[1]Sachin U. Balvir

[2]Mukesh M. Raghuwanshi

[3]Pradnya S. Borkar

Node2Vec and Machine Learning: A Powerful Duo for Link Prediction in Social Network

Abstract: - Link prediction in social networks is a challenging task that attempts to uncover hidden linkages and forecast future connections. The link prediction problem is addressed in this research article by utilizing the capabilities of Node2Vec and machine learning algorithms. To learn high-dimensional node representations that capture both local and global network structures, the Node2Vec technique is used. Then, in order to forecast potential connections, these node embedding's are put into various machine learning models. Two real-world social network datasets are used to test the suggested methodology, and the findings show a considerable improvement in link prediction accuracy. It achieves a deeper comprehension of the hidden relationships in social networks by fusing the semantic richness of Node2Vec embedding's with the predictive powers of machine learning methods. The results of this study extend link prediction approaches in social networks by revealing hidden ties and providing insightful predictions for upcoming connections. The suggested method indicates the potential for real-world applications in a number of fields, including recommender systems, targeted advertising, and social influence studies.

Keywords: Link Prediction, Node2Vec, Link Analysis, Social Network Analysis, Graph Mining

I. INTRODUCTION

In social networks, link prediction is a critical component of understanding and maximizing the power of social connections for a variety of applications. It also presents fascinating customers for future development. Researchers hope to find hidden connections and foresee the creation of new connections by studying the structure and dynamics of the network. This area of study holds significant importance as it provides insights into social network evolution, enables personalized recommendation systems, facilitates targeted advertising, and helps identify potential collaborations. Link prediction algorithms utilize various techniques such as similarity measures, graph-based algorithms, and machine learning approaches, along with feature extraction from the network. Social networks hold significant significance across various domains due to their ability to foster communication, information dissemination, collaboration, and community formation Platforms like Facebook, Twitter, and LinkedIn have revolutionized how individuals interact in online social networks, enabling cross-border communication, information exchange, and social mobility. Social networks are essential for the distribution of information, enabling targeted marketing, public health initiatives, and anticipating the uptake of technologies. Collaborative networks benefit from social connections, facilitating knowledge sharing, innovation, and collective problem-solving in domains such as scientific research, open-source software development, and crowd funding. Social networks also contribute to recommender systems, enhancing personalized recommendations for products, services, and content.

People make links to convey friendliness, support, or approval during social interactions on the Web, but they can also link to express disapproval of others or hostility to convey disapproval or mistrust of another's viewpoint. The vast majority of online social network research has solely taken into account positive ties, despite the fact that the interaction of good and negative interactions is obviously crucial in many social network contexts [1]. The author uses information from friendship connections, community memberships, co-authorship, and conference publications to examine the connection between social networks and user communities. They discover that the underlying network structure affects community development and engagement [2][3].

Link prediction describes the assignment of predicting future links in a network depending on the nodes' current connections. In other words, it involves predicting whether a link will exist or be formed between two entities in a network. In the context of social media, link prediction focuses on predicting future connections or relationships between users. These connections can include friendships, followership, collaborations, or interactions between

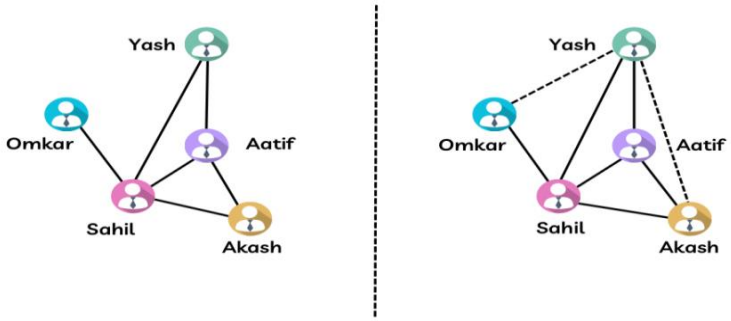
[1] [1]Research Scholar, Department of Computer Technology, Yeshwantrao Chavan College of Engineering, Nagpur, India

[2]Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International (Deemed University), Maharashtra, Pune, India

[3] Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International (Deemed University), Maharashtra, Pune, India

*Corresponding Author Email: sachin_balvir@yahoo.com

users on social media platforms. Link prediction algorithms analyze the structural characteristics of the network, as well as various attributes of the nodes and their relationships, to make predictions. These algorithms leverage machine learning techniques and statistical models to learn patterns and infer the possibility that two nodes may establish a link. The challenge of foreseeing upcoming or absent links between individuals or entities within a social network is known as the "problem of link prediction" in social networks.



(a) (b)

Figure 1. (a) Network structure at time "t0 and (b) Network structure at time "t1"

Figure 1 shows a social network with nodes (people) and edges (connections or relationships) connecting them. Initially, the diagram may have some existing connections which are represented in network at time t0, but there are also some missing links that we want to predict which are represented in network at time t1. Network at time t0 shows the connection between five nodes and at time t1 shows the possibility of future connection between Yash to Akash and Yash to Omkar. Link prediction algorithms aim to predict these missing links by analyzing the patterns and characteristics of the existing network. They utilize various techniques to estimate the likelihood or probability of a connection between node pairs that are not yet linked. By leveraging the available information in the network diagram, link prediction algorithms help uncover hidden relationships and contribute to a deeper understanding of social networks.

This paper is broken up into several pieces. The introduction provides background information about the link prediction and its importance in social networks. In second section we have given the literature review which provides a thorough examination of the available research. We discuss methodologies in third section includes the data collection, feature extraction and model description. Fourth section is for result and discussion, where we conducted experiments using a dataset and analyzed the results, including the best AUC, best ACC, best parameter, and best score. The paper concludes with a conclusion, followed by a list of references.

II. RELATED WORK

Social networking sites are crucial for tasks including information transmission, event prediction, and relationship prediction [4]. This method demonstrates how Anchor Link Prediction (ALP) in social networks is essential for inter-network applications. This approach indicates how the author discussed many approaches to link prediction that are based on learning, with a particular emphasis on supervised, unsupervised, and semi-supervised learning-based methods [5].The seven most prominent link prediction algorithms were used by the author [2] to compare the Link Prediction strategies that are currently in use on three real-world social networks. Proposed [6][4] the supervised link prediction approach with structural features i.e. topological features and features generated using node2vec. The theories propose that people often arrange their social interactions along five concentric circles of escalating intimacy. In order to anticipate new ties, they hypothesise that relationships between individuals in various circles have varying weights. Using this strategy, the author [7] discussed a social network that is graph-visualizable and that alters when new edges and nodes are introduced throughout time. People can be viewed as nodes in this graph and the connections between those nodes can be thought of as edges. In another approach the author [8] represent a Link prediction as a task in social network analysis that involves predicting the link potential using the network structure and known information. By using this method, we can learn how important link prediction is for users to be able to recommend connections to other users in online social networks.

As a path between two vertices can be rather frequent, the most popular method these days is to find the number and length of existing paths from one vertex to the other. Three categories are offered by the general path length for the area units of the similarity-based approach [9][30]. In order to determine how similar they are, the disc's inherent similarity-based algorithms only consider the direct relationships of the tray of vertices [10]. This method [11] demonstrates how the size of node coverage affects the link prediction performance of learning-based methods, which beat heuristic-based methods on five real-world dataset. The role of weights in complex networks is typically disregarded when link prediction is explored in unweighted networks [16]. Most studies of link prediction use unweighted networks, ignoring the significance of weights in complex networks [12]. Social media platforms play a significant role in bringing people together. Prediction of links and signs in complex networks is very helpful for decision-making and recommender systems, for example, in forecasting potential linkages or relative status levels [13][14]. This method demonstrates how social-based online network systems are quickly growing, leading to a significant data build-up. We encounter additional challenges while attempting to analyse the vast amount of data that has been gathered in these systems [20]. Traditional similarity-based algorithms have given way to embedding-based algorithms in link prediction methods. By suggesting potential associations in these networks, this method demonstrates how a link prediction has a significant impact on Facebook, Twitter, LinkedIn, and Koo, among other social networks [15]. This social network can be seen as a graph in which new edges and nodes are continuously added. Users are nodes in this graph, and the edges in this graph indicate the connections between these nodes (User) [16]. This approach shows how to predict interlayer links in multiplex social networks, an area that is still quite young. This method uses deep neural network-based link prediction architecture for a social network's friend recommendation system [12].

Modern normalized and unnormalized link prediction algorithms were examined by the author using five different datasets to see which technique outperformed the others. Their findings demonstrate that normalized similarity outperforms unnormalized similarity in link prediction [17]. The author reviews various node embedding techniques and design ideas for network representation learning over homogenous networks [18]. While several link prediction algorithms have been presented for single-layer social networks, there is currently little study on predicting interlayer ties in multiplex social networks [19][20]. The strategy is to approach the current link prediction methods from the perspective of nodes in order to identify the micro-level node evolution that generates different edges. We also suggest the edge generation coefficient to measure how well a link prediction approach can account for an edge's generation [21]. We find a node mapping to a low-dimensional space of features with node2vec that optimizes the likelihood of node network neighborhoods being maintained [6]. We investigate the relationship between these communities' evolution and many factors, such as the underlying social network's structure [2]. We have suggested that due to their simplicity, normalized techniques can outperform unnormalized ones [22]. We discover that, when applying models that generalize over this wide range of sites, the signals of linkages in the underlying social networks can be predicted with great accuracy [23].

Different link prediction algorithms have strengths and weaknesses. Structural-based algorithms excel in capturing connectivity patterns but struggle with sparse or noisy networks[5]. Attribute-based algorithms consider node attributes and are useful when they influence link formation, but they may face challenges with incomplete attribute data[4]. Machine learning algorithms capture complex patterns but depend on labeled data and feature quality. Graph-based algorithms capture network properties but may have scalability issues. Temporal models improve link prediction accuracy but require precise temporal information and may struggle with missing or sparse data. Choosing the right algorithm should align with network characteristics and research objectives, and combining techniques may enhance prediction accuracy. The existing literature on link prediction in social networks has made significant strides in understanding and predicting social connections. However, several gaps and limitations persist, which my research aims to address. By combining Node2Vec, a powerful network embedding technique, with machine learning, the paper proposes a robust approach for link prediction in social networks. The method leverages the structural properties of the network and effectively captures the underlying connectivity patterns. Including machine learning techniques enhances accuracy of prediction by leveraging the power of labeled training data.

III. METHOD

3.1 Data Collection:

We utilized two datasets for our project obtained from https://networkrepository.com. The details of each dataset are as follows:

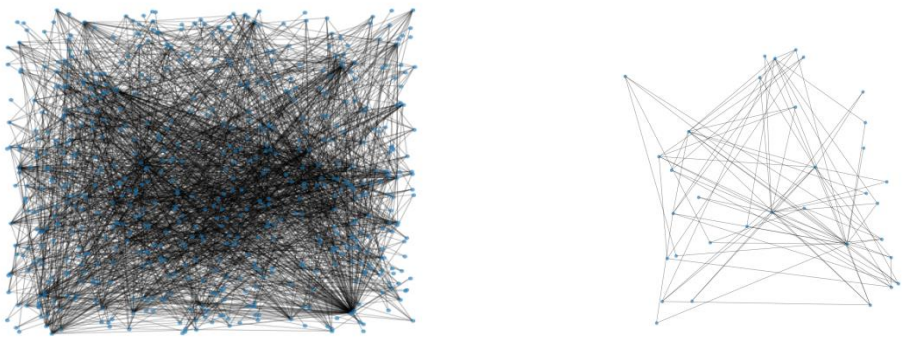
- *Dataset*: fb-pages-food (https://networkrepository.com/fb-pages-food.php)
  Description: This graph dataset consists of Facebook pages representing renowned restaurants and chefs worldwide. Nodes in the graph correspond to these pages, and connections (edges) exist between nodes that are similar to each other.
- *Dataset*: soc-karate's network dataset (https://networkrepository.com/fb-pages-food.php)
  Description: This dataset contains social connections among members of a university karate club, originally collected by Wayne Zachary in 1977.

  By utilizing these datasets, we aimed to analyze and study link prediction in the context of these social networks. The statistics of these network dataset is given in table 1 as below.These statistics provide insights into the size, connectivity, density, and other properties of the network structures within the fb-pages-food and soc-karate's datasets.

Table 1. Network statistics of dataset used for experimentation (https://networkrepository.com)

Network Data Statistics	***fb-pages-food***	**soc-karate's**
Nodes	620	34
Edges	2.1k	78
Density	0.0108969	0.139037
Maximum degree	132	17
Minimum degree	1	1
Average degree	6	4
Number of triangles	8.8k	135
Average number of triangles	14	3
Maximum number of triangles	461	18
Average clustering coefficient	0.330897	0.570638
Fraction of closed triangles	0.222641	0.255682
Maximum k-core	12	5

The representation of the fb-pages-food and soc-karate networks in the form of a graph is given in figure 2.



(a) (b)

Figure 2. (a) Network structure of fb-pages-food dataset, (b) Network structure of soc-karate dataset

3.2 Feature Extraction:

Node2Vec is an algorithm used for learning node embeddings, or vector representations, in a graph. It focuses on capturing structural information and similarities between nodes. Here's a simplified explanation of how Node2Vec works:

- *Random Walks*: By randomly traversing the graph and choosing neighbours at each step, Node2Vec creates random walks on the graph, which are collections of nodes. Depending on the properties of the graph, one can change the number and length of random walks.
- *Context Sampling*: Node2Vec builds pairs of nodes that appear close to one another from the random walks. Two factors, the "return parameter" (p) and the "in-out parameter" (q), govern this proximity. These variables affect the chance of returning to nodes and the ratio of local to global exploration.
- *Embedding Learning*: Similar to Word2Vec, Node2Vec uses a skip-gram model to learn node embedding's from the generated node pairs. Given a target node in the embedding space, the objective is to forecast the context nodes. Node2Vec develops low-dimensional vector representations for each node by optimizing this target.
- *Dimensionality Reduction*: Optionally, dimensionality reduction techniques like PCA or t-SNE can be applied to further reduce the dimensionality of the embedding's while preserving important structural information.
- *Node Similarity and Applications*: The resulting node embedding's capture the structural characteristics of the graph, making it possible to perform tasks like link prediction, node categorization, or clustering. Using similarity metrics in the embedding space, nodes with comparable roles or characteristics can be found.

Node2Vec is a scalable and adaptable method for learning node embedding's that works with different kinds of graphs. It makes use of context sampling and random walks to collect both local and global structural information, making it easier to create accurate representations for subsequent graph analysis tasks.

3.3 Model Selection:

The selected models for machine learning include:

- *Logistic Regression*: This statistical model is commonly used for predictive analytics and categorization. It makes predictions about how likely it is that an event will occur using independent factors. Logistic regression is a supervised learning technique frequently applied to forecast categorical dependent variables [15][16][24].
- *Random Forest*: The well-known machine learning method Random Forest was created by Adele Cutler and Leo Breiman. It creates an overall outcome by combining predictions from several decision trees. It is effective for solving classification and regression problems due to its flexibility and simplicity. Ensemble learning is a technique used by Random Forest to solve complicated problems and enhance overall model performance by combining many classifiers [25][26][27].
- *XgBoost*: XgBoost is an algorithm that leverages GPU acceleration, fast parallel prefix sum operations, and parallel radix sorting to efficiently evaluate potential splits and partition data. It constructs decision trees one level at a time for each boosting iteration while simultaneously processing all data on the GPU. XgBoost has gained recognition for its exceptional performance in machine learning hackathons and contests, tackling a wide range of tasks from high energy physics event identification to ad click-through rate prediction. It is a popular choice for ML hackathons[28][29].

These models have been selected based on their effectiveness and performance in various machine learning applications.

3.4 Evaluation Metrics:

We have used the following evaluation metrics for model evaluation.

- *AUC (Area Under the Receiver Operating Characteristic Curve)*: By altering the prediction link threshold, it assesses a classifier's capacity to discern between positive and negative linkages. It offers a thorough evaluation of the model's functionality.
- *Accuracy:* It takes into account both true positives and true negatives to determine how accurate the connection predictions are overall.
- *Best Score and Best Parameter*: These are not evaluation metrics themselves but rather indicators of model performance. The term "best score" refers to the highest performance achieved by a model during hyperparameter tuning or model selection, which could be the highest AUC or Accuracy value attained during the optimization process."Best parameter" refers to the set of hyperparameters that optimize the performance metric and enhance the model's predictive capability.

By using above evaluation metrics, such as AUC and Accuracy, and finding the best score and best parameter, link prediction models can be assessed and optimized for better predictive accuracy and performance.

3.5 Experimental Setup:

Several essential elements are usually included in the experimental setup for link prediction. Here's an outline of the typical steps involved is shown in figure 3. It illustrates the flow of data from collection to preprocessing, feature engineering, and data partitioning.

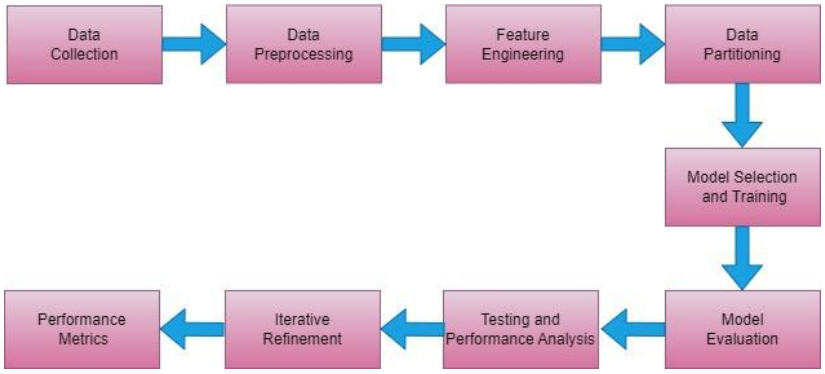


Figure 3. Key components in the process of link prediction

IV. RESULTS AND DISCUSSION

Table 2. Evaluations Metrics of models performed on fb-pages-food dataset

	Training dataset: 90% Testing dataset: 10%			Training dataset: 80% Testing dataset: 20%			Training dataset: 70% Testing dataset: 30%		
Model	**LR**	**RF**	**XGBoost**	**LR**	**RF**	**XGBoost**	**LR**	**RF**	**XGBoost**
Test AUC	0.799	0.945	1	0.817	0.942	0	0.821	0.945	1
Train AUC	0.826	1	1	0.823	1	1	0.821	1	1
Test Accuracy	0.742	0.943	0.074	0.745	0.945	0.926	0.0.743	0.943	0.074
Train Accuracy	0.731	1	0.072	0.736	1	0.928	0.736	1	0.072
Best Score	0.805093	0.933262	0.943784	0.801349	0.932287	0.939099	0.794058	0.933262	0.943784
Best Parameter	{'C': 0.30000000000000004}	{'n_estimators': 250, 'max_depth': 25}	{'n_estimators': 500, 'max_depth': 10}	{'C': 0.5}	{'n_estimators': 250, 'max_depth': 25}	'n_estimators': 500, 'max_depth': 15}	{'C': 0.3000000000000004}	{'n_estimators': 250, 'max_depth': 25}	{'n_estimators': 500, 'max_depth': 10}

Table 2 consists of the training and testing datasets, as well as the performance metrics (Test AUC, Train AUC, Test Accuracy, and Train Accuracy) for three different models (Logistic Regression, Random Forest, and XGBoost) on each dataset split. Additionally, the best scores and corresponding parameters for each model are included.

The Logistic Regression model scored Test AUC of 0.799 and Train AUC of 0.826 for the first dataset split (90% training, 10% testing). With a Test AUC of 0.945 and a Train AUC of 1, the Random Forest model performed better. The XGBoost model attained Test and Train AUCs of 1 in perfect harmony. The accuracy of Logistic Regression was 0.742 for the Test Accuracy and 0.731 for the Train Accuracy. With a Test Accuracy of 0.943 and

a Train Accuracy of 1, the Random Forest model was more accurate. Test Accuracy was 0.074 for the XGBoost model, and Train Accuracy was 0.072.

The Logistic Regression model produced a Test AUC of 0.817 and a Train AUC of 0.823 for the second dataset split (80% training, 20% testing). With a Test AUC of 0.942 and a Train AUC of 1, the Random Forest model fared marginally better. Test AUC was 0 and Train AUC was 1 for the XGBoost model. The accuracy of Logistic Regression was 0.745 for the Test Accuracy and 0.736 for the Train Accuracy. With a Test Accuracy of 0.945 and a Train Accuracy of 1, the Random Forest model was more accurate. A test accuracy of 0.926 and a train accuracy of 0.928 were attained by the XGBoost model.

The Logistic Regression model produced a Test AUC of 0.821 and a Train AUC of 0.821 for the third dataset split (70% training, 30% testing). With a Test AUC of 0.945 and a Train AUC of 1, the Random Forest model fared marginally better. The XGBoost model attained Test and Train AUCs of 1 in perfect harmony. The accuracy of Logistic Regression was 0.743 in the Test Accuracy and 0.736 in the Train Accuracy. With a Test Accuracy of 0.943 and a Train Accuracy of 1, the Random Forest model was more accurate. Test Accuracy was 0.074 for the XGBoost model, and Train Accuracy was 0.072.

Table 3. Evaluations Metrics of models performed on soc-karate dataset

	Training dataset: 90% Testing dataset: 10%			Training dataset: 80% Testing dataset: 20%			Training dataset: 70% Testing dataset: 30%		
Model	**LR**	**RF**	**XGBoost**	**LR**	**RF**	**XGBoost**	**LR**	**RF**	**XGBoost**
Test AUC	0.636	0.227	0.227	0.575	0.136	0	0.547	0.278	0.000
Train AUC	0.654	0.775	0.775	0.688	0.842	0.842	0.662	0.862	0.862
Test Accuracy	0.538	0.231	0.231	0.56	0.24	0.76	0.558	0.37	0.676
Train Accuracy	0.600	0.673	0.673	0.633	0.724	0.602	0.731	0.744	0.616
Best Score	0.466041	0.359570	0.417048	0.436675	0.390382	0.403964	0.512988	0.423420	0.444459
Best Parameter	{'C': 3}	{'n_estimators': 75, 'max_depth': 3}	{'n_estimators': 5, 'max_depth': 1}	{'C': 0.5}	{'n_estimators': 500, 'max_depth': 3}	{'n_estimators': 5, 'max_depth': 1}	{'C': 5}	'n_estimators': 100, 'max_depth': 3}	{'n_estimators': 100, 'max_depth': 1}

Table 3 includes the training and testing datasets, as well as the performance metrics (Test AUC, Train AUC, Test Accuracy, and Train Accuracy) for three different models (Logistic Regression, Random Forest, and XGBoost) on each dataset split. Additionally, the best scores and corresponding parameters for each model are included.

The Logistic Regression model produced a Test AUC of 0.636 and a Train AUC of 0.654 for the first dataset split (90% training, 10% testing). The test and train AUCs for the Random Forest model were 0.227 and 0.775, respectively. The Test AUC and Train AUC for the XGBoost model, respectively, were 0.227 and 0.775. The accuracy of Logistic Regression was 0.538 for the Test Accuracy and 0.600 for the Train Accuracy. The test accuracy and train accuracy of the Random Forest model were 0.231 and 0.673, respectively. The test accuracy and train accuracy for the XGBoost model were both 0.231 and 0.673, respectively. The Logistic Regression model scored the highest for this split, with a score of 0.466040904.

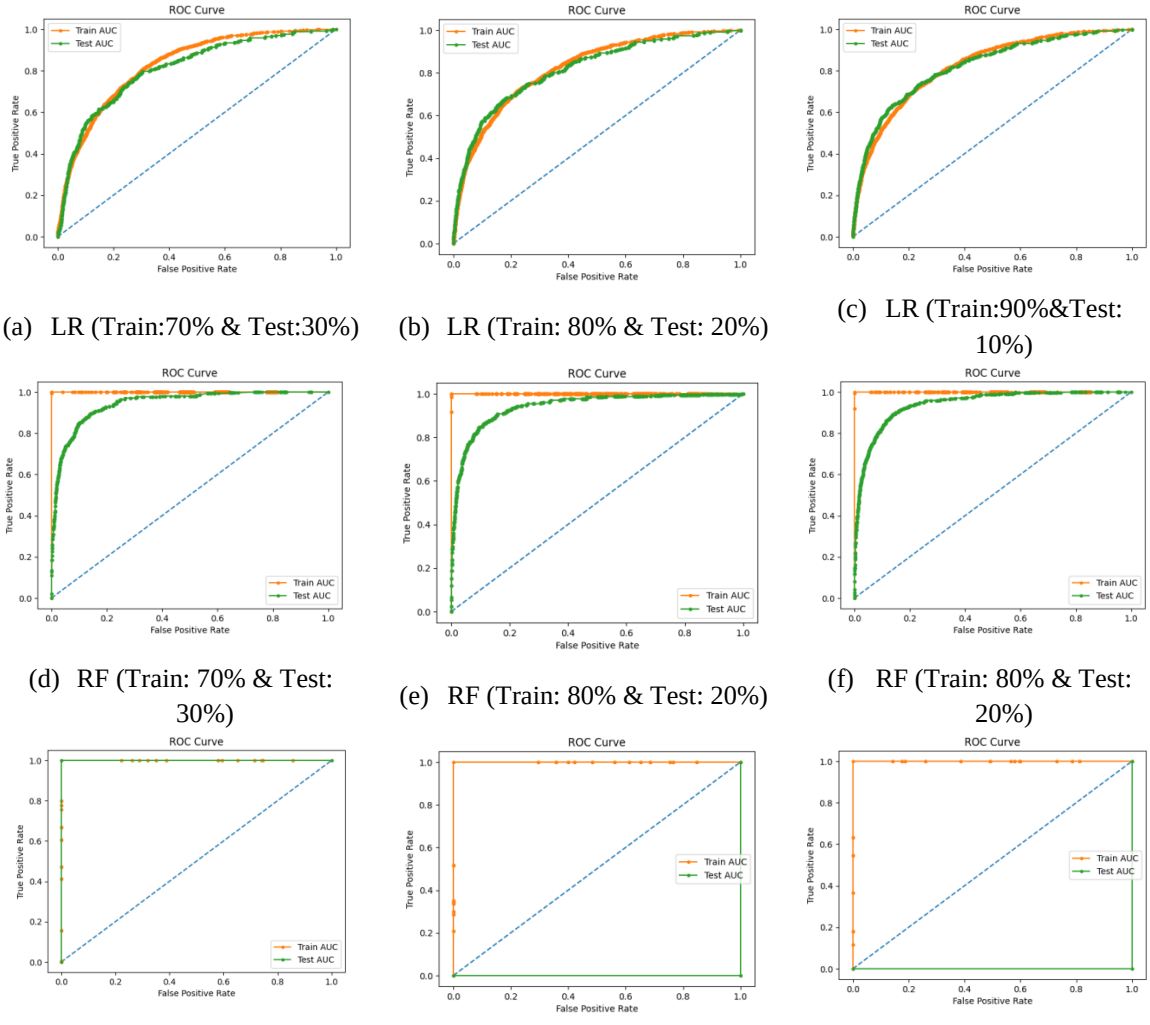
The Logistic Regression model produced a Test AUC of 0.575 and a Train AUC of 0.688 for the second dataset split (80% training, 20% testing). The test and train AUCs for the Random Forest model were 0.136 and 0.842, respectively. The test and train AUCs for the XGBoost model were both zero. The accuracy of Logistic Regression was 0.560 for the Test Accuracy and 0.633 for the Train Accuracy. The test accuracy and train

accuracy of the Random Forest model were 0.240 and 0.724, respectively. Test Accuracy and Train Accuracy for the XGBoost model were 0.760 and 0.602, respectively. The Random Forest model scored the highest for this split, with a score of 0.390381493506493.

The Logistic Regression model produced a Test AUC of 0.547 and a Train AUC of 0.662 for the third dataset split (70% training, 30% testing). The test and train AUCs for the Random Forest model were 0.862 and 0.370 respectively. The Test AUC and Train AUC for the XGBoost model, respectively, were 0.862 and 0.676. The accuracy of Logistic Regression was 0.558 for the Test Accuracy and 0.731 for the Train Accuracy. The test accuracy and train accuracy of the Random Forest model were 0.744 and 0.423419913419913, respectively. The test accuracy and train accuracy of the XGBoost model, respectively, were 0.616 and 0.444458874458874. The Logistic Regression model had the highest score for this split, 0.5129870129870129.

Overall, different dataset splits produced different results from the models. The Random Forest model generally achieved the highest AUC scores, while the Logistic Regression model had the highest accuracy in some cases. When compared to the other two models, the XGBoost model performed worse in terms of accuracy and AUC. The best parameters for each model varied across the splits.

Following Graph shows ROC curve of fb-pages-food dataset:



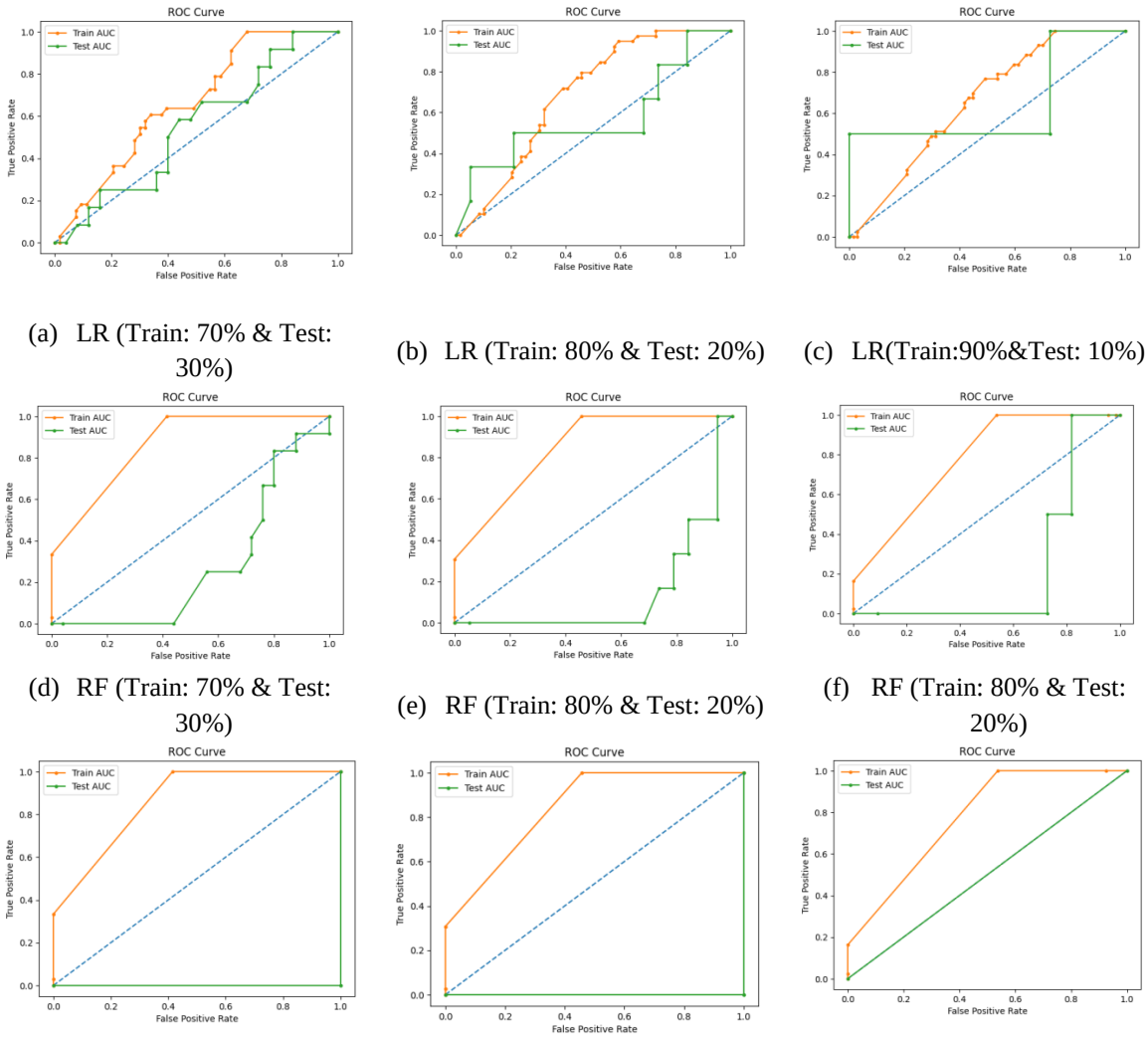
(a) LR (Train:70% & Test:30%) (b) LR (Train: 80% & Test: 20%) (c) LR (Train:90%&Test: 10%)

(d) RF (Train: 70% & Test: 30%) (e) RF (Train: 80% & Test: 20%) (f) RF (Train: 80% & Test: 20%)

(g) Xgboost (Train: 80% & Test: 20%) (h) Xgboost (Train: 80% & Test:20%) (i) Xgboost (Train:90%& Test:10%)

Figure 4. ROC curve of fb-pages-food dataset

Following Graph shows ROC curve of soc-karate dataset:



(a) LR (Train: 70% & Test: 30%)

(b) LR (Train: 80% & Test: 20%)

(c) LR(Train:90%&Test: 10%)

(d) RF (Train: 70% & Test: 30%)

(e) RF (Train: 80% & Test: 20%)

(f) RF (Train: 80% & Test: 20%)

(g) Xgboost (Train: 70% & Test: 30%)

(h) Xgboost (Train: 80% & Test:20%)

(i) Xgboost(Train:90%& Test:10%)

Figure 5. ROC curve of soc-karate dataset

V. CONCLUSION

Different outcomes were shown by the performance examination of the supplied models on the fb_food_pages dataset, across all dataset splits, the Random Forest model consistently beat the competition in terms of accuracy, Train AUC, and Test AUC. Strong predictive performance was demonstrated by the high Test AUC values of 0.945 and 1. With Test AUC values ranging from 0.799 to 0.821 and Train AUC values ranging from 0.823 to 0.826, the Logistic Regression model showed reasonable performance. The XGBoost model showed lower accuracy values, especially with a Test Accuracy of 0.074, but excellent Test AUC and Train AUC scores of 1 in all dataset splits. These findings suggest that the Random Forest model is the best at predicting links in this situation, demonstrating its potential.

Examining the provided data, we find that different models (Logistic Regression, Random Forest, and XGBoost) perform differently on the karate dataset. The model with the greatest AUC scores, which indicates the most predictive power, was Random Forest. The Logistic Regression model performed admirably, sometimes even obtaining the greatest accuracy. In contrast to the other models, the XGBoost model typically had lower AUC and accuracy scores. The most effective model parameters varied between dataset splits, indicating the necessity of fine-tuning the models for optimum performance. Overall, the Logistic Regression model shows promise for obtaining high accuracy, while the Random Forest model emerges as the most dependable alternative for link prediction in the context.

Conflicts of Interest: The authors declare no conflicts of interest.

REFERENCES

[1] J. L. D. H. ; J. Kleinberg, Predicting Positive and Negative Links in Online Social Networks. ACM, 2010.

[2] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan, "Group formation in large social networks: Membership, growth, and evolution," Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., vol. 2006, pp. 44–54, 2006.

[3] S. Haghani and M. R. Keyvanpour, "A systemic analysis of link prediction in social network," Artif. Intell. Rev., vol. 52, no. 3, pp. 1961–1995, 2019, doi: 10.1007/s10462-017-9590-2.

[4] H. Tiwari and S. Kumar, "Link Prediction in Social Networks Using Node Embeddings with Nature Inspired Approach," Proc. - 2021 3rd Int. Conf. Adv. Comput. Commun. Control Networking, ICAC3N 2021, pp. 145–150, 2021, doi: 10.1109/ICAC3N53548.2021.9725700.

[5] S. U. Balvir, M. M. Raghuwanshi, and K. R. Singh, "A Comprehensive Survey on Learning Based Methods for Link Prediction Problem," 2023 6th Int. Conf. Inf. Syst. Comput. Networks, ISCON 2023, pp. 1–7, 2023, doi: 10.1109/ISCON57294.2023.10112010.

[6] A. Grover and J. Leskovec, "node2vec: Scalable Feature Learning for Networks," pp. 855–864, 2016, doi: 10.1145/2939672.2939754.

[7] H. Yuliansyah, Z. A. Othman, and A. A. Bakar, "Taxonomy of link prediction for social network analysis: A review," IEEE Access, vol. 8, no. 1, pp. 183470–183487, 2020, doi: 10.1109/ACCESS.2020.3029122.

[8] L. Zhang, X. Zhu, and J. Ma, "Joint Connection and Content Embedding for Link Prediction in Social Networks," 2021 IEEE Glob. Commun. Conf. GLOBECOM 2021 - Proc., pp. 1–6, 2021, doi: 10.1109/GLOBECOM46510.2021.9685806.

[9] F. Che, D. Zhang, J. Tao, M. Niu, and B. Zhao, "ParamE: Regarding neural network parameters as relation embeddings for knowledge graph completion," AAAI 2020 - 34th AAAI Conf. Artif. Intell., pp. 2774–2781, 2020, doi: 10.1609/aaai.v34i03.5665.

[10] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "ConvE," 32nd AAAI Conf. Artif. Intell. AAAI 2018, pp. 1811–1818, 2018.

[11] D. Liu, "A Comparative Experimental Study of Link Prediction Methods with Structural Information," Proc. - 2022 IEEE/WIC/ACM Int. Jt. Conf. Web Intell. Intell. Agent Technol. WI-IAT 2022, pp. 572–576, 2022, doi: 10.1109/WI-IAT55865.2022.00090.

[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, doi: 10.3233/FAIA200308.

[13] S. Luthra, R. Kumar, and M. Gupta, "Future Link Prediction for Social Media Users with Community Detection Algorithm," IEEE Int. Conf. Distrib. Comput. Electr. Circuits Electron. ICDCECE 2022, pp. 1–5, 2022, doi: 10.1109/ICDCECE53908.2022.9793003.

[14] C. Liu, S. Yu, Y. Huang, and Z. K. Zhang, "Effective Model Integration Algorithm for Improving Link and Sign Prediction in Complex Networks," IEEE Trans. Netw. Sci. Eng., vol. 8, no. 3, pp. 2613–2624, 2021, doi: 10.1109/TNSE.2021.3100889.

[15] K. Li, L. Tu, and L. Chai, "Ensemble-model-based link prediction of complex networks," Comput. Networks, vol. 166, p. 106978, 2019, doi: 10.1016/j.comnet.2019.106978.

[16] Z. Lu, B. Savas, W. Tang, and I. S. Dhillon, "Supervised link prediction using multiple sources," Proc. - IEEE Int. Conf. Data Mining, ICDM, pp. 923–928, 2010, doi: 10.1109/ICDM.2010.112.

[17] M. Azam, M. Nouman, and Y. A. Saleh, ABEER, HMOUD AL- FAOURI, Ashraf Mousa HAYFA, "Evaluations of Similarity Base Link Prediction Techniques in Social Network," J. Eng. Sci. Technol., vol. 18, no. 2, pp. 1055–1082, 2023.

[18] J. Zhou, L. Liu, W. Wei, and J. Fan, "Network Representation Learning: From Preprocessing, Feature Extraction to Node Embedding," ACM Comput. Surv., vol. 55, no. 2, 2021, doi: 10.1145/3491206.

[19] X. Ding, C. Ma, X. Zhang, H. S. Chen, and H. F. Zhang, "SOIDP: Predicting Interlayer Links in Multiplex Networks," IEEE Trans. Comput. Soc. Syst., vol. 9, no. 4, pp. 997–1007, 2022, doi: 10.1109/TCSS.2021.3068468.

[20] L. Wang, J. Ren, B. Xu, J. Li, W. Luo, and F. Xia, "MODEL: Motif-Based Deep Feature Learning for Link Prediction," IEEE Trans. Comput. Soc. Syst., vol. 7, no. 2, pp. 503–516, 2020, doi: 10.1109/TCSS.2019.2962819.

[21] H. Wang, W. Hu, Z. Qiu, and B. Du, "Nodes' evolution diversity and link prediction in social networks," IEEE Trans. Knowl. Data Eng., vol. 29, no. 10, pp. 2263–2274, 2017, doi: 10.1109/TKDE.2017.2728527.

[22] L. L. Linyuan and T. Zhou, "Link prediction in complex networks: A survey," Phys. A Stat. Mech. its Appl., vol. 390, no. 6, pp. 1150–1170, 2011, doi: 10.1016/j.physa.2010.11.027.

[23] M. Pujari and R. Kanawati, "Link prediction in multiplex networks," Networks Heterog. Media, vol. 10, no. 1, pp. 17–35, 2015, doi: 10.3934/nhm.2015.10.17.

[24] J. Zhou, D. Y. Huang, and H. S. Wang, "A dynamic logistic regression for network link prediction," Sci. China Math., vol. 60, no. 1, pp. 165–176, 2017, doi: 10.1007/s11425-015-0807-8.

[25] R. Guns and R. Rousseau, "Recommending research collaborations using link prediction and random forest classifiers," Scientometrics, vol. 101, no. 2, pp. 1461–1473, 2014, doi: 10.1007/s11192-013-1228-9.

[26] A. Narayanan, E. Shi, and B. I. P. Rubinstein, "Link prediction by de-anonymization: How We Won the Kaggle Social

Network Challenge," Proc. Int. Jt. Conf. Neural Networks, pp. 1825–1834, 2011, doi: 10.1109/IJCNN.2011.6033446.
[27] S. Scellato, A. Noulas, and C. Mascolo, "Exploiting place features in link prediction on location-based social networks," Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., pp. 1046–1054, 2011, doi: 10.1145/2020408.2020575.
[28] D. K. Behera, M. Das, S. Swetanisha, J. Nayak, S. Vimal, and B. Naik, "Follower Link Prediction Using the XGBoost Classification Model with Multiple Graph Features," Wirel. Pers. Commun., vol. 127, no. 1, pp. 695–714, 2021, doi: 10.1007/s11277-021-08399-y.
[29] R. D. N. S. V. R. I. BV;, "Recommendations in Social Network using Link Prediction Technique Ramya," no. Icosec, pp. 782–786, 2020.
[30] S. U. Balvir, M. M. Raghuwanshi and K. R. Singh, "A Comprehensive Survey on Learning Based Methods for Link Prediction Problem," *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*, Mathura, India, 2023, pp. 1-7, doi: 10.1109/ISCON57294.2023.10112010.