applied sciences

MDPI

Article

DDMF: A Method for Mining Relatively Important Nodes Based on Distance Distribution and Multi-Index Fusion

Na Zhao [1,†], Qian Liu [1,†], Ming Jing [2], Jie Li [3], Zhidan Zhao [4] and Jian Wang [5,*]

[1] Key Laboratory in Software Engineering of Yunnan Province, School of Software, Yunnan University, Kunming 650091, China; zhaona@ynu.edu.cn (N.Z.); liu_antoni0409@163.com (Q.L.)
[2] School of Information Engineering, Kunming University, Kunming 650214, China; proofle@163.com
[3] Electric Power Research Institute of Yunnan Power Grid Co., Ltd., Kunming 650217, China; lj1226645407@163.com
[4] Department of Computer Science, School of Engineering, Shantou University, Shantou 515063, China; zzhidanzhao@gmail.com
[5] College of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650504, China
* Correspondence: jianwang@kust.edu.cn
† These authors contributed equally to this work.

Abstract: In research on complex networks, mining relatively important nodes is a challenging and practical work. However, little research has been done on mining relatively important nodes in complex networks, and the existing relatively important node mining algorithms cannot take into account the indicators of both precision and applicability. Aiming at the scarcity of relatively important node mining algorithms and the limitations of existing algorithms, this paper proposes a relatively important node mining method based on distance distribution and multi-index fusion (DDMF). First, the distance distribution of each node is generated according to the shortest path between nodes in the network; then, the cosine similarity, Euclidean distance and relative entropy are fused, and the entropy weight method is used to calculate the weights of different indexes; Finally, by calculating the relative importance score of nodes in the network, the relatively important nodes are mined. Through verification and analysis on real network datasets in different fields, the results show that the DDMF method outperforms other relatively important node mining algorithms in precision, recall, and AUC value.

Keywords: complex network; distance distribution; multi-index fusion; relatively important node

Citation: Zhao, N.; Liu, Q.; Jing, M.; Li, J.; Zhao, Z.; Wang, J. DDMF: A Method for Mining Relatively Important Nodes Based on Distance Distribution and Multi-Index Fusion. *Appl. Sci.* **2022**, *12*, 522. https://doi.org/10.3390/app12010522

Academic Editors: Giacomo Fiumara, Pasquale De Meo, Xiaoyang Liu and Annamaria Ficara

Received: 1 December 2021
Accepted: 3 January 2022
Published: 5 January 2022

1. Introduction

With the vigorous growth of network and information technology represented by the Internet, human society has entered a new and complex era of networks. Information mining in complex networks is important in theoretical research and offers great application and socioeconomic values [1–4]. For example, if users can unearth important nodes or edges in the spread network of a virus, then they can curb the spread of the virus in a short time by isolating or cutting off the important nodes or edges in the virus network at the beginning of the virus spread and thereby eliminate unnecessary economic losses [5]. Efficient information mining in complex networks has naturally become a key topic that continues to attract the attention of many scholars.

The existing studies on complex network information mining are generally ranked on the basis of the importance of all nodes and edges in the network [6–10]. However, determining which nodes are the most important in the network relative to one or one group of specific nodes presents an issue. This problem reminds us about the practical significance of mining relatively important information in networks, especially very large-scale ones.

The relative importance of nodes refers to the importance of nodes relative to known important nodes. It is also called proximity or similarity [11]. According to the key idea

of relative importance, information mining in a complex network can be described as a process in which the importance of a node in a network relative to a known important node is quantified and the importance of a node relative to a known important node set is calculated to identify the relatively important nodes in the network.

The central idea of relative importance can be widely used in many fields. For example, potential criminals can be found using known criminal data in the field of criminal networks, and terrorists in hiding can be captured on the basis of known terrorist data [12,13]. In the bionetwork field, people susceptible to diseases can be identified for timely treatment and isolation on the basis of relevant information on populations infected with known infectious diseases. Unknown pathogenic genes may be determined according to known pathogenic gene information in protein networks [14]. In the field of power grids, on the premise that the information on important power generation units or circuit breakers is known, finding relatively important power generation units, circuit breakers, etc. is prioritized for protection, in order to effectively avoid large-area power outages caused by successive faults. Mining relatively important nodes in complex networks obviously offers great research significance and application value [15].

The node distance distribution in a complex network quantifies many types of topological information in the network, including the degree of nodes, average degree of the network, diameter of the network, closeness centrality of nodes, and average path length of the network [16]. Therefore, the study on the relative importance of nodes in a network based on node distance distribution in the network will contribute to the accurate mining of relatively important nodes in networks. In the current study, the distance distribution of all nodes in a network is calculated. On the basis of known important node information, the differences in distance distribution between known important nodes and target nodes are measured from three dimensions, namely, direction, distance, and distribution. A relatively important node mining method based on distance distribution and multi-index fusion (DDMF) is proposed.

The DDMF method involves two main steps: First, the distance distribution of all nodes (including known important nodes and target nodes) is calculated on the basis of the shortest distance between nodes in the network. Then, the calculated results are converted into vector form. Second, multi-index fusion is made for cosine similarity, Euclidean distance, and relative entropy. The weights corresponding to different indexes are calculated using the entropy weight method to obtain the relative importance scores of the nodes. The nodes with high scores are regarded as a relatively important nodes in the network.

Our key contribution is in proposing a novel method based on network topology to find relatively important nodes in the network. The DDMF method not only fills the gap of relatively important node algorithms in the scientific field of complex network theory, but also provides a new idea for community detection and link prediction. Since the network in real life exists in different kinds of fields, we also conduct some experiments on different types of real network datasets to verify whether the method has practical application value in real life. Experiments demonstrate that DDMF method outperforms other relatively important node mining algorithms in terms of precision and applicability.

The remainder of this paper is organized as follows. In Section 2, works related to the proposed method are given. Section 3 deals with detailed descriptions of the proposed algorithm. The experimental results and analysis are presented in Section 4. Finally, we summarize in Section 5.

2. Related Work

At present, many researchers in the field of complex networks focus on the mining of important nodes in networks; that is, ranking the importance of all nodes in a network as a whole. Existing research has primarily aimed to develop an identification algorithm for influential nodes. Inspired by the heuristic scheme, Wang et al. [17] proposed the price-performance-ratio PPRank method, selecting nodes in a given range and aiming to

improve the performance of the diffusion. Yang et al. [18] proposed a method of ranking node importance based on multi-criteria decision-making (MCDM). The weight of each criterion is calculated by an entropy weighting method, which overcomes the impact of the subjective factor. Li et al. [19] proposed a method of calculating the importance degree of urban rail transit network nodes based on h-index, which considers the topology, passenger volume, and passenger flow correlation of the urban rail network. Luo et al. [20] proposed a relationship matrix resolving model to identify vital nodes based on community (IVNC), as an attempt to identify influential nodes in OSNs.

However, the study on node mining based on relative importance remains limited. The earliest study on relative importance in networks is that on a personalized variant HITS algorithm [21]. Haveliwala [22] and Jennifier et al. [23] later proposed their own variant PageRank algorithms, which consider the relative importance of nodes in a network. Alzaabi et al. [24] defined the universal framework of mining algorithms for relatively important nodes and proposed that the relative importance of nodes in a network relates to one node set or one group of specified node sets. Wang et al. [25] proposed a path probabilistic summation method, which defines the importance of any node relative to the nearest neighbor node as the probability of jumping from the node to the nearest neighbor node in the random walk process. Rodriguez et al. [26] proposed a cluster particle propagation method, which is used to evaluate the relative importance of nodes. Magalingam et al. [27] used shortest distance as a measurement indicator of relative importance. Langohr et al. [28] used the reciprocal of the P norm of the shortest distance as a measurement indicator of relative importance. In addition, some researchers have considered mining deep network information by using network embedded learning methods [29–35]. For example, some classical network-embedded learning algorithms have been used to mine relatively important nodes in networks.

Although some algorithms have been employed to mine relatively important nodes in networks, they suffer from problems that require immediate resolution, such as low accuracy and narrow use range. Therefore, novel and efficient methods for mining relatively important nodes need to be developed.

In the study of complex networks, the most classic and most widely used relative importance calculation indicators include the Ksmar index [11], PPR index [21], and Katz index [36]. Zhao et al. [37] proposed a relatively important node mining algorithm based on neighbor layer diffuse (NLD) in 2021, which is the latest relatively important node algorithm. In Section 4, we empirically compare our method with these methods using various real world networks.

3. Relative Importance Measure Based on Distance Distribution and Multi-index Fusion

To fully measure the impact of network structure information on the relative importance of nodes, this study proposes a relatively important node mining method based on distance distribution and multi-index fusion, i.e., the DDMF method. In this section, we first introduce the problem definition in complex networks and use a specific example to explain what is the distance distribution. Then three indicators of cosine similarity, Euclidean distance, and relative entropy are described in detail. Finally, we discuss how to calculate the relative importance score of a node based on multi-index fusion.

3.1. Problem Definition

Under normal conditions, a complex network can be represented by $G(V, E)$. Here, V refers to the node in the network G and E refers to the edge of the network G. The network G comprises n nodes. Among them, n nodes can be divided into important node set V_1 and unimportant node set V_2. The important node set V_1 has n_1 nodes, while the unimportant node set V_2 has n_2 nodes. The important node set V_1 includes known important node set R and unknown important node set U. The unimportant node set V_2 and unknown important node set U constitute target node set T, i.e., $T = V_2 \cup U$.

The key to finding the relatively important nodes in the target node set T is to first calculate the importance of a node in the target node set T relative to a known important node, and then calculate the importance of a node relative to all nodes in the known important node set R.

The main contents of this work include the following: For the information of known important node set R, the importance of any node in the target node set T relative to the node in the known important node set R is analyzed and calculated. The expectation is to find $top-k$ relatively important nodes in the target node set T. The final results are analyzed and evaluated on the basis of three evaluation indicators, namely, precision, recall, and area under the curve (AUC).

3.2. Distance Distribution

Distance distribution in complex networks is usually represented by the shortest path distribution between nodes. The node distance distribution in the network mainly considers the number of nodes with different shortest path lengths to the current node; thus, it can intuitively obtain the shortest path information of nodes in the network and reflect many important topological information in the network [38].

The distance distribution of each node v_i in the complex network can be represented as $P_i = \{p_i(j)\}$; the calculation formula of $p_i(j)$ is

$$p_i(j) = \frac{N_i(j)}{n} \tag{1}$$

where j represents the shortest path length with a value in the range of $0 \leq j \leq D(G)$. $D(G)$ refers to the diameter of the network G, and its value is the maximum distance between any two nodes in the network G. $N_i(j)$ represents the number of nodes with j of the shortest path length to node v_i in the network G; n represents the number of nodes in the network G.

Take a network $G_{example}$ as an example. The detailed calculation process of node distance distribution in $G_{example}$ is introduced as follows. In Figure 1, the red nodes are the nodes in the current study while the yellow, light green, blue, and pink nodes represent the nodes that can be reached by taking one, two, three, and four steps consecutively, starting from the nodes studied currently.

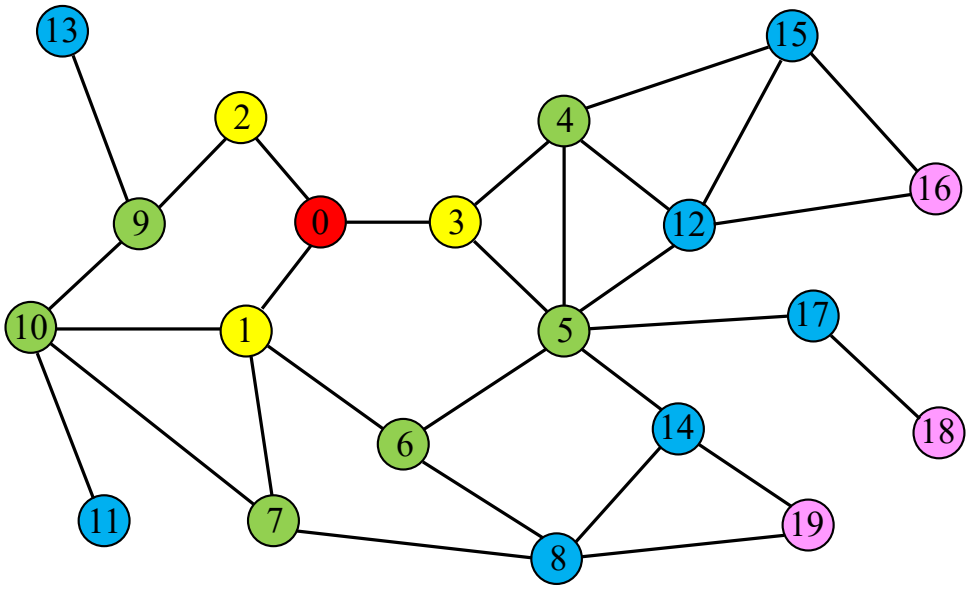

Figure 1. The topology of the example network.

The number of nodes n in the sample network $G_{example}$ is 20, and the diameter $D(G)$ is 7. The distance distribution dimension mainly depends on the diameter of the

network. Its value range is from 0 to $D(G)$, with a total of $D(G)+1$ cases. Therefore, the distance distribution dimension d of each node in $G_{example}$ is 8. Provided that node 0 is used as the starting research node, the set of nodes $N(i) = \{N_i(j)|0 \le j \le D(G)\}$ that can be reached by node 0 in turn can be obtained by calculating the shortest path length between this node and other nodes in $G_{example}$, that is, according to Formula (1) and $N(i)$ obtained through the above analysis, the distance distribution P_0 of node 0 can be obtained as: $P_0 = \{0.05, 0.15, 0.30, 0.35, 0.15, 0, 0, 0\}$.

Similarly, the distance distribution of any node in the sample network $G_{example}$ can be obtained. For a network G with n nodes, if the distance distribution of n nodes is known $P = \{P_0, P_1, \cdots, P_{n-1}\}$, then much important topology information in the network G can be obtained on the basis of the distance distribution of nodes. For example, the degree k_i of any node v_i in G, the average degree $\overline{k}$ of G, the average path length APL of G, and the closeness centrality CC_i corresponding to node v_i.

For the network G with n nodes, a distance distribution matrix $X = [x_{ij}] \in R^{n \times d}$ is established on the basis of the distance distribution information of all nodes in G; n refers to the total number of nodes in the network G.

(1) Degree k_i of node v_i

$$k_i = nx_{i1} \tag{2}$$

(2) Average degree $\overline{k}$ of network G

$$\overline{k} = \frac{1}{n}\sum_{i=0}^{n-1} nx_{i1} \tag{3}$$

(3) Average path length APL of network G

$$APL = \frac{2}{n(n-1)}\sum_{i=0}^{n-1}\sum_{j=1}^{D(G)} j \times nx_{ij} \tag{4}$$

(4) Closeness centrality CC_i of node v_i

$$CC_i = \frac{n}{\sum_{j=1}^{D(G)} j \times nx_{ij}} \tag{5}$$

The analysis indicates that the distance distribution of nodes contains abundant network topology information. Therefore, taking the distance distribution P_i of each node v_i in the network G as the main subject investigated and converting it into vector form, the difference in the distance distribution between nodes in the known important node set R and the target node set T is analyzed to find the relatively important nodes in the network G.

3.3. Introduction to Indicators

Cosine similarity is a measurement method for the difference between two individuals and involves calculating the cosine value of the angle between two vectors in the vector space, mainly focusing on the measurement of the difference between two individuals from the dimension of direction. The basic idea is to covert the individual's index data into the vector space and then measure the difference between individuals by comparing the cosine values of the angle in the inner product space between different individual vectors [39].

In a M-dimensional space, assuming that A and B are M-dimensional vectors, namely $A = [a_1, a_2, \cdots, a_M]$, $B = [b_1, b_2, \cdots, b_M]$, then the cosine similarity Cos_{AB} can be expressed as:

$$Cos_{AB} = \frac{\sum_{i=1}^{M} (A_i \times B_i)}{\sqrt{\sum_{i=1}^{M} (A_i)^2} \times \sqrt{\sum_{i=1}^{M} (B_i)^2}} = \frac{A \cdot B}{|A| \times |B|} \tag{6}$$

where the value range of Cos_{AB} is [–1, 1], that is, $Cos_{AB} \in [-1, 1]$.

In this work, the distance distribution of nodes in the network is converted into vector form; that is, in the network G with n nodes, the vectors of distance distribution of any node x and node y can be expressed as P_x and P_y, respectively. Then, the formula for the cosine similarity between nodes can be represented as:

$$D_{Cos}(P_x \| P_y) = \frac{P_x \cdot P_y}{|P_x| \times |P_y|} \tag{7}$$

$$C_{xy} = \frac{1 + D_{Cos}(P_x \| P_y)}{2} \tag{8}$$

Normalization is performed for the cosine similarity between nodes $D_{Cos}(P_x \| P_y)$ based on Equation (8), and C_{xy} is obtained. Among them, $C_{xy} \in [0, 1]$.

Euclidean distance, also called Euclidean metric, originates from the distance formula between two points in Euclidean geometry [40]. It is mainly used to measure the real distance between two points in M-dimension space; that is, focusing on the numerical difference between individuals.

In a M-dimensional space, assuming that A and B are M-dimensional vectors, namely $A = [a_1, a_2, \cdots, a_M]$, $B = [b_1, b_2, \cdots, b_M]$, then the Euclidean distance Euc_{AB} can be expressed as:

$$Euc_{AB} = \sqrt{\sum_{i=1}^{M} (a_i - b_i)^2} \tag{9}$$

Similarly, the distance distribution of nodes in the network G is first converted into vector form. Then, the Euclidean distance between any node x and node y can be represented as Euc_{xy}:

$$E_{xy} = \frac{Euc_{xy}}{Euc_{\max}} \tag{10}$$

Normalization is performed for Euclidean distance Euc_{xy} between node x and node y based on Equation (10), and E_{xy} is obtained. Among them, $E_{xy} \in [0, 1]$.

From information theory, relative entropy, also called KL divergence or information divergence, is generally used to measure the difference between two probability distributions [41]. In this work, the difference in the distance distribution between different nodes in the network is calculated from the dimension of distribution to effectively find relatively important nodes in the network.

For the network G with n nodes, the distance distributions of node x and node y are P_x and P_y, respectively. Then, relative entropy can be defined as the difference in the distance distribution between the two nodes. The formula is as follows:

$$D_{KL}(P_x \| P_y) = \sum_{j=0}^{D(G)} p_x(j) \ln \frac{p_x(j)}{p_y(j)} \tag{11}$$

If relative entropy $D_{KL}(P_x \| P_y)$ is small, then the difference in the distance distribution between node x and node y is small. The denominator of the logarithmic function cannot be 0. Therefore, in $p_x(j) = 0$ or $p_y(j) = 0$, the values of $\ln \frac{p_x(j)}{p_y(j)}$ are uniformly set to 0.

In addition, relative entropy is an asymmetric measure. Therefore, this study symmetrically converts the relative entropy between node distance distributions. The specific formula is as follows:

$$Q_{xy} = \frac{D_{KL}(P_x \| P_y) + D_{KL}(P_y \| P_x)}{2} \tag{12}$$

$$R_{xy} = 1 - \frac{Q_{xy}}{Q_{max}} = \frac{Q_{max} - Q_{xy}}{Q_{max}} \tag{13}$$

The relative entropy in asymmetric form is converted into symmetric form Q_{xy} in Equation (12). On the basis of Equation (13), normalization processing is implemented for the relative entropy in symmetric form, then R_{xy} is obtained. Among them, $R_{xy} \in [0, 1]$.

This study aims to find relatively important nodes in the network G by calculating the relative entropy of the distance distribution of nodes in the known important node set R and target node set T. If the relative entropy is small, then the difference in the distance distribution between different nodes is small. That is, the nodes with a smaller relative entropy in the target node set T compared to the known important node set R are more likely to be relatively important nodes in the network G.

3.4. Relative Importance Score Based on Multi-index Fusion

To fully integrate the advantages of cosine similarity, Euclidean distance, and relative entropy in the direction, distance, and distribution dimensions, this study performs the multi-index fusion of cosine similarity, Euclidean distance, and relative entropy and calculates the weights of the different indexes by using the entropy weight method [42] to maximize the advantages of the different indexes. The entropy weight method is an objective weighting method that is widely used and often depends on the discreteness of data. It mainly weighs different indexes according to the amount of information of different evaluation indexes.

Cosine similarity, Euclidean distance, and relative entropy are mainly considered in this work. Thus, weight allocation becomes necessary. A relative importance score matrix, $Z = [z_{tg}] \in R^{|T| \times 3}$, is defined herein.

$$z_{t1} = \frac{\sum_{r=1}^{|R|} C_{tr}}{|R|}, t = 1, 2, \cdots, |T| \tag{14}$$

$$z_{t2} = \frac{\sum_{r=1}^{|R|} E_{tr}}{|R|}, t = 1, 2, \cdots, |T| \tag{15}$$

$$z_{t3} = \frac{\sum_{r=1}^{|R|} R_{tr}}{|R|}, t = 1, 2, \cdots, |T| \tag{16}$$

where z_{t1}, z_{t2} and z_{t3} represent the arithmetic mean of cosine similarity, Euclidean distance and relative entropy between the $t-th$ node in the target node set T and all nodes in the known important node set R respectively. $|T|$ refers to the number of nodes in the target node set T, $|R|$ refers to the number of nodes in the known important node set R, and g refers to the number of indexes, $g = 1, 2, 3$.

Based on the relative importance score matrix, the entropy corresponding to cosine similarity, Euclidean distance, and relative entropy can be further calculated. The formulas are as follows:

$$e_g = -\frac{1}{\ln|T|} \sum_{t=1}^{|T|} p_{tg} \ln(p_{tg}) \tag{17}$$

$$p_{tg} = \frac{z_{tg}}{\sum_{t=1}^{|T|} z_{tg}} \tag{18}$$

where e_g represents the entropy of the index in the g column and p_{tg} represents the proportion of the index in the g column of the $t-th$ node in the target node set T in this column of indexes.

After the entropies of different indexes are obtained, the weight coefficient ω_g of each index can be further calculated. The weights corresponding to different indicators determine the relative importance scores of the target nodes in the network. The specific formula is:

$$\omega_g = \frac{1 - e_g}{\sum_{g=1}^{3} (1 - e_g)} \tag{19}$$

where $1 - e_g$ refers to information entropy redundancy. At the same time, ω_g should meet the restrictive conditions of $\sum \omega_g = 1, g = 1, 2, 3$.

Therefore, the relative importance score of $t - th$ node in the target node set T can be expressed as:

$$s_t = \omega_1 z_{t1} + \omega_2 z_{t2} + \omega_3 z_{t3} \tag{20}$$

Finally, the relative importance scores of all nodes in the target node set T are sorted in descending order, and the nodes with high scores can be regarded as relatively important nodes.

The calculation of the relative importance scores of the nodes in a network by using the DDMF method consists of the following steps:

First, on the basis of the information of the shortest distance between nodes in the network G, the distance distribution vectors of all nodes in the network G are calculated, along with all the nodes of the known important node set R and target node set T.

Second, the differences in the distance distribution of the nodes between the known important node set R and the target node set T are determined. The cosine similarity, Euclidean distance, and relative entropy of the distance distribution of the two node sets are then calculated and normalized.

Finally, multi-index fusion is made for cosine similarity, Euclidean distance, and relative entropy, and the weights corresponding to different indexes are calculated using the entropy weight method. The relative importance scores of all the nodes in the target node set T are further obtained. The nodes with high scores are regarded as relatively important nodes.

4. Experimental Results and Analysis

The data of four real networks are used to analyze and verify the accuracy of the DDMF method. The Node2vec algorithm [43] is a network-embedded learning algorithm that cannot be directly used to calculate the relative importance scores of nodes. Therefore, the NMF index is obtained on the basis of the improvement of the Node2vec algorithm. The basic idea of the NMF index is as follows: first, the Node2vec algorithm is adopted to generate the embedded vector of the network. Second, the multi-index fusion is made for the obtained vectors so as to calculate the relative importance scores of the nodes. The multi-index fusion method of the NMF index is consistent with proposed DDMF method.

The comparative algorithms included the Ksmar index, PPR index, Katz index, NLD algorithm, and NMF index obtained on the basis of the Node2vec algorithm improvement.

4.1. Datasets

Experimental analysis is performed for the selected algorithms by using four classical real network datasets. The selected datasets are of different sizes and come from different network fields as much as possible, including virus networks, gene networks, and

protein networks. The weight and direction of each network linking edge are ignored in this experiment.

(1) The international aviation network where the SARS virus spread [44] comprises 224 nodes and 2247 edges. The nodes represent the countries where flights arrived while the edges represent the routes between two countries. The important node set of the network is defined as the countries where the SARS virus spread at the early stage.

(2) The Genepath human gene signaling network [45] comprises 6306 nodes and 57,340 edges. Nodes represent genes while edges represent the relationship between nodes. The important node set of the network is defined as the Alzheimer's disease gene.

(3) The mouse protein interaction network [46] comprises 1187 nodes and 1557 edges. Nodes represent mouse proteins while edges represent the interaction between proteins. The important node set of the network is defined as mouse protein kinase.

(4) The yeast protein network [47] comprises 5093 nodes and 24,743 edges. The nodes represent proteins while edges represent the relationship between proteins. The important node set of the network is defined as the important protein of the yeast network.

The basic topology characteristics of the four real networks used in this work are shown in Table 1.

Table 1. Basic topological characteristics of real networks.

Dataset	n	m	n_1	$\bar{k}$	C
SARS	224	2247	18	20.06	0.65
Genepath	6306	57,340	51	18.19	0.32
Mouse	1187	1557	67	2.62	0.09
Yeast	5093	24,743	1167	9.72	0.1

Here, n refers to the number of nodes in the network, m refers to the number of edges in the network, n_1 refers to the number of important nodes in the network, $\bar{k}$ refers to the average degree of the network, and C refers to the average clustering coefficient of the network.

4.2. Evaluation Indexes

Precision, recall, and AUC are the three evaluation indexes used to quantify the relatively important nodes obtained by several algorithms in this work.

Precision is mainly used to measure whether the $top-L$ nodes in the results by the algorithm are predicted correctly. It is specifically defined as the proportion of correct predictions in $top-L$ nodes among the predicted results. The formula is defined as:

$$\text{precision} = \frac{N_r}{L} \tag{21}$$

where N_r refers to the frequency at which the $top-L$ nodes predicted by the algorithm occurred in the unknown important node set U.

Recall is mainly used to measure how many of the $top-L$ nodes predicted by the algorithm are correctly predicted. It is specifically defined as the proportion of the number of unknown important nodes n_r found in the $top-L$ nodes in the prediction results relative to all nodes in the unknown important node set U. The formula is defined as:

$$\text{recall} = \frac{n_r}{|U|} \tag{22}$$

AUC is mainly used to measure the precision of the algorithm as a whole. The formula is defined as:

$$\text{AUC} = \frac{0.5N_1 + N_2}{N} \tag{23}$$

The specific calculation process for AUC is as follows: one node is selected from the unknown important node set U, and another is selected from the unimportant node set V_2 in each experiment, and the relative importance scores of the two nodes are compared. If the two nodes receive the same score, then the score is recorded as 0.5 point; if the relative importance score of the node selected from the unknown important node set U is greater than that from the unimportant node set V_2, then the score is recorded as 1 point. N represents the number of all node combinations from the two sets U and V_2. After N independent experiments, the final AUC value is the sum of the scores of N experiments. Among them, the frequencies of getting 0.5 point and 1 point are N_1 and N_2, respectively.

4.3. Experimental Analysis

The core goal of this work is to find relatively important nodes from the target node set T. Therefore, the major subjects investigated from the four real networks selected, that is, all nodes of target node set T, need to be determined. From the important node set V_1, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, and 90% of the nodes are selected and used as known important nodes. The experiment in this paper treats the proportion of nodes equally; that is, the number of experiments corresponding to different proportion of nodes is the same. Different algorithms are used to find the relatively important nodes in the network. At the same time, precision, recall, and AUC values corresponding to different algorithms are calculated, and their values obtained from the experiments are averaged. Finally, the proposed DDMF method is used and compared with other comparative algorithms in terms of the three evaluation indexes.

The parameters of five other comparative algorithms are adjusted to be close to the optimal ones in the four networks. The specific values are as follows: $K = 3$ is taken from the Ksmar indexes, $S = 0.75$ is taken from the PPR indexes, and $\varphi = 0.0001$ is taken from the Katz indexes. In the NMF algorithm, random walk length *walk_length* is valued as 10, embedded vector length *size* is set to 128, and hyperparameters $p, q \in \{0.25, 0.50, 1, 2, 4\}$. In the NLD algorithm, the selection method of known important nodes *hub* is the same as that of the DDMF method. The experimental results of the three evaluation indexes are shown in Figures 2 and 3 and Table 2.

In this study, different proportions of nodes are selected from the important node set V_1 as the known important nodes R. The precision, recall, and AUC values are calculated by six relatively important node mining algorithms on the basis of experiments. The average value of 50 times in the experimental results is used as the final experimental result. Figure 2 shows the precision values of six relative importance node mining methods in the four networks. The X axis represents the proportion of nodes in the target node set T while the Y axis represents the precision of different node proportions. Figure 3 shows the recall rates of the six relative importance node mining algorithms in the four networks. The X axis represents the proportion of nodes in the target node set T while the Y axis represents the recall rates of different node proportions. Table 2 shows the AUC values obtained by the six relative importance node mining algorithms in the four networks.

The experimental results show that with the increase in the number of nodes in the target node set T, the precision of the algorithm decreases gradually while the recall rate increases gradually. In order to better simulate the actual situation of different real-world networks and to reduce accidental error, the important nodes of different batches are selected in different proportions from the important node set. Then the relatively important nodes corresponding to the important nodes of these different batches are calculated and mined. By calculating the arithmetic average of the relatively important nodes of different batches, the final relatively important nodes are obtained. In terms of precision, the proposed DDMF method is obviously better than the other five comparative algorithms in the SARS and Genepath networks, and all of them perform well in the mouse and yeast networks. In terms of recall, the DDMF method performs well in the SARS and mouse networks. Specifically, its recall, under multiple node proportions, is better than those of the comparative algorithms. The DDMF method ranks second for

the Genepath and yeast networks. In terms of the AUC, the DDMF method outperforms the others in the SARS, Genepath, and mouse networks and ranks second in the yeast network. In sum, the proposed DDMF method performs well in terms of all the evaluation indexes in the SARS, Genepath, and mouse networks and comes in second place in the yeast network. Specifically, the proportion of the important nodes in the yeast network is relatively large. Therefore, some errors may occur in calculating the distance distribution of important nodes.

In general, the proposed DDMF method achieves excellent performance in real and complex network datasets, especially in terms of the evaluation of precision and AUC. It is obviously better than several comparative algorithms. At the same time, the selected datasets come from different fields. The results indicate that the DDMF method is characterized by high precision and wide applicability in mining relatively important nodes in networks.

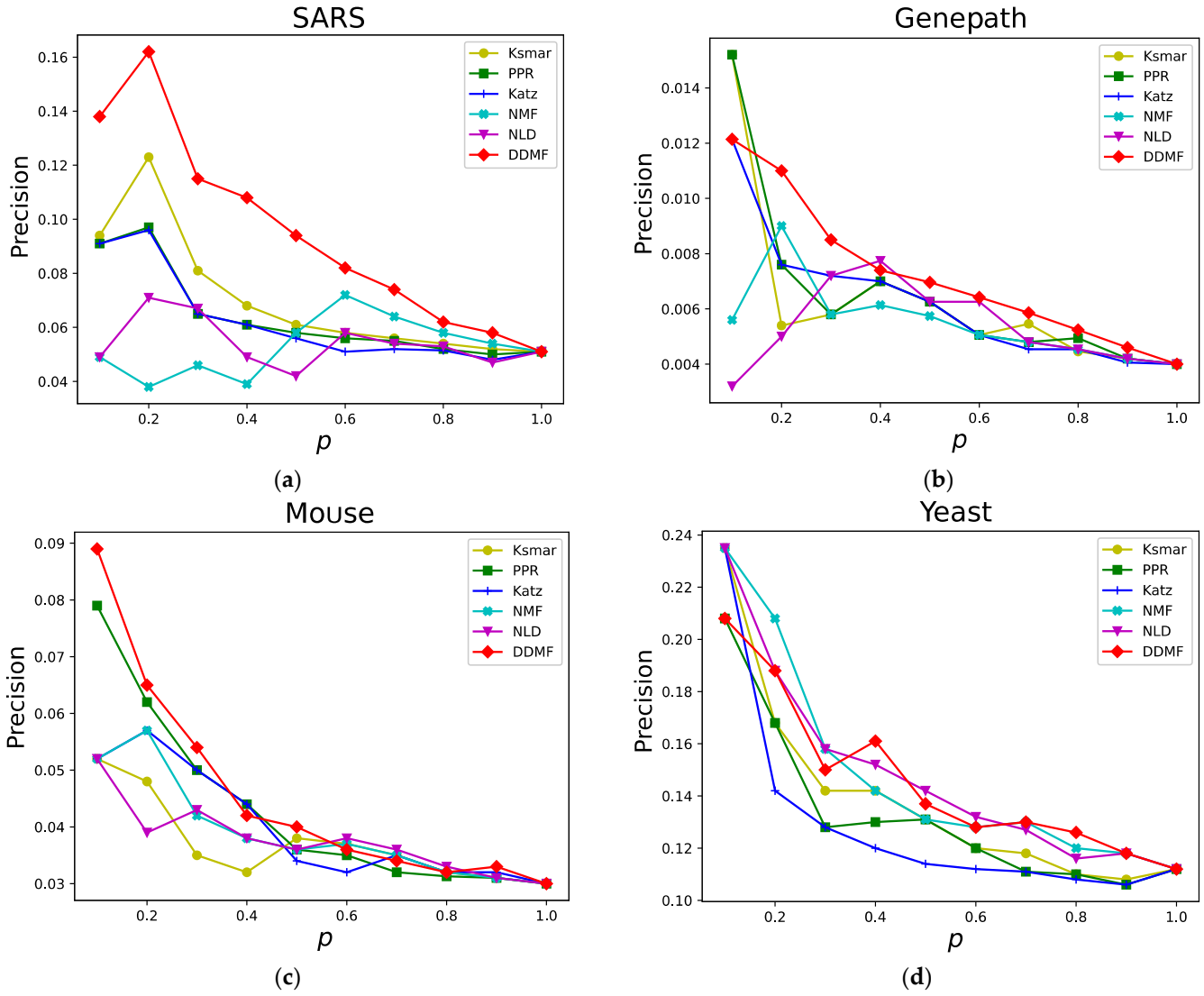

Figure 2. Precision rate results in four networks: (**a**) SARS network; (**b**) Genepath network; (**c**) Mouse network; (**d**) Yeast network.

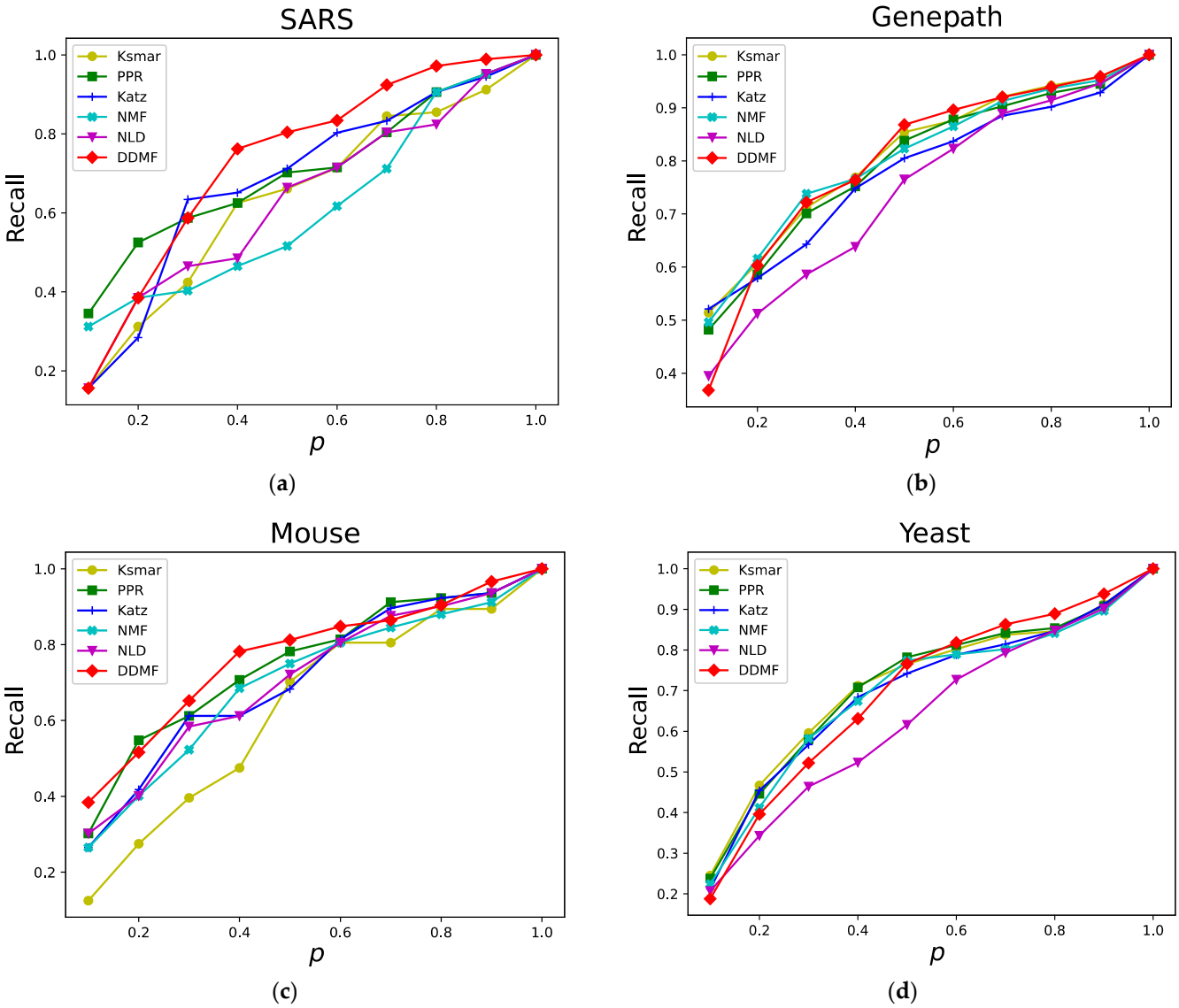

Figure 3. Recall rate results in four networks: (**a**) SARS network; (**b**) Genepath network; (**c**) Mouse network; (**d**) Yeast network.

Table 2. AUC results in four networks.

Dataset	Ksmar	PPR	Katz	NMF	NLD	DDMF
SARS	0.686	0.683	0.650	0.635	0.667	0.692
Genepath	0.545	0.526	0.482	0.568	0.565	0.675
Mouse	0.696	0.693	0.685	0.654	0.669	0.737
Yeast	0.596	0.582	0.564	0.686	0.665	0.669

5. Conclusions

A relatively important node mining method based on DDMF is proposed in this work. The DDMF method is mainly based on the distance distribution information of nodes. Starting from known important nodes, it aims to find relatively important nodes in a network. The detailed comparative experiments with five other algorithms for mining relatively important nodes in four real networks reveal that the DDMF method performs well in terms of precision and applicability. Moreover, the DDMF method can not only be used to mine the relatively important nodes in a network, but also be considered as a new idea for community detection and link prediction.

Mining relatively important nodes in complex networks is a challenging task with practical value. The DDMF method can be effectively used to find relatively important nodes in networks and provides a new idea and direction for the related work of network information mining in the future. With that being said, the limitation of the DDMF method can be summarized as something that it only considers mining relatively important nodes in single-layer networks. In the future, our relatively important nodes mining method can be applied to complex and diversified multilayer networks. Random walk could also be considered as a direction in future research.

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

Funding: This research was funded by the Special Plan of Yunnan Province Major Science and Technology Plan (202102AA100021), the National Natural Science Foundation of China (62066048), the Yunnan Natural Science Foundation Project (202101AT070167) and the Open Foundation of Key Laboratory in Software Engineering of Yunnan Province (2020SE311).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Ren, T.; Li, Z.; Qi, Y.; Zhang, Y.X.; Liu, S.M.; Xu, Y.J.; Zhou, T. Identifying vital nodes based on reverse greedy method. *Sci. Rep.* **2020**, *10*, 18. [CrossRef]
2. Li, A.W.; Xiao, J.; Xu, X.K. The Family of Assortativity Coefficients in Signed Social Networks. *IEEE Trans. Comput. Soc. Syst.* **2020**, *7*, 1460–1468. [CrossRef]
3. Liao, H.; Shen, J.; Wu, X.T.; Zhou, M.Y. Empirical topological investigation of practical supply chains based on complex networks. *Chin. Phys. B* **2017**, *26*, 144–150. [CrossRef]
4. Li, J.; Peng, X.Y.; Wang, J.; Zhao, N. A Method for Improving the Accuracy of Link Prediction Algorithms. *Complexity* **2021**, *2021*, 8889441. [CrossRef]
5. Paduraru, C.; Dimitrakopoulos, R. Responding to new information in a mining complex: Fast mechanisms using machine learning. *Min. Technol.* **2019**, *2019*, 1577596. [CrossRef]
6. Wang, T.; Chen, S.S.; Wang, X.X.; Wang, J.F. Label propagation algorithm based on node importance. *Phys. A: Stat. Mech. Its Appl.* **2020**, *551*, 124137. [CrossRef]
7. Meng, Y.Y.; Tian, X.L.; Li, Z.W.; Zhou, W.; Zhou, Z.J.; Zhong, M.H. Exploring node importance evolution of weighted complex networks in urban rail transit. *Phys. A: Stat. Mech. Its Appl.* **2020**, *558*, 124925. [CrossRef]
8. Liu, F.; Wang, Z.; Deng, Y. GMM: A generalized mechanics model for identifying the importance of nodes in complex networks. *Knowl.-Based Syst.* **2020**, *193*, 105464. [CrossRef]
9. Wen, T.; Jiang, W. Identifying influential nodes based on fuzzy local dimension in complex networks. *Chaos Solitons Fractals* **2019**, *119*, 332–342. [CrossRef]
10. Zhao, G.H.; Jia, P.; Zhou, A.M.; Zhang, B. InfGCN: Identifying influential nodes in complex networks with graph convolutional networks. *Neurocomputing* **2020**, *414*, 18–26. [CrossRef]
11. White, S.; Smyth, P. Algorithms for estimating relative importance in networks. In Proceedings of the 3th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 24 August 2003; pp. 266–275.
12. Magalingam, P.; Davis, S.; Rao, A. Ranking the importance level of intermediaries to a criminal using a reliance measure. *arXiv Preprint*, 2015; arXiv:1506.06221.
13. Magalingam, P. Complex network tools to enable identification of a criminal community. *Bull. Aust. Math. Soc.* **2016**, *94*, 350–352. [CrossRef]
14. Zhao, J.; Lin, L.M. A survey of disease gene prediction methods based on molecular networks. *J. Univ. Electron. Sci. Technol. China* **2017**, *46*, 755–765.
15. Zhu, J.F.; Chen, D.B.; Zhou, T.; Zhang, Q.M.; Luo, Y.J. A survey on mining relatively important nodes in network science. *J. Univ. Electron. Sci. Technol. China* **2019**, *48*, 595–603.

16. Schieber, T.A.; Carpi, L.; Díaz-Guilera, A.; Pardalos, P.M.; Masoller, C.; Ravetti, M.G. Quantification of network structural dissimilarities. *Nat. Commun.* **2017**, *8*, 110. [CrossRef]
17. Wang, Y.F.; Vasilakos, A.V.; Jin, Q.; Ma, J.H. PPRank: Economically Selecting Initial Users for Influence Maximization in Social Networks. *IEEE Syst. J.* **2017**, *11*, 2279–2290. [CrossRef]
18. Yang, Y.Z.; Yu, L.; Wang, X.; Zhou, Z.L.; Chen, Y.; Kou, T. A novel method to evaluate node importance in complex networks. *Phys. A: Stat. Mech. Its Appl.* **2019**, *526*, 121118. [CrossRef]
19. Li, X.L.; Zhang, P.; Zhu, G.Y. Measuring method of node importance of urban rail network based on h index. *Appl. Sci.* **2019**, *9*, 5189. [CrossRef]
20. Luo, J.W.; Wu, J.; Yang, W.Y. A relationship matrix resolving model for identifying vital nodes based on community in opportunistic social networks. *Trans. Emerg. Telecommun. Technol.* **2021**, *12*, e4389. [CrossRef]
21. Chang, H.; Cohn, D.; McCallum, A.K. Learning to create customized authority lists. In Proceedings of the Seventeenth International Conference on Machine Learning, San Francisco, CA, USA, 29 June–2 July 2000; pp. 127–134.
22. Haveliwala, T.H. Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *IEEE Trans. Knowl. Data Eng.* **2003**, *15*, 784–796. [CrossRef]
23. Jennifier, G.; Widom, J. Scaling personalized web search. In Proceedings of the 12th international conference on World Wide Web, New York, NY, USA, 20–24 May 2003; pp. 271–279.
24. Alzaabi, M.; Taha, K.; Martin, T.A. CISRI: A crime investigation system using the relative importance of information spreaders in networks depicting criminals communications. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2196–2211. [CrossRef]
25. Wang, H.; Chang, C.K.; Yang, H.I.; Chen, Y. Estimating the relative importance of nodes in social networks. *J. Inf. Processing* **2013**, *21*, 414–422. [CrossRef]
26. Rodriguez, M.A.; Bollen, J. An algorithm to determine peer-reviewers. In Proceedings of the 17th ACM conference on Information and knowledge management, New York, NY, USA, 26–30 October 2008; pp. 319–328.
27. Magalingam, P.; Davis, S.; Rao, A. Using shortest path to discover criminal community. *Digit. Investig.* **2015**, *15*, 117. [CrossRef]
28. Langohr, L. Methods for finding interesting nodes in weighted graphs. *Hels. Yliop.* **2014**, *11*, 145.
29. Cui, P.; Wang, X.; Pei, J.; Zhu, W.W. A survey on network embedding. *IEEE Trans. Knowl. Data Eng.* **2018**, *31*, 833–852. [CrossRef]
30. Zhang, P.Y.; Yao, H.P.; Li, M.Z.; Liu, Y.J. Virtual network embedding based on modified genetic algorithm. *Peer-Peer Netw. Appl.* **2019**, *12*, 481–492. [CrossRef]
31. Nelson, W.; Zitnik, M.; Wang, B.; Leskovec, J.; Goldenberg, A.; Sharan, R. To embed or not: Network embedding as a paradigm in computational biology. *Front. Genet.* **2019**, *10*, 381. [CrossRef]
32. Su, C.; Tong, J.; Zhu, Y.J.; Cui, P.; Wang, F. Network embedding in biomedical data science. *Brief. Bioinform.* **2020**, *21*, 182–197. [CrossRef]
33. Yao, H.P.; Ma, S.; Wang, J.J.; Zhang, P.Y.; Jiang, C.X.; Guo, S. A continuous-decision virtual network embedding scheme relying on reinforcement learning. *IEEE Trans. Netw. Serv. Manag.* **2020**, *17*, 864–875. [CrossRef]
34. Li, B.T.; Pi, D.C.; Lin, Y.X.; Cui, L. DNC: A Deep Neural Network-based Clustering-oriented Network Embedding Algorithm. *J. Netw. Comput. Appl.* **2021**, *173*, 102854. [CrossRef]
35. Song, G.J.; Wang, Y.; Du, L.; Li, Y.; Wang, J.S. Network Embedding on Hierarchical Community Structure Network. *ACM Trans. Knowl. Discov. Data* **2021**, *15*, 123. [CrossRef]
36. Zhao, J.; Yang, T.H.; Huang, Y.; Holme, P. Ranking candidate disease genes from gene expression and protein interaction: A Katz-centrality based approach. *PLoS ONE* **2011**, *6*, e0024306. [CrossRef]
37. Zhao, N.; Li, J.; Wang, J.; Peng, X.Y.; Jing, M.; Nie, Y.J.; Yu, Y. Relatively important nodes mining method based on neighbor layer diffuse. *J. Univ. Electron. Sci. Technol. China* **2021**, *50*, 121–126.
38. Mu, J.F.; Liang, J.Y.; Zheng, W.P.; Liu, S.Q.; Wang, J. Node similarity measure for complex networks. *J. Front. Comput. Sci. Technol.* **2019**, *14*, 749–759.
39. Liu, D.; Chen, X.; Peng, D. Some cosine similarity measures and distance measures between q-rung orthopair fuzzy sets. *Int. J. Intell. Syst.* **2019**, *34*, 1572–1587. [CrossRef]
40. Balaji, R.; Bapat, R.B.; Goel, S. Generalized Euclidean distance matrices. *arXiv Preprint*, 2021; arXiv:2103.03603. [CrossRef]
41. Gour, G.; Tomamichel, M. Entropy and relative entropy from information-theoretic principles. *IEEE Trans. Inf. Theory* **2021**, *67*, 6313–6327. [CrossRef]
42. Li, Y. Scheduling analysis of intelligent machining system based on combined weights. In Proceedings of the 2nd International Conference on Frontiers of Materials Synthesis and Processing, Sanya, China, 10–11 November 2018; Volume 493, p. 012146.
43. Grover, A.; Leskovec, J. node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 855–864.
44. Jani, P. Airport, Airline and Route Data. Available online: https://openflights.org/data.html (accessed on 28 December 2021).
45. Krauthammer, M.; Kaufmann, C.A.; Gilliam, T.C.; Rzhetsky, A. Molecular triangulation: Bridging linkage and molecular-network information for identifying candidate genes in Alzheimer's disease. *Proc. Natl. Acad. Sci. USA* **2004**, *101*, 15148–15153. [CrossRef]
46. Xenarios, I.; Rice, D.W.; Salwinski, L.; Baron, M.K.; Marcotte, E.M.; Eisenberg, D. DIP: The database of interacting proteins. *Nucleic Acids Res.* **2000**, *28*, 289–291. [CrossRef]
47. Li, M.; Zhang, H.H.; Wang, J.X.; Pan, Y. A new essential protein discovery method based on the integration of protein-protein interaction and gene expression data. *BMC Syst. Biol.* **2012**, *6*, 15. [CrossRef]