ICIC Express Letters
Part B: Applications
Volume **14**, Number **3**, March **2023**

ICIC International ©2023 ISSN 2185-2766
pp. **303–311**

LIGHTGBM FOR LINK PREDICTION BASED ON GRAPH STRUCTURE ATTRIBUTES

Asia Mahdi Naser Alzubaidi

College of Computer Science and Information Technology
University of Kerbala
Employees Neighborhood, Street Number 95, Home 41, Kerbala 56001, Iraq
asia.m@uokerbala.edu.iq

Received September 2022; accepted December 2022

Abstract. *Link prediction in networks is the process of predicting whether or not a link will exist in the future. Previous link prediction studies used heuristic methods. Because of their simplicity, interpretability, in some cases, scalability, they have a wide range of practical applications. However, their performance in other forms of networks is restricted. This paper focuses on predicting the missing link by formalizing and developing an approach that can extract features from network topology for analyzing the proximity scores of nodes, links, and their attributes in a network. The main step in this paper is to discover a new missing edge finding algorithm by randomly introducing an equal number of missing links or 'false links' into the dataset which are not in the graph and whose shortest path length is greater than 2. Then, intensive feature engineering is performed to extract over 38 new features introduced as the main dataset for the learning method, including graph-specific characteristics, node-like features, and a few more scores. Finally, the LightGBM algorithm was used to train and test the constructed dataset. Experiment-based performance measures like AUC, F1, Accuracy, and AP have achieved impressive results on various datasets and surpass all baseline techniques, containing prior state-of-the-art models.*

Keywords: Social network analysis, Structural information, Heuristic scores, Feature selection, Relational data approach, Community based approach, LightGBM method

1. **Introduction.** Social networking is a popular method for modeling the interactions among users in a specific community. It may be represented as a social graph, with each node representing a network member and each edge indicating a type of relationship between the persons involved [1]. Social networks are exceptionally energetic structures, sparse, and have collective structures over time that evolve rapidly by creating new associates, understanding how these networks change provided a novel percentage of awareness of the mechanisms that motivate them [2]. Link prediction is a branch of social network analysis that analyzes whether two nodes in the network are more closely linked soon. The link prediction can apply in various domains like knowledge graph, completion, information retrieval to analyze the hyperlink structure of the web and recommendation frameworks to propose modern companions or common interests, bio-informatics within the think about of the protein-protein interaction network, connected examination, and mining for recognizing covered up hoodlums in terrorist networks, link analysis and mining for recognizing hidden criminals in terrorist networks and E-commerce to facilitate purchasing a value of the customer by recommending products to consumers via over-targeting on past basis purchase history and general customer data [3]. Various heuristics methods were proposed in early research to handle the link prediction problem from different areas, which finds proximity between potential nodes and predicts connection presence based on the metrics. Heuristics approaches, on the other hand, fared well in some social

DOI: 10.24507/icicelb.14.03.303

networks, such as protein-protein interaction networks, when two proteins with a high number of shared neighbors have a low likelihood of interacting [4]. Later studies offer latent methods to increase the accuracy of connection prediction. Those approaches, on the other hand, may be capable of improving accuracy in some varieties of social networks, but they fared worse in others than the simple heuristics approach [5]. Generally, the probable link prediction task is mainly deliberated from two views: structure-based prediction and features of nodes-based and edges-based prediction, where network structure denoted the method for which nodes that comprised the social network are consistently based on the typical assumption that the more similarities a node pair has, the more likely they are to connect [6]. The link prediction has also been considered from a machine learning perspective, by utilizing features of nodes in the graph and applying various algorithms such as decision tree, support vector machine, Naïve Bayes, deep learning, convolutional neural network, graph neural network, and random forest [7]. Despite improvements in prediction accuracy in similarity-based methods, balancing performance and computational complexity for attributes-based metrics is difficult. This is because the output is dependent on matrix measurement [8]. The contributions of this work are as follows.

1) Construct a multiple features dataset that required the knowledge of predecessors and successors of each node in the directed graph. The dataset is firstly containing the source and distention nodes and then by performing extensive feature engineering, we added over 38 new heuristic features such as Jaccard Index, Adar index, Page Rank index, Shortest path length, and Katz Centrality.
2) Build a link prediction model that computes the closeness metric of two nodes and concludes that very comparable nodes are more likely to be connected throughout the entire network.
3) A link prediction technique based on the LightGBM learning methodology is developed, which incorporates distinct features. To enhance link prediction performance, the approach thoroughly considers network topology and node attributes.

Deep learning is a distinct route in machine learning that was recently presented in the literature. One limitation of typical deep learning models is that the input is dispersed independently and uniformly, rendering them incapable of representing relational data. To address this issue, Wang et al. [9] used a Bayesian deep learning framework that successfully learns relational data. Li et al. [10] proposed a new deep learning framework, conditional temporal Restricted Boltzmann Machine (ctRBM), that captures network evolution trends or dynamic networks. Graph Convolution Networks (GCN) are a recent family of deep network methods that are utilized in network embedding, node categorization, and link prediction. A localized first-order approximation of spectral convolutions is used to train the model's representation [11]. Ragunathan et al. [12] proposed a framework called PLACN and compared this method with the state-of-the-art methods and reached above 96% of AUC. Recently, a new subgraph method Weisfeiler-Lehman Neural Machine (WLNM) was considered to be a state-of-the-art link prediction method based on its high accuracy [13]. This necessitates a large number of hops from the enclosing subgraph to the entire network, as well as additional computation time and memory. To overcome this issue, SEAL (learning from Subgraphs, Embeddings, and Attributes for Link prediction) employed graph neural networks to learn overall graph structural properties from local enclosing subgraphs [14]. They considered first-order, second-order, and high-order heuristic scores to create a vector. The SEAL model achieves state-of-the-art performance for the link prediction. Finding an acceptable hop number for a specific network is a trial-and-error approach. Another disadvantage of SEAL is the loss of topological information produced by pooling layers, as well as the inefficiency of graph convolution layers for learning edge embeddings from graphs.

2. **Main Method.** This section described our approach for enhancing link prediction accuracy in social networks by assessing the common neighbors of each node's predecessors and successors. We propose a link prediction framework as depicted in Figure 1 using the tree-based learning method. To begin, the missing edge detection algorithm is used by categorizing the directed graph edges as positive or negative connections. The desired dataset is then obtained by extracting and aggregating graph features from the entire graph and using the LightGBM approach to categorize positive and negative links. Throughout the whole section, the symbols X and Y have represented two nodes from the graph, N represents the number of nodes in the network, and K represents the average degree. $\Gamma(X)$ and $\Gamma(Y)$ signify the first-order neighbor sets of these nodes. The suggested link prediction algorithm takes the whole directed graph as input of the edges list and applies the algorithm to predecessors and successors of source and destination nodes to avoid information loss as accrued in most state-of-the-art methods which are performed

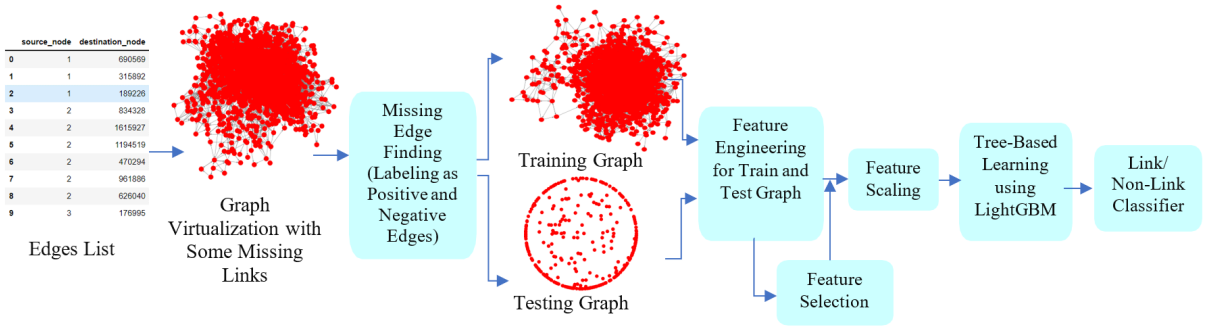

FIGURE 1. The proposed link prediction framework

Algorithm 1: Missing edges finding

```
Input: G: graph, R: edge list.
Output: Missing_edges.
Let edges = dict()
Let edges_num = G.edges
Let nodes_num = G.nodes
For edge in R do
        Edges[(edge[0],edge[1])] = 1
Missing_edges = set([])
While |Missing_edges| < edges_num
          Ui = random(1, edges_num)
          Uj = random(1, nodes_num)
            T = edges.get((Ui, Uj), −1)
            If T = −1 and Ui ≠ Uj
          Try:
                If shortest_path_length(G, Ui, Uj) >= 2
                        Missing_edges.add((Ui, Uj))
                Else
                        Continue
          except:
                Missing_edges.add((Ui, Uj))
            Else
              Continue
Pickle.dump(Missing_edges)   # take pickle dump of missing edges
End
```

in subgraph and applies the algorithms on target nodes only. The proposed technique takes the nodes with an indicator labeling for positive link class and negative link class by using missing edges finding as in Algorithm 1.

Our dataset consists of pair nodes that are already linked. This can be assigned class label = 1, as there is already a link present between them. To convert the given task into a machine learning classification problem, we randomly generated a pair of nodes from the existing set of all nodes, such that they are not already linked. For pair of nodes, we assigned class label = 0. Next, we combine these positive and negative class label data points. Before going to build the machine learning algorithm, we performed extensive feature engineering and added over 38 new features to the dataset. Extended features engineering is used to build training and testing data by extracting and aggregating graph

TABLE 1. The heuristic attributes extraction

#	***Extended attributes***	***Equation***	
1.	Common Neighbors (CN) [16]	$S(X,Y) = \lvert\Gamma(X) \cap \Gamma(Y)\rvert$	(1)
2.	Jaccard's Coefficient (JC) [17]	$S(X,Y) = \lvert\Gamma(X) \cap \Gamma(Y)\rvert / \lvert\Gamma(X) \cup \Gamma(Y)\rvert$	(2)
3.	Cosine (Salton) Index (CI) [18]	$S(X,Y) = \frac{\Gamma(X)\lvert\Gamma(Y)}{\lVert\Gamma(X)\rVert * \lVert\Gamma(Y)\rVert}$	(3)
4.	Sørensen Index (SI) [18]	$S(X,Y) = \frac{2\lvert\Gamma(X)\cap\Gamma(Y)\rvert}{\lvert\Gamma(X)+\Gamma(Y)\rvert}$	(4)
5.	Adamic-Adar Score for Source and Destination (AA) [19]	$S(X,Y) = \sum_{Z\in\Gamma(X)\cap\Gamma(Y)} \frac{1}{\log\lvert\Gamma(Z)\rvert}$	(5)
6.	Resource Allocation Index (RA) [20]	$S(X,Y) = \sum_{Z\in\Gamma(X)\cap\Gamma(Y)} \frac{1}{\lvert\Gamma(Z)\rvert}$	(6)
7.	Preferential Attachment Index (PAI) [16]	$S(X,Y) = \Gamma(X) * \Gamma(Y)$	(7)
8.	Resource Allocation index Soundarajan Hopcroft (RA_SH) [21]	$S(X,Y) = \sum_{w\in\Gamma(u)\cap\Gamma(v)} \frac{f(w)}{\lvert\Gamma(w)\rvert}$	(8)
9.	Common Neighbor Soundarajan Hopcroft (CN_SH) [22,23]	$S(X,Y) = \lvert\Gamma(u) \cap \Gamma(v)\rvert + \sum_{w\in\Gamma(u)\cap\Gamma(v)} f(w)$	(9)
10.	Within and Inter-Cluster Index (WIC) [24]	$WIC_{S_{X,Y}} = \frac{\lvert A^w_{x,y}\rvert}{\lvert A^{IC}_{x,y}\rvert+\delta}$	(10)
11.	Deductive Metric (DED) [25]	$DED_{S_{X\to Y}} = \frac{\lvert A(X)\cap D(Y)\rvert}{\lvert A(X)\rvert}$	(11)
12.	Inductive Metric (IND)	$IND_{S_{X\to Y}} = \frac{\lvert D(X)\cap D(Y)\rvert}{\lvert D(X)\rvert}$	(12)
13.	Inductive LOG (IND_LOG)	$S_{X\to Y} = \frac{\lvert D(X)\cap D(Y)\rvert}{\lvert D(X)\rvert} * \log(\lvert D(X)\rvert)$	(13)
14.	Deductive LOG (DED_LOG)	$S_{X\to Y} = \frac{\lvert A(X)\cap D(Y)\rvert}{\lvert A(X)\rvert} * \log(\lvert A(X)\rvert)$	(14)
15.	The INF Score	$INF_{S_{X\to Y}} = DED_{S_{X\to Y}} + IND_{S_{X\to Y}}$	(15)
16.	INF_2D Score	$INF\_2D_{S_{X\to Y}} = DED * 2_{S_{X\to Y}} + IND_{S_{X\to Y}}$	(16)
17.	PageRank for Source and Destination (PR) [26]	$PR_i(t) = \sum_{j=1}^{N} a_{i,j} PR_i(t-1)/K_j^{out}$	(17)
18.	Shortest Path for Source and Destination		
19.	The User Following Back Metric		
20.	Same Community for Source and Destination Nodes		
21.	Katz Centrality for Source and Destination Nodes [27]	$X_i = \alpha \sum_j A_{ij} X_j + \beta$	(18)
22.	The Hyper-Link Induced Topic Search Metric (HITS)		
23.	Weight Features for Source and Destination	$w = \frac{1}{\sqrt{1+\lvert X\rvert}}$	(19)

structure features. The graph provides information about how the nodes are connected, and similarity scores indicate how strongly they are related to one another [15]. Some of extended attributes are depicted in Table 1.

To build the model, feature scaling is performed. For each feature, the Standard Scaler scales the values such that the mean (M) is 0 and the standard deviation (Std) is 1.

$$X_{scaled} = X - M/Std \tag{20}$$

3. **Results and Discussions.** The proposed link prediction model objectives are to adjust with diverse types of social networks and improve the link prediction model's efficiency. To evaluate the suggested model, we are using AUC (Area Under Curve), F1 score, Accuracy (ACC), and the Average Precision score (AP) as evaluation performance metrics.

3.1. **Real-world datasets.** We chose 15 different types of real-world network datasets from varied locations and sizes. All the datasets are publicly available online. Table 2 shows the datasets and their details.

Table 2. The statistical information of each real-world network

Dataset	***Nodes***	***Edges***	***Degree***	***Type***
Celegans	297	4296	14.4646	Biology
USAir	332	4252	12.8072	Transportation dataset
NSC	1461	5484	3.7536	Co-authorship
Yeast	2375	23386	9.8467	Biology network
Power	4941	6594	1.3345	Electrical grid network
PB	1222	33428	27.3552	US political
Router	5022	12516	2.4922	Internet routing
E.coli	1805	14660	8.1219	Pairwise reaction
arXiv	17421	396160	22.7404	Collaboration network
Facebook	4039	176468	43.6910	Friendship network
Wikipedia	4777	184812	38.6879	Online encyclopedia
PPI	3890	76584	19.6874	Protein-protein interactions
PubMed	19717	88651	4.4962	Citations for biomedical literature
Citeseer	3327	9228	2.7737	Citation network
Cora	2708	10556	3.8981	Citation network

3.2. **Results.** We compare the suggested link prediction model against WLNM, SEAL and PLACN methodologies and repeat all of the trials ten times and give the average assessment metrics outcomes. The experimental results show that our model is a fantastic and flexible framework for link prediction.

Table 3 and Table 4 depicted the results of AUC and AP scores, in general, the WLNM, SEAL, and our method achieve much better than only using the baseline heuristics scores. Among learning-based methods, we can observe that our model has the best performance and has superior graph feature learning ability over graph kernels, fully connected neural networks, graph neural networks, and convolution neural networks. This implies finding new heuristics scores for networks to catch more and more of their structural features and using them in the learning classifier where no existing heuristics work can significantly enhance model performance. Moreover, the experimental findings of similarity-based connection prediction indicated that small differences in overall AUC value such as in Wikipedia, Power, and Router datasets do not necessarily imply low predictability for that dataset. This is due to the variety in size of each network. Overall, the findings of this experiment confirm that measuring correlation using a similarity vector for edges is

Table 3. The comparison using AUC score

Dataset	*WLNM*	*SEAL*	*PLACN*	*Our method*
Celegans	86.18	90.30	96.08	**98.4270**
USAir	95.95	96.62	98.36	**98.9440**
NSC	98.61	98.85	99.53	**99.7730**
Yeast	95.62	97.91	98.87	**99.1560**
Power	84.76	87.61	**98.78**	98.7480
PB	93.49	94.72	96.67	**98.6160**
Router	94.41	96.38	98.40	98.2400
E.coli	97.21	97.64	–	**99.3090**
arXiv	99.19	99.40	–	**99.8660**
Facebook	99.24	99.40	–	**99.8640**
Wikipedia	99.05	**99.63**	–	99.2060
PPI	88.79	93.52	–	**98.9100**
PubMed	–	–	–	99.0050
Citeseer	–	–	–	98.4850
Cora	–	–	–	97.8190

–: Not calculated

Table 4. The comparison using AP score

Dataset	*WLNM*	*SEAL*	*Our method*
Celegans	85.08	89.48	**97.4950**
USAir	95.95	96.80	**98.5530**
NSC	98.81	99.06	**99.6300**
Yeast	96.40	98.33	**98.7490**
Power	87.16	89.55	**98.0730**
PB	92.69	94.31	**97.5730**
Router	93.53	96.23	**97.2380**
E.coli	97.50	98.03	**98.9680**
arXiv	–	–	99.8080
Facebook	–	–	99.8010
Wikipedia	–	–	98.7780
PPI	–	–	98.2830
PubMed	–	–	98.4090
Citeseer	–	–	97.8760
Cora	–	–	96.8180

–: Not calculated

an effective way of distinguishing between test and non-existing node edges. Moreover, the state-of-the-art methods use only AUC and AP as performance measurements, while our model can be evaluated by other metrics such as accuracy and F1 scores as shown in Table 5, respectively.

As we can see from Table 5, our model shows significant improvement and high values of AUC and F1 scores. One reason for this is that our model learns from both graph topologies in terms of node and edge characteristics at the same time. Our predictive modeling has 38 attributes and this could be many features as compared with state-of-the-art methods and can slow the development and training of the model. Additionally, the performance of some models can degrade when including input features that are not relevant to the label. However, when we implemented feature selection in our model, we can note that it reduces all the performance measurements as depicted in Table 6.

Table 5. Accuracy and F1 scores for our model

Dataset	ACC	F1 score
Celegans	98.429	98.444
USAir	98.942	98.947
NSC	99.772	99.769
Yeast	99.156	99.157
Power	98.749	98.715
PB	98.607	98.597
Router	98.243	98.264
E.coli	99.309	99.298
arXiv	99.866	99.865
Facebook	99.864	99.864
Wikipedia	99.206	99.209
PPI	98.910	98.900
PubMed	99.005	99.005
Citeseer	98.483	98.493
Cora	97.821	97.865

Table 6. Evaluation scores for our model with feature selection

Dataset	AUC	AP	ACC	F1 score
Celegans	98.247	96.797	98.255	98.288
USAir	98.355	97.638	98.354	98.364
NSC	99.086	98.704	99.088	99.074
Yeast	99.156	98.749	99.156	99.157
Power	97.621	96.524	97.631	97.559
PB	98.107	96.702	98.094	98.085
Router	97.759	96.477	97.764	97.794
E.coli	98.888	98.137	98.883	98.87
arXiv	99.825	99.761	99.825	99.824
Facebook	99.823	99.732	99.823	99.822
Wikipedia	99.074	98.530	99.075	99.08
PPI	98.910	98.900	98.577	98.577
PubMed	98.412	97.788	98.412	98.404
Citeseer	97.265	96.054	97.264	97.286
Cora	96.18	94.437	96.188	96.272

The feature importance analysis reveals that "same community", which is a binary feature representing whether the two nodes are in the same community or not is the most important feature in our prediction model. Other important features include preferring attachment, shortest path length, weight features, and resource allocation. In summary, these findings indicate that combining the influence of multiple types of features can result in much better model accuracy than subgraphing methods.

4. **Conclusion.** The link prediction has attracted growing interest from a variety of scientific disciplines as a key research issue in complex network analysis. Heuristics-based methods have gained the majority among diverse approaches due to their minimal complexity and excellent interpretability. In this paper, we implemented extended feature extraction and found 38 different heuristic features. Our model outperforms both state-of-the-art and baseline approaches, according to performance measurements. However, we should agree that assessing connections using heuristics scores as features in the learning model is a reliable way to differentiate between the test and non-existent node edges. The

goal in future work will be to make better predictions of node associations, building and adding new node features instead of the graph structure attributes to the model may help improve performance as it will add information.

Acknowledgment. The author gratefully acknowledges the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] M. Hasan, *Social Network Data Analytics*, Springer US, Boston, MA, 2011.
[2] A. Samad, M. Qadir, I. Nawaz, M. Islam and M. Aleem, A comprehensive survey of link prediction techniques for social network, *EAI Endorsed Trans. Ind. Networks Intell. Syst.*, vol.7, no.23, 163988, 2020.
[3] K. Prajapati, H. Shah and R. Mehta, A survey of link prediction in social network using deep learning approach, *Int. J. Sci. Technol. Res.*, vol.9, no.4, pp.2540-2543, 2020.
[4] L. Cai, J. Li, J. Wang and S. Ji, Line graph neural networks for link prediction, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.8828, 2021.
[5] A. Gupta and Y. Raghav, Deep learning roles based approach to link prediction in networks, *Computer Science & Information Technology (CS&IT)*, pp.203-222, 2020.
[6] Y. Xiao, R. Li, X. Lu and Y. Liu, Link prediction based on feature representation and fusion, *Inf. Sci.*, vol.548, pp.1-17, 2021.
[7] T. Poštuvan, S. Salkić and L. Šubelj, Learning-based link prediction analysis for Facebook100 network, *Uporab. Inform.*, vol.29, no.2, 2021.
[8] A. K. Rai, A survey on link prediction problem in social networks, *Int. J. Res. Appl. Sci. Eng. Technol.*, vol.5, no.9, pp.1875-1883, 2017.
[9] H. Wang, X. Shi and D. Y. Yeung, Relational deep learning: A deep latent variable model for link prediction, *The 31st AAAI Conf. Artif. Intell. (AAAI2017)*, pp.2688-2694, 2017.
[10] X. Li, N. Du, H. Li, K. Li, J. Gao and A. Zhang, A deep learning approach to link prediction in dynamic networks, *Proc. of the 2014 SIAM International Conference on Data Mining*, vol.1, pp.289-297, 2014.
[11] S. Fu, W. Liu, K. Zhang, Y. Zhou and D. Tao, Semi-supervised classification by graph p-Laplacian convolutional networks, *Inf. Sci.*, vol.560, pp.92-106, 2021.
[12] K. Ragunathan, K. Selvarajah and Z. Kobti, Link prediction by analyzing common neighbors based subgraphs using convolutional neural network, *Front. Artif. Intell. Appl.*, vol.325, pp.1906-1913, 2020.
[13] M. Zhang and Y. Chen, Weisfeiler-Lehman neural machine for link prediction, *Proc. of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.575-583, 2017.
[14] M. Zhang and Y. Chen, Link prediction based on graph neural networks, *Adv. Neural Inf. Process. Syst.*, pp.5165-5175, 2018.
[15] K. Ragunathan, *Convolutional Neural Network for Link Prediction Based on Subgraphs in Social Networks*, Master Thesis, University of Windsor, 2020.
[16] F. Gao, K. Musial, C. Cooper and S. Tsoka, Link prediction methods and their accuracy for different social networks and network metrics, *Sci. Program.*, vol.2015, pp.1-13, 2015.
[17] A. Saxena, G. Fletcher and M. Pechenizkiy, NodeSim: Node similarity based network embedding for diverse link prediction, *EPJ Data Sci.*, vol.11, 24, doi: 10.1140/epjds/s13688-022-00336-8, 2022.
[18] L. Lü and T. Zhou, Link prediction in complex networks: A survey, *Phys. A Stat. Mech. Its Appl.*, vol.390, no.6, pp.1150-1170, 2011.
[19] D. Liben-Nowell and J. Kleinberg, The link-prediction problem for social networks, *J. Am. Soc. Inf. Sci. Technol.*, vol.58, no.7, pp.1019-1031, 2007.
[20] S. Liu, X. Ji, C. Liu and Y. Bai, Extended resource allocation index for link prediction of complex network, *Phys. A Stat. Mech. Its Appl.*, vol.479, pp.174-183, 2017.
[21] I. Ahmad, M. U. Akhtar, S. Noor and A. Shahnaz, Missing link prediction using common neighbor and centrality based parameterized algorithm, *Sci. Rep.*, vol.10, no.1, 364, 2020.
[22] S. Soundarajan and J. Hopcroft, Using community information to improve the precision of link prediction methods, *WWW'12 – Proc. of the 21st Annual Conference on World Wide Web Companion*, vol.1, pp.607-608, 2012.
[23] Z. Wu, Y. Lin, H. Wan and W. Jamil, Predicting top-L missing links with node and link clustering information in large-scale networks, *J. Stat. Mech. Theory Exp.*, vol.2016, no.8, 083202, 2016.

[24] J. C. Valverde-Rebaza and A. de Andrade Lopes, Link prediction in complex networks based on cluster information, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol.7589, pp.92-101, 2012.

[25] D. Garcia-Gasulla and U. Cortés, Link prediction in very large directed graphs: Exploiting hierarchical properties in parallel, *CEUR Workshop Proc.*, vol.1243, 2014.

[26] S. Priyanta and I. Nyoman, Social network analysis of twitter to identify issuer of topic using PageRank, *Int. J. Adv. Comput. Sci. Appl.*, vol.10, no.1, pp.107-111, 2019.

[27] H. Wang and Z. Le, Seven-layer model in complex networks link prediction: A survey, *Sensors*, vol.20, no.22, 6560, 2020.