Follower Link Prediction using the XGBoostClassification Model with Multiple Graph Features

Dayal Kumar Behera (dayalbehera@gmail.com)
KIIT University
Madhabananda Dash
KIIT University
Subhra Swetanisha
Trident Academy of Technology
Janmenjoy Nayak
AITAM: Aditya Institute of Technology and Management
S Vimal
National Engineering College
Bighnaraj Naik
Veer Surendra Sai University of Technology

Research Article

Keywords: Social network, Follower recommendation, Link prediction, Graph-based features, XGBoost, Classification Model.

Posted Date: February 23rd, 2021

DOI: https://doi.org/10.21203/rs.3.rs-239295/v1

Version of Record: A version of this preprint was published at Wireless Personal Communications on April 7th, 2021. See the published version at https://doi.org/10.1007/s11277-021-08399-y.

Follower Link Prediction using the XGBoost Classification Model with Multiple Graph Features

Dayal Kumar Behera[1,*], Madhabananda Das[1], Subhra Swetanisha[2], Janmenjoy Nayak[3], S. Vimal[4], Bighnaraj Naik[5]

[1]School of Computer Engineering, KIIT University, Bhubaneswar, India

[2]Department of CSE, Trident Academy of Technology, Bhubaneswar, India

[3]Department of Computer Science and Engineering, Aditya Institute of Technology and Management (AITAM), Tekkali-532201, Andhra Pradesh, India

[4]Department of Information Technology, National Engineering College, Kovilpatti, Tamilnadu 628503, India

[5]Department of Computer Application, Veer SurendraSai University of Technology, Burla, Sambalpur 768018, Odisha, India

dayalbehera@gmail.com, mndas_prof@kiit.ac.in, sswetanisha@gmail.com, mailforjnayak@gmail.com, vimal28.05.1984@gmail.com, mailtobnaik@gmail.com

Corresponding Author

Dayal Kumar Behera

School of Computer Engineering, KIIT University, Bhubaneswar, India

E-mail: dayalbehera@gmail.com

Follower Link Prediction using the XGBoost Classification Model with Multiple Graph Features

Abstract: The Follower Link Prediction is an emerging application preferred by social networking sites to increase their user network. It helps in finding potential unseen individual and can be used for identifying relationship between nodes in social network. With the rapid growth of many users in social media, which users to follow leads to information overload problems. Previous works on link prediction problem are generally based on local and global features of a graph and limited to a smaller dataset. The number of users in social media is increasing in an extraordinary rate. Generating features for supervised learning from a large user network is challenging. In this paper, a supervised learning model (LPXGB) using XGBoost is proposed to consider the link prediction problem as a binary classification problem. Many hybrid graph feature techniques are used to represent the dataset suitable for machine learning. The efficiency of the LPXGB model is tested with three real world datasets Karate, Polblogs and Facebook. The proposed model is compared with various machine learning classifiers and also with traditional link prediction models. Experimental results are evident that the proposed model achieves higher classification accuracy and AUC value.

Keywords: Social network, Follower recommendation, Link prediction, Graph-based features, XGBoost, Classification Model.

I. Introduction

In the age of the intelligent web, many users connected to the social media across a heterogeneous network leads to an information overload problem. Social networking services attract many researchers for both ego and complete networking analysis. Ego network analysis deals with individuals of social network whereas complete network analysis deals with group or community.Facebook is one of the biggest social network service providers expanding over more than one thousand million users. In Facebook, individual user can create or join communities for specific interest and goals[1].The rapid growth of the network creates many opportunities for information sharing. However, it’s difficult to find a potential friend or whom to follow in an

extensive user network.Link Prediction (LP) problem [2]is useful to explore unseen links in a social network.As the number of users in social media is increasing day by day, the complexity of this problem is also increasing. This is an active research area in academia and also preferred by many social networkingsites like Facebook,Instagram, Twitter, etc.It has been successfully applied to recommend friends for gaming communities [3]. The graph-based structure is generally preferred to represent usersocial network. In the graph-based model to recommend follower link, the system needs to identify feasible relationships at the time 't' of the social network [4]. To predict the people sharing a common purpose, the link prediction technique plays a vital role. In this paper Kartate and Polblogs dataset are used for analysis along with facebook Dataset.

In a social network internal representation of the user/follower graph is $G\ (U, L)$ where U represents node set of users and L represents a link set of edges. At the time 't' in the given follower graph edge represents a direct relationship between follower and followee. In the future time t' where $t' > t$, the possible edges associated with the active user is known as indirect or predicted relationship. Figure 1 depicts the direct and indirect relationship of the active user u_1. An LP problem deals with finding an indirect relationship using the same graph snapshot at time t. Graph-based representation attracts many researchers to apply graph theory for the LP problem (Y. Li, Luo, Fan, Chen, & Liu, 2017) [6].

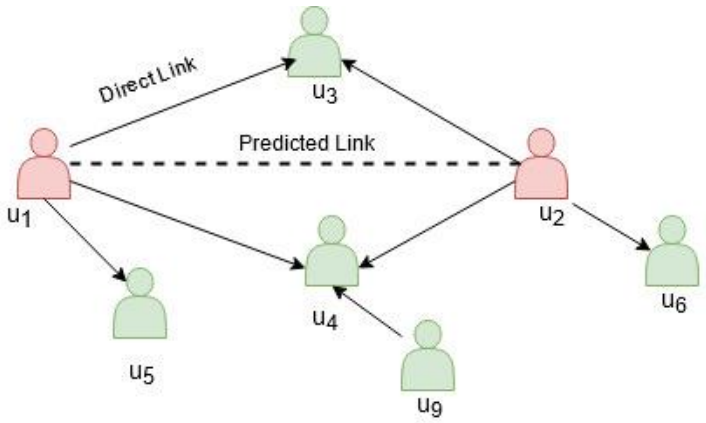


Fig. 1: User graph showing direct and predicted links

Various approaches to an LP problem are broadly classified into similarity-based [7][8]and learning-based system (Han & Xu, 2016) [10]. Similarity-based link prediction problem works

on finding graph-based features, and prediction does not require training data. [11]reviewed different similarity-based features for link prediction problem and found that Common Neighbors, Adamic/Adar, Katz and Jaccard index achieve better performance.[12]had proposed an integrated approach by considering both neighbors and common features of a node (FINN) for prediction of missing links in a social network.Learning-based System uses classifier to train, and the trained model is used for prediction.[9]uses machine learning-based classifier for predicting potential links in Micro-blog network dataset.

Contributions of the work are listed below:

i. Feature set for the classification modeling of the problem is designed by using various path-based and weighted features of the social graph.
ii. Proposed a learning based classification model for predicting the missing potential linkin the user graph.
iii. Studying the impact of combining the features of the social graph.Designing a hybrid feature model by combining both similarity-based and learning-based model.
iv. After preparing the feature set, XGBoost-based classifier is used to train the classifier.

In section II, the related work of the link prediction problem has been discussed. Section III describes the background of graph-based features and the proposed classification model. Implementation steps and result analysis have been discussed in Section IV. Section V concludes the paper with lights on the future work.

II. Related Work

We can view popular strategies for link prediction as a similarity-based traditional approach, supervised and unsupervised learning-based ML approach.

A. Traditional Approach

Traditional link prediction methods are based on calculating the similarity between vertices of the user graph. The similarity calculation is based on the graph's topological property. The topological property can be used for calculation of local similarity index or overall similarity index. Common neighbors, preferential attachment, Adamic-Adar are used to calculate local

similarity index whereas Katz is used to calculate overall similarity index based on the global path[13].Prediction in LP problem is generally based on the topological structure of the graph and influenced by the number of common friends. To address these features [14]proposed two algorithms for local and global link prediction. They have used “cScore” (Correlation Score) and “iScore” (Influential Score) to calculate similarity between two nodes.[15]have also worked on local and global features of the graph. They find similarity between every pair of vertices in the graph having length 2 and 3. Social network changes overtime and dynamic either by addition of new users and by new links across heterogeneous networks. [16]captured information across a heterogeneous network for link prediction.Datasets from various networks such as Epinions, Slashdot, Wikivote and Twitter are used to predict links in one network by extracting the information from the other network. Homophily (similarity) based features of a graph are taken as baseline predictor. [17]introduces multitasking factor graph model by considering both user-user and user-item interaction for prediction of both social edges and rating edges of the graph. They have also studied feature extraction across a heterogeneous network. Their experiments on dataset of two veritable world networks like Epinion and FriendFeed outperforms common neighbor, Jaccard and Logistic Regression based model. [18]represent users of different networks in a multilayer representation and uses proximity based features like Adar and Jaccard to propose an interlayer similarity based link prediction. Traditionally social networks are modeled as a single layer network. But multiplex network deals with multiple layers of information. For example, a user may be connected to users of two different networks which can be represented as multiple layers. They have tested their model by combining data of Twitter, Instagram and Foursquare.

B. Machine Learning (ML) Approach

Currently, machine learning-based models are preferred for an LP problem and successfully used by many online social networking sites. Broadly ML-based models are classified under both supervised classification model and unsupervised learning model.

W. Cukierski *et al.* applied graph-based features [10] for supervised learning based LP problems. They have participated in “IJCNN Social Network” challenge and worked on the Flickr dataset. Random forest model classifier shows higher accuracy than similarity-based model. Area under ROC curve for Flickr dataset achieved was 0.9695. [5] applied utility analysis using Bayesian

Inference in the LP problem with the thought that link formation is done on the basis of individual preferences. Every user in the network works as an intelligent agent for building links with other users by using an Expectation-Maximization algorithm. [19]worked on LP problem for new users or the users having few social links using multilevel deep belief network (DBN). If user u consumes a product$p$, it shows user u likes p. These behaviors are represented in two matrices such as social link and consumption matrix. They focused on how to recommend links for a newly created user in online social networking site. The proposed model has been compared with models for featurization such as Naïve Bayes, C4.5 and SVC classifier in Google+ dataset.[20]proposed a model using node ranking feature by identifying influential nodes in view of common neighbours. Influential node identification plays a vital role in social network analysis. It assigns a rank to every node in the network. [21]focused on an unsupervised model to consider LP problem as a matrix denoising (MD) problem. Topological structure of the social graph is used in designing a mapping function that maps the existing network to a network consisting of all the links. [22]proposed a semi-supervised model to work on a dynamic link prediction problem. By using previous 't' network snapshots, network structure at time t+1 is predicted. The prediction is based on link formation and dissolution network.A loss function was designed for learning the proposed model "SemiGraph".

III. Classification Model for Link Prediction

A. Problem Statement:

Given an edge set, predict the potential missing links or edges. The edge set can be mapped to a directed user graph. Each user is represented as a node or vertex.

Remark 1: If there is a connection between users u_i to u_j, means u_i and u_j are associated with a certain relationship. This direct relationship indicates u_i is following u_j in which u_i is known as follower and u_j is known as the followee.

Remark 2: The link prediction problem over the social graph can be mapped to a classification problem by assigning class 1 for the given edges and class 0 for non-available edges.

B. Dataset

Three real world datasets from smaller to very large user networks are chosen for the experiments. Karate Club is having 34 nodes and 156 edges. Polblogs is having 1490 nodes and 19025 edges. The third dataset is collected from Facebook Recruiting Challenge on Kaggle at a certain time period. This dataset contains two columns which is a pair of vertices as Source node and Destination node. Each user-pair (a, b) represents a direct relationshipand can be viewed as a directed edge from user u_a to user u_b.Facebook dataset contains 1862220 nodes and 9437519 edges. All the users available in the dataset are considered for analysis.

C. Features

A very common feature could be common friends to u_i and u_j which is highly indicative of an edge between $u_i$and u_j. In Figure 1, $u_1$is following $\{u_3, u_4, u_5\}$ and u_2 isfollowing$\{u_3, u_4, u_6\}$. In order to predict whether there could be any potential edge possible from u_1 to u_2, the following steps could be followed as per the common neighbor feature.

Step1: Find out set of vertices or users that u_1 and u_2 follows.

Step 2: Determine the common vertices/users in both the sets.

Step 3: Looking to the common users, prediction can be taken for the users to be recommended as friends or the missing links can be found out.

As because u_1 and u_2 have lots of overlap or many similar users, it shows thatu_1 and u_2 have similar interests. By looking into the behavior pattern of both the users u_2 may have an interest in u_5, u_1 may follow u_6 and they may follow each other.

Remark 3: if u_i is following u_j there is a prime chance that user u_j will start following back. This is called a following back feature. Other graph-based features have been discussed below.

1. Jaccard Similarity

It is based on the common user between u_i and u_j divided by the total number of users associated with u_i and u_j. Mathematically, it is formulated as in Equation (1).

$$Jsim = \frac{|u_i \cap u_j|}{|u_i \cup u_j|} \quad (1)$$

Jaccard distance is computed both for the followers and the followee. The higher the 'jsim' value, there is a higher probability of an edge between u_i and u_j. Implementation of Jaccard similarity is represented in Algorithm 1.

Algorithm 1: Jaccard Similarity-jsim ($\boldsymbol{u_i}$, $\boldsymbol{u_j}$)

Input: g_train, vertices $\boldsymbol{u_i}$ and $\boldsymbol{u_j}$
Output: jsim

1. p ← successor ($\boldsymbol{u_i}$)
2. q ← successor ($\boldsymbol{u_j}$)
3. If len(p) or len(q) == 0
   return 0
4. Jsim←$\frac{|\boldsymbol{p} \cap \boldsymbol{q}|}{|\boldsymbol{p} \cup \boldsymbol{q}|}$
5. return Jsim

2. Cosine distance (Otsuka-Ochiai Coefficient)

It is based on the number of common neighbors between u_i and u_j divided by the square root of multiplication of the number of neighbors of u_i and the number of neighbors ofu_j.

If the Cosine distance value is larger, then there is a larger overlap. The calculation of Ochiai similarity is represented in Equation (2). Algorithm 2 shows the cosine_for_followees.

$$Csim = \frac{|u_i \cap u_j|}{sqrt(|u_i| \times |u_j|)} \quad (2)$$

Algorithm 2: cosine_for_followees

Input: g_train (Train set of the Graph)
Output:cosine distance

1. p ← successor ($\boldsymbol{u_i}$)
2. q ← successor ($\boldsymbol{u_j}$)
3. If len(p) or len(q) == 0
   return 0
4. Csim←$\frac{|\boldsymbol{p} \cap \boldsymbol{q}|}{\boldsymbol{sqrt}(|\boldsymbol{p}| \times |\boldsymbol{q}|)}$
5. **return**Csim

In the same way,cosine_for_followers are found out. The only difference from the algorithm of cosine_for_followees is that instead of successor, predecessors are considered.

3. Ranking Measures(Page Rank)

Page rank is a popular traditional approach, generally followed by Google. A page P_i have larger value if, lots of pages are linking to them, which means P_i must be a trusted source. Also P_i score can be large as other important pages are linking to it. Ultimately, quality and number of links to a page determine the importance of a page. If this is the situation, there is a very high chance that u_i is to be followed by an unknown useru_j.

In this paper, using Page Rank for every pair of vertex, two features are created. It means for (u_i, u_j) pair f_i and f_j are created where f_i is page rank of the user u_i and f_j is Page rank of the user $u_{j,}$ respectively.

4. Shortest Path

The shortest path is calculated by removing an edge between two nodes if exist, i.e. Trivially connected. It is used to determine how far one user is from another.

5. Adamic/ Adar Index

It is used to predict the links in Social networks. Mathematically, it is represented in Equation (3).

$$A(u_i, u_j) = \sum_{u \in N(u_i) \cap N(u_j)} \frac{1}{\log(|N(u)|)} \quad (3)$$

Where u_i and u_j are the users, N represents the neighbors, u represents any random user that is a neighbor of bothu_i and u_j. According to this concept, if the neighbor of u itself is very large, then there is the least chance of friendship between u_i and u_j and vice versa.

6. Katz Centrality

Katz centrality is found out by calculating the influence of the user. In order to do that, the influence of its neighbors is also to be taken for consideration. It is formulated as in Equation (4) for node u_i.

$$K_i = \alpha \sum_j A_{ij} K_j + \beta \quad (4)$$

Where A represents the Adjacency matrix of the graph having eigen values λ. β controls the initial centrality and the value of α is less than 1/ λ_{max}.

7. Adding a new set of features using SVD in the directed graph:

First, the Adjacency matrix is calculated by taking 1.78 million nodes. This matrix is a binary matrix of order 1.78M× 1.78M. As it is a binary matrix, the result is 1, if there exists a directed edge $u_i \rightarrow u_j$, 0 otherwise as in Equation (5). The resultant adjacency matrix becomes a sparse matrix.

$$A_{ij} = \begin{cases} 1, & if\ u_i \rightarrow u_j \\ 0, & otherwise \end{cases} \quad (5)$$

By applying SVD with six components to this matrix, the structure becomes as in Equation (6).

$$A = U \sum V^T \quad (6)$$

Left Singular Matrix U becomes the order of 1.78M× 6 and Right Singular Matrix V^T becomes the order of 6 ×1.78M. $\sum$ matrix is an order of 6 ×6.

8. Weight Features

If the number of followers or incoming weight of a person (u_i) is very high as in case ofcelebrity, there is a very high chance that any two random people follow u_i may not know each other. On the other hand, if the number of followers of a person $u_j$is less, then it has the higher probability that any two random people, that follow u_j know each other because all may belong to a small friend circle. This weight can be calculated for incoming edges to a vertex (in-weight) or outgoing edges (out-weight) from a vertex.

The in-weight is calculated as in Equation (7).

$$w_{in}(u_i) = \frac{1}{\sqrt{1+|L_{in}|}} \quad (7)$$

Where L_{in} = Set of all vertices linking into/ in-degree of u_i.The out-weight is calculated as in Equation (8).

$$w_{out}(u_i) = \frac{1}{\sqrt{1+|L_{out}|}} \quad (8)$$

Where L_{out} = Set of all vertices linking out to u_i. The implementation of weight feature is represented in Algorithm 3.

Algorithm 3: Weight Features

Input: g_train (Train set of the Graph)
Output: weight for source and destination of each link

```
1. for each node i in g_train
        L_in= predecessors(i)
        w_in = 1.0/(sqrt(1+len(L_in)))
        L_out= sucessors(i)
        w_out = 1.0/(sqrt(1+len(L_out)))
2. calculate mean of w_in
3. calculate mean of w_out
```

The features are created for both train and test data points taking weight features. The selected features arew_{in},w_{out}, $w_{in} + w_{out}$, $w_{in} * w_{out}$, $w_{in} + 2 * w_{out}$and $2 * w_{in} + w_{out}$.

D. Proposed Framework

In similarity based model, the input to the model is the adjacency matrix and then similarity score is calculated. Popular similarity score are Adar, Katz and Jaccard. After calculation of similarity score top scored links are recommended. In this work, a classification model is used to predict the missing link. The collected social data that indicates user pair is represented as a graph structure. User-pair (a, b) represents an edge between user 'a' and user 'b'.Available edges are labeled as class 1 where as missing edges are labeled with class 0. The task is to consider all the points to the dataset that does not have an edge between the user pair. This leads to a highly unbalanced dataset. As the number of edges possible in a user graph is O (n^2).

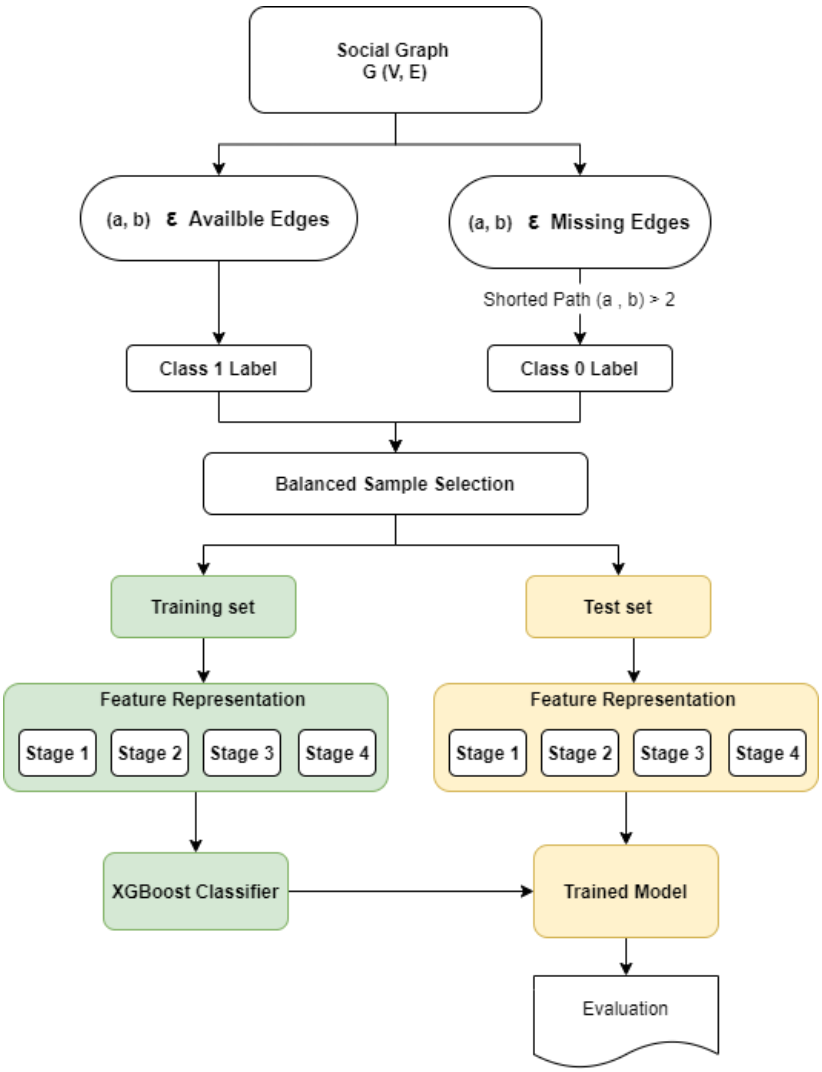


Fig. 2: Proposed Framework for link prediction

In order to deal with this situation, the training data is generated by taking the random sample of all possible edges. Also, the inclusion of those pairs of vertices is avoided that are having the shortest path length <=2.It is seen that 'friend of a friend' (FOAF) is trivial case to be followed hence path length >2 is considered for prediction model. The pre-processed dataset is split into 80: 20 train and test set. Then the feature set X $\{x_1, x_2, \ldots, x_{51}\}$ is constructed by applying different graph-based feature techniques discussed in the previous section. As many features are considered, the basic features are combined with many other features and categorized into four stages. Features in different stages are represented in table 1. Features in different stages are Stage 1$\{x_1, x_2, \ldots, x_{10}\}$, Stage 2 $\{x_1, x_2, \ldots, x_{14}\}$, Stage 3 $\{x_1, x_2, \ldots, x_{28}\}$ and Stage 4$\{x_1, x_2, \ldots, x_{51}\}$. Node based features are called local based feature and graph based features

are called global based feature. We have considered many local and global based features also hybrid features.

Table 1: Features set

Notation	Description
x_1	Follower of node/user by calculating Jacard Distance
x_2	Followees of node/user by calculating Jacard Distance
x_3	Follower of node/user by calculating Cosine Distance
x_4	Followees of node/user by calculating Cosine Distance
x_5	Number of followers of source node
x_6	Number of followers of destination node
x_7	Number of followees of destination node
x_8	Number of followees of destination node
x_9	Common followers of source node
x_{10}	Common followees of destination node
x_{11}	Mapping adar index on train and test
x_{12}	Mapping the user is following back or not on train and test
x_{13}	Mapping same component of weakly connected component(WCC)
x_{14}	shortest path between source and destination
x_{15}	weight of incoming edges
x_{16}	weight of outgoing edges
x_{17}	weight of incoming edges + weight of outgoing edges
x_{18}	weight of incoming edges * weight of outgoing edges
x_{19}	2 * weight of incoming edges + weight of outgoing edges
x_{20}	weight of incoming edges + 2 * weight of outgoing edges
x_{21}	page ranking of source
x_{22}	Page ranking of destination
x_{23}	Katz centrality of source
x_{24}	Katz centrality of destination
x_{25}	Hubs of source
x_{26}	Hubs of destination
x_{27}	Authorities of source
x_{28}	Authorities of destination
$[x_{29...} x_{51}]$	SVD features for both source and destination

After preparation of the feature dataset, XGBoost-based classifier proposed in [23] is used for training and validating the model. The proposed framework is depicted in Figure 2. The XGBoost based model for predicting the link in a social network can be expressed as in Equation (9).

$$\hat{Y}_E = \sum_{e=1}^{E} f_e(X) \tag{9}$$

Where $f_e()$ is the eth tree in the forest and E is the number of estimator. $\hat{Y}_E$ is the predicted link and X is the feature vector.

The objective function shown in Equation (10) is optimized in an additive manner across E number of trees. Finally, the prediction is based on the results of all the trees altogether.

$$Obj = \sum_{i=1}^{M} L\left(Y_i, \hat{Y}_i\right) + \sum_{e=1}^{E} \Omega\left(f_e\right) \tag{10}$$

Where $\Omega\left(f_e\right) = \gamma T + \frac{1}{2}\lambda \parallel w \parallel^2$ is the regularization term. T is the number of leaf node and γ, λ is the weight parameters. L is the sum of square of residuals loss function. M is the number of instances in the train set. The error is minimized in a step-by-step manner by optimizing the objective function as a minimization problem. In iteration *t* objective function used to minimize is represented in Equation (11).

$$Obj^{(t)} = \sum_{i=1}^{M} L\left(Y_i, \hat{Y}_i\right) + f_t(X_i) + \sum_{e=1}^{E} \Omega\left(f_e\right) \tag{11}$$

IV. Experimental Set Up and Result Analysis

A. Data Pre-processing

To study the impact of the model in large dataset facebook data is considered. We have used following pre-processing steps using python libraries: i) loading the dataset into a data frame using the pandas library, ii) removing all the duplicate and unavailability entries and iii) Create a DiGraph using NetworkX library. After the pre-processing step, the information about the graph created is illustrated in Table 2.

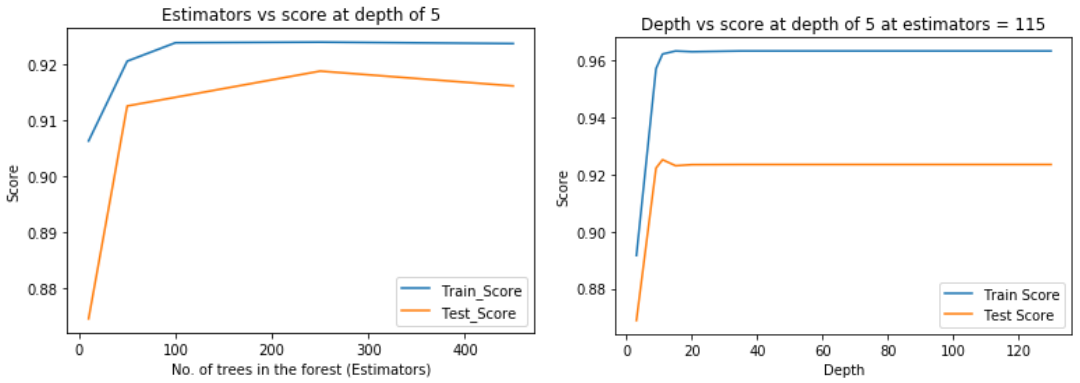
Table 2: Constructed social graph info

Type	Karate	Polblogs	Facebook
Number of Nodes/Users	34	1490	1862220
Number of Edges/Relation	156	19025	9437519
Average in-degree	4.5882	12.7685	5.0679
Average out-degree	4.5882	12.7685	5.0679

B. Results and Discussion

To validate the effectiveness of the proposed work (LPXGB), the model is tested with three real world datasets and is compared with KNN, SVM, logistic regression (LR), Decision Tree (DT) and Random Forest (RF), Gradient Boosting Classifier(GBC) Machine Learning models. Also these Machine Learning models are compared with traditional link prediction models such as Jaccard Similarity (JS), Katz similarity (KS) and Adar Index (AI).

In the KNN classifier, number of neighbors (K value) is set to 5. SVM model is training using a linear kernel. In the RF model with Facebook data initially, no of estimator (#estimator) is calculated by comparing against different score values by setting the depth to 5. The score against no of trees in the forest is depicted in Figure 3(a). From the Figure 3(a) by looking into the score #estimator is taken 115. By setting estimator to 115 impacts of depth is plotted in Figure 3(b).

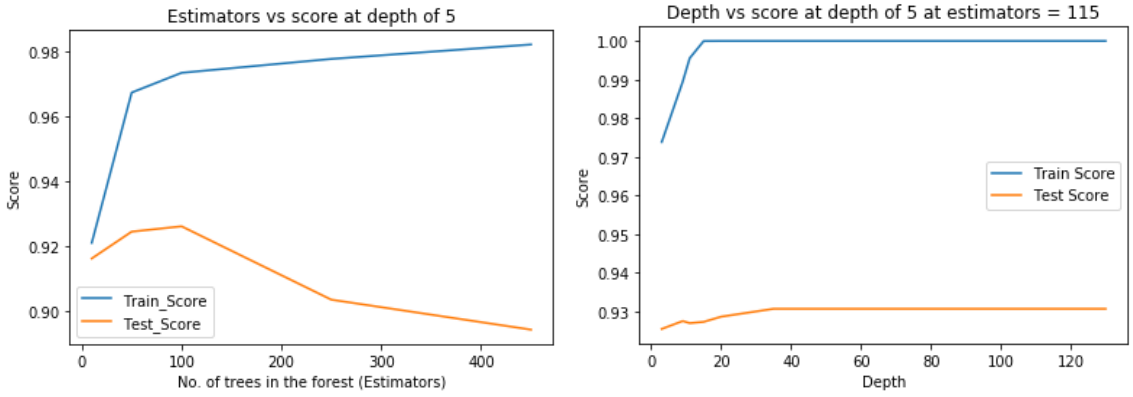


(a) (b)

Fig. 3: Parameter setting in RF (a) #estimator vs. score (b) max depth vs. score

Then by looking into the pattern, the #estimator range is set to 100 to 125 and the max depth range is set to 10 to 15. With the setting of different range value, two hyper parameters #estimator and max depth are fine-tuned by using the Randomized Search cross-validation technique. The final value of #estimator and max depth is set to 121 and 14, respectively.

Similarly, for LPXGB model as depicted in Figure. 4 (a) and (b), #estimator range is set to 90 to 120 and max depth range is set to 10 to 40. The final value of #estimator and max depth is set to 115 and 15, respectively. Other parameters such as learning rate, γ, λ are set to 0.09, 0 and 1 respectively. In GBC model, #estimator, max depth, learning rate are set as of LPXGB model.



(a) (b)

Fig. 4: Parameter setting in LPXGB(a) #estimator vs. score (b) max depth vs. score

Table 3: Test Accuracy score on various feature categories

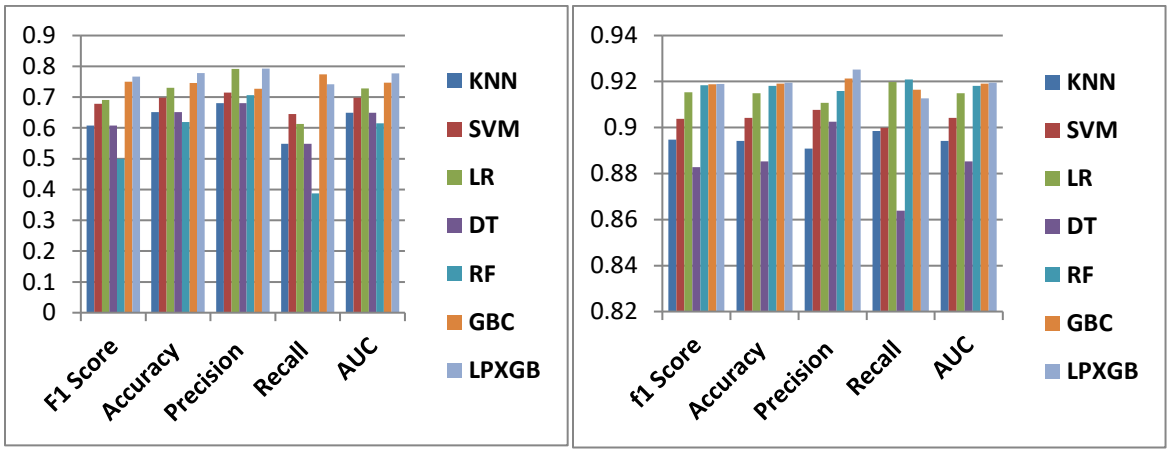
Dataset	Stage	KNN	SVM	LR	DT	RF	GBC	LPXGB
karate	1	0.6667	0.6825	0.5873	0.4921	0.6508	0.6508	0.6508
	2	0.6508	0.6984	0.7619	0.746	0.7619	0.8095	0.8095
	3	0.6349	0.6984	0.7619	0.6349	0.6349	0.7619	0.7619
	4	0.6508	0.6984	0.7302	0.6508	0.619	0.746	**0.7778**
Polblogs	1	0.8915	0.8974	0.8816	0.8689	0.9102	**0.9116**	0.911
	2	0.8943	0.9037	0.8982	0.8832	0.9146	0.9158	**0.9163**
	3	0.8932	0.9039	0.9151	0.8833	0.9162	0.9175	**0.92**
	4	0.8942	0.9042	0.9148	0.8853	0.9181	0.9191	**0.9194**
facebook	1	0.8019	0.8362	0.8304	0.8225	0.8348	0.8334	**0.8349**
	2	0.8776	0.9176	0.9106	0.9186	0.9253	0.921	**0.9248**
	3	0.8833	0.9159	0.9152	0.9219	0.9329	0.9239	**0.9295**
	4	0.8835	0.9156	0.9153	0.9212	0.9279	0.9239	**0.9304**

Table 4: Comparison with traditional link prediction model

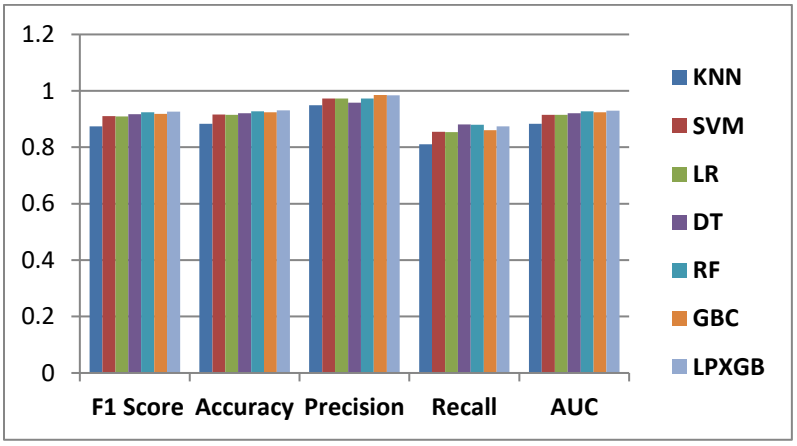
Model	Karate		Polblogs		Facebook	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
JS	0.4921	0.4929	0.7495	0.7495	0.7056	0.7051
AI	0.619	0.6169	0.7344	0.7344	0.673	0.6724
KS	0.5079	0.5	0.5	0.5	0.5009	0.5
LPXGB	**0.7778**	**0.7772**	**0.9194**	**0.9194**	**0.9304**	**0.9303**

Test accuracy score of various models are illustrated in Table 3. Higher value indicates better result. The accuracy score is calculated based on the different stages of features in a forward selection manner. In stage 4, XGBoost-based classifier achieves better result. Performance of LPXGB compared with other link prediction model is shown in Table 4. Proposed model attains best accuracy score and with highest AUC value. Traditional approaches are simple and do not require any training but achieves worst result as compared to machine learning based classifier.

From Figure 5, It is found that the model also achieves impressive precision, recall and AUC value when all the hybrid features are taken in account (stage 4).



(a) (b)



(c)

Fig. 5: Classification metrics at stage 4(a) Karate (b) Polblogs (c) Facebook

Confusion matrixes of facebook data over test data of various models are depicted in Figure 6.

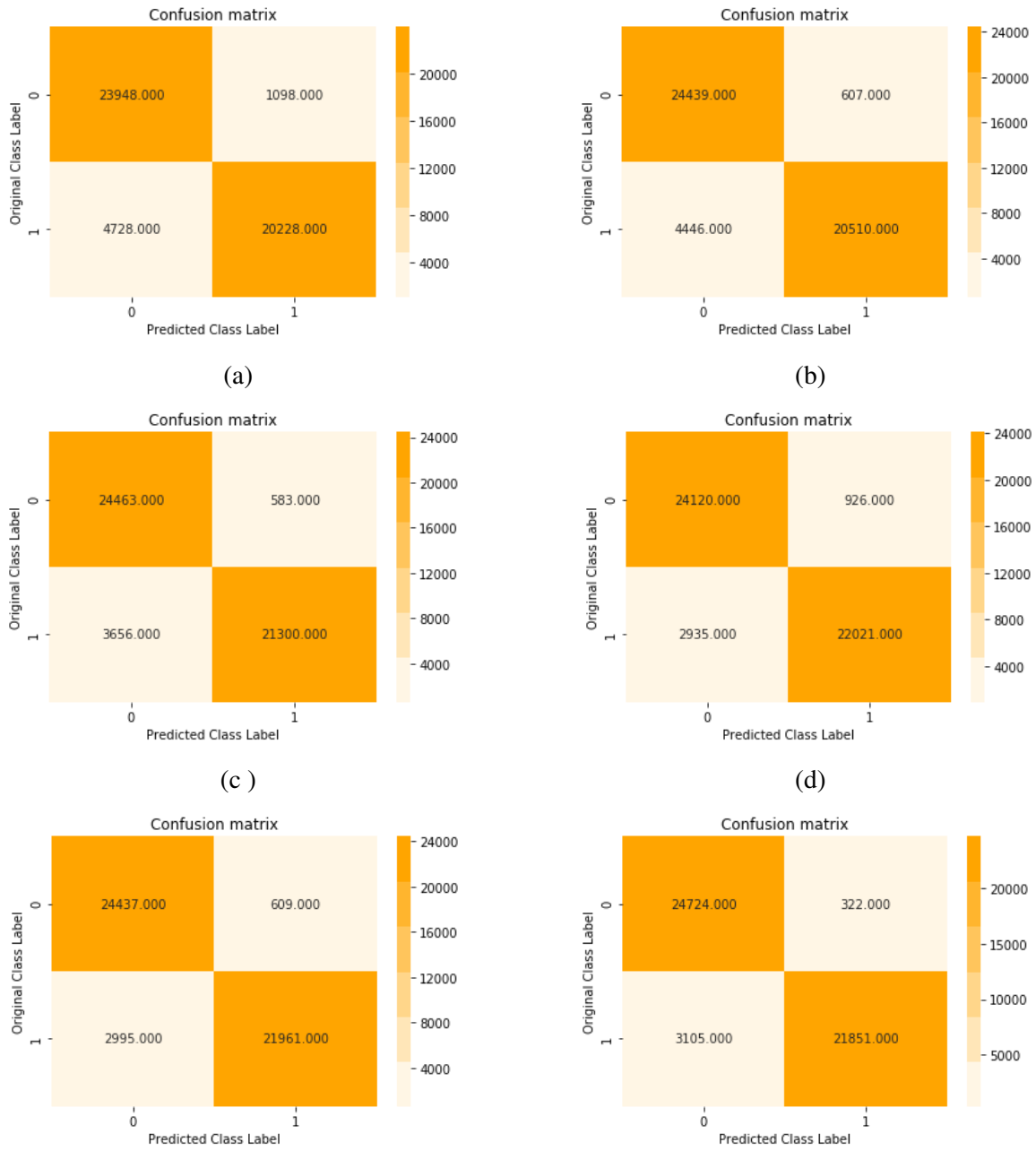
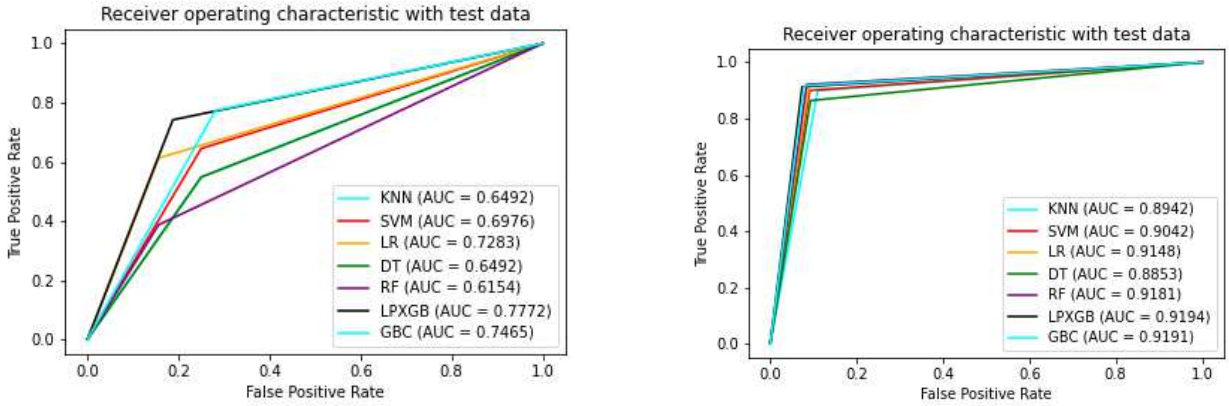
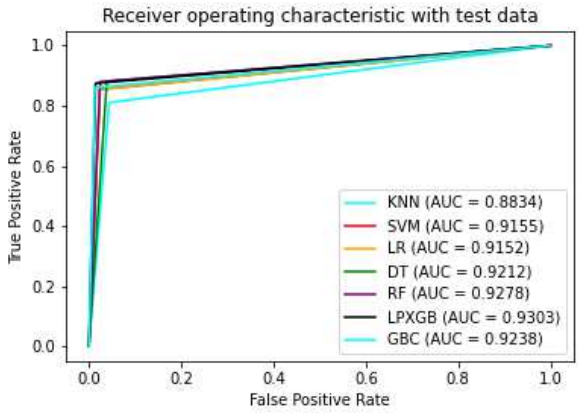


Fig. 6: Facebook data Confusion Matrix (a) KNN (b) SVM (c) LR (d) DT (e) RF (f) LPXGB

It is seen that in the case of the RF-based model in facebook data the Train F1 score is 0.9652 whereas the test score is 0.9241. In the proposed model, the F-Measure of Train and Test set are 0.9996 and 0.9272, respectively. ROC of the test dataset on stage 4 features for all the datasets are represented in Figure. 7. As the proposed model works better even in the large dataset, it can be used in any social networks.



(a) (b)



(c)

Fig. 7: ROC curve (a) Karate (b) Polblogs (c) Facebook

V. Conclusion

Web 2.0 evolution facilitates users of web communities to share information through social networking sites in real time.Link prediction is an emerging approach to discover new potential vertices in a social graph. By exploring new popular users over the web, the reputation of the communityspreadbeyond the group and it also facilitates the collaborative ways of working across users. In this paper, a supervised classification model is proposed to predict the unseen link. To view the problem as a supervised model, the collected dataset is represented in a two-class binary classification problem. Graph representation of the dataset reflects that the number of missing links is much higher than the given link. A shorted path algorithm on randomized

samples is applied to make it balanced. Many local and graph-based featuresare generated and when all the features are considered into account, the ML models exhibit better results as shown in table 3. The proposed XGBoost based classification model achieves betteraccuracy than other learning-based models like KNN, SVM, logistic regression, decision tree, random forest and Gradient boosting classifier. In karate, polblogs and facebook data the AUC value of the proposed models are 0.7772, 0.9194 and 0.9303, respectively.In larger network the model gives impressive results, hence it can be used in any social networks. By the use of link prediction, people with similar needs are identified effectively that helps to explore more web communities in a social network.In future LSTM-based deep learning approach may be used to deal with dynamic changes of the social network.

References

1. Wu, K.S., Chang, P.C.: (2016) Identifying communities and influential node in Facebook fan page - A case study of FJU 2013 ad camp. *Int. J. Web Based Communities*. 12, 376–392. https://doi.org/10.1504/IJWBC.2016.080813.

2. Papadimitriou, A., Symeonidis, P., Manolopoulos, Y.: (2012) Fast and accurate link prediction in social networking systems. *J. Syst. Softw.* 85, 2119–2132. https://doi.org/10.1016/j.jss.2012.04.019.

3. Watson, B., Watson, T., Zheng, J.: (2019) A study of friend recommendations for gaming communities. *Int. J. Web Based Communities*. 15, 292–314.

4. Anandhan, A., Shuib, L., Ismail, M.A., Mujtaba, G.: (2018) Social Media Recommender Systems: Review and Open Research Issues. *IEEE Access*. 6, 15608–15628. https://doi.org/10.1109/ACCESS.2018.2810062.

5. Li, Y., Luo, P., Fan, Z. ping, Chen, K., Liu, J.: (2017) A utility-based link prediction method in social networks. *Eur. J. Oper. Res.* 260, 693–705. https://doi.org/10.1016/j.ejor.2016.12.041.

6. Papadimitriou, A., Symeonidis, P., Manolopoulos, Y.: (2012) Scalable link prediction in social networks based on local graph characteristics. *Proc. 9th Int. Conf. Inf. Technol. ITNG 2012*. 738–743. https://doi.org/10.1109/ITNG.2012.145.

7. Marjan, M., Zaki, N., Mohamed, E.A.: (2018) Link Prediction in Dynamic Social Networks: A Literature Review. *Colloq. Inf. Sci. Technol. Cist*. 2018-Octob, 200–207. https://doi.org/10.1109/CIST.2018.8596511.

8. Kumar, A., Singh, S.S., Singh, K., Biswas, B.: (2020) Link prediction techniques, applications, and performance: A survey. *Phys. A Stat. Mech. its Appl.* 124289. https://doi.org/10.1016/j.physa.2020.124289.

9. Han, S., Xu, Y.: (2016) Link Prediction in Microblog Network Using Supervised Learning with Multiple Features. *J. Comput.* 11, 72–82. https://doi.org/10.17706/jcp.11.1.72-82.

10. Cukierski, W., Hamner, B., Yang, B.: (2011) Graph-based features for supervised link prediction. *Proc. Int. Jt. Conf. Neural Networks*. 1237–1244. https://doi.org/10.1109/IJCNN.2011.6033365.

11. Liben-Nowell, D., Kleinberg, J.: (2007) The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.* 58, 1019–1031. https://doi.org/10.1002/asi.20591.

12. Gupta, A.K., Sardana, N.: (2018) Prediction of missing links in social networks: Feature integration with node neighbour. *Int. J. Web Based Communities*. 14, 38–53. https://doi.org/10.1504/IJWBC.2018.090917.

13. Dong, L., Li, Y., Yin, H., Le, H., Rui, M.: (2013) The algorithm of link prediction on social network. *Math. Probl. Eng.* 2013,. https://doi.org/10.1155/2013/125123.

14. Rahman, M.S., Dey, L.R., Haider, S., Uddin, M.A., Islam, M.: (2017) Link prediction by correlation on social network. In: 20th International Conference of Computer and Information Technology, ICCIT 2017. pp. 1–6. https://doi.org/10.1109/ICCITECHN.2017.8281812.

15. Yadav, R.K., Tripathi, S.P., Rai, A.K., Tewari, R.R.: (2020) Hybrid feature-based approach for recommending friends in social networking systems. *Int. J. Web Based Communities*. 16, 51–71.

16. Dong, Y., Tang, J., Wu, S., Tian, J., Chawla, N. V., Rao, J., Cao, H.: (2012) Link prediction and recommendation across heterogeneous social networks. *Proc. - IEEE Int.*

Conf. Data Mining, ICDM. 181–190. https://doi.org/10.1109/ICDM.2012.140.

17. Lin, S., Liu, C., Zhang, Z.K.: (2017) Multi-Tasking link prediction on coupled networks via the factor graph model. *Proc. IECON 2017 - 43rd Annu. Conf. IEEE Ind. Electron. Soc.* 2017-Janua, 5570–5574. https://doi.org/10.1109/IECON.2017.8216964.

18. Najari, S., Salehi, M., Ranjbar, V., Jalili, M.: (2019) Link prediction in multiplex networks based on interlayer similarity. *Phys. A Stat. Mech. its Appl.* 536, 120978. https://doi.org/10.1016/j.physa.2019.04.214.

19. Sharma, P.K., Rathore, S., Park, J.H.: (2019) Multilevel learning based modeling for link prediction and users' consumption preference in Online Social Networks. *Futur. Gener. Comput. Syst.* 93, 952–961. https://doi.org/10.1016/j.future.2017.08.031.

20. Wu, J., Shen, J., Zhou, B., Zhang, X., Huang, B.: (2019) General link prediction with influential node identification. *Phys. A Stat. Mech. its Appl.* 523, 996–1007. https://doi.org/10.1016/j.physa.2019.04.205.

21. Hao, Z.: (2019) Link Prediction in Online Social Networks Based on the Unsupervised Marginalized Denoising Model. *IEEE Access*. 7, 54133–54143. https://doi.org/10.1109/ACCESS.2019.2912662.

22. Hisano, R.: (2018) Semi-supervised graph embedding approach to dynamic link prediction. *Springer Proc. Complex.* 109–121. https://doi.org/10.1007/978-3-319-73198-8_10.

23. Chen, T., Guestrin, C.: (2016) XGBoost: A scalable tree boosting system. In: Int. Conf. Knowl. Discovery Data Mining. pp. 785–794.

24. Behera, D.K., Das, M., Swetanisha, S., Naik, B.: (2018) Collaborative filtering using restricted boltzmann machine and fuzzy C-means. In: Advances in Intelligent Systems and Computing. pp. 723–731. *Springer Verlag*. https://doi.org/10.1007/978-981-10-7871-2_69.

Figures

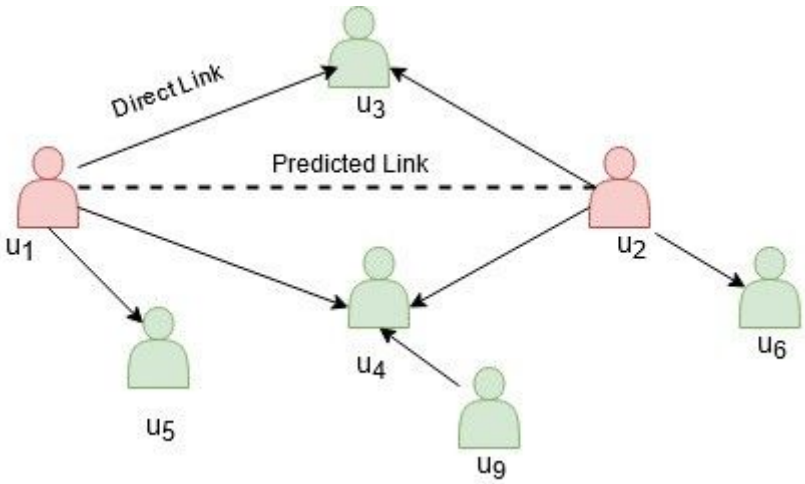


Figure 1

User graph showing direct and predicted links

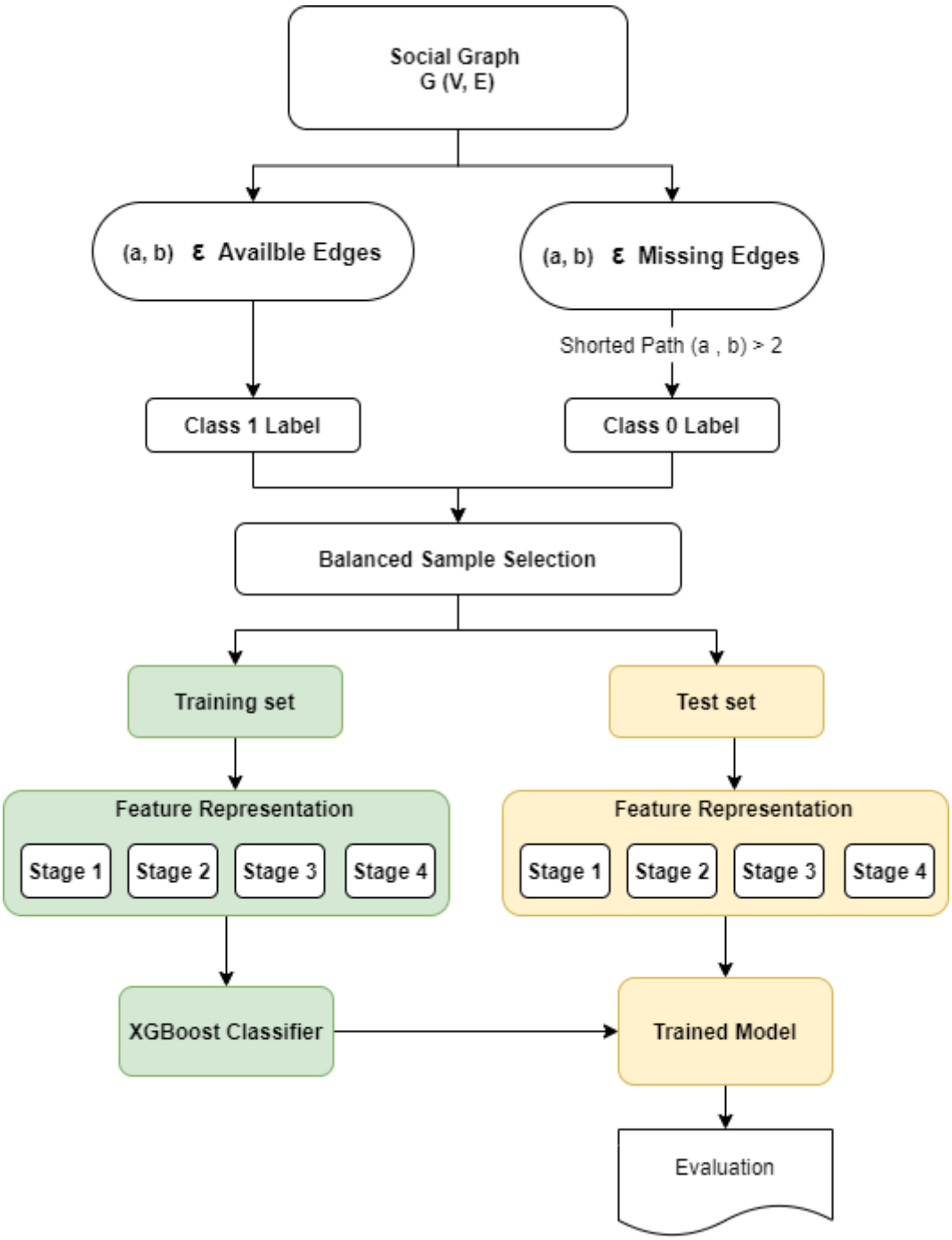


Figure 2

Proposed Framework for link prediction

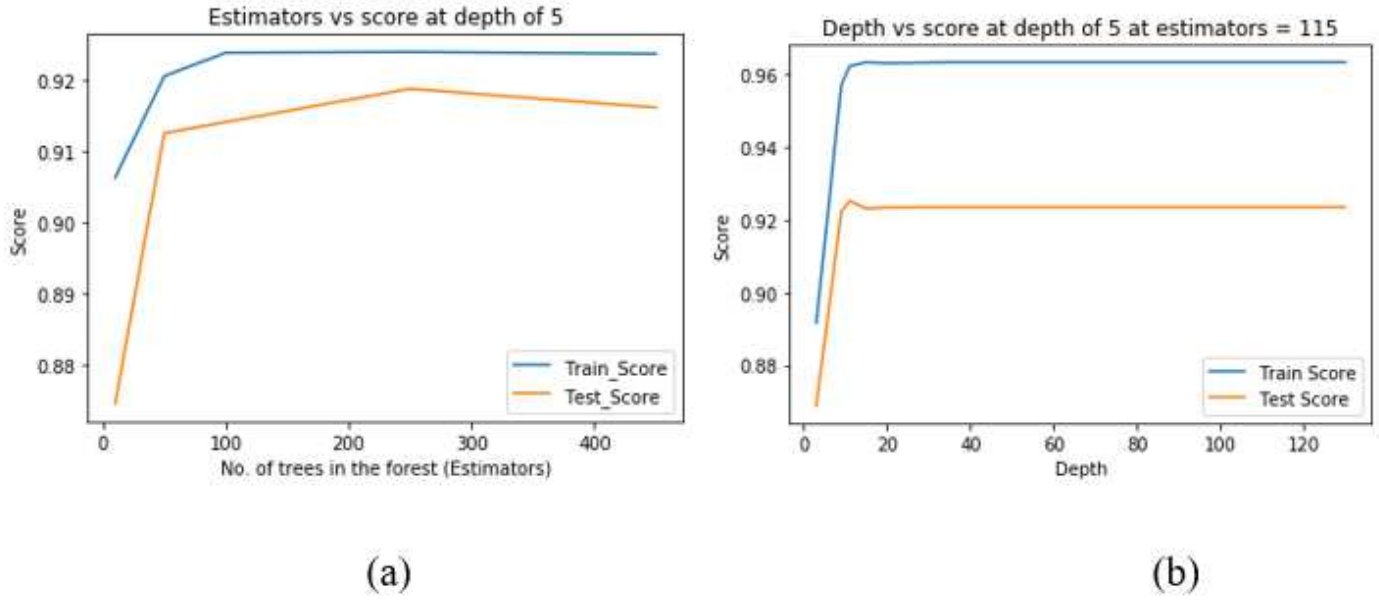


Figure 3

Parameter setting in RF (a) #estimator vs. score (b) max depth vs. score

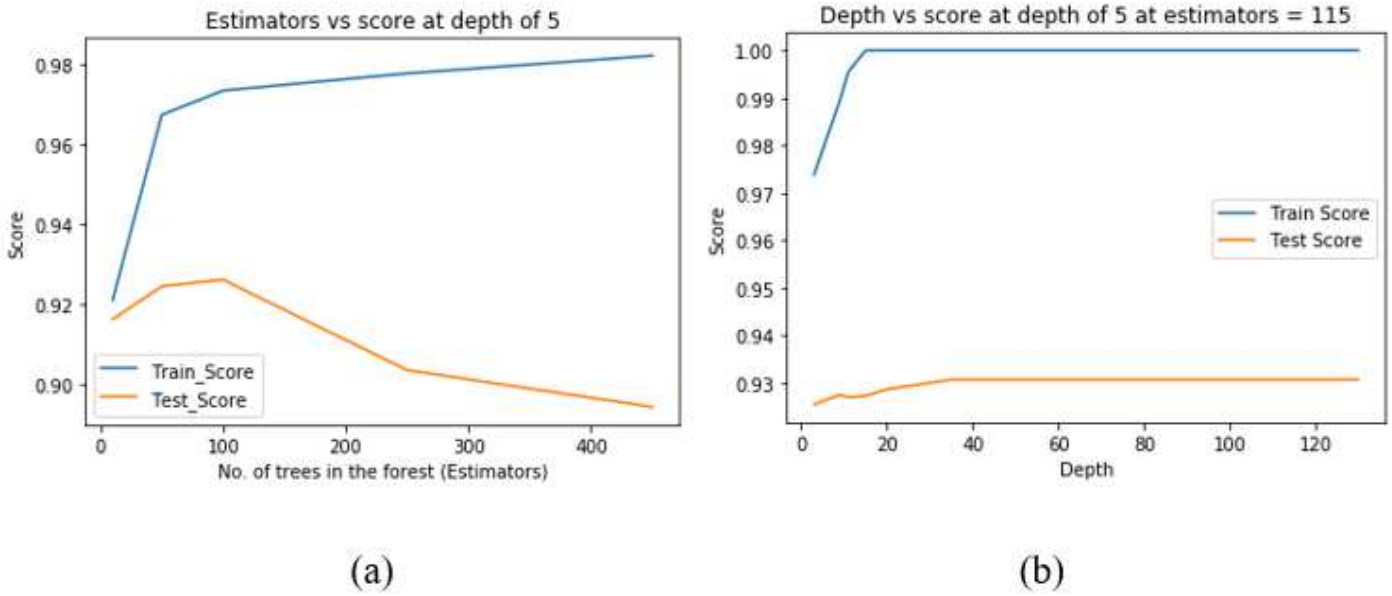
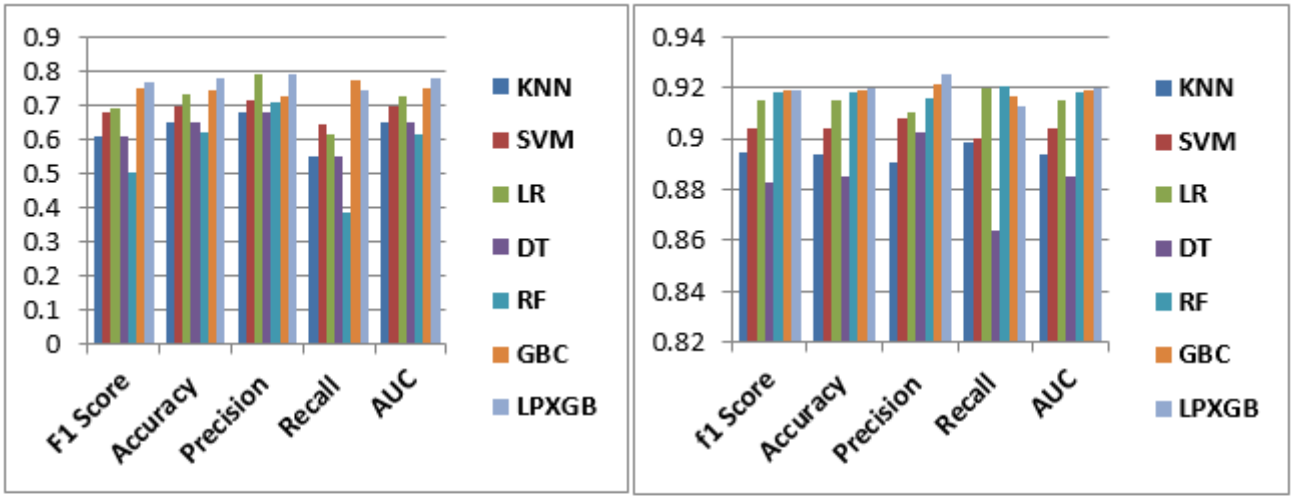
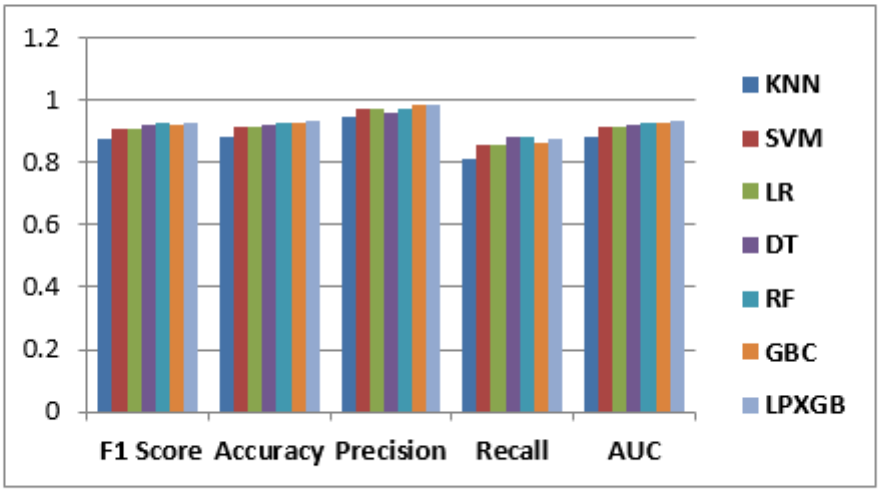


Figure 4

Parameter setting in LPXGB(a) #estimator vs. score (b) max depth vs. score



(a) (b)



(c)

Figure 5

Classification metrics at stage 4(a) Karate (b) Polblogs (c) Facebook Confusion matrixes of facebook data over test data of various models are depicted in Figure 6.

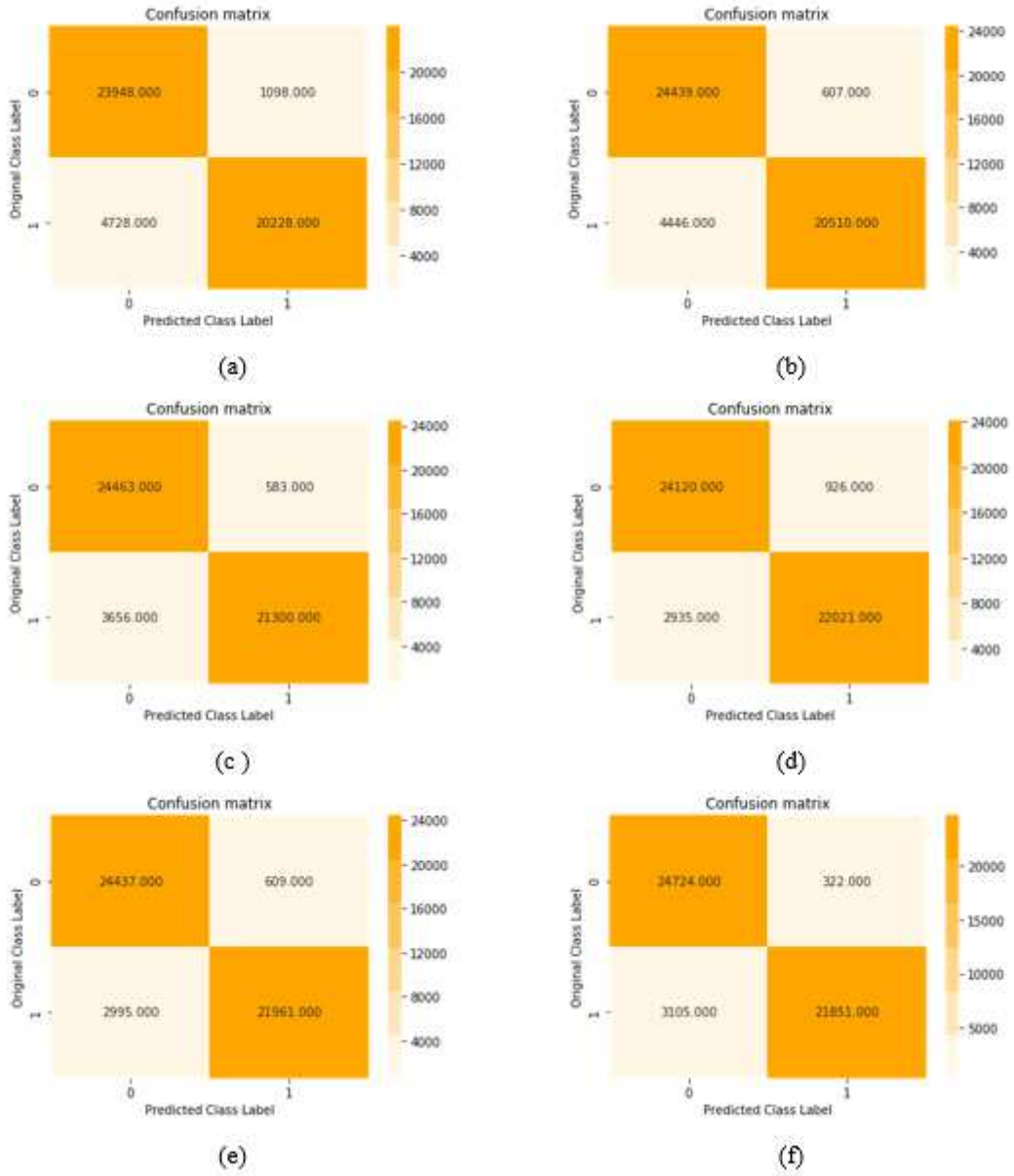
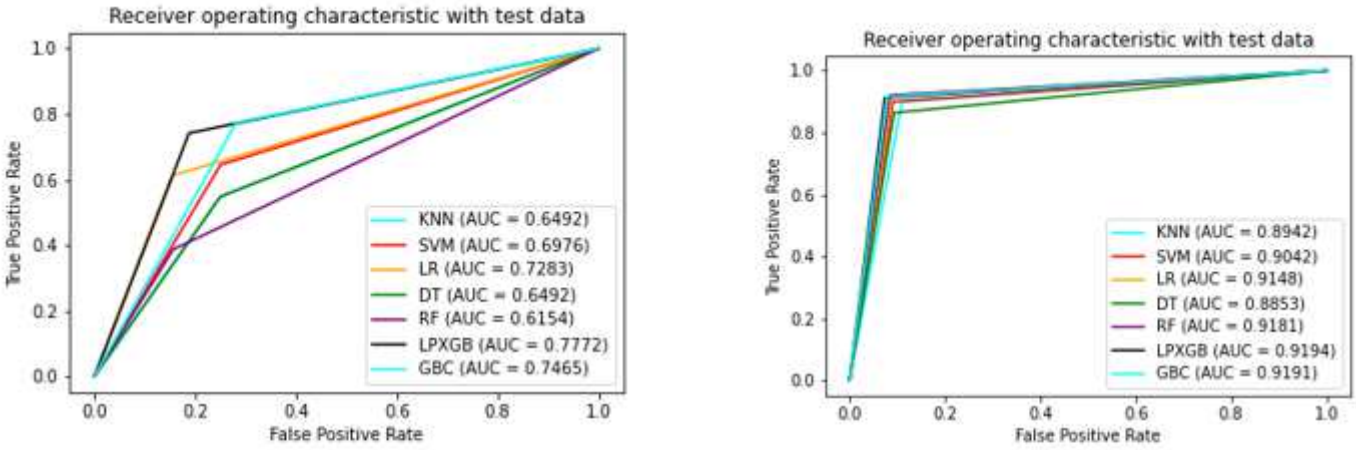
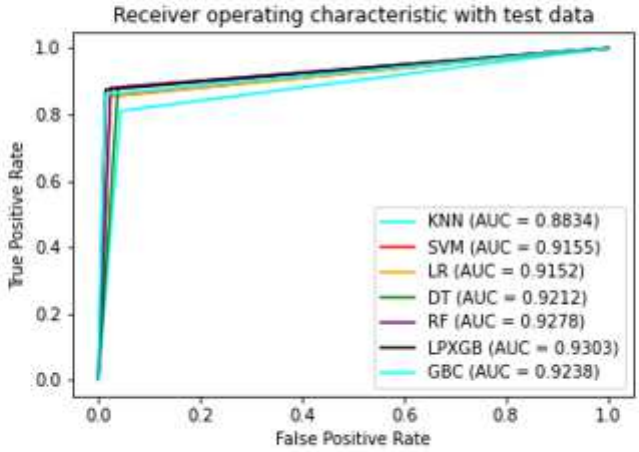


Figure 6

Facebook data Confusion Matrix (a) KNN (b) SVM (c) LR (d) DT (e) RF (f) LPXGB



(a) (b)



(c)

Figure 7

ROC curve (a) Karate (b) Polblogs (c) Facebook