Comparison Analysis of Link Prediction Algorithms in Social Network

Sahil Gupta
Department of Computer Science and Engineering
IIT (BHU), Varanasi, India

Shalini Pandey
Department of Computer Science and Engineering
IIT (BHU), Varanasi, India

K.K.Shukla
Department of Computer Science and Engineering
IIT (BHU), Varanasi, India

ABSTRACT
Social Network depicts the relationship like friendship, common interests etc. among various individuals. Social Network Analysis deals with analysis of these social relationships. Link prediction algorithms are used to predict these social relationships. Given a social network graph in which a node represents a user and an edge represents the relationship between the users, link prediction algorithm predicts the possible new relationships that can be created in the future. This paper compares these link prediction algorithms on the basis of performance metrics like accuracy, precision, specificity and sensitivity.

Keywords
Social Network Analysis, Link Prediction, Performance Evaluation

1. INTRODUCTION
Nowadays social network has become part and parcel of day to day life. Online social networks are developing rapidly. Due to the increasing popularity of the social networking sites, the field of social network analysis is developing and expanding. A social network is represented by a graph in which nodes represent a user and edges represent relationship among various users. A very common example of social network is Facebook in which edges represent friendship among nodes which are the users. In recent days it has been used in social searches such as Facebook Graph Search as well as linking goods and services based on their common features .

Social networks are dynamic in nature, as they grow over time through the addition of new users, creation of new relationships and ending of some old relationships. This dynamic change forms the base of link prediction algorithms. These algorithms involve trying to understand the process of these dynamic changes and try to replicate them. The problem statement is that given a social network graph at a time t, predict the new relationships (edges) that will be created after an interval of time t_1. The algorithms try to use different features to predict new relationships with maximum accuracy.

Link prediction is not only used in the field of social network but can also be used to find persons for a certain job profile. It can also be used in bioinformatics to discover interactions between proteins. In Security it can be used to monitor terrorist groups.

Link Prediction algorithm is based on three different approaches:

- In local Similarity exist the common neighbours between two nodes decide the similarity of the two nodes.
- In global similarity the contribution of all the nodes in the path between two nodes is taken into consideration.
- Feature Vector (such as age group of two persons or their places)also decide the presence of linkages between two nodes

2. ALGORITHMS FOR LINK PREDICTION
We will look at various methods for link prediction in this section. The algorithms we will see work on the principle of similarity. Similarity is the measure calculated for all possible non-existing edges in the graph. It is calculated based on the current features of the existing graph. The values obtained are sorted in decreasing order. The more is the value of similarity of an edge, the more is the chance of it being formed in the future.

We are given a graph G = (V, E) at a time t and a set of edges, we need to predict the probability of link formation for the given set of edges. An edge in the set is given by two nodes u and v. We calculate similarity for the edges by using link prediction similarity features and node guidance capability.

2.1 Local Similarity Features
Let us define N(u) as the set of neighbors of node u in the graph G. These algorithms are based on the idea that two nodes u and v are more likely to form a relationship in the future if their neighbor sets have large number of common nodes.

2.1.1 Common Neighbors
The idea is that more is the number of common neighbors of two nodes u and v, more is the probability of a future relationship between nodes u and v [1]. For an undirected graph, similarity for an edge is defined as follows:

$$\text{Similarity}(u, v) = \left| N(u) \cap N(v) \right| \quad (1)$$

2.1.2 Jaccard's Coefficient
It is one of the most commonly used similarity metrics used in information retrieval. It calculates the probability that a random node x is neighbor of both u and v, if it is neighbor of either u or v [2]. We define similarity as:

$$\text{Similarity}(u, v) = \left| N(u) \cap N(v) \right| / \left| N(u) \cup N(v) \right| \quad (2)$$

2.1.3 Adamic-Adar
This method follows the principle that a common node of nodes u and v with low degree will contribute more towards future relationship between nodes u and v as compared to a common node with high degree [3]. We define d(x) as the degree of node x. Similarity is defined as:

$$Similarity(u, v) = \sum_{z \in |\, N(u) \cap N(v)\,|} \frac{1}{\log d(z)} \quad (3)$$

2.1.4 Preferential Attachment

This method works on the idea that the probability of the relationship formation between node u and v is directly proportional to the degree of the nodes [4]. This method is only dependent on the nodes between which an edge may be formed. The similarity is defined as:

$$\text{Similarity}(u, v) = d(u) \times d(v) \quad (4)$$

2.1.5 Resource Allocation

For nodes u and v which are not connected, we consider that node u can allocate some resources to node v through their common neighbor. We assume that each node has one resource only which it assigns to its neighbors evenly [5]. Similarity between node u and node v can be defined as:

$$Similarity(u, v) = \sum_{z \in | N(u) \cap N(v)|} \frac{1}{d(z)} \quad (5)$$

2.2 Overall Similarity Features

2.2.1 Random Walk with Restart [6]

This method is based on the probability that a node will visit its neighbor. For stationary state e define the similarity as

$$Similarity(u, v) = \frac{d(u) * d(v)}{2 * |E| * |E|} \quad (6)$$

where |E| is number of edges in the graph.

2.3 Node Guidance Capability

This method is based on the density of the common neighbor sub-graph.We extract the sub-graph containing nodes u, v and their common neighbors. If the common neighbors sub-graph is denser, the nodes in the sub-graph made more contribution for link formation. We assign the density of the sub-graph to each node. If the common neighbor occupied greater proportion in the neighbor of the node, it has greater ability to form new link between node u and v [7]. We define guidance force formula of the node:

$$Guidance - Force(z) = \frac{|\varphi(z)|}{\log d(z)} \quad (7)$$

where $|\varphi(z)|$ denotes the degree of node z in the extracted subgraph.

2.3.1 CNGF

This method is based on the guidance capability of the common nodes of nodes u and v. Similarity is calculated by adding the guidance capabilities of all the common neighbors of nodes u and v. The formula of similarity is:

$$Similarity(u, v) = \sum_{z \in N(u) \cap N(v)} \frac{|\emptyset(z)|}{\log d(z)} \quad (8)$$

where$|\emptyset(z)|$ is the number of links that the node connected with the common neighbors.

The complexity of this method is $O(N^2)$; N is maximum of all node's degree.

2.3.2 KatzGf

This method takes into account the path between two nodes u and v. The idea is that if there are more paths between two nodes, the possibility of a new link existing in the two nodes is greater. The contribution of paths with different pathlengths are different, also paths with same pathlengths have different contributions because of different guidance capability of nodes in the path.

The formula used in this method is

$$Similarity(u, v) = \sum_{l=1}^{l=\infty} \beta^l \left(\sum_{z \in path(uv)} \frac{|\emptyset(z)|}{\log d(z)}\right)| \quad (9)$$

where$|\emptyset(z)|$ is the number of links that the node is connected with the nodes in the paths between u and v.

3. DATASETS AND SETUP

For our analysis, we have used Facebook dataset from University of Koblenz-Landau [8],[9] to test various link prediction algorithms. This dataset contains an undirected network of friendship data of Facebook users. A node represents a user and an edge represents friendship between two users. The dataset contains Facebook graph at time 0 and then new edges are formed with passage of time. We have used the following steps to process and modify the dataset as to make it easy for our use.

Step1. We first preprocess the data and separate the graph (edges) which existed at time 0. We will use the rest of the data as edges created after time interval t.

Step2. We form two sets from the above preprocessed data (one small and one large set).

Step3. For the sets created we form testing dataset which comprises of edges which are formed after time t and edges which are not formed after time t.

Step4. Determine threshold values for the algorithms to be compared with the help of pre-existing edges.

Step5. Run the algorithms for testing datasets.

Step6. Performance Evaluation is done for the observations recorded in the above step.

Step7.Based on the results of performance evaluation, conclusions are drawn.

4. OBSERVATIONS AND RESULTS

By using the calculated threshold values and various other possibilities, we get the number of correct and incorrect predictions. Based on this, we calculate the true positive rate, true negative rate, accuracy and precision.

We tested the algorithms for two datasets one small comprising of 1215 nodes and one large comprising of 10026 nodes.

The test dataset had 3448 edges for small dataset and 158430 edges for large dataset. The table shown below shows performance metrics obtained for various algorithms for the large and small datasets. The metrics used are Sensitivity or True Positive Rate (TPR), Specificity (SFC), Precision (PRE) and Accuracy (ACC).

Table 1 : Performance Metrics for Small Dataset

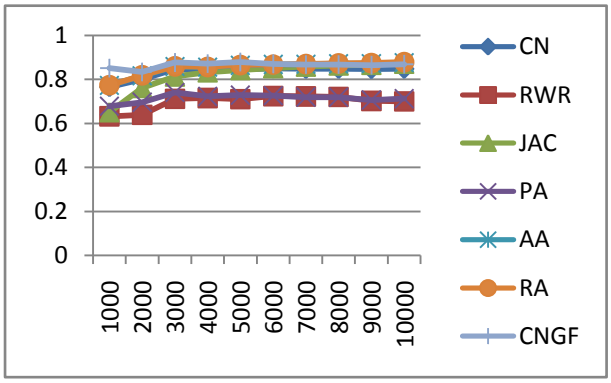
	TPR	SFC	PRE	ACC
Common Neighbour	0.588	0.942	0.910	0.765
Jaccard's Coefficient	0.803	0.488	0.611	0.646
Adamic Adar	0.614	0.932	0.900	0.773
Random Walk	0.657	0.689	0.678	0.673
Preferential Attchment	0.628	0.716	0.688	0.672
Resource	0.613	0.932	0.900	0.773

Allocation				
CNGF	0.713	0.963	0.951	0.838
KatzGF	0.769	0.970	0.962	0.869

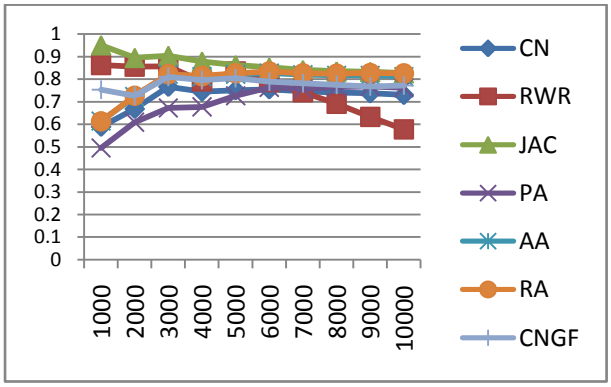
Table 2: Performance Metrics for Large Dataset

	TPR	SFC	PRE	ACC
Common Neighbour	0.729	0.963	0.952	0.846
Jaccard's Coefficient	0.862	0.888	0.885	0.875
Adamic Adar	0.862	0.888	0.885	0.875
Random Walk	0.557	0.819	0.761	0.698
Preferential Attachment	0.762	0.660	0.691	0.711
Resource Allocation	0.824	0.930	0.922	0.877
CNGF	0.812	0.975	0.971	0.893

The following graphs show performance measures sensitivity and accuracy for the above implemented algorithms with increasing number of nodes in the graph.



Graph1 :Accuracy vs Number of nodes in the graph



Graph 2 : Sensitivity vs Number of nodes in the graph

5. CONCLUSION AND FUTURE WORK

We used performance metrics like accuracy, precision, specificity and sensitivity to analyse the performance of the eight similarity indexes.

The results obtained from short dataset shows that KatzGf algorithm gives the best results. Remaining most of the algorithm have very similar accuracies.We observe that for large datasets, threshold values can be calculated better and hence we get better results. For large dataset, CNGF algorithm gave the best results.

We also obtained that KatzGF algorithm despite being the most accurate algorithm, had the maximum time complexity. We were unable to run it for large dataset because the algorithm grows exponentially with increasing number of nodes.

This analysis is important since it will help in deciding which similarity index should be used for any given graph .

Future work includes combining the three signals of Link Prediction in order to bring out an approach which has better time complexity as well as accuracy.Also a rank will be given to every node which will decide the priority of recommending that node . This will improve the recommendation system .

6. REFERENCES

[1] G. Kossinets, "Effects of missing data in social networks," Social Networks, vol. 28, no. 3, pp. 247–268, 2006.

[2] Gerard Salton and Michael J. McGill. Introduction to Modern Information Retrieval. McGraw- Hill, 1983.

[3] Y. D. Jin, T. Zhou, B. H. Wang, and B. Q. Yin, "Power-law strength-degree correlation from resource-allocation dynamics on weighted networks," Physical Review Letters, no. 15, pp. 021–029, 2007.

[4] A.-L. Barab´asi and R. Albert, "Emergence of scaling in Random networks," American Association for the Advancement of Science, vol. 286, no. 5439, pp. 509–512, 1999.

[5] L. Katz, "A new status index derived froms ociometric analysis," Psychometrika, vol. 18, no. 1, pp. 39–43, 1953.

[6] Weiping Liu and Linyuan Lu" "Link prediction based on Local random walk " Department of Physics, University of Fribourg - Chemin du Mus´ee 3, CH-1700 Fribourg – Chemin, Switzerland.

[7] Liyan Dong, Yongli Li, Han Yin Huang Le and Mao Rui" The Algorithm of Link Prediction on Social Network" College of Computer Science and Technology, Jilin University, Changchun 130012, China.00

[8] Facebook friendships network dataset - KONECT, November 2014.

[9] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P. Gummadi. On the evolution of user interaction in Facebook. In Proc. Workshop on Online Social Networks, pages 37-42, 2009.