pandemic. In *Advances in Social Science, Education and Humanities Research, Proceedings of the International Scientific and Practical Conference "Sustainable Development of Environment after COVID-19", Yekaterinburg, Russia, 7–8 December 2021*; Atlantis Press: Paris, France, 2022; pp. 249–252. [CrossRef]
4. Sugiyantoro, N.L.A.; Wijaya, M.; Supriyadi, S. Benefits of WhatsApp as a Communication Media on Small Business Social Networks. *J. Soc. Media* **2022**, *6*, 1–16. [CrossRef]
5. Luo, J.; Wu, J.; Yang, W. A relationship matrix resolving model for identifying vital nodes based on community in oppor-tunistic social networks. *Trans. Emerg. Tel. Tech.* **2022**, *33*, e4389.
6. Zhang, Q.; Li, X.; Fan, Y.; Du, Y. An SEI3R information propagation control algorithm with structural hole and high influential infected nodes in social networks. *Eng. Appl. Artif. Intell.* **2022**, *108*, 104573.
7. Bouyer, A.; Beni, H.A. Influence maximization problem by leveraging the local traveling and node labeling method for discovering most influential nodes in social networks. *Phys. A Stat. Mech. Its Appl.* **2022**, *592*, 126841. [CrossRef]
8. Bahutair, M.; Al Aghbari, Z.; Kamel, I. NodeRank: Finding influential nodes in social networks based on interests. *J. Supercomput.* **2021**, *78*, 2098–2124. [CrossRef]
9. Aghaalizadeh, S.; Afshord, S.T.; Bouyer, A.; Anari, B. Improving the stability of label propagation algorithm by propagating from low-significance nodes for community detection in social networks. *Computing* **2021**, *104*, 21–42. [CrossRef]
10. Borgatti, S.P.; Ofem, B. Social network theory and analysis. *Soc. Netw. Theory Educ. Chang.* **2010**, *17*, 29.
11. Pei, S.; Muchnik, L.; Tang, S.; Zheng, Z.; Makse, H.A. Exploring the Complex Pattern of Information Spreading in Online Blog Communities. *PLoS ONE* **2015**, *10*, e0126894. [CrossRef]
12. Onnela, J.-P.; Saramäki, J.; Hyvönen, J.; Szabó, G.; Lazer, D.; Kaski, K.; Kertész, J.; Barabási, A.-L. Structure and tie strengths in mobile communication networks. *Proc. Natl. Acad. Sci. USA* **2007**, *104*, 7332–7336. [CrossRef] [PubMed]
13. Clauset, A.; Shalizi, C.R.; Newman, M.E.J. Power-Law Distributions in Empirical Data. *SIAM Rev.* **2009**, *51*, 661–703. [CrossRef]
14. Bornholdt, S.; Ebel, H. World Wide Web scaling exponent from Simon's 1955 model. *Phys. Rev. E* **2001**, *64*, 035104. [CrossRef] [PubMed]
15. Barabaási, A.-L.; Albert, R. Emergence of Scaling in Random Networks. *Science* **1999**, *286*, 509–512. [CrossRef]
16. Nadaraja, R.; Yazdanifard, R. *Social Media Marketing: Advantages and Disadvantages*; Center of Southern New Hampshire University: Manchester, NH, USA, 2013; pp. 1–10.
17. Ellison, N.B.; Steinfield, C.; Lampe, C. The benefits of Facebook "friends:" Social capital and college students' use of online social network sites. *J. Comput.-Mediat. Commun.* **2007**, *12*, 1143–1168. [CrossRef]
18. Sun, M.; Zhang, H.; Kang, H.; Zhu, G.; Fu, X. Epidemic spreading on adaptively weighted scale-free networks. *J. Math. Biol.* **2016**, *74*, 1263–1298. [CrossRef]
19. Dong, S.; Deng, Y.-B.; Huang, Y.-C. SEIR Model of Rumor Spreading in Online Social Network with Varying Total Population Size. *Commun. Theor. Phys.* **2017**, *68*, 545. [CrossRef]
20. Zhu, Y.-X.; Zhang, X.-G.; Sun, G.-Q.; Tang, M.; Zhou, T.; Zhang, Z.-K. Influence of Reciprocal Links in Social Networks. *PLoS ONE* **2014**, *9*, e103007. [CrossRef]
21. Liu, C.; Zhang, Z.K. Information spreading on dynamic social networks. *Commun. Nonlinear Sci. Numer. Simul.* **2014**, *19*, 896–904. [CrossRef]
22. Lü, L.; Chen, D.-B.; Zhou, T. The small world yields the most effective information spreading. *New J. Phys.* **2011**, *13*, 123005. [CrossRef]

23. Haythommthwaite, K. *Characterized Social Networks as Having the Following Components: Actors*; Nodes: New York, NY, USA, 2005.
24. Barnes, J.A. Class and Committees in a Norwegian Island Parish. *Hum. Relat.* **1954**, *7*, 39–58. [CrossRef]
25. Kempe, D.; Kleinberg, J.; Tardos, É. Maximizing the spread of influence through a social network. In Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 24–27 August 2003.
26. Litvak, N.; Scheinhardt, W.R.; Volkovich, Y. In-degree and PageRank: Why do they follow similar power laws? *Internet Math.* **2007**, *4*, 175–198. [CrossRef]
27. Weng, J.; Lim, E.P.; Jiang, J.; He, Q. Twitterrank: Finding topic-sensitive influential twitterers. In Proceedings of the Third ACM International Conference on Web Search and Data Mining, New York, NY, USA, 3–6 February 2010.
28. Xing, W.; Ghorbani, A. Weighted pagerank algorithm. In Proceedings of the Second Annual Conference on Communication Networks and Services Research, Fredericton, NB, Canada, 19–21 May 2004.
29. Liu, L.; Tang, J.; Han, J.; Jiang, M.; Yang, S. Mining topic-level influence in heterogeneous networks. In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, Toronto, ON, Canada, 26–30 October 2010.
30. Chen, S.; Fan, J.; Li, G.; Feng, J.; Tan, K.-L.; Tang, J. Online topic-aware influence maximization. *Proc. VLDB Endow.* **2015**, *8*, 666–677. [CrossRef]
31. Barbieri, N.; Bonchi, F.; Manco, G. Topic-aware social influence propagation models. *Knowl. Inf. Syst.* **2013**, *37*, 555–584. [CrossRef]
32. Agirre, E.; Soroa, A. Personalizing pagerank for word sense disambiguation. In Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), Athens, Greece, 30 March–3 April 2009.
33. Xie, W.; Bindel, D.; Demers, A.; Gehrke, J. Edge-weighted personalized pagerank: Breaking a decade-old performance barrier. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, Australia, 10–13 August 2015.
34. Pirouz, M.; Zhan, J. Toward Efficient Hub-Less Real Time Personalized PageRank. *IEEE Access* **2017**, *5*, 26364–26375. [CrossRef]
35. Clauset, A.; Moore, C.; Newman, M.E.J. Hierarchical structure and the prediction of missing links in networks. *Nature* **2008**, *453*, 98–101. [CrossRef]
36. Backstrom, L.; Leskovec, J. Supervised random walks: Predicting and recommending links in social networks. In Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, Hong Kong, China, 9–12 February 2011.
37. Yeh, W.-C. Novel binary-addition tree algorithm (BAT) for binary-state network reliability problem. *Reliab. Eng. Syst. Saf.* **2021**, *208*, 107448. [CrossRef]
38. Tarjan, R. Depth-First Search and Linear Graph Algorithms. *SIAM J. Comput.* **1972**, *1*, 146–160. [CrossRef]
39. Bundy, A.; Wallen, L. *Breadth-first Search, in Catalogue of Artificial Intelligence Tools*; Springer: Berlin/Heidelberg, Germany, 1984; p. 13.
40. Garton, L.; Haythornthwaite, C.; Wellman, B. Studying online social networks. *J. Comput.-Mediat. Commun.* **1997**, *31*, JCMC313. [CrossRef]
41. Tripathi, S.; Verma, S. *Analysing Technological Dimensions for Engagement with NGOs on Social Media*; John Wiley & Sons: Hoboken, NJ, USA, 2017; pp. 1–11.
42. Khoo, D. Travel Planning for Social Networks. Google Patents US20040193488A1, 15 March 2010.
43. Qiu, T.; Chen, B.; Sangaiah, A.K.; Ma, J.; Huang, R. A Survey of Mobile Social Networks: Applications, Social Characteristics, and Challenges. *IEEE Syst. J.* **2017**, *12*, 3932–3947. [CrossRef]
44. Rahim, A.; Kong, X.; Xia, F.; Ning, Z.; Ullah, N.; Wang, J.; Das, S.K. Vehicular Social Networks: A survey. *Pervasive Mob. Comput.* **2018**, *43*, 96–113. [CrossRef]
45. Alavi, S.S.; Mehdinezhad, I.; Kahshidinia, B.; Jenab, K. A trend study on the impact of social media on advertisement. *Int. J. Data Netw. Sci.* **2019**, *3*, 185–200. [CrossRef]
46. Khoo, D. Travel Planning for Social Networks. Google Patents US20060004590A1, 19 December 2013.
47. Pei, S.; Makse, H.A. Spreading dynamics in complex networks. *J. Stat. Mech. Theory Exp.* **2013**, *2013*, P12002. [CrossRef]
48. Moreno, Y.; Vazquez, A. Disease spreading in structured scale-free networks. *Eur. Phys. J. B-Condens. Matter Complex Syst.* **2003**, *31*, 265–271. [CrossRef]
49. Boguñá, M.; Pastor-Satorras, R.; Vespignani, A. Absence of Epidemic Threshold in Scale-Free Networks with Degree Correlations. *Phys. Rev. Lett.* **2003**, *90*, 028701. [CrossRef] [PubMed]
50. Faloutsos, M. On power-law relationships of the internet topology. *ACM SIGCOMM Comput. Commun. Rev.* **2011**, *29*, 195–206. [CrossRef]
51. Albert, R.; Jeong, H.; Barabasi, A. Error and attack tolerance of complex networks. *Nature* **2000**, *406*, 378–382. [CrossRef]
52. Blanka, G.; Reisdorf, B. The participatory web: A user perspective on Web 2. *Information. Commun. Soc.* **2012**, *15*, 537–554. [CrossRef]
53. Kleinberg, J.M.; Kumar, R.; Raghavan, P.; Rajagopalan, S.; Tomkins, A.S. *The Web as a Graph: Measurements, Models, and Methods*; Springer: Berlin/Heidelberg, Germany, 1999.
54. Broder, A.; Kumar, R.; Maghoul, F.; Raghavan, P.; Rajagopalan, S.; Stata, R.; Tomkins, A.; Wiener, J. *Graph Structure in the Web, in the Structure and Dynamics of Networks*; Princeton University Press: Princeton, NJ, USA, 2011; pp. 183–194.
55. Adamic, L.A.; Huberman, B.A. Power-Law Distribution of the World Wide Web. *Science* **2000**, *287*, 2115. [CrossRef]
56. Lin, F.R. Contagion Dynamics of Scale-Free Networks with Resource Limitation. Master Thesis, Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan, 2005.

57. Barabási, A.-L. Scale-Free Networks: A Decade and Beyond. *Science* **2009**, *325*, 412–413. [CrossRef]
58. Bollobás, B.; Riordan, O.; Spencer, J.; Tusnády, G. The degree sequence of a scale-free random graph process. *Random Struct. Algorithms* **2001**, *18*, 279–290. [CrossRef]
59. Chen, Q.; Shi, D. The modeling of scale-free networks. *Phys. A Stat. Mech. Its Appl.* **2004**, *335*, 240–248. [CrossRef]
60. Albert, R.; Barabási, A.-L. Statistical mechanics of complex networks. *Rev. Mod. Phys.* **2002**, *74*, 47–97. [CrossRef]
61. Holme, P.; Kim, B.J. Growing scale-free networks with tunable clustering. *Phys. Rev. E* **2002**, *65*, 026107. [CrossRef]
62. Brin, S.; Page, L. The anatomy of a large-scale hypertextual Web search engine. *Comput. Netw. ISDN Syst.* **1998**, *30*, 107–117. [CrossRef]
63. Luo, S. Distributed PageRank Computation: An Improved Theoretical Study. *Proc. Conf. AAAI Artif. Intell.* **2019**, *33*, 4496–4503. [CrossRef]
64. Langville, A.N.; Meyer, C.D. Deeper inside pagerank. *Internet Math.* **2004**, *1*, 335–380. [CrossRef]
65. Valente, T. *Network Models of the Diffusion of Innovations*; Hampton Free Press: Creswell, NJ, USA, 1995.
66. Bharat, K.; Mihaila, G.A. When experts agree: Using non-affiliated experts to rank popular topics. In Proceedings of the 10th International Conference on World Wide Web, Hong Kong, China, 1–5 May 2001.
67. Haveliwala, T. Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *IEEE Trans. Knowl. Data Eng.* **2003**, *15*, 784–796. [CrossRef]
68. Page, L.; Brin, S.; Motwani, R.; Winograd, T. *The PageRank Citation Ranking: Bringing Order to the Web*; Stanford University InfoLab: Stanford, CA, USA, 1999.
69. Scozzafava, F.; Maru, M.; Brignone, F.; Torrisi, G.; Navigli, R. Personalized PageRank with syntagmatic information for multilingual word sense disambiguation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Online, 5–10 July 2020; pp. 37–46.
70. Hou, G.; Chen, X.; Wang, S.; Wei, Z. Massively Parallel Algorithms for Personalized PageRank. *VLDB* **2021**, *14*, 1668–1680. [CrossRef]
71. Wang, H.; Yuan, Y.; Wei, Z.; Du, X.; Gan, J.; Wen, J.R. Edge-based Local Push for Personalized PageRank. *VLDB* **2022**, *15*, 1376–1389.
72. Mo, D.; Luo, S. Agenda: Robust Personalized PageRanks in Evolving Graphs. *CIKM* **2021**. [CrossRef]
73. Wu, H.; Gan, J.; Wei, Z.; Zhang, R. Unifying the Global and Local Approaches: An Efficient Power Iteration with Forward Push. In Proceedings of the 2021 International Conference on Management of Data, Xi'an, China, 20–25 June 2021; pp. 1996–2008. [CrossRef]
74. Aven, T. Availability evaluation of oil/gas production and transportation systems. *Reliab. Eng.* **1987**, *18*, 35–44. [CrossRef]
75. Bhavathrathan, B.; Patil, G.R. Analysis of Worst Case Stochastic Link Capacity Degradation to Aid Assessment of Transportation Network Reliability. *Procedia-Soc. Behav. Sci.* **2013**, *104*, 507–515. [CrossRef]
76. Yeh, W.-C. A Squeezed Artificial Neural Network for the Symbolic Network Reliability Functions of Binary-State Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *28*, 2822–2825. [CrossRef] [PubMed]
77. Kakadia, D.; Ramirez-Marquez, J.E. Quantitative approaches for optimization of user experience based on network resilience for wireless service provider networks. *Reliab. Eng. Syst. Saf.* **2019**, *193*, 106606. [CrossRef]
78. Yeh, W.-C.; Lin, J.-S. New parallel swarm algorithm for smart sensor systems redundancy allocation problems in the Internet of Things. *J. Supercomput.* **2016**, *74*, 4358–4384. [CrossRef]
79. Yeh, W.C. A Novel Generalized Artificial Neural Network for Mining Two-Class Datasets. *arXiv* **2019**, arXiv:1910.10461.
80. Yeh, W.-C. A Modified Universal Generating Function Algorithm for the Acyclic Binary-State Network Reliability. *IEEE Trans. Reliab.* **2012**, *61*, 702–709. [CrossRef]
81. Yeh, W.C. An Evaluation of the Multi-state Node Networks Reliability Using the Traditional Binary-State Networks Reliability Algorithm. *Reliab. Eng. Syst. Saf.* **2003**, *81*, 1–7. [CrossRef]
82. Yeh, W.-C. A new exact solution algorithm for a novel generalized redundancy allocation problem. *Inf. Sci.* **2017**, *408*, 182–197. [CrossRef]
83. Zhou, J.; Coit, D.W.; Felder, F.A.; Wang, D. Resiliency-based restoration optimization for dependent network systems against cascading failures. *Reliab. Eng. Syst. Saf.* **2020**, *207*, 107383. [CrossRef]
84. Singh, A.P.; Luhach, A.K.; Gao, X.-Z.; Kumar, S.; Roy, D.S. Evolution of wireless sensor network design from technology centric to user centric: An architectural perspective. *Int. J. Distrib. Sens. Netw.* **2020**, *16*, 1550147720949138. [CrossRef]
85. Yeh, W.C. Novel bounded binary-addition tree algorithm for binary-state network reliability problems. *arXiv* **2020**, arXiv:2011.14832.
86. Su, Y.-Z.; Yeh, W.-C. Binary-Addition Tree Algorithm-Based Resilience Assessment for Binary-State Network Problems. *Electronics* **2020**, *9*, 1207. [CrossRef]
87. Yeh, W.-C.; Kuo, C.-C. Predicting and Modeling Wildfire Propagation Areas with BAT and Maximum-State PageRank. *Appl. Sci.* **2020**, *10*, 8349. [CrossRef]
88. Yeh, W.C. Self-Adaptive Binary-Addition-Tree Algorithm-Based Novel Monte Carlo Simulation for Binary-State Network Reliability Approximation. *arXiv* **2022**, arXiv:2201.05764.
89. Yeh, W.-C.; Tan, S.-Y.; Forghani-Elahabad, M.; El Khadiri, M.; Jiang, Y.; Lin, C.-S. New binary-addition tree algorithm for the all-multiterminal binary-state network reliability problem. *Reliab. Eng. Syst. Saf.* **2022**, *224*, 108557. [CrossRef]

90. Yeh, W.-C.; Hao, Z.; Forghani-Elahabad, M.; Wang, G.-G.; Lin, Y.-L. Novel Binary-Addition Tree Algorithm for Reliability Evaluation of Acyclic Multistate Information Networks. *Reliab. Eng. Syst. Saf.* **2021**, *210*, 107427. [CrossRef]
91. Yeh, W.C. Application of Long Short-Term Memory Recurrent Neural Networks Based on the BAT-MCS for Binary-State Network Approximated Time-Dependent Reliability Problems. *arXiv* **2022**, arXiv:2202.07837.
92. Su, Y.Z.; Yeh, W.C. The protection and recovery strategy development of dynamic resilience analysis and cost consideration in the infrastructure network. *J. Comput. Des. Eng.* **2022**, *9*, 168–186. [CrossRef]
93. Yeh, W.-C.; Tan, S.-Y.; Zhu, W.; Huang, C.-L.; Yang, G.-Y. Novel binary addition tree algorithm (BAT) for calculating the direct lower-bound of the highly reliable binary-state network reliability. *Reliab. Eng. Syst. Saf.* **2022**, *223*, 108509. [CrossRef]
94. Yeh, W.-C.; Lin, E.; Huang, C.-L. Predicting Spread Probability of Learning-Effect Computer Virus. *Complexity* **2021**, *2021*, 6672630. [CrossRef]
95. Yeh, W.-C. New Method in Searching for All Minimal Paths for the Directed Acyclic Network Reliability Problem. *IEEE Trans. Reliab.* **2016**, *65*, 1263–1270. [CrossRef]
96. Yeh, W.-C. A Novel Cut-Based Universal Generating Function Method. *IEEE Trans. Reliab.* **2013**, *62*, 628–636. [CrossRef]
97. Hagberg, A.; Swart, P.; Chult, D.S. *Exploring Network Structure, Dynamics, and Function Using NetworkX*; No. LA-UR-08-05495; LA-UR-08-5495; Los Alamos National Lab. (LANL): Los Alamos, NM, USA, 2008.