











Research Article

LRDC: Learning Resource Diffusion Based on Credibility for Computer-Supported Collaborative Learning

Peng Li,^{1,2,3,4,5} Yuanru Cui ,⁵ Qian Liu ,⁵ Meirui Ren ,^{1,2,3,4,5} Longjiang Guo ,^{1,2,3,4,5}
Hong Liu ,⁵ Lichen Zhang ,^{1,2,3,4,5} Xiaojun Wu ,^{1,2,3,4,5} Xiaoming Wang ,^{1,2,3,4,5}
Ning An ,⁵ and Jinhu Sun ,⁵

¹Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi'an 710062, China

²Key Laboratory of Intelligent Computing and Service Technology for Folk Song, Ministry of Culture and Tourism, Xi'an 710119, China

³Engineering Laboratory of Teaching Information Technology of Shaanxi Province, Xi'an 710119, China

⁴Xi'an Key Laboratory of Culture Tourism Resources Development and Utilization, Xi'an 710062, China

⁵School of Computer Science, Shaanxi Normal University, Xi'an 710119, China

Correspondence should be addressed to Meirui Ren; meirui@snnu.edu.cn and Longjiang Guo; longjiangguo@snnu.edu.cn

Received 2 January 2021; Revised 6 September 2021; Accepted 23 October 2021; Published 30 November 2021

Academic Editor: Xiao Zhang

Copyright © 2021 Peng Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Computer-supported collaborative learning (CSCL) is a learning strategy that gathers students together on campus through mobile application software on intelligent handheld devices to carry out creative exploration learning activities and social interaction learning activities. Learning resource diffusion is a very important constituent part of CSCL mobile software. However, learners will receive or forward a large number of learning resources such that short video, images, or short audio which will increase the energy consumption of forwarding nodes and reduce the message delivery success rate. How to improve the message delivery success rate is an urgent problem to be solved. To solve the aforementioned problem, this paper mainly studies the diffusion of learning resources in campus opportunistic networks based on credibility for CSCL. In campus opportunistic networks, learners who participate in collaborative learning can obtain the desired learning resources through the distribution and sharing of learning resources. Learning resource diffusion depends on the credibility of learners who participate in collaborative learning. However, the existing classical algorithms do not take into account the credibility between learners. Firstly, the concept of credibility in campus opportunistic networks is proposed, and the calculation method of credibility is also presented. Next, the problem of node initialization starvation is solved in this paper. The node initialization starvation phase of collaborative learning is defined and resolved in campus opportunistic networks. Based on the information of familiarity and activity between nodes formed in the process of continuous interaction, a learning resource diffusion mechanism based on node credibility is proposed. Finally, the paper proposes a complete learning resource diffusion algorithm based on credibility for computer-supported collaborative learning (LRDC for short) to improve the delivery success rate of learning resources on the campus. Extensive simulation results show that the average message diffusion success rate of LRDC is higher than that of classical algorithms such as DirectDeliver, Epidemic, FirstContact, and SprayAndWait under the different transmission speed, buffer size, and initial energy, which is averagely improved by 46.83%, 44.43%, and 45.6%, respectively. The scores of LRDC in other aspects are also significantly better than these classical algorithms.

1. Introduction

Collaborative learning (CL) is a strategy that organizes some learners to form some groups or teams to carry out learning tasks [1]. In order to achieve the learning goal, all members

of a collaborative learning group work together as a team to solve problems. The individual in a collaborative learning group can share learning resources with other members in the group, or with students outside the group. With the continuous innovation of educational ideology, the idea of

collaborative learning has been applied more and more widely, because it can better reflect the learner-centred educational ideology. In the process of group cooperation and interaction between learners, learners' participation and problem-solving skills are improved. Collaborative learning enhances the learning efficiency of individuals and the cohesion of the team in the collaborative group, which is also advantageous to enhancing learners' social skills. Therefore, CL is an effective way to improve learning efficiency.

Computer-supported collaborative learning (CSCL) is a learning strategy that gathers students together through mobile application software on intelligent handheld devices and promotes learners to carry out creative exploration activities and social interaction activities [2, 3]. Learning resource diffusion is a very important constituent part of CSCL mobile software. However, learners will receive or forward a large number of learning resources such that short video, images, or short audio which will increase the energy consumption of forwarding nodes and reduce the message delivery success rate. How to improve the message delivery success rate is an urgent problem to be solved. This is the motivation of this paper.

To solve the aforementioned problem, this paper mainly studies the diffusion of learning resources in campus opportunistic networks based on credibility for CSCL. In campus opportunistic networks, learners who participate in collaborative learning can obtain the desired learning resources through the distribution and sharing of learning resources. Learning resource diffusion depends on the credibility of learners who participate in collaborative learning. However, the existing classical algorithms do not take into account the credibility between learners.

First of all, this paper presents the definition of campus opportunistic networks. Opportunistic network [4, 5] is a kind of self-organizing network, which does not need a complete link between the source node and the destination node and uses the encounter opportunity brought by node movement to realize communication. The campus opportunistic network is a wireless self-organization network on campus, in which many learner nodes can participate in campus collaborative learning by using handheld smart devices. Every learner is a learner node participating in communication, and the communication manner depends on learner's community and the opportunistic networks formed by the information of the learners in the community.

Next, the concept of credibility in campus opportunistic network is proposed, and the calculation method of credibility is also presented in this paper. In campus opportunistic networks, learning resource diffusion is a very important constituent part for CSCL mobile software. In campus opportunistic networks, the learners who participate in collaborative learning can obtain the desired learning resources through the distribution and sharing of learning resources. Learning resource diffusion depends on credibility of learners who participate in collaborative learning. The diffusion of campus learning resources depends on the credibility between nodes. Because there is no credibility, resource diffusion between nodes is meaningless. However, the existing classical algorithms such as Epidemic [6], DirectDeliver

[7], FirstContact [8], and SprayAndWait [9] do not take into account the credibility between learners. These algorithms are not suitable for resource diffusion on campus opportunistic network. This paper studies the diffusion of learning resources in campus opportunistic networks based on credibility for CSCL.

Then, this paper solves the problem of the node initialization starvation on campus opportunistic networks. Campus opportunistic networks have the characteristics of high node density, complex social relations among individuals and communities, and the complicated links among nodes. So, it is particularly difficult to select the suitable node to forward learning resources. The selection of the suitable forwarding node depends on community to which the learner belongs. The community difference between nodes is a key factor for selecting suitable forwarding nodes [10]. However, community difference among nodes is not significant in the early stage forming collaborative learning community. Therefore, it is difficult to select forwarding nodes in this stage. In this paper, in the early stage of community formation, the selection of forwarding nodes is called as the node initialization starvation stage of collaborative learning in campus opportunistic networks. In social networks, the current mainstream solutions of node initialization starvation problem focus on the problems of information overload, accuracy, and efficiency of recommendation. However, node initialization starvation problem of collaborative learning in opportunistic networks is quite different from that of mainstream social networks, because in the general social networks [11], there will be no message forwarding delay which caused by insufficient contact between mobile nodes. Therefore, the solution of node initialization starvation problem in the general social networks cannot be directly applied to campus opportunistic networks.

Finally, for node initialization starvation problem, this paper defines the node initialization starvation stage of collaborative learning in campus opportunistic networks and proposes LRDC algorithm. In the early stage of collaborative learning in campus opportunistic networks, the effective information collected by mobile learner nodes is less, and the community formed is relatively unstable. In order to obtain the effective information of nodes in the community as soon as possible, complete the information transmission between nodes in the node initialization starvation stage, and improve the transmission efficiency of learner nodes, this paper defines the node initialization starvation stage of collaborative learning in campus opportunistic networks.

In this paper, activity and familiarity are comprehensively considered as credibility to select forwarding nodes. According to the rule of node movement, the process of learning resource diffusion is divided into node initialization starvation stage and community operation stage. A novel learning resources diffusion algorithm based on credibility named LRDC is proposed in this paper. For collaborative learning in campus opportunistic networks, the contributions of this paper are given as follows:

- (i) First of all, this paper defines the concept of campus opportunistic network and proposes the concept of

credibility in campus opportunistic network and gives the calculation method of credibility, which is not found in other papers. According to the continuous interaction information between the learner nodes, the familiarity and activity of the nodes are calculated, the credibility of the nodes is accurately evaluated, and the diffusion of learning resources is completed efficiently based on the credibility

- (ii) Next, this paper effectively solves the problem of node initialization starvation. The learner node can quickly obtain the effective information of the node initialization starvation stage in the learning community, complete the information transmission between the nodes in the node initialization starvation stage, and improve the transmission efficiency
- (iii) Finally, this paper proposes a complete LRDC algorithm to diffuse learning resources on campus opportunistic networks and improve the delivery success rate of learning resources on the campus. An extensive simulation results show that the average message diffusion success rate of LRDC is higher than that of classic algorithms such as DirectDeliver, Epidemic, FirstContact, and SprayAndWait under the different transmission speed, buffer size, and initial energy, which is averagely improved by 46.83%, 44.43%, and 45.6%, respectively. The scores of LRDC in other aspects is also significantly better than these algorithms

The rest of this paper is organized as follows. The second section introduces related work. The third section describes the starvation phase division of learner node initialization. The fourth section introduces modelling process of credibility degree for campus collaborative learning nodes. The fifth section describes the diffusion mechanism of learning resources in the collaborative learning environment of campus opportunity network. The sixth section carries on the simulation and the result analysis. At last, the seventh section summarizes the full paper.

2. Related Work

In order to solve the information transmission between the nodes in the node initialization starvation stage, researchers propose different routing algorithms for different situations. In social networks, initialization stage refers to the use of a large number of newly created services and these services of no usage records [12], the node initialization starvation problem can be divided into new project initialization starvation problem and new user initialization starvation problem [13]. In order to solve the node initialization starvation problem in social networks, Huang et al. proposed and implemented the prediction of the potential collaboration relationship between users and the relevant recommendation mechanism based on the model of service relationship [14]; Zhang et al. parsed user's demand information based on the LDA (latent Dirichlet allocation) model and keyword information to improve accuracy of the recom-

mendation [15]. In social networks, the above mainstream solutions of the node initialization starvation problem are mainly aimed at providing customers with product information and suggestions and dealing with the problems of information overload, accuracy, and efficiency of recommendation and do not consider the node initialization starvation problem in collaborative learning opportunistic networks and the distribution and forwarding of learning resources with social intimacy and node influence. In campus opportunistic networks, the node initialization starvation problem refers to that in the initial stage of opportunistic network; due to the insufficient information of link connection and the delay in obtaining the information of each node, the interaction between nodes and data diffusion are blind, and the message delivery rate is low; at the same time, the transmission opportunity is wasted. Therefore, the above schemes are not suitable for collaborative learning opportunistic networks. This paper considers the node initialization starvation problem of campus collaborative learning opportunistic networks, make full use of contact information between nodes and the characteristics of nodes, and promote the efficient and rapid transmission of node information in the campus collaborative environment.

In the opportunistic network, relevant scholars have carried out a series of explorations on familiarity and activity of nodes. For example, Wang et al. [16] calculated the node activity according to the area covered by moving trajectory of the node and the average dwell time of the node. Feng et al. [17] proposed a delayed back off algorithm based on the active state of neighbour nodes; in the algorithm, the delayed back off window is adaptively adjusted according to the active number of neighbour nodes. Rao et al. [18] considers the link similarity of nodes according to the node link condition in k -hop. The LFM algorithm proposed by Lancichinetti et al. [19] is based on the consideration of the similarity between the feature of nodes in contact with each other. However, these algorithms are not suitable as the diffusion mechanism of collaborative learning resources in campus opportunistic networks.

Among the classical algorithms in opportunistic networks, Amin and David [6] proposed a flooding algorithm named Epidemic similar to the spread of infectious diseases. When two nodes meet, they exchange packets carried by each other. This algorithm has large network load and poor scalability. Shah et al. [7] proposed an algorithm named DirectDeliver that the source node forwards the data only when it meets the destination node. This algorithm has poor performance in delivery rate and delay. Zhao et al. [8] proposed FirstContact algorithm that the source node forwards the data to the meeting node, and then, these mobile nodes forwarded the data to the destination node. Spyropoulos et al. [9] proposed SprayAndWait algorithm to limit the number of data backup. When the number of data backup in the network reaches a certain value, the data will stop spreading and adopt the strategy of direct waiting. The above algorithms have defects in message delivery rate and delivery delay, and they all do not consider credibility between learners in campus opportunistic networks. Therefore, this paper proposes LRDC algorithm to solve the problem of learning resource diffusion based on creativity for computer-supported collaborative learning, and after

TABLE 1: Node initialization information table.

Variable name	Meaning
NS_{ij}	Number of encounters between node i and node j
Sig_i	The signal value of the node i
Init $i \{ NS_{ij_1}, \dots, NS_{ij_p} \} \{ 0 \leq j_1 \leq j_p \leq n \}$	Initial queue
Q	Community modularity
m	Total number of edges in the network
d_j	Degree of node j
A	Initial matrix

verification in extensive simulation results, LRDC is better than these classical algorithms.

3. The Learner Node Initialization Starvation Phase

In the mobile social network, a single learner node is used as a unit for message transmission. Therefore, in the collaborative learning community based on mobile social networks, the running process of the network can be divided into two phases such as the node initialization starvation phase and the normal operation phase according to the communication situation of a single node. In the campus, the learner nodes are denser, and every learner's daily schedule has more obvious regularity and shows more obvious characteristics of group activities, so the community between learner nodes is more obvious.

Campus collaborative learning community is a virtual learning organization in which students participate in groups to maximize the acquisition results under a certain incentive mechanism. In the learning community, students cooperate each other based on their own will and the common goal.

As previously mentioned, in the early stage of collaborative learning in campus opportunistic networks, the effective information collected by mobile learner nodes is less, and the formed community is relatively unstable, so this paper will divide the running process of campus opportunistic networks into two phases: the node initialization starvation phase and the normal operation phase.

The node initialization starvation phase is the phase in which the community of the node is in an unstable state. When the community of a learner node is in a stable state, it can be considered that the node initialization starvation phase will end.

Adjacency matrix is the most suitable data structure to express social network relation, NS represents the set of the number of encounters between two nodes, the element NS_{ij} indicates the number of encounters between node i and node j . For any node i , set the initial value of its signal $Sig_i = n - 1$ (n is the upper limit of the number of students in the campus collaboration network), initial queue, initial matrix, and other parameters are maintained (as shown in Table 1).

Next, we show how to determine whether the community in which the node is located is stable, that is, the node initialization starvation phase ends. Firstly, we introduce the initialization adjacency matrix to represent the encounter situation between nodes in the starvation initialization phase. The formation process of the initialization adjacency matrix is as follows:

Case 1. When node p encounters to node q and the signal value of node p (p can be any node) and the signal value of node q are not greater than 0, the signal values of p and q will not be changed.

Case 2. When the signal value of node p is greater than 0 (p can be any node) and node p encounters to node q whose signal value is greater than 0, the signal value of node p and node q are subtracted by one, and the signal value of node q in node p 's signal queue is increased by one, simultaneously, and the signal value of node p in node q 's signal queue is also increased by one.

Case 3. When the signal value of node p is not greater than 0 (p can be any node), and node p encounters to node q whose signal value is greater than 0:

- (a) If the queue of node p does not contain node q , the signal value of node p will be kept original, and the node q will be added to the queue of node p , and the signal value of the node q in the node p 's queue will be added by one. At the same time, node p is added to the queue of node q , the signal value of node q is subtracted by two, and the signal value of node p in node q 's queue is added by one
- (b) If the initial queue of node p contains node q , the signal value of node q in node p 's signal queue is increased by one, and the signal value of node p in node q 's signal queue is added by one, and the signal value of node q itself is subtracted by two

Case 4. When the signal value of node p is greater than 0 (p can be any node), and node p encounters to node q whose signal value is not greater than 0, node p and node q do nothing.

When the signal values of all nodes are less than or equal to 0, the initialization phase is completed. At this time, we need to further judge through the community modularity.

In this section, we use modularity which is a common method to measure the stability of network community to detect the initial communities [20, 21]. The definition of community modularity Q is as follows:

$$Q = \frac{1}{2m} \sum_{ij} \left(NS_{ij} - \frac{d_i d_j}{2m} \right) \varepsilon(c_i, c_j), \quad (1)$$

where m is the total number of edges in the network, d_i and d_j , respectively, represent the degree of node i and node j . NS_{ij} represents the number of encounters between node i and node j ; if there is no encounter between node i and node j , the value of NS_{ij} is 0; otherwise, it is equal to the number of encounters. c_i represents the community where node i is located, and c_j represents the community where node j is located. If node i and node j are in the same community, the value of $\varepsilon(c_i, c_j)$ is 1; otherwise, $\varepsilon(c_i, c_j)$ is 0.

In this scenario, c_i and c_j are unknown, so the value of $\varepsilon(c_i, c_j)$ cannot be confirmed, but it can be deduced according to the number of encounters between node i and node j . If the number of encounters between node i and node j is greater than the average number of encounters between node i and all other nodes in the community, it indicates that node i and node j are in the same community.

Take node i as an example, and the number of encounters between node i and node j is 5, and the average number of encounters between node i and all initial nodes is 2; it is obvious that node i and node j are in the same community.

The value range of Q is [0, 1], but in the actual network, the value of Q is usually between (0.3, 0.7) [22, 23]. So, in this paper, we set that when the Q value is greater than 0.3, the node starvation phase ends, and the network is in a relatively stable community.

This paper simulates and calculates the value of modularity in campus collaborative learning community, and through calculation, it is concluded that the Q is greater than 0.3. At this point, it is determined that the community is in a stable state and obtained the following initialization matrix NS which represents the community network structure.

4. Model for Credibility between Nodes

In this section, we introduce the concept of credibility between nodes. This concept serves as a metric for selecting the next forwarding node in the normal operation phase. In this section, there basic variables are shown in Table 2.

4.1. Familiarity

4.1.1. *Matching Matrix between Nodes.* Suppose that NS represents the community network structure. A specific com-

TABLE 2: Variable name table.

Variable name	Meaning
C_{ij}	The credibility of node j to node i
AD_j	Activity of node j
FD_{ij}	Familiarity between node i and node j
S_i	The sum of weights of all edges connected to node i
WE_{ij}	The edge weight between nodes i and j
i_Asb	The direct node set of node i

munity network structure is given as follows.

$$NS = \begin{bmatrix} i_1 & i_2 & i_3 & i_4 & i_5 & i_6 & i_7 & i_8 & i_9 & i_{10} & i_{11} & i_{12} & i_{13} & i_{14} \\ i_1 & 0 & 4 & 2 & 1 & 3 & 4 & 3 & 2 & 0 & 0 & 3 & 0 & 0 & 0 \\ i_2 & 4 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ i_3 & 2 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ i_4 & 1 & 0 & 3 & 0 & 2 & 0 & 2 & 0 & 0 & 0 & 4 & 3 & 0 & 0 \\ i_5 & 3 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 & 0 & 0 \\ i_6 & 4 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 2 & 0 & 4 & 0 & 0 & 0 \\ i_7 & 3 & 2 & 0 & 2 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ i_8 & 2 & 0 & 0 & 0 & 2 & 2 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ i_9 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ i_{10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ i_{11} & 3 & 0 & 0 & 4 & 2 & 4 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ i_{12} & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ i_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ i_{14} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (2)$$

Matrix A_{i_1} represents the adjacency relationship between nodes in the community with the node i_1 as the center node. Suppose that A_{i_1} based on the above NS is given as follows.

$$A_{i_1} = \begin{bmatrix} i_1 & i_2 & i_3 & i_4 & i_5 & i_6 & i_7 & i_8 & i_{11} \\ i_1 & 0 & 4 & 2 & 1 & 3 & 4 & 3 & 2 & 3 \\ i_2 & 4 & 0 & 0 & 3 & 0 & 0 & 2 & 0 & 0 \\ i_3 & 2 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ i_4 & 1 & 3 & 3 & 0 & 2 & 0 & 2 & 0 & 4 \\ i_5 & 3 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 2 \\ i_6 & 4 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 \\ i_7 & 3 & 2 & 0 & 2 & 0 & 1 & 0 & 1 & 0 \\ i_8 & 2 & 0 & 0 & 0 & 2 & 2 & 1 & 0 & 2 \\ i_{11} & 3 & 0 & 0 & 4 & 2 & 4 & 0 & 2 & 0 \end{bmatrix}$$

Taking the adjacency matrix of aforementioned community network structure NS as an example, suppose that node i_1 carries messages, and nodes i_1 and i_6 are candidate transmission nodes. The matching matrix for node i_6 to node i_1 is calculated as follows: the number of rows and columns of matrix $A_{i_6 \& i_1}$ is the same as that of A_{i_1} , and if there is a nonzero element in matrix A_{i_1} and its corresponding position in matrix A_{i_6} is also a nonzero element, then the value of the element at that position in matrix $A_{i_6 \& i_1}$ is the same as that

of A_{i_1} ; otherwise, the value is 0 in matrix $A_{i_6 \& i_1}$. In the same way as above, based on $A_{i_6 \& i_1}$ and A_{i_7} , the matching matrix $A_{i_7 \& (i_6 \& i_1)}$ can be obtained. For example, the matching matrix $A_{i_6 \& i_1}$ and $A_{i_7 \& (i_6 \& i_1)}$ are as follows:

$$\begin{aligned}
 A_{i_6 \& i_1} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 4 & 3 & 2 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 \\ 3 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 2 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 2 \\ 3 & 0 & 0 & 0 & 0 & 4 & 0 & 2 & 0 \end{bmatrix}, \\
 A_{i_7 \& (i_6 \& i_1)} &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 4 & 3 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 \\ 3 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{3}
 \end{aligned}$$

4.1.2. Familiarity between Nodes. In this section, this paper calculates the social familiarity between nodes through the situation of nodes' common neighbors. The familiarity between node i and node j is to be calculated as follows:

$$FD_{ij} = SN_{ij} \cdot SC_{ij}, \tag{4}$$

where FD_{ij} is the familiarity between node i and node j and S_{ij} is the proportion of the common neighbors of nodes i and j to all the nodes contacted by nodes i and j , and SC_{ij} is the link contribution of the common neighbors of nodes i and node j to nodes i and j . WE_{ij} is the edge weight between node i and node j , and S_i is the sum of the weights of all edges connected to node i . The edge strength ES_{ij} is defined as the ratio of WE_{ij} to the sum of S_i and S_j . The calculations of SN_{ij} , ES_{ij} , and SC_{ij} are shown in formulas (5), (6), and (7), respectively.

$$SN_{ij} = \frac{|i_Asb \cap j_Asb|}{|i_Asb \cup j_Asb|}, \tag{5}$$

$$ES_{ij} = \frac{WE_{ij}}{S_i + S_j}, \tag{6}$$

$$SC_{ij} = \sum_{c \in i_Asb \cap j_Asb} \left\| A_{c \& (i \& j)} \right\|_F^2 \cdot ES_{ij}. \tag{7}$$

In formula (7), c is any node in the set of common neighbor nodes of nodes i and j , i_Asb is the direct node set of node i , $A_{c \& (i \& j)}$ is the matching matrix of c for nodes i and j , and $\|A_{c \& (i \& j)}\|_F^2$ represents the F -norm of the matrix $A_{c \& (i \& j)}$. It can be considered that the larger the SC_{ij} is, the higher the contribution of the shared neighbor nodes to the links of nodes i and j . Let us take adjacency matrix NS in Section 3 as an example to calculate the familiarity between node i_1 and node i_6 .

$$ES_{i_1 i_6} = \frac{4}{(4+2+1+3+4+3+2+3) + (4+1+2+2+4)} = \frac{4}{35},$$

$$SC_{i_1 i_6} = \left(\|A_{i_7 \& (i_6 \& i_1)}\|_F^2 + \|A_{i_8 \& (i_6 \& i_1)}\|_F^2 + \|A_{i_{11} \& (i_6 \& i_1)}\|_F^2 \right) \cdot ES_{i_1 i_6}$$

$$= \left[\begin{array}{c} \left\| \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 4 & 3 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 \\ 3 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \right\|_F^2 \\ + \left\| \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 4 & 3 & 2 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 \\ 3 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 2 \\ 3 & 0 & 0 & 0 & 0 & 4 & 0 & 2 & 0 \end{bmatrix} \right\|_F^2 \\ + \left\| \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 4 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 2 \\ 3 & 0 & 0 & 0 & 0 & 4 & 0 & 2 & 0 \end{bmatrix} \right\|_F^2 \end{array} \right] \times \frac{4}{35} = 32.46, \tag{8}$$

$$SN_{i_1 i_6} = \frac{4}{22+11} = \frac{4}{33},$$

$$FD_{i_1 i_6} = \frac{4}{33} \times 32.46 = 3.93.$$

4.2. Activity of Node. In the early stage of learning community in campus opportunistic networks, because any node does not contain or only contains less communication characteristic information of other nodes, when any node needs to send messages, it cannot select the forwarding node correctly according to the node information table it maintains, which will cause a lot of contact opportunity waste, message redundancy, and the increase of message transmission delay.

When the node in initialization starvation phase performs message transmission, it will select the next hop node by comprehensive consideration of its activity and familiarity.

When calculating the activity of node i (marked as AD_i), the value of AD_i can be divided into direct contact degree AD_{dir_i} and indirect contact degree AD_{rel_i} as shown in formula (9); AD_{dir_i} represents the ability that node i contacts directly other nodes, and AD_{rel_i} reflects the ability that node i to contact indirectly other nodes through its direct neighbors, as shown in the formulas (10) and (11), respectively. In formula (11), $NS_{ij(c)}$ represents the number of times that node j contacts the common neighbors of node i and node j , and it can be calculated by formula (12).

$$AD_i = AD_{dir_i} + AD_{rel_i}, \quad (9)$$

$$AD_{dir_i} = \sum_{j \in i_{Asb}} NS_{ij}, \quad (10)$$

$$AD_{rel_i} = \sum_{j \in i_{Asb}} \frac{AD_{dir_j} - NS_{ij(c)} - NS_{ij}}{AD_{dir_j}} \times NS_{ij}, \quad (11)$$

$$NS_{ij(c)} = \sum_{k \in (i_{Asb} \cap j_{Asb})} NS_{jk}. \quad (12)$$

4.3. Credibility of Nodes. In order to solve the problem of finding learning resources for learners in collaborative learning environment, this paper proposes a learning resource diffusion strategy based on credibility for computer-supported collaborative learning. According to common sense, a node that you are familiar with and active, you will believe that it has the ability to spread messages. So, a learner node wants to forward its learning resources as soon as possible; it needs to find a node that is familiar with it and highly active. Therefore, the paper defines the credibility as the weighted sum of familiarity and activity (as shown in formula (13)). The credibility of nodes i to j is denoted C_{ij} , in particular, C_{ij} and C_{ji} are not necessarily equal. AD_j is the activity of node j , the familiarity between nodes i and j is represented by FD_{ij} , and θ represents the weight coefficient.

$$C_{ij} = \theta \cdot AD_j + (1 - \theta) \cdot FD_{ij}. \quad (13)$$

This paper uses formula (14) to normalize the value of C_{ij} , where max and min represent, respectively, the maximum and minimum values of credibility.

$$X^* = \frac{x - \min}{\max - \min}. \quad (14)$$

5. Learning Resource Diffusion Algorithm Based on Credibility

In the context of collaborative learning on campus, the movement of learners' nodes has strong regularity and community. Based on the initialization process of node starvation phase proposed in Section 3 and the calculation method of credibility proposed in Section 4, this section proposes LRDC (learning resource diffusion based on credibility) algorithm and discusses the learning resource diffusion strategy in cooperative learning community as follows.

For any node i , establish direct contact node linked list $\psi_{i_all} = (i, c_1, c_2, \dots, c_m, d_1, d_2, \dots, d_n)$, community node linked list $\psi_{i_com} = (i, c_1, c_2, \dots, c_m)$, and noncommunity node linked list $\psi_{i_dir} = (i, d_1, d_2, \dots, d_n)$, respectively. In ψ_{i_all} , $c_i (i = 1, 2, \dots, m)$ indicates the node whose contact times with node i is greater than the average contact number between node i and all nodes, and $c_i (i = 1, 2, \dots, m)$ will be added to the community node linked list ψ_{i_com} ; $d_j (j = 1, 2, \dots, n)$ indicates the node whose contact times with node i is not greater than the average contact number between node i and all nodes, and $d_j (j = 1, 2, \dots, n)$ will be added to the community node linked list ψ_{i_dir} .

The first element in the above each linked list is the central node i , and the other elements are sorted according to the credibility of the node i to them from large to small. When the credibility is the same, they are sorted according to the times of contact with node i . When any two nodes contact each other, they update their node linked lists according to the contact information of each other. The contact linked list representation of node i is shown in Figure 1.

Region 1 is the community node linked list of node i , region 2 is the noncommunity node linked list $\psi_{i_dir} = (i, d_1, d_2, \dots, d_n)$ of node i and the community node linked list of each element except node i in ψ_{i_com} , and region 3 is the noncommunity node linked list of each element except node i in ψ_{i_com} .

According to the elements in the linked lists of region 2 and region 3, the matrices A_{id} and A_{ic} are established, respectively, as follows:

$$A_{id} = \begin{bmatrix} d_1 & c_1 c_1 & c_2 c_1 & \dots & c_m c_1 \\ d_2 & c_1 c_2 & c_2 c_2 & \dots & c_m c_2 \\ \dots & \dots & \dots & \dots & \dots \\ d_n & c_1 c_m & c_2 c_m & \dots & c_m c_m \end{bmatrix}, \quad (15)$$

$$A_{ic} = \begin{bmatrix} c_1 d_1 & c_2 d_1 & \dots & c_m d_1 \\ c_1 d_2 & c_2 d_2 & \dots & c_m d_2 \\ \dots & \dots & \dots & \dots \\ c_1 d_n & c_2 d_n & \dots & c_m d_n \end{bmatrix}.$$

Taking node i as an example, the priority determination strategy for selecting forwarding nodes is as follows: the priority of nodes in the linked list ψ_{i_com} is the highest, the

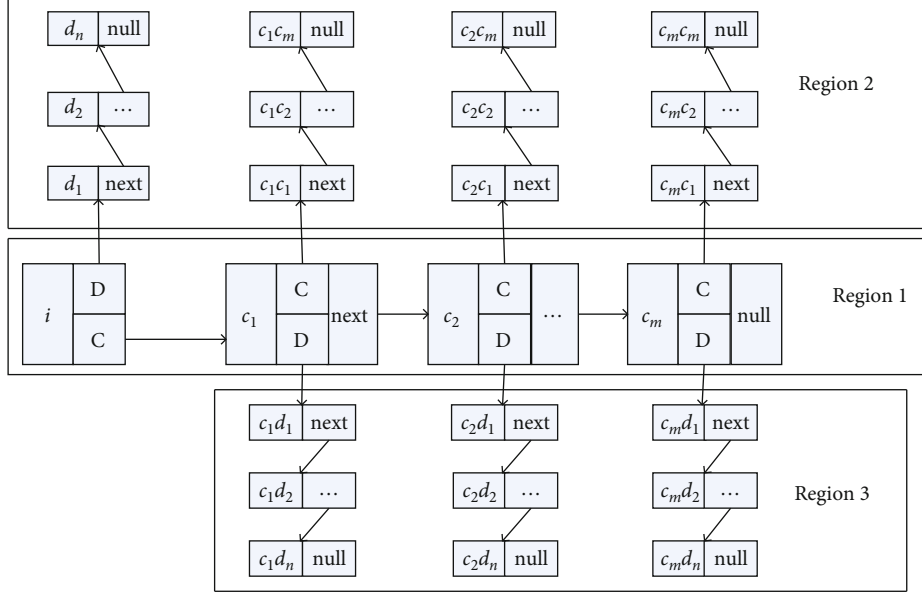


FIGURE 1: Node contact linked list diagram.

priority of nodes in A_{id} matrix is the second, and the priority of nodes in A_{ic} matrix is the lowest.

Suppose that node i is the sending node and j is the destination node. The routing and forwarding scheme is described as below:

- (1) If there is a destination node j in the community node linked list of node i , the node i 's message is sent directly to the destination node j ; otherwise, the matrix A_{ic} and A_{id} are traversed according to the priority order, respectively, and if the node j exists in matrix A_{ic} or A_{id} , the node i 's message is forwarded to the destination node j by the corresponding relay nodes
- (2) If the situation in (1) does not occur, the message is not forwarded

The LRDC algorithm is described in Figure 2.

6. Simulation and Result Analysis

6.1. Simulation Setup. In order to evaluate the performance of the algorithm described in this paper, ONE (Opportunistic Network Environment) simulator is used for experimental simulation. In this paper, the real dataset CRAWAD dataset Cambridge/haggle [24] is used for simulation experiment, and the main experimental parameters are set as shown in Table 3.

6.2. The Evaluation on Message Transmission. This section presents a set of experiments comparing the performance of LRDC, DirectDeliver, Epidemic, FirstContact, and SprayAndWait and considering the effect of simulation time on the message delivery success rate, the average residual energy of nodes, and the average hops of message transmission.

This section has two goals. The first goal is to quantify the effect of simulation time on the message delivery success

rate; the second goal is to quantify the average residual energy of nodes and the average hops of message transmission after 90000 s simulation time. Set other main parameters of simulation environment to $initialEnergy = 50000$, $bufferSize = 100$ MB, $transmitSpeed = 150$ KB/s, and $msgttl = 900$ min; the simulation results are given in Tables 4 and 5.

For the first goal, with the increase of simulation time, the success rate of message delivery of five routes is increasing continuously. And in the same simulation time period, LRDC's message delivery success rate is the highest among the five routing algorithms. It can be seen that LRDC can consume less energy and deliver messages to the destination node at a faster speed without causing message redundancy when the community of the node is not fully mature, and its delivery time can approach the theoretical limit.

For the second goal, this section has three subgoals. For the first subgoal, the higher the average residual energy, it indicates that the performance of routing algorithm is better. It can be seen from Table 5 that the average residual energy of DirectDeliver is the highest, but the routing algorithm is not suitable for the campus collaborative learning situation based on opportunistic network. The average residual energy of SprayAndWait and LRDC is also high, but the average delay of message transmission of SprayAndWait is poor. For the second subgoal, the shorter the average delay time of message transmission, it indicates that the performance of routing algorithm is better. It can be seen from Table 5 that the average delay time of message transmission of Epidemic is the shortest and that of LRDC is also shorter, but the average residual energy of nodes of LRDC is much higher than that of Epidemic. For the third subgoal, the lower the average hops of message transmission, the lower the network overhead; it indicates that the performance of routing algorithm is better. The average hops of message

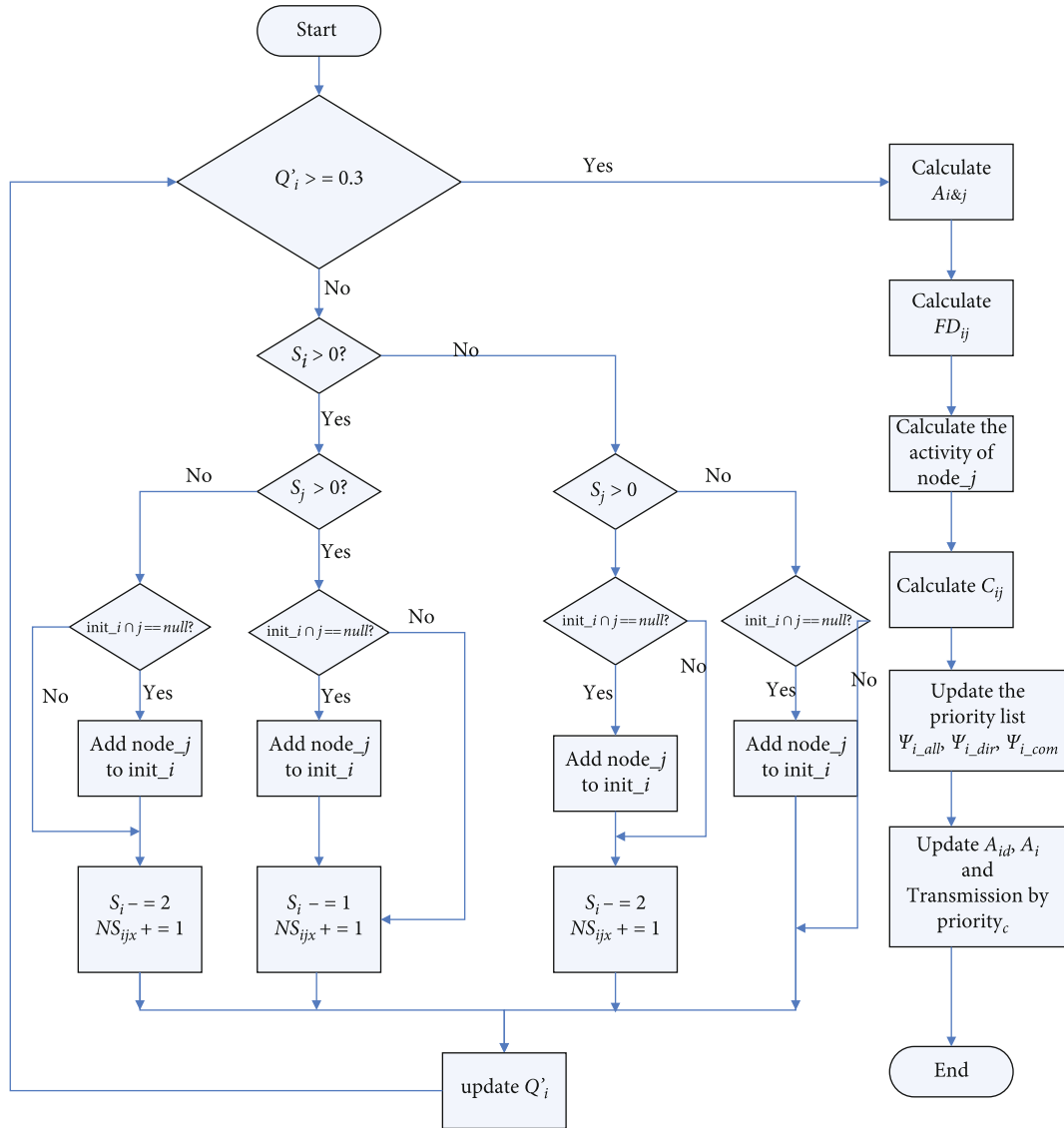


FIGURE 2: The flow chart of node initialization and learning resource diffusion in LRDC.

TABLE 3: The simulation parameter settings.

Parameter type	Value
Data set	CRAWDAD dataset Cambridge/haggle
Simulation duration/s	92000
Transfer file size/kB	50 k, 5000 k
Message generation interval/s	300, 5000
Number of nodes	98
Number of message copies from	20

transmission of DirectDeliver is the lowest, but the routing algorithm is not suitable for the campus collaborative learning situation based on opportunistic network. The average residual energy of LRDC and SprayAndWait is also low, but the average delay of message transmission of SprayAndWait is poor.

It can be seen from Tables 4 and 5 that the performance of LRDC, the average delay of message transmission, and the average hop number of message transmission are the best, so the performance of LRDC is the best.

This paper uses data normalization to process Table 5 and get the normalized score table in Table 6. Data

TABLE 4: The effect of simulation time on message delivery success rate.

Algorithm/simulation time (ten thousand seconds)	1	2	3	4	5	6	7	8	9	Average success rate	Relative promotion ratio (%)
LRDC	0.07	0.1	0.2	0.27	0.29	0.53	0.62	0.74	0.94	0.42	0
DirectDeliver	0	0	0	0.12	0.18	0.32	0.35	0.35	0.35	0.15	176
Epidemic	0.06	0.09	0.18	0.27	0.29	0.53	0.62	0.74	0.94	0.35	19
FirstContact	0	0	0	0.12	0.24	0.35	0.5	0.62	0.62	0.27	53
SprayAndWait	0	0.06	0.09	0.18	0.26	0.47	0.59	0.68	0.68	0.33	25
Average relative promotion ratio(%)											68.25

TABLE 5: The evaluation on message transmission.

Algorithm	The average residual energy	The average delay	The average hops
LRDC	40970.775	8864.43	5.75
DirectDeliver	45288.5	15562.93	1
Epidemic	19691.025	8816.55	6.22
FirstContact	40186	22133.09	28.29
SprayAndWait	42298.025	14085.71	4.52

normalization is as follows: because the higher the average residual energy, it indicates that the performance of routing algorithm is better. This paper uses formula (16) to deal with the data of the average residual. Because the shorter the average delay time of message transmission, it indicates that the performance of routing algorithm is better. Because the lower the average hops of message transmission, the lower the network overhead, it indicates that the performance of routing algorithm is better. This paper uses formula (17) to deal with the data of the average delay and the average hops. This section uses max and min to represent the maximum and minimum values of five kinds of routing data, respectively. For the calculation of total score, we use formula (18) to deal with it. The higher the total score, the better the routing algorithm performance. It can be seen from Table 6 that the LRDC has the highest total score, so the performance of LRDC is the best.

$$X^* = \frac{x - \min}{\max - \min}, \quad (16)$$

$$X = 1 - \frac{x - \min}{\max - \min}, \quad (17)$$

$$\text{Total score} = \text{TNARE} + \text{TNAD} + \text{TNAH}, \quad (18)$$

6.3. *The Effect of Message Lifetime on LRDC Message Delivery Success Rate.* When other main parameters of the simulation environment are set to $\text{initialEnergy} = 50000$, $\text{bufferSize} = 100$ MB, and $\text{transmitSpeed} = 150$ KB/s, the effect of msgTtl on LRDC is tested. The simulation results are as follows:

This section presents a set of experiments comparing the performance of LRDC, DirectDeliver, Epidemic, FirstContact, and SprayAndWait and considering the effect of message lifetime on message delivery success rate.

This paper uses data normalization to process Table 7 and uses formula (16) to deal with the data of the RECMSR (ratio of energy consumption to message delivery success rate). And define the following formula:

$$\text{RECMSR} = \frac{\text{the average success delivery rate}}{5000 - \text{the residual energy}}. \quad (19)$$

As can be seen from Table 7, when TTL is less than 400 min, the impact on each routing algorithm is greater. When TTL is greater than 400 min, the performance of each routing algorithm is gradually stable. Combining with Table 5, it can be seen that the message transmission delay of LRDC and Epidemic is the lowest. Since the message transmission delay of the FirstContact is the highest, the change of TTL also has a great influence on it. The change of TTL also has a great impact on FirstContact, and has little impact on LRDC and Epidemic. The higher the normalized RECMSR, the better the routing algorithm performance. Among them, the normalized RECMSR of LRDC is the highest, so the performance of LRDC is the best.

6.4. *The Effect of Transmission Speed on LRDC Message Delivery Success Rate.* Set other main parameters of simulation environment to $\text{initialEnergy} = 50000$, $\text{bufferSize} = 100$ MB, and $\text{msgTtl} = 900$ min, the effect of transmitSpeed on LRDC was tested. The simulation results are as follows.

This section presents a set of experiments comparing the performance of LRDC, DirectDeliver, Epidemic, FirstContact, and SprayAndWait and considering the influence of transmission speed on message delivery success rate.

It can be seen from Table 8 that when the transmitSpeed is less than 50 kb/s, the impact on LRDC, DirectDeliver, Epidemic, FirstContact, and SprayAndWait is greater than that on DirectDeliver. When the transmitSpeed is greater than 50 kb/s, the performance of each routing algorithm gradually stabilizes. It can be seen from Table 5 that the DirectDeliver has less hops for message delivery. It directly transmits the message to the destination node. Therefore, when the transmitSpeed is limited, the effect on the DirectDeliver is low, but the overall performance of the routing algorithm is poor.

$$\begin{aligned} &\text{Relative promotion ratio} \\ &= \frac{\text{the difference average success delivery rate between LRDC and R}}{\text{the average success delivery rate of R}}. \end{aligned} \quad (20)$$

TABLE 6: The normalized score table.

Algorithm/scoring project	The normalized average residual energy (TNARE)	The normalized average delay (TNAD)	The normalized average hops (TNAH)	Total score
LRDC	0.8313	0.9964	0.8259	2.6536
DirectDeliver	1	0.4934	1	2.4934
Epidemic	0	1	0.8087	1.8087
FirstContact	0.8007	0	0	0.8007
SprayAndWait	0.8832	0.6043	0.8710	2.3585

TABLE 7: The effect of message lifetime on message delivery success rate.

Algorithm/ttl (min)	30	60	120	240	360	600	900	1200	RECMSR	The normalized RECMSR	Average success rate	Relative promotion ratio (%)
LRDC	0.33	0.34	0.49	0.67	0.78	0.86	0.90	0.94	0.000074	1	0.66	0
DirectDeliver	0.03	0.03	0.14	0.18	0.21	0.32	0.35	0.35	0.00004	0.33	0.20	228
Epidemic	0.35	0.38	0.53	0.71	0.82	0.88	0.94	0.94	0.000023	0	0.7	-5
FirstContact	0.12	0.21	0.27	0.32	0.32	0.47	0.5	0.65	0.000036	0.26	0.36	85
SprayAndWait	0.12	0.21	0.35	0.5	0.68	0.77	0.82	0.85	0.00007	0.92	0.54	23
Average relative promotion ratio (%)												82.75

TABLE 8: The effect of transmission speed on message delivery success rate.

Algorithm/transmitSpeed (kb/s)	12.5	25	50	87.5	125	187.5	250	Average success rate	Relative promotion ratio (%)
LRDC	0.441	0.735	0.882	0.912	0.941	0.941	0.941	0.828	0
DirectDeliver	0.324	0.324	0.353	0.353	0.353	0.353	0.353	0.345	140
Epidemic	0.412	0.735	0.882	0.941	0.941	0.941	0.941	0.828	0
FirstContact	0.412	0.588	0.677	0.647	0.618	0.647	0.647	0.605	36.9
SprayAndWait	0.435	0.653	0.753	0.853	0.853	0.853	0.853	0.75	10.4
Average relative promotion ratio (%)									46.83

The higher the average success rate, the better the routing algorithm performance. For the calculation of relative promotion ratio, we use formula (20) to deal with it. R is any of the four routing algorithms of DirectDeliver, Epidemic, FirstContact, and SprayAndWait. Among them, the average success rate of LRDC is the highest, it can be seen that the performance of LRDC is the best, and the average relative promotion ratio is 46.83%.

6.5. The Effect of BufferSize on LRDC Message Delivery Success Rate. Set other main parameters of simulation environment to $initialEnergy = 50000$, $transmitSpeed = 150$ KB/s, and $msgTtl = 900$ min, the effect of bufferSize on LRDC was tested. The simulation results are as follows. This section presents a set of experiments comparing the performance of LRDC, DirectDeliver, Epidemic, FirstContact, and SprayAndWait and considering the effect of bufferSize on message delivery success rate.

It can be seen from Table 9 that bufferSize has a greater effect on LRDC, Epidemic, and SprayAndWait, while it has less effect on Epidemic and FirstContact. The higher the average success rate, the better the routing algorithm perfor-

mance. For the calculation of relative promotion ratio, we use formula (20) to deal with it. R is any of the four routing algorithms of DirectDeliver, Epidemic, FirstContact, and SprayAndWait. Among them, the average success rate of LRDC is the highest, it can be seen that the performance of LRDC is the best, and the average relative promotion ratio is 44.43%.

6.6. The Effect of Initial Node Energy on LRDC Message Delivery Success Rate. Set other main parameters of simulation environment to $bufferSize = 100$ MB, $transmitSpeed = 150$ KB/s, and $msgTtl = 900$ min, the effect of $initialEnergy$ on LRDC was tested. The simulation results are as follows. This section presents a set of experiments comparing the performance of LRDC, DirectDeliver, Epidemic, FirstContact, and SprayAndWait and considering the effect of $initialEnergy$ of nodes on message delivery success rate.

It can be seen from Table 10 that when the $initialEnergy$ is less than 20000, all routing algorithms will be greatly affected; when the $initialEnergy$ is greater than 20000, the performance of each routing algorithm will gradually stabilize. The lower $initialEnergy$ has the greatest effect on the

TABLE 9: The effect of bufferSize on message delivery success rate.

Algorithm/bufferSize (MB)	5	10	20	40	60	80	100	120	Average success rate	Relative promotion ratio (%)
LRDC	0.405	0.482	0.601	0.748	0.856	0.911	0.941	0.941	0.736	0
DirectDeliver	0.323	0.353	0.353	0.353	0.353	0.353	0.353	0.353	0.349	110.9
Epidemic	0.147	0.235	0.322	0.469	0.606	0.794	0.941	0.941	0.557	32.1
FirstContact	0.303	0.529	0.618	0.618	0.618	0.618	0.618	0.618	0.568	29.6
SprayAndWait	0.415	0.504	0.613	0.723	0.783	0.853	0.853	0.853	0.7	5.1
Average relative promotion ratio (%)										44.43

TABLE 10: The effect of initial energy of nodes on message delivery success rate.

Algorithm/initial energy	5000	10000	20000	30000	40000	50000	60000	70000	Average success rate	Relative promotion ratio (%)
LRDC	0.453	0.629	0.824	0.882	0.892	0.912	0.941	0.941	0.809	0
DirectDeliver	0.273	0.353	0.353	0.353	0.353	0.353	0.353	0.353	0.343	135.9
Epidemic	0.165	0.271	0.494	0.682	0.802	0.901	0.941	0.941	0.853	-5.2
FirstContact	0.235	0.411	0.558	0.618	0.618	0.618	0.618	0.618	0.537	50.7
SprayAndWait	0.618	0.725	0.803	0.853	0.853	0.853	0.852	0.853	0.801	1
Average relative promotion ratio (%)										45.6

Epidemic. The average success rate of Epidemic is the highest, and that of LRDC was the second. It can be seen from Table 5 that the average energy consumption of nodes using Epidemic is higher, which is mainly because nodes routing by Epidemic will transmit all messages to all contact nodes, because it will cause a great waste of energy. The higher the average success rate, the better the routing algorithm performance. For the calculation of relative promotion ratio, we use formula (20) to deal with it. R is any of the four routing algorithms of DirectDeliver, Epidemic, FirstContact, and SprayAndWait. Among them, the average success rate of Epidemic is the highest, but it can be seen from Table 5 that the average energy consumption of nodes using Epidemic is higher, and nodes routing by Epidemic will transmit all messages to all contact nodes, so the performance of LRDC is still the best, and the average relative promotion ratio is 45.6.

6.7. Summary of Experiment Results. In the above simulation experiments, we can see that the proposed LRDC algorithm has achieved good experimental results. The reason is that this paper considers the concept of credibility between nodes in the campus opportunistic network, and effectively solves the problem of resource diffusion in the stage of node starvation. The higher credibility between nodes, the more easily the spread resources are accepted and forwarded by each other. It can effectively improve the message delivery rate of the whole network and reduce the redundancy of messages and the delay of message delivery and the energy consumption of nodes.

7. Conclusions

The campus opportunistic networks are that many learner nodes use handheld smart devices and participate in campus

collaborative learning. In the campus opportunistic network, the LRDC algorithm is proposed to solve the stability problem of initialization node community and the diffusion problem of learning resources. After a large number of simulation experiments, the comparison results between the proposed LRDC algorithm and the classical algorithms DirectDeliver, Epidemic, FirstContact, and SprayAndWait are as follows: in aspects of average delay and the average hop number, the performance of LRDC is the best; in aspect of RECMSR, LRDC is the highest; in addition, when the transmission speed is the same, LRDC is the best in terms of message delivery success rate, and the average relative promotion ratio is 46.83%. When the buffer size is same, the average success rate of LRDC is also the highest, and the average relative promotion ratio is 44.43%. This paper solves the initialization and stability problem of node community and the diffusion problem of learning resources.

Data Availability

The CRAWDDAD dataset Cambridge/haggle (v. 2009-05-29) used to support the findings of this study has been deposited in the <https://crawdad.org/cambridge/haggle/20090529> repository ([doi:10.15783/C70011]).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Peng Li, Yuanru Cui, and Qian Liu contributed equally to this work.

Acknowledgments

This work is partly supported by the National Key R&D Program of China under grant no. 2020YFC1523305; the National Natural Science Foundation of China under grant nos. 61877037, 61872228, 61977044, and 62077035; the Key R & D Program of Shaanxi Province under grant nos. 2020GY-221, 2019ZDLSF07-01, and 2020ZDLGY10-05; the Natural Science Basis Research Plan in Shaanxi Province of China under grant nos. 2020JM-302, 2020JM-303, and 2017JM6060; the S&T Plan of Xi'an City of China under grant no. 2019216914GXRC005CG006-GXYD5.1; the Fundamental Research Funds for the Central Universities of China under grant nos. GK201903090 and GK201801004; the Shaanxi Normal University Foundational Education Course Research Center of Ministry of Education of China under grant no. 2019-JCJY009; and the second batch of new engineering research and practice projects of the Ministry of Education of China under grant no. E-RGZN20201045.

References

- [1] D. Colomé Cedeño, A. Palmero Ortega, A. Granda Dihigo, and T. Faife Rodríguez, "The collaborative learning of the analysis and modeling of software with the use of Facebook," in *2020 17th International Conference on Remote Engineering and Virtual Instrumentation (REV)*, pp. 718–728, Athens, GA, USA, 2020.
- [2] C. Liu and K. Wang, "Analysis and modeling of computer-supported collaborative learning system," in *2012 IEEE International Conference on Control Engineering and Communication Technology (ICCECT)*, pp. 1026–1028, Shenyang, Liaoning, China, 2012.
- [3] J. Laru, P. Näykki, and S. Järvelä, "Four stages of research on the educational use of ubiquitous computing," *IEEE Transactions on Learning Technologies*, vol. 8, no. 1, pp. 69–82, 2015.
- [4] M. Cuka, D. Elmazi, M. Ikeda, K. Matsuo, L. Barolli, and M. Takizawa, "Application of fuzzy logic for IoT node elimination and selection in opportunistic networks: performance evaluation of two fuzzy-based systems," *World Wide Web*, vol. 24, no. 3, pp. 929–940, 2021.
- [5] G. Costantino, R. Maiti, F. Martinelli, and P. Santi, "A locality sensitive routing protocol in opportunistic networks with contact profiles," *IEEE Transactions on Mobile Computing*, vol. 19, no. 10, pp. 2392–2408, 2020.
- [6] V. Amin and B. David, "Epidemic routing for partially-connected ad hoc networks," *Handbook of Systemic Autoimmune Diseases*, vol. 6, pp. 1–16, 2000.
- [7] R. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: modeling and analysis of a three-tier architecture for sparse sensor networks," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 215–233, 2003.
- [8] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *2004 ACM 5th International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pp. 187–198, Roppongi, Tokyo, Japan, 2004.
- [9] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the single-copy case," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 63–76, 2008.
- [10] S. Oramas, V. Ostuni, T. Noia, X. Serra, and E. D. Sciascio, "Sound and music recommendation with knowledge graphs," *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no. 2, pp. 1–21, 2017.
- [11] H. Chen, C. Yin, R. Li, W. Rong, Z. Xiong, and B. David, "Enhanced learning resource recommendation based on online learning style model," *Tsinghua Science and Technology*, vol. 25, no. 3, pp. 348–356, 2020.
- [12] Y. Hao and Y. Fan, "Mining and prediction of service cooperation relationship of node initialization starvation problem in service system," *Journal of Tsinghua University*, vol. 59, no. 11, pp. 917–924, 2019.
- [13] H. Yu and J. Li, "A recommended algorithm to solve the node initialization starvation problem of new projects," *Journal of Software*, vol. 26, no. 6, pp. 1395–1408, 2015.
- [14] K. Huang, Y. Fan, and W. Tan, "Recommendation in an evolving service ecosystem based on network prediction," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 3, pp. 906–920, 2014.
- [15] Y. Zhang, T. Lei, and Y. Wang, "A service recommendation algorithm based on modeling of implicit demands," in *2016 IEEE International Conference on Web Services (ICWS)*, pp. 17–24, San Francisco, CA, USA, 2016.
- [16] G. Wang, J. Zhang, and B. Wang, "Sa-dtn: research of dtn routing based on node social activity," *Computer Application Research*, vol. 28, no. 4, pp. 1524–1526, 2011.
- [17] J. Feng, Y. Zhang, and P. Fan, "An adaptive backoff algorithm based on neighbor activity in ad hoc networks," *Journal of System Simulation*, vol. 2008, no. 5, pp. 1348–1352, 2008.
- [18] F. Rao, F. Meng, and Y. Xing, "Overlapping community discovery algorithm based on node importance and similarity," *Computer Engineering*, vol. 44, no. 9, pp. 192–198, 2018.
- [19] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, no. 3, pp. 1–20, 2009.
- [20] U. Brandes, D. Delling, M. Gaertler et al., "On modularity clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 2, pp. 172–188, 2008.
- [21] B. H. Good, Y. D. Montjoye, and A. Clauset, "Performance of modularity maximization in practical contexts," *Physical Review E*, vol. 81, no. 4, pp. 1–19, 2009.
- [22] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, pp. 1–16, 2004.
- [23] S. Li, H. Lou, W. Jiang, and J. Tang, "Detecting community structure via synchronous label propagation," *Neurocomputing*, vol. 151, pp. 1063–1075, 2015.
- [24] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD dataset Cambridge/haggle (v. 2009-05-29)," May 2009, <https://crawdad.org/cambridge-haggle/20090529>, 10.15783/C70011.