IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received November 20, 2018, accepted December 4, 2018, date of publication December 12, 2018,
date of current version January 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2886360

Link Prediction Approach for Opportunistic Networks Based on Recurrent Neural Network

XULIN CAI, JIAN SHU, AND MANAR AL-KALI
School of Software, Nanchang Hangkong University, Nanchang 330063, China

Corresponding author: Jian Shu (shujian@nchu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61762065, Grant 61363015, and Grant 61262020, in part by the Natural Science Foundation of Jiangxi Province under Grant 20171ACB20018, Grant 20171BAB202009, and Grant 20171BBH80022, and in part by the Innovation Foundation for Postgraduate Student of Jiangxi Province under Grant YC2017-S344.

ABSTRACT The target of link prediction is used to estimate the possibility of future links among nodes through known network structures and nodes information. According to the time-varying characteristics of the opportunistic network, the historical information of node pairs has a significant influence on the future connection state. We propose a novel link prediction approach which is based on the recurrent neural network link prediction (RNN-LP) framework. With the help of time series method, we define the vector that is made up of the node information and historical connection information of the node pairs, in which a sequence vector is constructed. Benefiting from RNN in sequence modeling, the time domain characteristics were extracted in the process of the dynamic evolution of the opportunistic network. Hence, the future link prediction becomes significantly better. By utilizing iMote traces Cambridge and MIT reality datasets, experimental results are obtained to reveal that RNN-LP method gives better accuracy and stability than the prediction techniques of the common neighbor, Adamic-Adar, resource allocation, local path, and Katz.

INDEX TERMS Opportunistic network, link prediction, recurrent neural network, historical connection information, vector sequence.

I. INTRODUCTION

Opportunistic network (ON) [1] is a new type of mobile Ad Hoc networks [2] that establishes communication through the movement of network nodes without the need of perfect communication domain between the source and the target node. The data transmission of the ON is realized by the Story-Carry-Forward routing mechanism. According to the time-varying characteristics of the network topology, node mobility and intermittent connectivity, ON has irreplaceable advantages in the application fields of non-fully connected networks. In recent years, ON is widely applied in many fields such as Vehicular Ad Hoc Networks [3], Mobile Data Offloading [4], Information Sharing [5], Mobile Computing [6] and so on.

Recently, link prediction become an important topic of the opportunistic network field. It estimates the possibility of the existing links between couple nodes by known network structure, nodes attributes and network historical information. The excellent link prediction algorithm does not only mine the potential relationships of the nodes in the network but also help to understand the evolution of the network structure, hence, the routing algorithm is supported.

According to the time-varying characteristics of the ON, this study combines the historical information of node pairs which have an impact on its connection state at the next moment. We define the vector that is made up of the node data and historical connection information of the pairs, in which a sequence vector is constructed. As well as define the link prediction window, which is the length of vector sequence. Link prediction method based on RNN is proposed, which extract the time domain characteristics, then, it gives a significantly high prediction performance.

The rest of the paper is organized as follows. Related work are discussed in Section 2. Problems associated with link prediction are identified in Section 3. The data representation is described in Section 4. The prediction model based on RNN is constructed in Section 5. The experimental data and analysis are described in Section 6. Section 7 makes the conclusion.

II. RELATED WORK

Link prediction methods of ON fall into the following categories– predictions based on similarity indices, probabilistic model, mixed frameworks, and machine learning.

A. SIMILARITY INDICES-BASED PREDICTION METHODS

Similarity indices-based prediction methods means that the higher the degree of similarity between two nodes the greater the possibility of a connection between them. It is divided into local information-based similarity indices and path-based similarity indices.

1) LOCAL INFORMATION-BASED SIMILARITY INDICES

Local information-based similarity indices calculate the similarity between node pairs using local information, such as the degree of nodes, common neighbors, etc. There are some common similarity indices, such as common neighbor (CN) index [7], Adamic-Adar (AA) index [8], Resource allocation (RA) [9] and so on. These similarity indices are defined in (1), (2) and (3) respectively:

$$S_{xy}^{CN} = |\Gamma(x) \cap \Gamma(y)| \tag{1}$$

$$S_{xy}^{AA} = \sum\nolimits_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\log k_z} \tag{2}$$

$$S_{xy}^{RA} = \sum\nolimits_{z\in\Gamma(x)\cap\Gamma(y)} \frac{1}{k_z} \tag{3}$$

where $\Gamma(x)$ and $\Gamma(y)$ are the neighbor set of nodes x and y, k_x, k_y, $\mathrm{k}_z$denote the degree of nodes x, y and z respectively. The local information-based similarity indices are suitable for large-scale network applications because of its lower computational complexity. However, it has lower accuracy.

2) PATH-BASED SIMILARITY INDICES

Path-based similarity indices fall into the following categories– local path (LP) index [10], Katz index [11], LHN-II index [12].

The local path similarity index adds the path information with length 3 to the index CN, as it is defined in (4):

$$S_{xy}^{LP} = \boldsymbol{A}^2 + \alpha \cdot \boldsymbol{A}^3 \tag{4}$$

where α is an adjust parameter and $\boldsymbol{A}$ is an adjacency matrix of the network

The Katz index considers the n-hop paths, and its computational complexity increases in sequence, as it is defined in (5):

$$\mathrm{S}_{\mathrm{xy}}^{katz} = \sum\nolimits_{t=1}^{\infty} \alpha^l \cdot \left|paths_{x,y}^{\langle l\rangle}\right| \tag{5}$$

where $\alpha > 0$ denotes an adjust parameter that controls the weight of paths, $\left|paths\ _{x,y}^{\langle l\rangle}\right|$ indicates the set of paths with length l $(l = 1, 2, \ldots, n)$ between x and y.

With the help of the regular equivalence, the LHN-II index is proposed in [12]. If the neighbors of$x$are similar to$y$, then$x$and$y$are similar. Hence, the similarity of the nodes is transitive. LHN-II index is defined in (6):

$$S_{xy}^{LHN-II} = 2M\lambda_1 \boldsymbol{D}^{-1}\left(\boldsymbol{I} - \frac{\phi}{\lambda_1}\boldsymbol{A}\right)^{-1}\boldsymbol{D}^{-1} \tag{6}$$

where $\boldsymbol{D}$ is the degree matrix, ϕ is an adjust parameter $0 < \phi < 1$, λ_1 is the largest eigenvalue of the adjacent matrix $\boldsymbol{A}$, M is the total number of network edges.

B. PROBABILISTIC MODEL-BASED PREDICTION METHODS

According to the network structure, the probabilistic model-based prediction methods calculate the probability that the link exists between two nodes. Das *et al.* [13] proposes a Markov prediction model based on time-varying maps of the network, which considers the impact of multiple time scale on the correlation of predicting time analysis, and estimates the probability of communication between node pairs and the time of future connection by simulating the speed of the start and the end time respectively. Sharma and Singh [14] puts forward the concept of multi-layer network, the links of social network are divided into several layers with respect to different characteristics. One layer serves as target layer and others serve as the prediction layers. The relationship among different layers is extracted from the network structure and the link relation between two nodes. The links are predicted by constructing a probabilistic model. These approaches need to select the appropriate model using the actual characteristics of the target network and the prior probability, hence, the range of the application is limited.

C. MIXED FRAMEWORKS-BASED PREDICTION METHODS

The mixed frameworks-based prediction methods split the network using the different cases of the network, then many verities are utilized to predict the link with the help of the characteristics of each part. Huang *et al.* [15] proposes a mixed link prediction idea that combines the time series model and some prediction approaches with respect to the static graph. It shows better prediction performance than other prediction schemes based on the static graph. Li *et al.* [16] applies the periodic pattern mining based decision tree based and AA index link prediction methods for frequent and periodic frequent and non-periodic and non-frequent to the nodes contact respectively. The efficiency and accuracy of the link prediction are improved by selecting the optimal threshold, jitter tolerance ratio and the time slice length. This technique predicts links effectively by decomposing the network according to the network's characteristic, however this link prediction is relatively poor.

D. MACHINE LEARNING-BASED PREDICTION METHODS

The essence of the machine learning-based prediction technique is to transform the link prediction issue into a classification problem. Hasan [17] discovers that support vector machine (SVM) has a good performance in link prediction by comparing with several supervised learning algorithms. Yang *et al.* [18] proposes an algorithm for link prediction using clustering and decision tree. The two types of objects in the network are served as feature vectors and cluster them. Then, three heuristic rules are applied to construct the decision tree, and finally the link prediction is performed. The authors also define the concept of potential link nodes and introduces the number of layers, which can reduce the running time and improve the accuracy. At present, most of the link prediction algorithms based on machine learning are shallow learning algorithms. These algorithms are not applicable to large-scale or frequently-changing networks because their feature extraction capabilities are obviously inadequate.

The above-mentioned prediction methods show a good prediction effect on a static network or a social network whose topology change slowly. However, these approaches ignore the relationship between network topology and time information, so they are not useful to the ON whose network topology change over time. Therefore, according to the time-varying characteristics of the ON, this paper combines the historical information of node pairs which have an impact on its connection state at the next moment. We define the vector that is made up of the node data and historical connection information of the pairs, in which a vector sequence is constructed. As well as define the link prediction window, which is the length of vector sequence. We utilize the advantage of the RNN-LP model in sequence modeling to extract the time domain characteristics in the process of dynamic evolution of the ON, thus to realize the future link prediction better.

III. PROBLEM DESCRIPTION

ON is a typical, dynamic network that its topology changes frequently over time. The network topology is divided into a series of network snapshots as $G = (G_1, G_2, G_3, \ldots G_{t-1}, G_t)$ using the time series theory to depict the dynamics of ON, where $G_t = (V_t, E_t)$, V_t and E_t denote the network topology map, the set of nodes and the set of edges at time t.

The link prediction of ON is a process of mining the potential relationships between couple of nodes in the network. The essence is to detect which edges disappear and which edges increase in the future for ON. Therefore, this paper transforms the link prediction issue into whether the link of single node pairs has a connection or not at the next moment, thus realizes the link prediction of ON.

The link prediction of the node pairs in the ON is to predict connection state between them in the $(T+1)^{th}$ network snapshot according to the node data and the historical connection information of the node pairs in the previous t network snapshots. Figure 1 shows the evolution of the link in the ON.

IV. DATA REPRESENTATION

The sample data in the ON data sets exists in chronological order. The data processing is divided into two steps. The first step is to transform the sample data. The historical information in time dimension is considered in the second step, and define the vector that is made up of the node data and historical connection information of the pairs, in which a vector sequence is constructed. We also define the link prediction window, which is the length of vector sequence.

The real data set in iMote Traces Cambridge (ITC) [19] is selected from a simulation experiment of student trajectory visualization at Cambridge University campus. The data format is shown in figure 2.

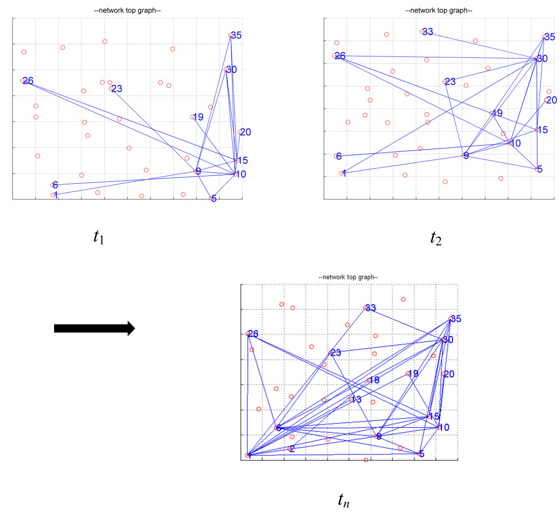

FIGURE 1. The evolution of link in the ON.

Device ID	Device ID	Start Time	End Time

FIGURE 2. Communication data format.

where Device ID denotes the device number, Start Time and End Time denote the time that two devices are connected and disconnected respectively.

A. DATA CONVERSION

As shown in figure 2, the contact time (Start Time, End Time) of the data set in the ON exists in the form of a timestamp. Through analysis, it has no regularity, but the communication timestamps of the node pairs in the data set are arranged in ascending order. Therefore, based on the time series method, this paper splits the data set to obtain a series of network snapshots in time. As shown in figure 1, the start time and end time in the data set are divided into corresponding network snapshots. In order to find the hidden features of the links between any two nodes in the time domain, the form of the Start Time Timestamp and the End Time Timestamp are converted into common time (e.g. a certain time in a certain day).

If there is an intersection between the time interval from End Time to Start Time (contact time) and the time period corresponding to the network snapshot, there is a link between node pairs, otherwise, it is not exist. Then the intersection has the following three cases. As shown in figure 3, a time slice $[T_S, T_E]$ with a selected length T is taken as an example, where T_S and T_E are the start and end time of the time slice, T_0 and T_1 are the moments when the link for node pairs (x, y) is generated and disconnected.

(1). If $T_0 < T_S$ or $T_1 > T_E$, it means that the contact time between x and y partly belongs to the time period corresponding to the network snapshot, then $T_0 = T_S$ or $T_1 = T_E$.

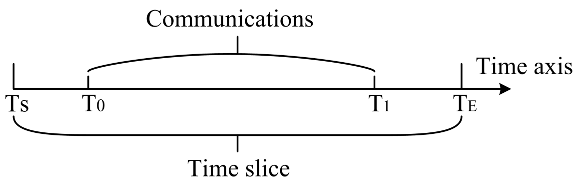

FIGURE 3. The segment of contact time in the time axis.

(2). If $T_0 > T_S$ and $T_1 < T_E$, it means that the contact time betweenx and y is a subset of the time period corresponding to the network snapshot.

(3). If $T_0 < T_S$ and $T_1 > T_E$, it means that the time period corresponding to the network snapshot is a subset of the contact time between x and y, then $T_0 = T_S$ and $T_1 = T_E$.

B. DATA REPRESENTATION

The sample data after the data conversion cannot be used as the input of the prediction model. Thus, the Device ID, Start Time, End Time, the frequent of contact f and the node-to-connection state S in the sample data are mapped into a vector, as shown in (7):

$$\vec{V}_{<i,j>} = \{N_i, N_j, T_s, T_e, f, S\} \tag{7}$$

where Ni denotes the number of the node i, Nj denotes the number of the node j, Ts, Te denote the time at which the pairs $\langle Ni, Nj\rangle$ are connected and disconnected respectively, f denotes the frequent of contact in the time slice T, S denotes the state at which the pairs is connected. The state is represented by 0 or 1, 0 means that $\langle Ni, Nj\rangle$ is not connected, and 1 means that $\langle Ni, Nj\rangle$generate a link.

Moreover, we define the link prediction window as the input the RNN-LP model, and the vector $\vec{V}$ is consisted of the vector sequenceSeq, the length of vector sequence is the link prediction window size. The vector sequence is defined in (8).

$$\text{Seq}= [\text{V}_1, V_2, V_3, \ldots, V_n] \tag{8}$$

The specific length of the vector sequence is verified by experiments.

V. THE CONSTRUCTION OF RECURRENT NEURAL NETWORKS PREDICTION MODEL

The vector sequence Seq is made up of the node data and historical connection information of the pairs at different moment, and it is used as the input of the RNN-LP model, the connection state of node pairs is the output of the RNN-LP model. Benefiting from the RNN in sequence modeling to capture the hidden features of the vector sequence data in the time domain, hence it predict the connection state of node pairs at the next moment.

The main purpose of constructing RNN-LP model is to learn the correlation between nodes in the connection state and the historical information in the time dimension, and is to extract the inherent law of node pairs which change frequently over time in opportunistic communication.

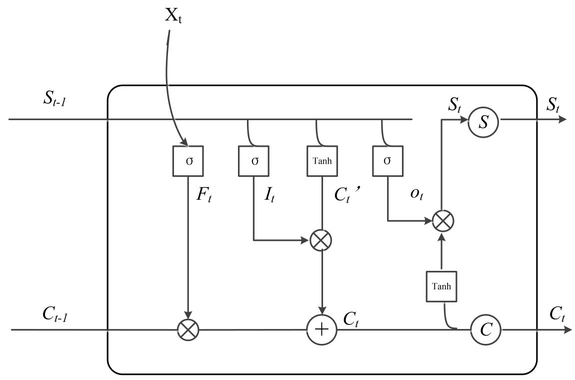

FIGURE 4. Long Short-Term Memory Network.

The RNN-LP model is constructed as follow: network structure, hyper-parameters setting, training algorithm and parameters, and performance optimization methods.

A. NETWORK STRUCTURE

The RNN-LP aims to extract the dependence between the node pairs' historical information and its connection state in time dimension, thus to obtain the hidden key features. However, the traditional RNN has the problem of the gradient vanishing and long-term dependence. Such as, if the length of the input vector sequence is very long, the traditional RNN may discard the previous part of the historical information, and lose the corresponding features, thus lead to the poor prediction performance. This study selects the Long Short-Term Memory Network (LSTM) [20] that is widely used in the RNN, which solves the problem of vanishing gradient and long-term dependence using the "gate" mechanism. A typical LSTM is shown in figure 4.

where Xt, Ft, It, Ct', Ot, Ct and St represent the input, the forget gate, the input gate, the candidate gate, the output gate, the memory cell and the output at time t respectively, $Ct - 1$ and $St - 1$ denote the memory cell and the output at time $t-1$ respectively, lastly σ and $Tanh$denote the Activation function of $Sigmoid$ and $Tanh$.

The design of our link prediction model needs to determine the network layer number, hidden layer activation function, output layer activation function and so on. The network structure of RNN-LP model based on LSTM is shown in figure 5. The first layer is the input layer, the middle three layers are the LSTM hidden layer, the activation function of the hidden layer is $Sigmoid$, the last layer is the output layer. We transforms the link prediction issue into a binary classification problem, therefore, the output layer is classified by Logistic Regression which is applicable to binary classification problem, thus the prediction results are obtained.

B. HYPER-PARAMETERS SETTING

The main problem is the hyper-parameters setting in constructing the RNN-LP model, which includes the initialization weight, the length of the input sequence and so on.

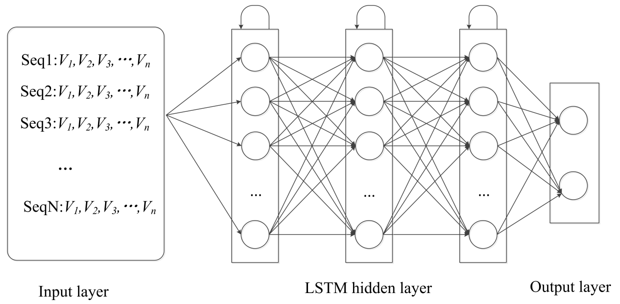

FIGURE 5. Network structure.

1) THE INITIALUZATION WEIGHT

The initialization weight is very important in deep neural network. If the weight is too small, it may easily lead to the gradient vanishing, and then cause the model iterate slowly. On the contrary, it may easily lead to the gradient exploding, and then cause the model converge to a poor local optimal value. This paper initializes the weight based on the orthogonal initialization [21] which initializes the transition matrix to an orthogonal matrix to avoid the gradient vanishing or exploding at the beginning. Thus, it ensures that the total number of training iteration required to reach convergence is less than the depth (network layers).

2) THE LENGTH OF THE INPUT SEQUENCE

The RNN is used to process the sequence data. The length of the input sequence is defined that how many historical data which the nodes are connected is to predict the connection state of the nodes at the next moment. Different length of the sequence have different temporal characteristics. If the length is very long, it leads to too many temporal characteristics, so that the training complexity of the model increases, and it is easily to over-fitting. On the contrary, the model cannot fully learn the correlation of before and after the time step. The length of the input sequence is determined at the beginning according to the empirical value. In this paper, we determine the length of the input sequence of the RNN-LP model using the comparison experiment in the second experiment.

C. TRAINING ALGORITHM AND PARAMETERS

The purpose of the training process is to select the optimal parameters for the network. The common algorithms are as follow: SGD, Adam, Adadelta, Adagrad, RMSprop and so on. We select the adaptive moment estimation (Adam) [22]. Several factors are taken into the account such as the batch training size (batch size), iteration number (epochs), initial learning rate (η), momentum parameter (beta1), momentum parameter (beta2) and so on.

We select the common value 64 recommended as the batch size. The epochs are determined according to the experimental results (loss function curve). According to [22], the value of beta1 and beta2 are equal to 0.9 and 0.999 respectively.

The Adam algorithm uses the first moment estimation and the second moment estimation of the gradient to adjust dynamically the adaptive learning rate of each parameter. The advantages of Adam is that the each iteration learning rate has a certain range after the offset correction, so as to realize the parameters more stable.

TABLE 1. Information of datasets.

Data Sets	Device	Mobile Nodes	Duration (days)	Network Type	Sampling Interval(s)
ITC	iMote	50	12	Bluetooth	10

D. THE OPTIMIZATION OF MODEL PERFORMANCE

The essence of the deep learning is to learn the distribution characteristics of the data. If the training data and the test data have the same distribution in the training process, the generalization ability of the model is greatly enhanced. However, the generalization ability of the model is greatly decreased. If the distribution of the training data is exactly the same, the training speed of the network is significantly improved. While, the training speed is notability decreased. Therefore, we selects the Layer Normalization (LN) [23], which calculates the mean and variance of the all neurons within each layer, so that the same-level neurons inputs using LN have the same mean and variance. In a LN RNN, the LN makes the training and testing preform the same calculations and also estimates the normalized statistical information at each time step, then to be applied to the RNN-LP. LN improves the training speed and generalization ability of the RNN-LP model.

VI. EXPERIMENTS AND ANALYSIS

For the sake of providing actual condition results, we select iMote Traces Cambridge (ITC) as a data set for the experiments. The Area under the Receiver Operating Characteristic Curve (AUC), Accuracy and Precision of the Receiver Operating Characteristic Curve (ROC) are adopted as evaluation indices. Moreover, this paper verifies the rationality and effectiveness of the proposed RNN-LP model by designing different comparison experiments.

A. EXPERIMENTAL DATASETS

The ITC dataset records a 12-day visual trace of students on the Cambridge University campus. The experiment collects the movement trajectories and interconnection using 50 mobile iMote devices. The dataset is short duration and intensive data. The information about the ITC data set is shown in table 1

B. EXPERIMENTAL RESULTS AND ANALYSIS

This experiment is mainly divided into two steps. The first step is to determine an optimal RNN-LP by comparing its performance under different realistic parameters. In the second step, the effectiveness and rationality of the model are verified by comparing the optimal RNN-LP with the traditional

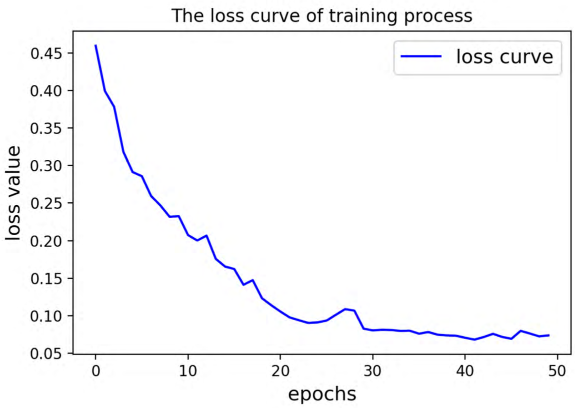

FIGURE 6. The loss value curve of training process.

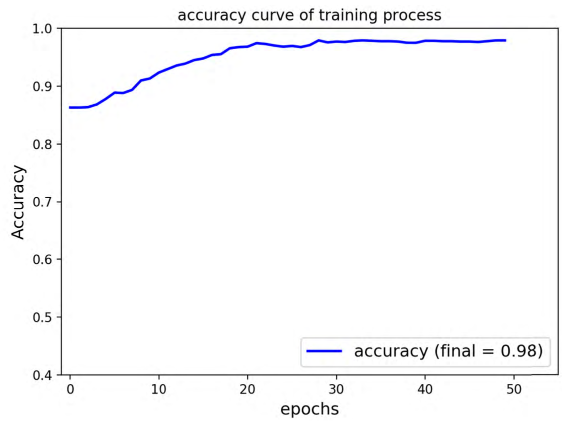

FIGURE 7. The accuracy curve of training process.

similarity indices. In all the experiments, 80% of training and 20% of testing are used in the ITC data set.

1) PERFORMANCE OF RNN-LP MODEL UNDER DIFFERENT PARAMETERS

The experiment in this section mainly verifies the influence of different parameters on our approach. By verifying the influence of the number of iterations in training and different input sequence lengths on the performance of RNN-LP, the prediction accuracy and precision of the model is obtained under the optimal iterations and sequence length.

a: THE EFFECT OF THE ITERATIONS IN TRAINING ON THE PERFORMANCE OF THE RNN-LP

The appropriate iterations training has a certain impact on the generalization ability of the proposed model. If the iterations of training is too large, the training time of the RNN-LP model becomes longer and over-fitting occurs. However, it has an under-fitting phenomenon, which leads to the prediction accuracy to be lower. Hence, the experiments in this study set different iterations of 10, 20, 30, 40, and 50 times respectively. By comparing the influence of different iterations on RNN-LP to determine the appropriate number of iterations, and finally improve its prediction accuracy. The objective loss function (loss value) and the accuracy (Accuracy) curve in RNN-LP training are shown in figure 6 and 7.

Through experiments, we found that the loss value of RNN-LP gradually decreases and the accuracy rate gradually increases. The final loss value and accuracy tend to be in a stable range. Figures 6 and 7 show that when the iterations of training is between 0 and 30, the loss value of RNN-LP has reduced and its accuracy has improved. This phenomenon indicates that the RNN-LP model is in an under-fitting state at the beginning of training. When the number of iterations increase to about 40, its loss value has roughly in a stable state and its accuracy tends to be fixed. When the number of iterations is between 40 and 50, RNN-LP loss value begins to fluctuate up and down and its accuracy of the model is basically the same as the number of iterations at 40. This shows that RNN-LP has converged. Therefore, considering its performance and the training time, the optimal iterations of RNN-LP is 40.

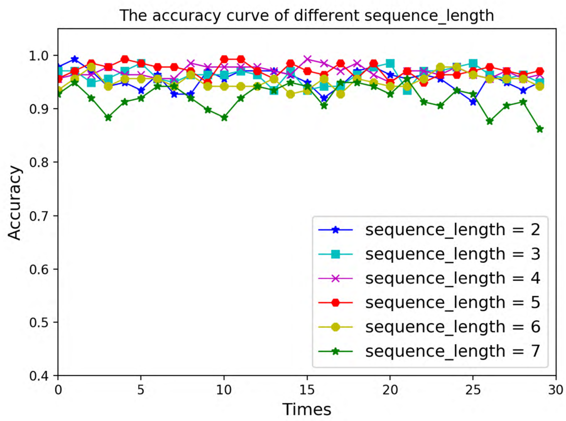

FIGURE 8. The accuracy of different sequence lengths.

b: THE EFFECT OF THE INPUT SEQUENCE LENGTH ON THE PERFORMANCE OF THE RNN-LP

The different length of the sequence have different temporal characteristics. If the length is very long, it leads to too many temporal characteristics, so that the training complexity of the model increases, and it is easily to over-fitting. Moreover, it cannot fully learn the correlation of before and after the time step. In this paper, the length of the sequence is set to 2, 3, 4, 5, 6, and 7, respectively. Accuracy and precision of the prediction model obtained by 30 random samplings are shown in figures 8 and 9, respectively.

We saw from figure 8 and figure 9 that the length of input sequence has certain influence on the performance of our model. Comparing the accuracy and precision of the different length input sequence, it is found that the temporal characteristics are extracted well when the input sequence length of RNN-LP model is 5. Therefore, it is optimal input sequence length.

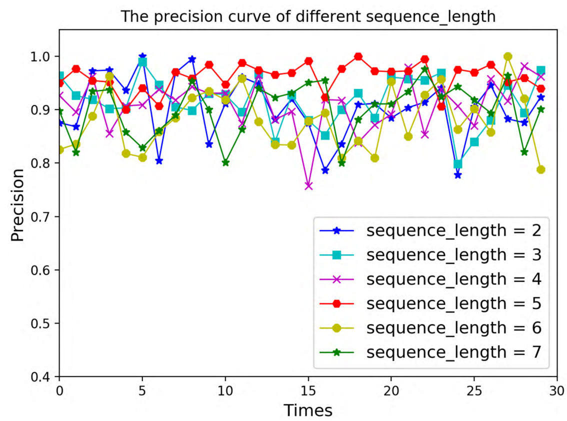

FIGURE 9. **The precision of different sequence lengths.**

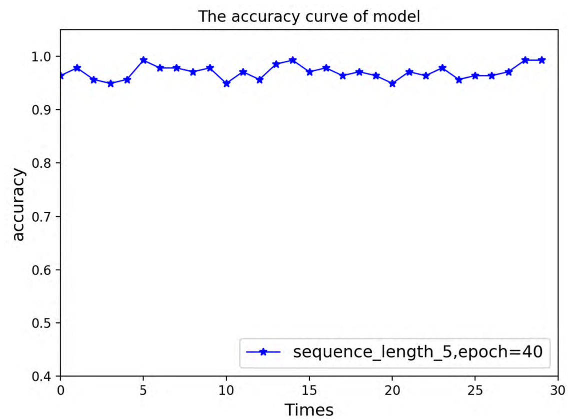

FIGURE 10. **The accuracy of the optimal model.**

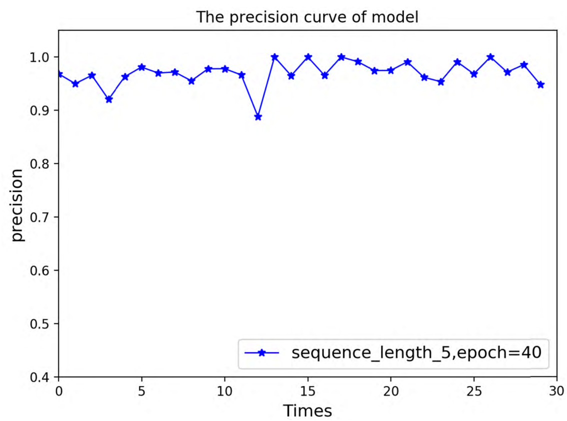

FIGURE 11. **The precision of the optimal model.**

c: DETERMINATION OF THE OPTIMAL MODEL

The optimal model is determined based on the optimal training iterations and the optimal input sequence length which are determined by the previous experiments. The accuracy and precision of the RNN-LP obtained by 30 random sampling are shown in figures 10 and 11.

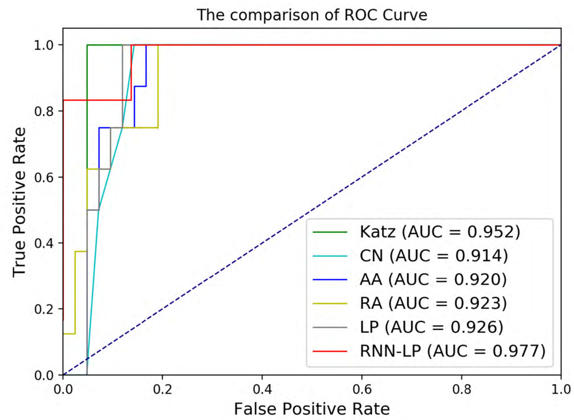

FIGURE 12. **The comparison experiment with a sample size of 50 in ITC dataset.**

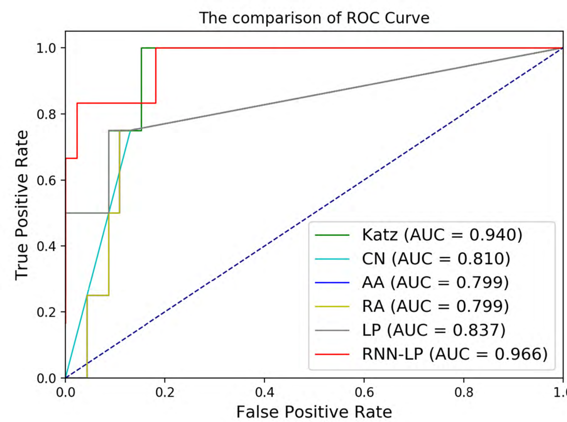

FIGURE 13. **The comparison experiment with a sample size of 50 in MIT dataset.**

Through the analysis of the above experimental results, the accuracy and precision of the RNN-LP have exceeded 90%. This phenomenon indicates that our method is feasible under the optimal training iterations and the optimal input sequence length.

2) COMPARISON OF DIFFERENT PREDICTION METHODS

In current link prediction studies, the most widely used approaches for link prediction is based on the similarity indices. In our work, the effectiveness and rationality of RNN-LP model are verified by comparison experiments based on the similarity indices of CN, AA, RA, LP, Katz.

These comparison experiments in the ITC and MIT Reality (the dataset is sparse data) datasets, when the iterations of training is 40 and the length of the input sequence is 5, the dimension of the training sample data which is randomly extract from the all sample from 50 to 200 (the step length is 50). In this paper, four rounds tests are carried out, and AUC is adopted as the evaluation index. The experimental results are shown in figures 12 to 19.

According to the above four-round comparison experiments, the predicted performance of each similarity index

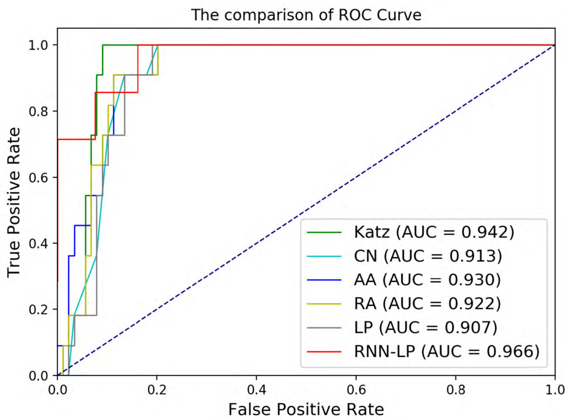

FIGURE 14. The comparison experiment with a sample size of 100 in ITC dataset.

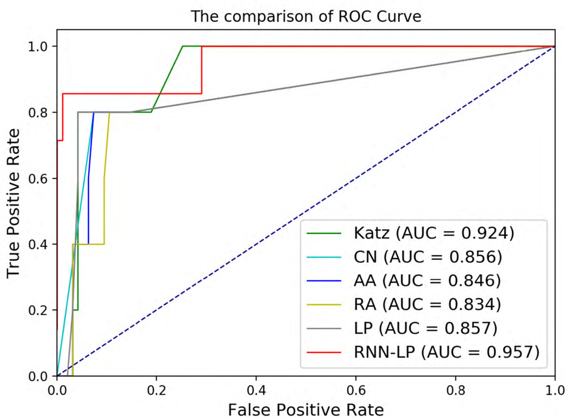

FIGURE 15. The comparison experiment with a sample size of 100 in MIT dataset.

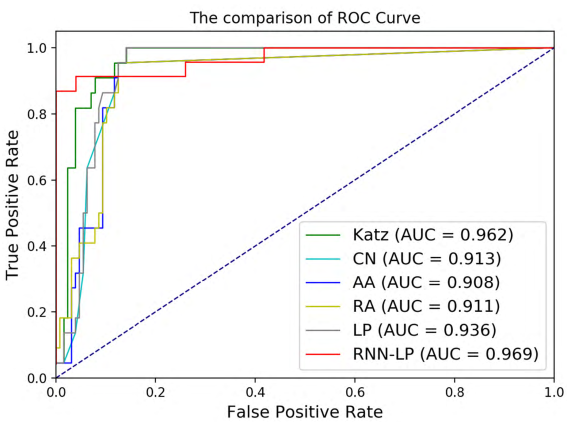

FIGURE 16. The comparison experiment with a sample size of 150 in ITC dataset.

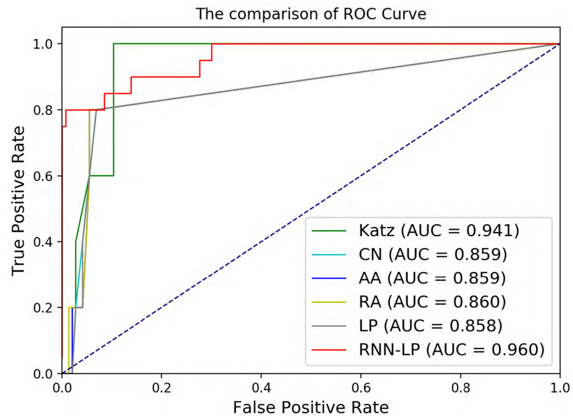

FIGURE 17. The comparison experiment with a sample size of 150 in MIT dataset.

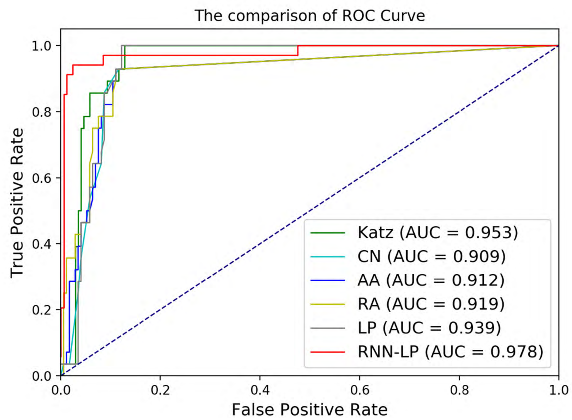

FIGURE 18. The comparison experiment with a sample size of 200 in ITC dataset.

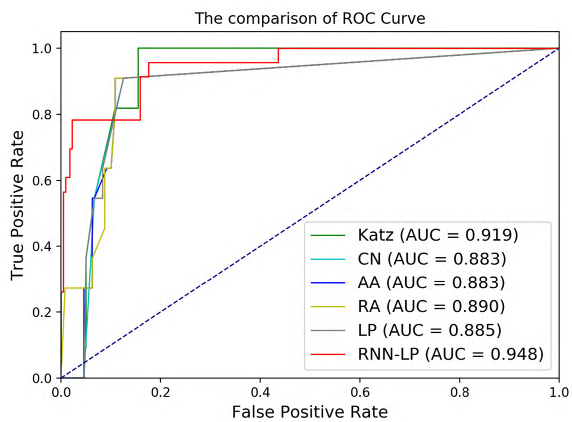

FIGURE 19. The comparison experiment with a sample size of 200 in MIT dataset.

is different under different sample sizes. Comparing with other traditional similarity indices, the similarity index Katz with global information is better than others, but its stability and accuracy is not good compared with the RNN-LP. These experimental results show that RNN-LP has the best stability and accuracy, which indicates that our proposed method can predict the link of the ON effectively.

Comprehensive analysis of the above experimental results shows that our proposed model have significantly high prediction performance in these datasets with different density.

Hence, our model is effective and reasonable in the link prediction of the ON.

VII. CONCLUSION

This paper elaborates on the characteristics of the ON and the significance of its influence on link prediction. For the reason, we consider the historical information of node pairs which have an impact on its connection state at the next moment. We define the vector that is made up of the node data and historical connection information of the pairs, in which a vector sequence is constructed. We also define the link prediction window, which is the length of vector sequence. Then, benefiting from the RNN framework, we propose RNN-LP method to extract the time domain characteristics in the dynamic evolution process of the ON. Experiments show that the RNN-LP model achieves better link prediction in the ITC and MIT reality datasets, moreover, our proposed method provides better accuracy and stability compared with these similarity indices of CN, AA, RA, LP and Katz.

REFERENCES

[1] P. Jadhav and R. Satao, "A survey on opportunistic routing protocols for wireless sensor networks," *Procedia Comput. Sci.*, vol. 79, no. 1, pp. 603–609, 2016.
[2] H. Mansouri, N. Badache, M. Aliouat, and A.-S. K. Pathan, "Checkpointing distributed application running on mobile ad hoc networks," *Int. J. High Perform. Comput. Netw.*, vol. 11, no. 2, pp. 95–107, Jan. 2018.
[3] A. Prado, S. Ruj, M. Stojmenovic, and A. Nayak, "Applying transmission-coverage algorithms for secure geocasting in VANETs," *Int. J. Comput. Sci. Eng.*, vol. 16, no. 1, pp. 17–26, Nov. 2018.
[4] B. Han, P. Hui, V. S. A. Kumar, M. V. Marathe, J. Shao, and A. Srinivasan, "Mobile data offloading through opportunistic communications and social participation," *IEEE Trans. Mobile Comput.*, vol. 11, no. 5, pp. 821–834, May 2012.
[5] S. Jung, U. Lee, A. Chang, D.-K. Cho, and M. Gerla, "BlueTorrent: Cooperative content sharing for Bluetooth users," *Pervasive Mobile Comput.*, vol. 3, no. 6, pp. 609–634, Mar. 2007.
[6] Y. Li and W. Wang, "Can mobile cloudlets support mobile applications?" in *Proc. IEEE INFOCOM*, Toronto, ON, Canada, Apr. 2014, pp. 1060–1068.
[7] M. Zhou, R. Liu, X. Zhao, and Q. Zhu, "Link prediction algorithm based on local centrality of common neighbor nodes using multi-attribute ranking," in *Proc. 12th Int. Conf. Comput. Sci. Educ.*, Houston, TX, USA, Oct. 2017, pp. 506–511.
[8] B. Moradabadi and M. R. Meybodi, "Link prediction in weighted social networks using learning automata," *Eng. Appl. Artif. Intell.*, vol. 70, no. 1, pp. 16–24, Apr. 2018.
[9] S. Liu, X. Ji, C. Liu, and Y. Bai, "Extended resource allocation index for link prediction of complex network," *Phys. A, Stat. Mech. Appl.*, vol. 479, no. 1, pp. 174–183, Aug. 2017.
[10] P. P. Pei, B. Liu, and L. C. Jiao, "Link prediction in complex networks based on an information allocation index," *Phys. A, Stat. Mech. Appl.*, vol. 470, pp. 1–11, Mar. 2017.
[11] B. Moradabadi and M. R. Meybodi, "Link prediction in stochastic social networks: Learning automata approach," *J. Comput. Sci.*, vol. 24, no. 1, pp. 313–328, Jan. 2018.
[12] E. A. Leicht, P. Holme, and M. E. J. Newman, "Vertex similarity in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 73, no. 2, pp. 026026–026120, Feb. 2006.
[13] S. Das and S. K. Das, "A probabilistic link prediction model in time-varying social networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Paris, France, May 2017, pp. 1–6.
[14] S. Sharma and A. Singh, "An efficient method for link prediction in complex multiplex networks," *Comput. Social Netw.*, vol. 3, no. 1, pp. 453–459, Feb. 2016.
[15] D. Huang, S. Zhang, and Z. Chen, "An optimal stopping decision method for routing in opportunistic networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Shanghai, China, Apr. 2013, pp. 2074–2079.
[16] Y. Li and S. Zhang, "Combo-pre: A combination link prediction method in opportunistic networks," in *Proc. 24th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Las Vegas, NV, USA, Oct. 2015, pp. 1–6.
[17] Hasan, A. Mohammad, "Link prediction using supervised learning," in *Proc. SDM Workshop Link Anal. Counterterrorism Secur.*, vol. 30, no. 9, Feb. 2006, pp. 798–805.
[18] N. Yang, T. Peng, and L. Liu, "Link prediction method based on clustering and decision tree," *J. Comput. Res. Develop.*, vol. 54, no. 8, pp. 1795–1803, Aug. 2017.
[19] F. Takens, "Detecting strange attractors in turbulence," in *Dynamical Systems and Turbulence, Warwick 1980* (Lecture Notes in Mathematics), vol. 898, Berlin, Germany: Springer, 1981, pp. 366–381.
[20] A. Graves, "Long short-term memory," in *Supervised Sequence Labelling With Recurrent Neural Networks*. Berlin, Germany: Springer, 2012, pp. 1735–1780.
[21] A. M. Saxe, J. L. McClelland, and S. Ganguli. (2013). "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks." [Online]. Available: https://arxiv.org/abs/1312.6120
[22] D. P. Kingma and J. Ba. (2015). "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980
[23] J. L. Ba, J. R. Kiros, and G. E. Hinton. (2016). "Layer normalization." [Online]. Available: https://arxiv.org/abs/1607.06450

XULIN CAI was born in Jiujiang, Jiangxi, in 1993. He is currently pursuing the master's degree with Nanchang Hangkong University. His main research interest is opportunity networks.

JIAN SHU was born in 1964. He received the M.Sc. degree in computer networks from Northwestern Polytechnical University. He is a Professor with Nanchang Hangkong University. He is a Senior Member of CCF. His main research interests include wireless sensor networks, embedded systems, and software engineering.

MANAR AL-KALI received the M.S. and Ph.D. degrees from the Department of Telecommunication and Information Engineering, Huazhong University of Science and Technology, China, in 2011 and 2015, respectively, majoring in communication and information systems. He is currently a Researcher and a Lecturer with the School of Software, Nanchang Hangkong University. His research interests include wireless sensor networks, cognitive radio, cooperative communications, MIMO OFDM, and information theory.