ORIGINAL RESEARCH ARTICLE UHD JOURNAL OF SCIENCE AND TECHNOLOGY

Link Prediction in Dynamic Networks Based on the Selection of Similarity Criteria and Machine Learning

Karwan Mohammed HamaKarim

Department of Information Technology, University of Human Development, Kurdistan Regional Government, Iraq

ABSTRACT

The study's findings showed that link prediction utilizing the similarity learning model in dynamic networks (LSDN) performed better than other learning techniques including neural network learning and decision tree learning in terms of the three criteria of accuracy, coverage, and efficiency., Compared to the random forest approach, the LSDN learning algorithm's link prediction accuracy increased from 97% to 99%. The proposed method's use of oversampling, which improved link prediction accuracy, was the cause of the improvement in area under the curve (AUC). To bring the ratio of the classes closer together, the suggested strategy attempted to produce more samples from the minority class. In addition, similarity criteria were chosen utilizing feature selection techniques based on correlation that had a strong link with classes. This technique decreased over-fitting and improved the suggested method's test data generalizability. Based on the three criteria (accuracy, coverage, and efficiency), the research's findings demonstrated that link prediction utilizing the similarity LSDN outperformed other learning techniques including neural network learning and decision tree learning. Compared to the random forest algorithm, the LSDN algorithm's link prediction accuracy increased from 97% to 99%. The oversampling in the suggested strategy, which increased link prediction accuracy, is what caused the increase in AUC. To bring the ratio of the classes closer together, the suggested strategy attempted to produce more samples from the minority class. In addition, similarity criteria were chosen utilizing feature selection techniques based on correlation that had a strong link with classes. This technique decreased over-fitting and improved the suggested method's test data generalizability.

Index Terms: Dynamic Social Network, Link Prediction, Machine Learning Algorithms, Similarity Learning Model in Dynamic Networks, Neural Network

1. INTRODUCTION

A group of social players make up a social structure known as a social network. The edges of a network are the interactions, collaborations, or influences between things, and the nodes themselves are individuals or entities found in social contexts. These communities typically develop as a result of shared interests within a strong group. People's relationships are constantly evolving; therefore, as time goes on, new edges and nodes are added to the network while maybe removing some of the older ones. Social networks are so frequently intricate and dynamic [1]. In many applications, online social networks are ambiguous and unpredictable, and those network structures and parameters vary over time. The majority of prior link prediction approaches are based on static network representations. Therefore, it is constrained to solve actual social network problems using deterministic social network models with fixed values for linkages. In other words, when the online social network behaves arbitrarily, link prediction

Access this article online

DOI:10.21928/uhdjst.v7n2y2023.pp32-39	**E-ISSN:** 2521-4217 **P-ISSN:** 2521-4209

Corresponding author's e-mail: Karwan Mohammed HamaKarim, Department of Information Technology, University of Human Development, Kurdistan Regional Government, Iraq. E-mail: karwan.mohammad@uhd.edu.iq

Received: 04-01-2023 Accepted: 17-06-2023 Published: 03-09-2023

approaches based on static graph representation fail [2]. To better understand network evolution and the connections between topologies and functions, link prediction in dynamic networks makes predictions about the network's future structure based on previous data. For instance, we can foresee which linkages will be made in the near future in online social networks. This means that based on their past activities, we can infer who the target user is most likely to be friendships with or even with a specific person. In addition, it can be applied to research on protein-protein interactions, the spread of illness, and many other aspects of evolution [3]. Links that are likely to arise in the future can be predicted by examining the network's structure and extracting its properties at various time intervals. On the basis of time labels, various machine learning models have been employed to forecast missing links in the network [4]. The feature vector can be constructed by extracting similarity-based features from connected and unconnected node pairs based on the network topological structure at various time intervals, and by doing so, the training data produces a machine learning model for identifying links that could be generated in the future [5]. Links in dynamic social networks have been predicted using supervised machine learning methods. The findings demonstrate that whereas these models typically exhibit acceptable accuracy in educational data, their accuracy has declined in experimental data. These models struggle with over-fitting in link prediction because of the abundance of features. In other words, the designed machine learning model performs well on educational data but less well on experimental data [6]. In homogeneous networks, there is also the issue of an unbalanced data set. Unbalanced data have been used to describe a set of data where fewer samples are used to represent one class than other samples used to represent other classes. When a class, which is typically an absolute or minority class, is underrepresented in the data set — in other words, when the number of inaccurate observations in a class outweighs the number of correct observations — this condition in time classification becomes problematic. Therefore, the performance of link prediction is significantly impacted by the class imbalance or incorrect item labeling [7]. To lower the dimensions of input structures (feature reduction) in machine learning algorithms that aim to improve the accuracy of link prediction in dynamic social networks, a new method based on feature selection algorithms is provided in this paper.

2. RELATED WORKS

Common neighborhood (CN) [8] and resource allocation index (RAI) [9] are two similarity indicators that are frequently employed in static network link prediction [10]; however, they cannot be used for link prediction in dynamic social networks. Yao *et al.* [11] identified the time-varying weights for the prior graphs to understand the temporal dependencies. They then predicted the connection using a modified CN, taking into consideration the neighbors situated between the two hops.

In an enhanced RA-based dynamic network, Zhang *et al.* [12] link prediction algorithm updated the similarity between pairs of nodes as the network structure changed. These techniques, however, are less effective when dealing with strong nonlinearity and rely more on basic network statistics.

3. PROPOSED METHOD

Because static networks have a fixed structure, link prediction algorithms can only produce results by learning about the network's spatial organization and determining its links based on the distribution of the observed edges. Dynamic network links, however, undergo constant modification. The temporal characteristics of the network should also be learned by dynamic link prediction algorithms from the network sequence. Based on the selection of similarity criteria in various time intervals in accordance with the dynamics of the social network, link prediction is carried out in this paper. Using a pattern of changes in similarity criteria based on time, it is feasible to forecast the future structure of the graph and the presence or absence of linkages between two nodes.

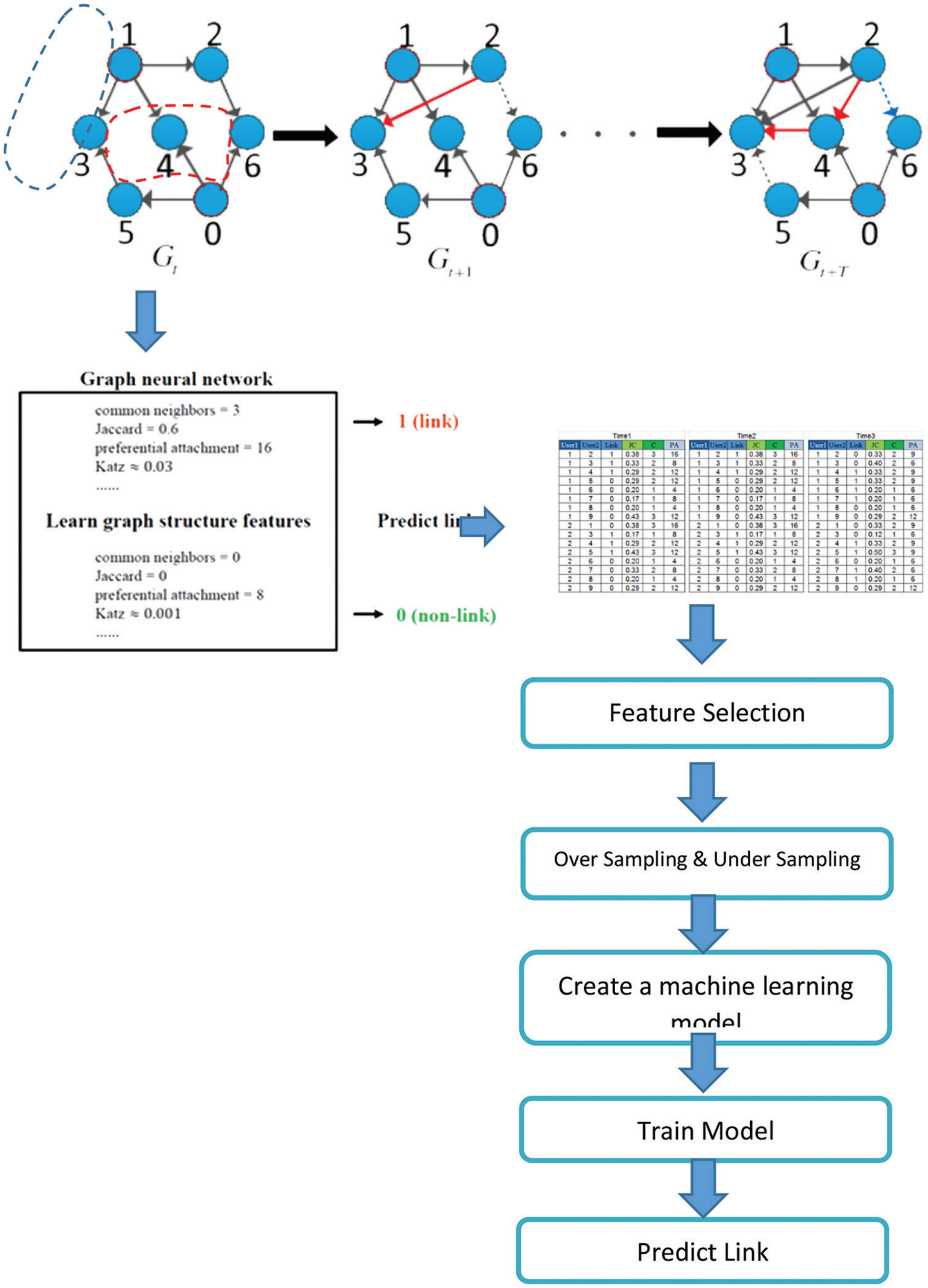
According to the suggested method, a dynamic network is represented as a progression of graphs that are captured over time at regular intervals. A network sequence with many photographs of a discrete network taken across the time intervals G 1, G 2, and G end can be thought of as a dynamic social network. When G t = (V,E t), the k'th image of a dynamic network that reflects the network at time t, is represented by a sequence of graphs (t1,2,end). When a set of vertices, V, is represented by a set of edges, E t, which contains the connections between each pair of vertices in the network, is represented by V. The adjacency matrix G t is displayed with A t and a (t; i, j)=1 if there is a directed connection between v i and v j; otherwise, a (t; i, j) = 0. The suggested technique makes use of extracted data.

The suggested link prediction model, which makes use of correlation-based feature selection and data balancing, was designated as a similarity learning model in the dynamic network (LSDN). As a result, the issues of enormous dimensions, non-linearity, and data dispersion can be efficiently handled by this approach. Fig. 1 shows the general layout of the suggested model.

In the following, the steps of implementing the proposed method are explained:

3.1. Creating a Subgraph

Given a sequence of graphs of length N, $S=\{G_t, G_{t-1}\}$, the purpose of predicting a dynamic network link is to learn a function that represents the input sequence S to G_t. In predicting link in dynamic network, time labels T in the network sequence are used to calculate the probability of all links in the next time label t through network information, which can be shown as follows:



Time1						Time2						Time3					
User1	User2	Link	JC	C	PA	User1	User2	Link	JC	C	PA	User1	User2	Link	JC	C	PA
1	2	1	0.38	3	16	1	2	1	0.38	3	16	1	2	0	0.33	2	9
1	3	1	0.33	2	8	1	3	1	0.33	2	8	1	3	0	0.40	2	6
1	4	1	0.29	2	12	1	4	1	0.29	2	12	1	4	1	0.33	2	9
1	5	0	0.29	2	12	1	5	0	0.29	2	12	1	5	1	0.33	2	9
1	6	0	0.20	1	4	1	6	0	0.20	1	4	1	6	1	0.20	1	6
1	7	0	0.17	1	8	1	7	0	0.17	1	8	1	7	1	0.20	1	6
1	8	0	0.20	1	4	1	8	0	0.20	1	4	1	8	0	0.20	1	6
1	9	0	0.43	3	12	1	9	0	0.43	3	12	1	9	0	0.29	2	12
2	1	0	0.38	3	16	2	1	0	0.38	3	16	2	1	0	0.33	2	9
2	3	1	0.17	1	8	2	3	1	0.17	1	8	2	3	0	0.12	1	6
2	4	1	0.29	2	12	2	4	1	0.29	2	12	2	4	1	0.33	2	9
2	5	1	0.43	3	12	2	5	1	0.43	3	12	2	5	1	0.50	3	9
2	6	0	0.20	1	4	2	6	0	0.20	1	4	2	6	0	0.20	1	6
2	7	0	0.33	2	8	2	7	0	0.33	2	8	2	7	1	0.40	2	6
2	8	0	0.20	1	4	2	8	0	0.20	1	4	2	8	1	0.20	1	6
2	9	0	0.29	2	12	2	9	0	0.29	2	12	2	9	0	0.29	2	12

Fig. 1. Learning similarity in dynamic network model.

$$A'_t = argmaxP(A_t \big| A_{t-T}, A_{t-T+1}, \cdots, A_{t-1}) \qquad (1)$$

that $(A_t \big| A_{t-T}, A_{t-T+1}, \cdots, A_{t-1})$ represents the network adjacency matrix in the previous T, A_t represents real adjacency matrix in time t, A'_t is the adjacency matrix predicted by dynamic network link prediction.

The structure of a dynamic network evolves over time. As shown in Fig. 2, some links may appear while others may disappear. According to the variable structure of the social network, we can extract multiple graphs at any time.

3.2. Creating Feature Vector

Pairs of related nodes are found by converting the subgraph intended for implementation in the preprocessing step to edge list form. To show that a link exists between these node pairs, they are given the number 1. On the other hand, there can be a lot of node pairs in the network for which there is not a link yet. The processing of missing node pairs may not be complete. Only node pairs that fall inside the double-hops interval are taken into account to ensure steady computational complexity. These node pairs are marked with the number 0, which denotes the absence of linkages.

A subgraph produced by the correlation of neighbors x and y with h hops from the pair's origin is known as an enclosing subgraph for a pair of nodes (x;y).

Fig. 3 shows the one-hop enclosing subgraphs for (A, B) and (C, D). These enclosing subgraphs are very instructive for link prediction – all similarity-based link prediction methods can be calculated directly from single-hop enclosing subgraphs.

3.3. Creating a Training Set from Subgraphs

At time t, the training set is taken from the subgraphs. There are two training sets in this set: NL and HL. In the NL training set, node pairs without linkages are taken out of the graph. For these node pairs, the similarity criterion is calculated and extracted as one of their attributes. The following phase involves extracting connected pairs of nodes from the network. In addition, retrieved and saved as an HL training set are their similarity features.

3.4. Correlation-Based Feature Selection Method

A subset of characteristics is deemed a good subset in this feature selection approach if, on the one hand, they have a strong correlation with the target feature and, on the other, they are not correlated with one another [8]. The relationship

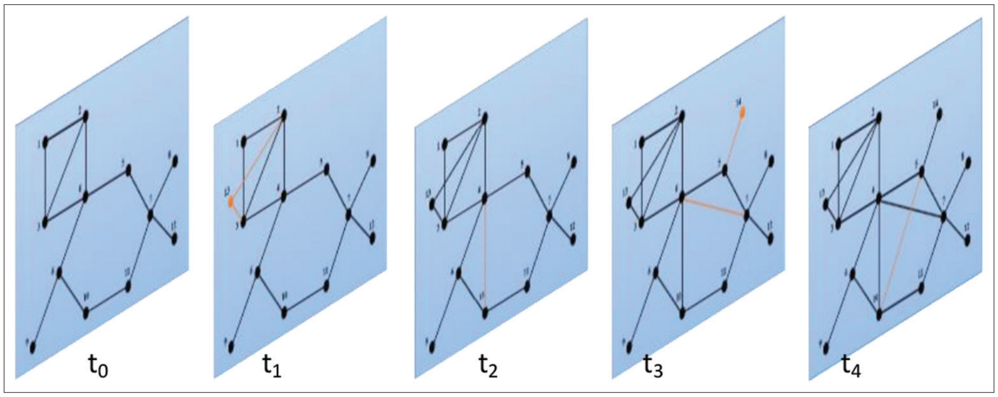


Fig. 2. The change of the graph structure over time and the extraction of subgraphs.

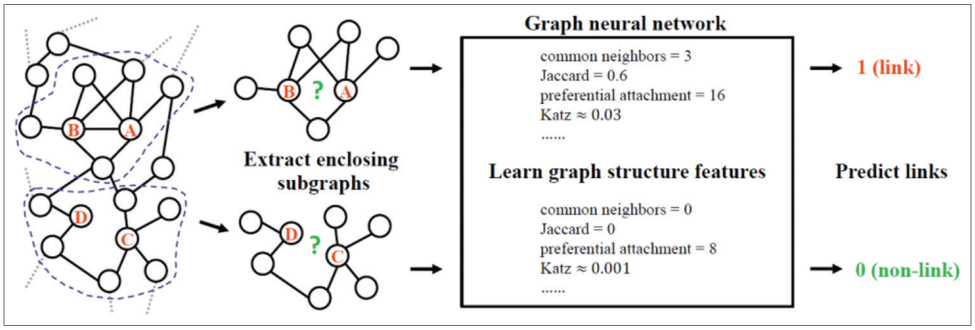


Fig. 3. The enclosing subgraphs.

between node-pair attributes (similarity criteria) and the presence or absence of a link (response) is taken into account in this method as a scenario. In this feature selection scenario, our goal is to find similarity criteria that are significantly dependent on the response. Equation (2) determines a subset of characteristics' Merit:

$$Merit_{S_k} = \frac{k\,\overline{r}_{cf}}{\sqrt{k + k(k(k-1)\,\overline{r}_{ff})}} \tag{2}$$

In this equation, $\overline{r}_{cf}$ is the mean of the calculated correlation between the target feature and all the features in the data set and $\overline{r}_{ff}$ is the mean of the one-to-one correlation calculated between the features [8]. Finally, the correlation-based method is formulated as follows:

$$CFS = \max_{Sk}\left[\frac{r_{cf_1} + r_{cf_2} + \ldots\ldots + r_{cf_k}}{\sqrt{k + 2(r_{f_2f_1} + \ldots + r_{f_if_j} + \ldots + r_{f_kf_1})}}\right] \tag{3}$$

In this regard, the variables $\overline{r}_{cf_i}$ and $\overline{r}_{f_if_i}$ are called correlation values.

3.5. Balancing Data

Data classification issues include unbalanced data. Data that are out of balance have drastically differing class proportions. The data are unbalanced if one class (the dominating or majority class) comprises 90% of the data and the remaining 10% of the data. Under-sampling and over-sampling are two techniques used in machine learning to handle uneven data. To put it another way, either the dominant class is under-sampled, or the minority class is over-sampled, or both approaches are combined. The performance of machine learning algorithms in mistake detection may be impacted, which makes the unbalanced categorization a concern. To make the class ratios more similar, oversampling seeks to provide more samples from the minority class. In addition, the goal of under-sampling is to take fewer samples from the majority class. To get the ratio of classes closer to one another, we actually do not use all the samples from the majority class in this technique [9].

Oversampling is the process of selecting random samples from the minority class to select the same sample more than once by replacing and adding to the training data. In fact, to increase the number of samples, this method reproduces minority samples. Many machine learning algorithms are highly adept at handling uneven data by using sampling techniques. However, because we are either copying current data or deleting samples, properly adjusting the data ratio is crucial. The outcome may be significantly impacted if these samples are taken carelessly. The rising cost of computation is another issue we need to be mindful of. There may be more calculations required if there are more samples in the minority class, especially if the dataset is very imbalanced. When combined, the under-sampling and over-sampling random approaches in the suggested method perform better as a whole than when used alone. This implies that we can reduce the size of the majority class while increasing the size of the minority class. This approach makes an effort to utilize the benefits of the earlier approaches while attempting to avoid their drawbacks.

3.6. Training Learning Model

The dataset is constructed using a vector set of features and node-pair labels (link = 1 and non-link = 0). The training dataset is chosen and the validation dataset is chosen from the remaining portions of the dataset. To provide the ability of fitting model parameters (the weight of connections between neural cells in artificial neural networks) using a set of feature vectors and each vector's label, the learning model is taught on a set of training data using a supervised learning method, such as optimization methods like gradient descent or random gradient descent (link or not-link). The similarity criterion of the two nodes in the training data set is compared with the label of each input vector to produce a result.

4. EVALUATION OF THE PROPOSED METHOD

4.1. Description of the Dataset

Now, working on Facebook data begins. In this example, the combined "Ego Networks" dataset is used. This dataset contains the aggregated network of ten Facebook friends list. The facebook_combined.txt file can be downloaded from the website https://snap.stanford.edu/data/egonets-Facebook.html. The interested people can receive their Facebook/Twitter data using the Facebook/Twitter API and use it to do this example and analyze their data. To do this project, first, the file is read and then its graph is created.

Fig. 4 shows the Ego Networks dataset graph. This dataset contains the aggregated network of ten Facebook friends. The color and size of the nodes vary according to the degree the centrality, respectively.

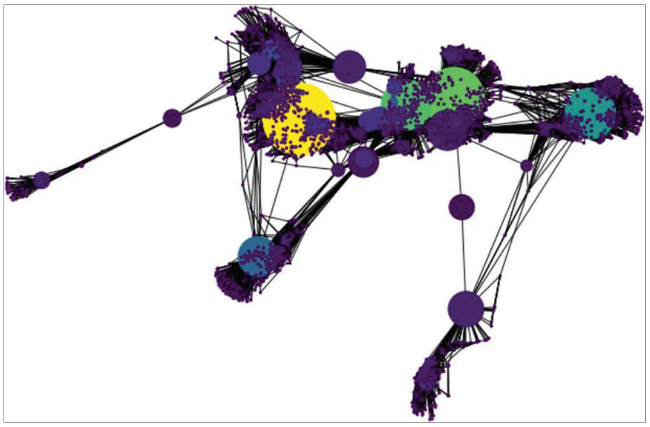


Fig. 4. Ego networks dataset graph.

The feature vector is constructed by considering all pairs of nodes that exist at double-hop intervals. Therefore, various global similarity criteria have been adopted in the methodology section. Fig. 5 shows the simulation result that we gave him the graph that created a several subgraphs for prediction between anode and neighbors. In the proposed framework, seven local similarity indicators involving CN, Jaccard coefficient, Adamic and Adar criterion, preferred connection, RAI, Salton index), and Sorenson index are considered as a measure of similarity in the feature construction process.

4.2. Evaluated Methods

In this section, the evaluated methods of this research are presented in Table 1.

4.3. Evaluation Metrics

4.3.1. Area under the precision-recall curve

Recall quantifies the proportion of positive results that were obtained. The degree of precision indicates the proportion of real positive findings. A Precision-Recall Curve that shows how an increase in recall impacts precision is created using these metrics. This curve's area under it serves as a measure of link equality.

4.3.2. Recall

The ratio of the number of correctly predicted results to the total number of predicted results.

$$recall = \frac{tp}{tp + fn} \tag{4}$$

4.3.3. Precision

The ratio of the number of correctly predicted results to the total number of predicted results is related.

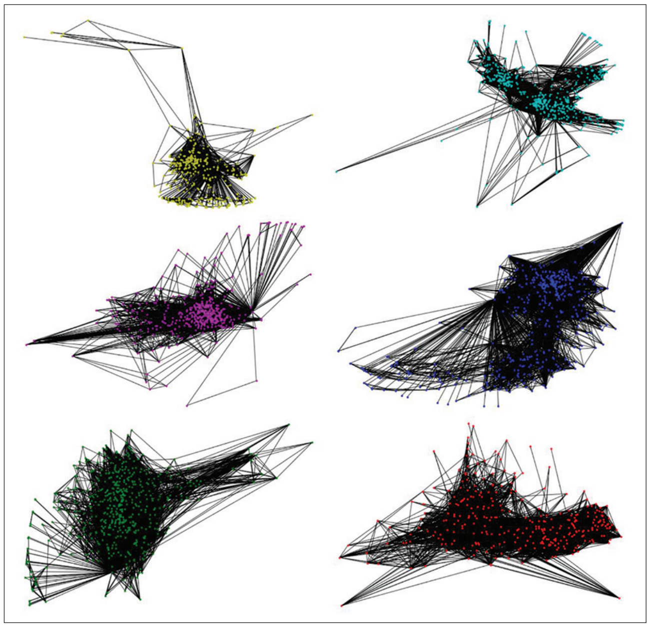


Fig. 5. Subgraphs extracted from ego network.

Table 1: Evaluated methods

Evaluated methods
Support vector machine
Decision tree
Random forest
Bayes naïve
K-nearest neighbor
MLP
LSDN

MLP: Multi-layer perceptron, LSDN: Learning similarity in dynamic network

$$precision = \frac{tp}{tp + fp} \tag{5}$$

4.3.4. Performance (F-score)

Measures system performance by considering recall and precision; the formula is as follows:

$$F - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{6}$$

4.4. Experimental Setting

The proposed work has been done on the CPU i5 2.29 GHz. For experimental purposes, we used the Networkx package to obtain structure-based similarity among network nodes. To use link prediction models, we have written programs in Python.

Table 2: Evaluate the proposed method in the ego network

Methods	Class	Precision	Recall	F-Score	Accurcy
SVM	Link	0.9302	0.9884	0.9584	0.9225
	No link	0.7387	0.3057	0.4325	
DT	Link	0.9799	0.9797	0.9798	0.9635
	No link	0.8105	0.8119	0.8112	
RF	Link	0.9785	0.99	0.9842	0.9712
	No link	0.8946	0.7962	0.8425	
KNN	Link	0.9346	0.9786	0.9561	0.9188
	No link	0.6423	0.3597	0.4611	
MLP	Link	0.9627	0.9776	0.9701	0.9455
	No link	0.7549	0.6454	0.6959	
NB	Link	0.9667	0.9484	0.9575	0.9238
	No link	0.5898	0.6945	0.6379	
LSDN	Link	0.9932	0.9933	0.9931	0.994
	No link	0.9971	0.9978	0.9975	

MLP: Multi-layer perceptron, LSDN: Learning similarity in dynamic network, SVM: Support vector machine

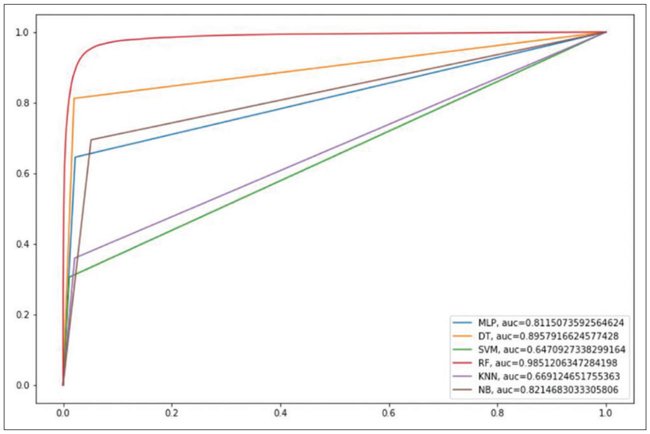


Fig. 6. The ROC curve evaluation of the proposed method.

5. RESULTS AND ANALYSIS

Based on the results obtained from Table 2, it can be seen that the similarity LSDN and KNN have the best and worst performance in terms of the accuracy of the prediction model, respectively. According to the results of three criteria (accuracy, coverage, and efficiency), link prediction using similarity LSDN provides better results than other learning methods such as neural network learning and decision tree. The accuracy of the link prediction in the LSDN algorithm has been improved from 97% to 99% compared to the random forest algorithm.

We also evaluated performance in terms of area under the curve (AUC) values. Fig. 6 shows the analysis of the receiver operating characteristic graph link prediction in terms of AUC value.

From (Fig. 6), it can be seen that the LSDN model has the highest AUC value and the SVM model has the lowest AUC values compared to other algorithms. Based on the obtained AUC criteria, the LSDN is 99%, which is a better result than the decision tree 89%, simple Bayesian 82%, MLP 81%, KNN 66%, SVM 64%, and RF 97%.

6. CONCLUSION

One of the quickly developing study areas in the field of social network analysis is link prediction, which has a wide range of applications, including identifying and simulating the growth of various social network types. To apply the model to successfully identify lost ties, social network interpretation may be helpful. To forecast missing links, a link prediction framework based on the global similarity criterion is proposed in this research. According to research records, several scholars have chosen to use a variety of structure-based similarity criteria to solve the link prediction problem. For capturing structural data in complicated networks, we suggest link-based predictive machine learning algorithms. Through a variety of machine learning models, we attempted to aggregate all the global similarity criteria to forecast future linkages. It may be inferred from this experiment that none of the distinct similarity criteria (by themselves) can reliably forecast the missing links. However, the prediction of missing links becomes more precise when these universal standards are taken into account as aspects of machine learning algorithms. Standard evaluation parameters such as AUC, accuracy, F-criteria, recall, and accuracy have been used for validation purposes. For ease of operation, we have provided only a weightless and static network for implementation purposes. In the future, this can be extended to a dynamic network where nodes and edges are added over time.

REFERENCES

[1] C. Muro, B. Li, K. He. Link prediction and unlink prediction on dynamic networks. *IEEE Transactions on Computational Social Systems*. vol. 10, no. 2, pp. 590-601, 2022.

[2] A. Mohan and K. V. Pramod. "Link prediction in dynamic networks using time-aware network embedding and time series forecasting". *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 1, pp. 1981-1993, 2021.

[3] L. Hu, X. Wang, Y. A. Huang, P. Hu and Z. H. You. "A survey on computational models for predicting protein-protein interactions". *Briefings in Bioinformatics*. vol. 22, p. bbab036, 2021.

[4] A. K. Singh and L. Kailasam. "PILHNB: Popularity, interests, location used hidden Naive Bayesian-based model for link prediction in dynamic social networks". *Neurocomputing*, vol. 461,

pp. 562-576, 2021.

[5] M. Zhang, Y and Chen. "Link prediction based on graph neural networks". *Advances in Neural Information Processing Systems*, vol. 31, pp. 5165-5175, 2018.

[6] G. J. De Bruin, C. J. Veenman, H. J. van den Herik and F. W. Takes. Experimental Evaluation of Train and Test Split Strategies in Link Prediction. In: *International Conference on Complex Networks and their Applications*. Springer, Cham, 2020, pp. 79-91.

[7] T. D. Hua, A. T. Nguyen-Thi and T. A. H. Nguyen. Link prediction in weighted network based on reliable routes by machine learning approach. In: *2017 4th NAFOSTED Conference on Information and Computer Science*. IEEE, 2017, pp. 236-241.

[8] M. E. Newman. “Clustering and preferential attachment in growing networks”. *Physical Review E*, vol. 64, no. 2, p. 025102. 2001.

[9] T. Zhou, L. Lü and Y. C. Zhang. “Predicting missing links via local information”. *The European Physical Journal B*, vol. 71, no. 4, pp. 623-630, 2009.

[10] L. Lü and T. Zhou. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 6, pp. 1150-1170, 2011.

[11] L. Yao, L. Wang, L. Pan and K. Yao, K. Link prediction based on common-neighbors for dynamic social network. *Procedia Computer Science*, vol. 83, pp. 82-89, 2016.

[12] Z. Zhang, J. Wen, L. Sun, Q. Deng, S. Su and P. Yao. “Efficient incremental dynamic link prediction algorithms in social network”. *Knowledge-Based Systems*, vol. 132, pp. 226-235, 2017.