Link Prediction in Social Networks

Abstract—**Being able to recommend links between users in online social networks is important for users to connect with like-minded individuals as well as for the platforms themselves and third parties leveraging social media information to grow their business. Link prediction is an important issue in complex network analysis and mining. Given the structure of a network, a link prediction algorithm obtains the probability that a link is established between two non-adjacent nodes in the future snapshots of the network. We have discussed different methods of link prediction for different types of networks.In the first method, link prediction for ego networks has been discussed.In the second method, attempt to predict link for dynamic and scale free time evolving network has been made. And finally, in the third one, a hybrid method for directed graphs has been explained.**

Index Terms—**Link Prediction, Structural Similarity, Local Similarity Measure, Common Neighborhood, Supervised Learning, Unsupervised Learning, Node Ranking**

I. Background

Link prediction and entity resolution are two ways to identify missing information in networks. Link prediction helps identify edges that are likely to appear in the future, if they do not exist already. Entity resolution uses attributes and network structure data to link nodes that represent the same individual.

These techniques take advantage of many network features like degree, clustering, and path lengths. This introduced some of the basic methods for both tasks. Many more sophisticated computational approaches to both link prediction and entity resolution exist, and those will make excellent further reading for computer scientists interested in this topic.

The results of link prediction can be applied in many areas. Two short case studies discussed how link prediction can be used to recommend connections in social media and how entity resolution is useful for identifying duplicate accounts belonging to the same person. Link prediction is also particularly useful for network forecasting. Knowing which people in an organization are likely to connect can be used in many ways. Within companies, for example, this could be leveraged to make introductions and get collaborations moving faster. Within a criminal or terrorist organization, the predicted links could provide interesting intelligence about how the group will evolve. Entity resolution also has many other applications in social media and online. It is often applied to Census records, where data about people in multiple locations should be connected. It has similar anti-crime and anti-terrorism applications, linking aliases with true identities. Other applications include merging duplicate products in online shopping, merging duplicate web search results, and detecting spam

II. Main Methods Used

Several link predication approaches have been proposed including unsupervised approaches such as similarity measures computed on the entity attributes, random walk and matrix factorization based approaches, and supervised approaches based on graphical models and deep learning.

A. Topology Based Methods

- Common neighbors(CN): This is a common approach to link prediction that computes the number of common neighbors. Entities with more neighbors in common are more likely to have a link. It is computed as follows:

$$CN(A, B) = |A \cap B|$$

- Jaccard measure(JA): The Jaccard Measure addresses the problem of Common Neighbors by computing the relative number of neighbors in common:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

- Adamic–Adar measure(AA): The Adamic–Adar measure is the sum of the log of the intersection of the neighbors of two nodes. This captures a two-hop similarity, which can yield better results than simple one-hop methods. It is computed as follows: where N(u) is the set of nodes

$$A(x, y) = \sum_{u \in N(x) \cap N(y)} \frac{1}{\log |N(u)|},$$

adjacent to u.

- Katz measure: It is computed by searching the graph for paths of length t in the graph and adding the counts of each path length weighted by user specified weights.

$$C_{\mathrm{Katz}}(i) = \sum_{k=1}^{\infty} \sum_{j=1}^{n} \alpha^k (A^k)_{ji}$$

III. Main Technical Content Of the Papers

Mainly 3 papers have been discussed in this report which are given below as:

Identify applicable funding agency here. If none, delete this.

A. *Harnessing the Power of Ego Network Layers for Link Prediction in Online Social Networks*

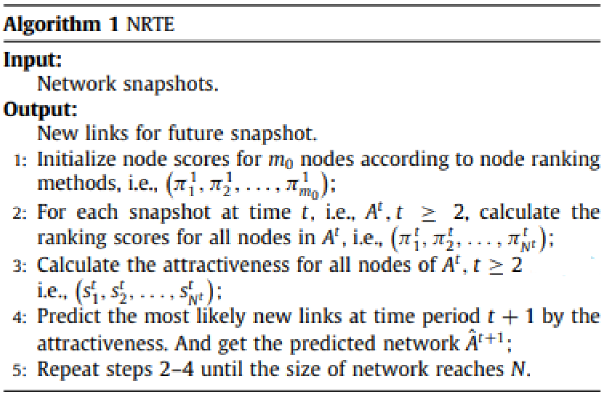
In this article, the performance of circle-aware feature extraction link prediction algorithms have been studied. Specifically, relying on very well-established models from anthropology, the social circles in individual ego networks has been considered, using the circle as a proxy of intimacy. After selecting four benchmark heuristics, we have modified them to include awareness of the social circles. Our results show that social-circle-based link prediction is generally extremely effective. Specifically, in the majority of cases, regardless of the prediction approach (unsupervised or supervised), the specific heuristic or learning algorithm, and the metric (precision, AUC, F1 score) considered, leveraging social circles' information outperforms the corresponding baseline in which circles are ignored. In addition, using only information about the innermost social circles guarantees the same performance achieved when using the whole network. Using social circles information also seems to provide the same performance as using additional classifiers on nodes, which might be impractical or costly to set up. Finally, and most importantly, circle-aware supervised link prediction outperformed recent state-of-the-art feature learning-link prediction approaches, including a GNN-based solution. Interestingly, the best performing circle is C1, which comprises only the two or three strongest relationships of the ego: using knowledge on a few common strong ties, circle-aware link prediction consistently beats black-box approaches.

B. *Link prediction of time-evolving network based on node ranking*

The traditional methods such as AA, JA, RA etc only considered local information or paths of a single snapshot, which is not enough for dynamic network. Here, eigenvector-based node ranking methods, such as Eigenvector Centrality (EC), Cumulative Nomination, PR, Leader Rank are used. The main reasons for adoption of node-ranking-based methods are as follows: Firstly, from a statistical point of view, the importance of a node is like the probability of attracting other nodes to connect with it and the derivative value of a node-pair is like the probability of attracting each other. Secondly, these node ranking methods consider not only the number of neighbours, but also the importance of neighbours when calculating the importance of nodes. Thirdly, the process of solving the importance of each node iteratively in a network is closely related to the process of dynamic network evolution.

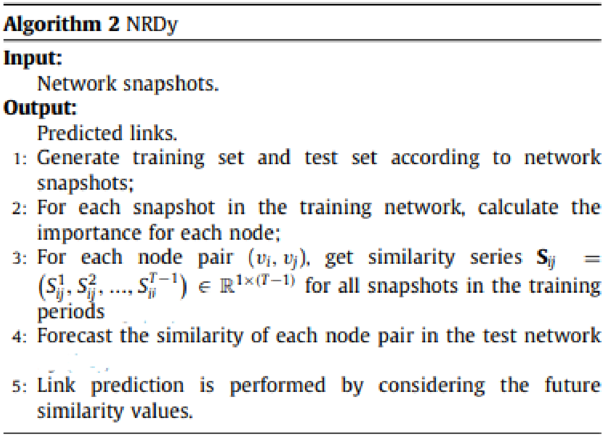
Here, the author have proposed an a new model named Adaptive weighted model(AWM) to forecast the node-pair similarity of the snapshot to be estimated. For different dynamic networks, the weights of the know snapshot can be learned adaptively through some regression methods, like Least squares(LS), ridge, lasso. etc.

Given time periods, t = 1, . . . , T , we define an evolving network as $G^T = (V^T , E^T)$. Here, V^t is the set of nodes in a time period t. N is the scale of network. The set of edges is denoted as E^T , where E^t is the set of undirected edges which occurred, whether new or recurring, between nodes in V^t within time period t. We present a snapshot of network in a given time period as an adjacency matrix.



```
Algorithm 1 NRTE
Input:
    Network snapshots.
Output:
    New links for future snapshot.
1: Initialize node scores for $m_0$ nodes according to node ranking
   methods, i.e., $(\pi_1^1, \pi_2^1, \ldots, \pi_{m_0}^1)$;
2: For each snapshot at time $t$, i.e., $A^t, t \geq 2$, calculate the
   ranking scores for all nodes in $A^t$, i.e., $(\pi_1^t, \pi_2^t, \ldots, \pi_{N^t}^t)$;
3: Calculate the attractiveness for all nodes of $A^t, t \geq 2$
   i.e., $(s_1^t, s_2^t, \ldots, s_{N^t}^t)$;
4: Predict the most likely new links at time period $t + 1$ by the
   attractiveness. And get the predicted network $\hat{A}^{t+1}$;
5: Repeat steps 2–4 until the size of network reaches $N$.
```

Fig. 1. Algorithm for Scale-Free network



```
Algorithm 2 NRDy
Input:
    Network snapshots.
Output:
    Predicted links.
1: Generate training set and test set according to network
   snapshots;
2: For each snapshot in the training network, calculate the
   importance for each node;
3: For each node pair $(v_i, v_j)$, get similarity series $\mathbf{S}_{ij} =$
   $(S_{ij}^1, S_{ij}^2, \ldots, S_{ij}^{T-1}) \in \mathbb{R}^{1\times(T-1)}$ for all snapshots in the training
   periods
4: Forecast the similarity of each node pair in the test network

5: Link prediction is performed by considering the future
   similarity values.
```

Fig. 2. Algorithm for Dynamic network

How it works: i) Choose a similarity metric (e.g. CN, AA, . . .) ii) For each pair of nodes, generate a time series by applying the chosen metric to the known snapshots iii) Choose a forecasting model (e.g. MA,..), and then take the chosen model in its application to the series to perform one-step-ahead prediction.

After many comparisons, a conclusion can be drawn that Pagerank and Leaderrank methods with good performance and high speed can be selected to predict the links for time-evolving networks.

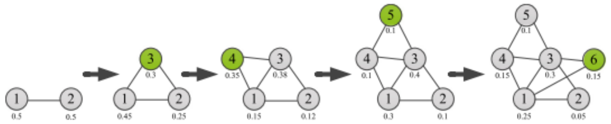


Fig. 3. A toy example of the time-evolving SF network link prediction. The value next to each node is its attractiveness. The new node in green color is more likely to be attracted by the existent nodes (gray color) with higher attractiveness. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

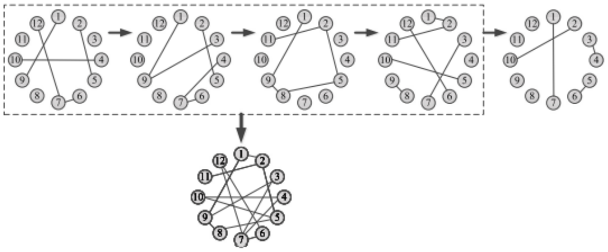


Fig. 4. Snapshots of a sample dynamic network. The union of snapshots in dotted box can be taken as the training network. The last snapshot can be taken as the test network whose links will be estimated

$$\hat{Z}^t = \frac{\omega_1 Z^{t-1} + \omega_2 Z^{t-2} + \cdots + \omega_p Z^{t-p}}{\omega_1 + \omega_2 + \cdots + \omega_p},$$

Fig. 5.

The baselines, such as CN, AA, JA, have worse prediction results than our proposed NRTE-PR. It is because that they consider the similarity of node pair while neglect the evolutionary rule of SF networks.

Therefore, a new forecasting model has been proposed for time series in this paper named AWM(Adaptive Weighted Moving) given in figure 5:

where p N+ is the step of moving and the weight (= 1, . . . , p) of each element can be learned by some regression methods, such as LS, ridge, lasso and so on. Specifically, we build a regression model to learn the relationships between the current element and all past elements Thus, the total computational complexity of our NRDy-RW-PR is about O (M) + O (M) O (M) which is smaller than the computational complexity O (N^2) of CN. When the scale of network is large (e.g., Realcall or Wiki), the time loss in the simulation process of our proposed NRDy-RW-PR is no longer obvious, which shows the real efficient of proposed NRDy-RW-PR

C. A hybrid method of link prediction in directed graphs

Most local measures are based on common neighbors. A problem with these methods is that if two nodes do not have a common neighbor, the probability of their friendship in the future will be assumed to be zero. However, there are some instances in real systems where nodes with no common neighbors have established friendships. Such local measures cannot describe this phenomenon, and one has to use richer information on the connectivity. The method proposed in this paper introduces a novel measure by combining the features obtained based on common neighbors and the hubs and authorities of the nodes. The proposed method overcomes the limitations of common neighbor based measures, while maintaining their simplicity.The rank score of a page depends on the number of input links and the rank scores of the pages of the neighbors that have provided links to that page. If there is a link on page u to page v, the author of page u has actually confirmed the importance of page v. According to this idea, node u is referred to as hub, and node v is known as authority. The hubs point to the authorities. Authority←Hub In the proposed measure, the hub, authority, and direction of the connection (in-out-neighbor) are used along with the information on the common neighbors.

- Common neighbors hub authority (CN-HA) The similarity between two nodes x and y based on both common neighbors and information on hub-ness and authority-ness is defined as:

$$CN - HA_{(xy)} = \left(\sum_{z \in \Gamma_o^{(x)} \cap \Gamma_o^{(y)}} Auth(z) \right) + (Hub(x) + Hub(y)\,)$$

- Common neighbors authority hub (CN-AH) : This measure is defined as follows.

$$CN - AH_{(xy)} = \left(\sum_{z \in \Gamma_i^{(x)} \cap \Gamma_i^{(y)}} Hub(z) \right) + (Auth(x) + Auth(y)\,)$$

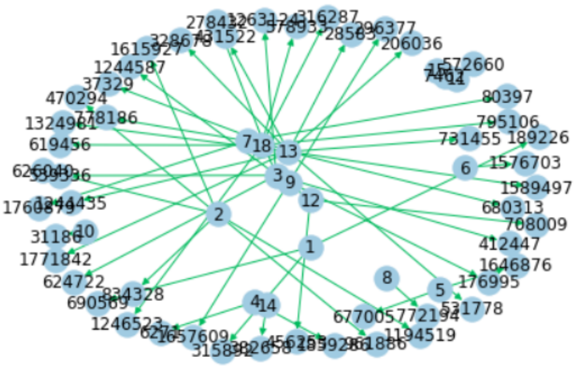
- Sum of common neighbors with hub and authority (SCNHA) : This measure is a combination of the above two similarity metrics, defined as follows :

$$\mathrm{SCNHA}_{(xy)} = CN_H A_{(xy)} + CN_A H_{(xy)}$$

IV. Experiment Setup

We are taken the dataset from Stanford snap website which have following aspects: Type: DiGraph
Number of nodes: 66
Number of edges: 50
Average in degree: 0.7576
Average out degree: 0.7576

The network given as:



A. *Observation*

The number of unique persons 1862220
We find that majority of the people have less than 100 followers. after removing outliers, majority of the people have less than 10 followers.
We find that 75 percent have less than 100 followers.
99We find that majority of the people have less than 100 followers.
We find that majority of the people follow less than 200 people.
After removing the outliers, majority of the people follow less than 10 people.
We find that 75We find that majority of the people follow less than 10 people.
No of persons those are not following anyone are 274512 and No of persons having zero followers are 188043 and No of persons those are not not following anyone and also not having any followers are 0.
We find that majority of the people follow and get followed together less than 200 people.
We find that majority of the people follow and get followed together less than 14 people after removing outliers.
Min of no of followers + following is 1
334291 persons having minimum no of followers + following
Max of no of followers + following is 1579
1 persons having maximum no of followers + following
No of persons having followers + following less than 10 are 1320326
No of weakly connected components: 45558
weakly connected components with 2 nodes: 32195.

B. *Posing a problem as classification problem*

Step 1: Generating some edges which are not present in graph for supervised learning so that we can generate the train-positive, train-negative for training the model and test-positive, test-negative data for testing the accuracy.

Step 2: Generated Bad links from graph which are not in graph and whose shortest path is greater than 2. Removed edges from Graph and used as test data and after removing used that graph for creating features for Train and test data. Data points in train data (15100029, 2) Data points in test data (3775007, 2) Shape of traget variable in train (15100029, 1) Shape of traget variable in test (3775007, 1)

Similarity Measures Used to generate the links: 1.Jaccard Distance
2.Cosine Distance
Ranking Measures Used:
1.Page Ranking
Other Graph Features:
1. Shortest Path
2. Checking For same community
3. Adamic/Adar Index
4. Is Person was following back
5. Katz Centrality
6. Hits Score

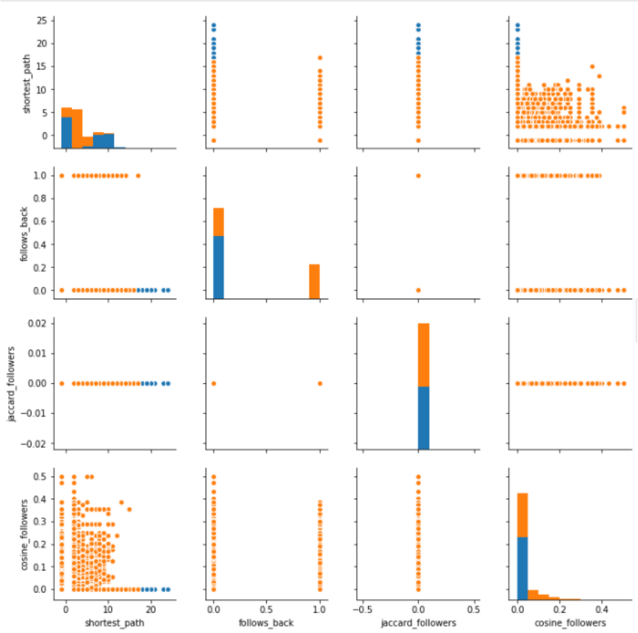
Step 3: Featurization:
Reading a sample of data from both train and test
Add a set of features:
1. jaccard followers
2. jaccard followees
3. cosine followers
4. cosine followees
5. num followers-s
6. num followees-s
7. num followers-d
8. num followees-d
9. inter-followers
10. inter-followees

After Adding new set of features we will create each of these features for both train and test data points:
1. adar index
2. is following back
3. belongs to same weakly connect components
4. shortest path between source and destination

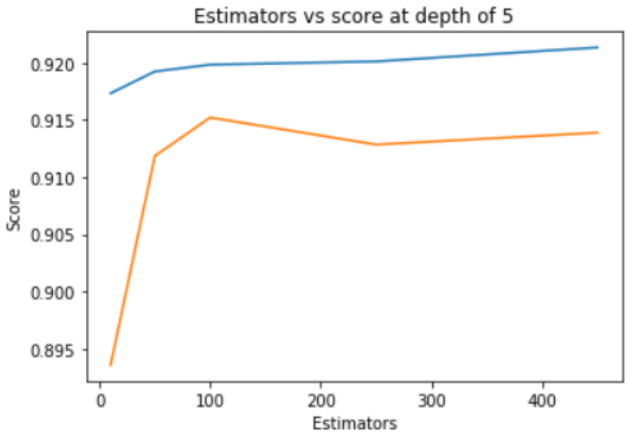
we will create each of these features for both train and test data points
1. Weight Features
• weight of incoming edges
• weight of outgoing edges
• weight of incoming edges + weight of outgoing edges
• weight of incoming edges * weight of outgoing edges
• 2*weight of incoming edges + weight of outgoing edges
• weight of incoming edges + 2*weight of outgoing edges
2. Page Ranking of source
3. Page Ranking of dest
4. katz of source
5. katz of dest
6. hubs of source
7. hubs of dest
8. authorities-s of source
9. authorities-s of dest
Weight Features

In order to determine the similarity of nodes, an edge weight value was calculated between nodes. Edge weight decreases as the neighbor count goes up. Intuitively, consider one million people following a celebrity on a social network then chances are most of them never met each other or the celebrity. On the other hand, if a user has 30 contacts in his/her social network, the chances are higher that many of them know each other. credit - Graph-based Features for Supervised Link Prediction William Cukierski, Benjamin Hamner, Bo Yang
Adding new set of features
we will create these each of these features for both train and test data points
1. SVD features for both source and destination
2. Preferential Attachment(for both followees and followers)
3. SVD Dot

EDA on important extracted features:

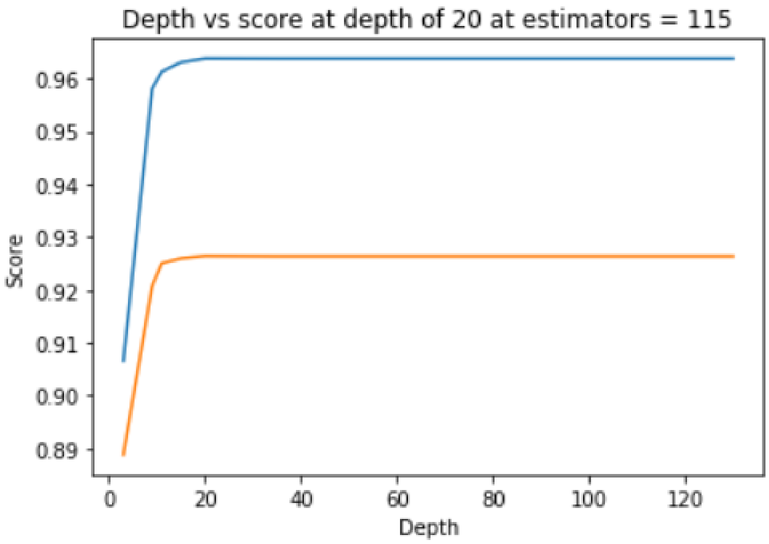


1. We find that using cosine followers and shortest path,we can seperate the classes to some extent.
2. It's not clear to concretely ascertain whether this patterns are seperable or not.
Machine Learning Model used for prediction is Random Forest Classifier

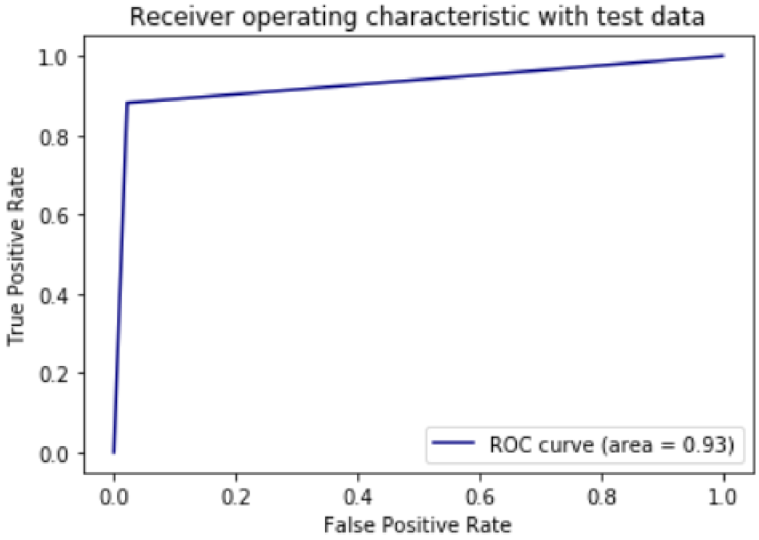


1.We've performed hyper parameter tuning for randomf classifier using for loop.
2. For estimators=115, We find the best train f1 score is 0.915.

1.In the previous cell,we did hyper parameter tuning for optimal estimators.
2. Now, We've performed hyper parameter tuning with multi hyperparameters using for loop.
3. For max-depth=20 and Max-estimators=115, We find the best train f1 score.
Train f1 score 0.9641732843187033
Test f1 score 0.9263485914010191

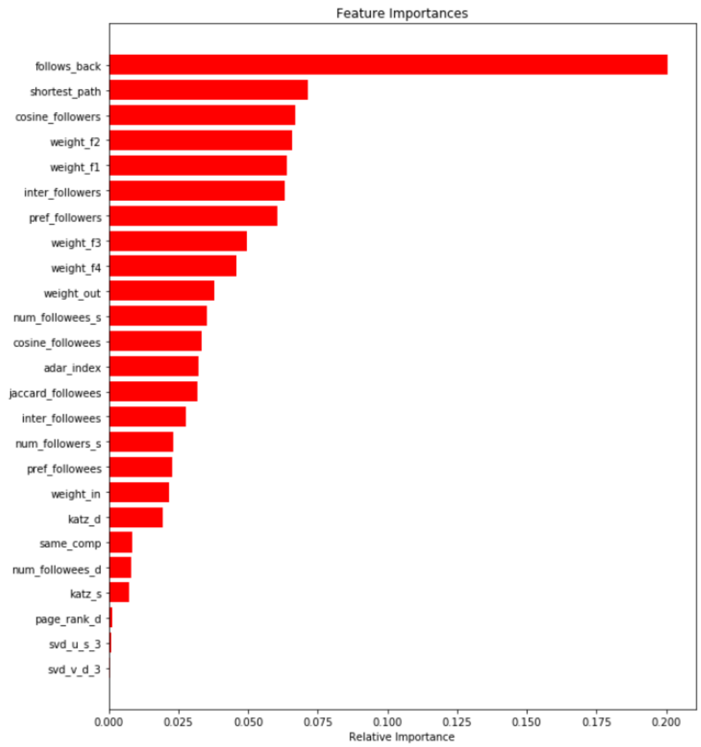


Observations:
1. We have plotted confusion matrix for both train and test.
2. We find that precision and recall are fairly good for both train and test.



1. Here we have plotted the auc value for test values.
2. The value obtained is 0.93

Observations: 1. We tried to plot the important features based on feature importance.
2. The most important feature is follow-back



3. The least important features are svd features.

V. Connection Between the papers

All the papers have same node and link features for comparison i.e., Jaccard similarity, Adamic adar measure and Resource allocation etc. And each paper has considered the importance of the neighbour unlike the prior work. In 1st paper, the ego network is used and the best approach for this is Resource allocation method. In 2nd paper, for scale-free evolving network and dynamic network, a new method called AWM has been proposed and in the third paper a hybrid method has been introduced which takes into account similarity as well as directions of the graph and it seems to work well for balanced and large data set.
In all papers, similarity-based methods have seemed to have outperformed the learning-based method e.g., node2vec, spectral clustering, etc.

VI. Strength and weaknesses of each approach

A. Harnessing the Power of Ego Network Layers for link prediction in online social networks

- Weakness:Not every network is ego network, so still there is great deal of research pending to study whether dynamic ego networks (i.e., time-varying ego network models) can be used for link prediction in dynamic social graphs.
- Strength: results show that social-circle-based link prediction is generally extremely effective. Specifically, in the majority of cases, regardless of the prediction approach (unsupervised or supervised), the specific heuristic or learning algorithm, and the metric (precision, AUC, F1 score) considered, leveraging social circles' information outperforms the corresponding baseline in which circles are ignored. In addition, using only information about the innermost social circles guarantees the same performance achieved when using the whole network. Using social circles information also seems to provide the same performance as using additional classifiers on nodes, which might be impractical or costly to set up.

B. Link prediction of time-evolving network based on node ranking

- Weakness: i)There is a certain amount of training data which is required to achieve good performance.Also the data should be balanced or it may lead to skewed classification. ii)Also calculating the node ranking for each node may become computationally very costly for large sized networks.
- Strength:In this paper, several link prediction algorithms based on node ranking are proposed for time-evolving networks. Through theoretical and experimental analyses, some conclusions can be drawn as follows. Firstly, some eigenvector-based node ranking methods such as PR and LR which compute the importance of each node iteratively are effective and reasonable for SF evolving network or dynamic network link prediction. Because they consider the structural and temporal information of the time evolving network. Secondly, our proposed AWM series forecasting model can predict the score of each node-pair adaptively, which is very suitable for dynamic sequence without specific laws. Thirdly, our proposed algorithms achieve better link pre diction results and have great advantages for sparse dynamic networks with new nodes continuously joining than some state-of the-art methods. Furthermore, our proposed algorithms are more efficient than many other methods especially for the large-size networks.

C. A hybrid method of link prediction in directed graphs

- Weakness:Like most similarity based measures,our proposed method predicts only the presence or absence of edges in a directed graph but not the direction of the link.
- Strength:A problem with common neighborhood based methods is that if two nodes do not have any common neighbors, they always predict a chance of zero for establishment of a link between them; however, such nodes have been shown to establish links in some real systems. Another issue with these measures is that they often disregard the connection direction. The method proposed in the above paper resolves these issues.

VII. Conclusion

All papers have discussed different approaches for different type of networks. But applying this method on real-world network can be computationally high.In the past, lot of work has been done under link prediction in social network but the research that investigates issues related to link prediction in signed networks has been very scarce. Only recently have there was an increase in the number of studies that discuss issues related to link prediction in signed networks.

Low link prediction accuracy remains a major challenge that requires attention. Link prediction within the context of social networks is by no means a novel research topic. However, the greatest challenge associated with predicting new or missing links in dynamic SNs characterized by ongoing evolution is yet to be adequately addressed.

References

[1] Ghorbanzadeh, H., Sheikhahmadi, A., Jalili, M., Sulaimany, S. (2021). A hybrid method of link prediction in directed graphs. Expert Systems with Applications, 165, 113896.

[2] Toprak, M., Boldrini, C., Passarella, A., Conti, M. (2022). Harnessing the Power of Ego Network Layers for Link Prediction in Online Social Networks. IEEE Transactions on Computational Social Systems.

[3] Wu, X., Wu, J., Li, Y., Zhang, Q. (2020). Link prediction of time-evolving network based on node ranking. Knowledge-Based Systems, 195, 105740.