

# **Link Prediction in Complex Networks**

Report submitted in fulfillment of the requirements  
for the Exploratory Project of

**Second Year**

By

**Md Zuhair**

19074008

**Naresh Kumar**

19075047

**Shaan Kumar**

19074015

Under the guidance of

**Dr. Bhaskar Biswas**

*Professor*



---

Department of Computer Science and Engineering  
INDIAN INSTITUTE OF TECHNOLOGY (BHU) VARANASI,  
Varanasi 221005, India

**Dedicated to**

***Our parents, teachers ...***

## **Declaration**

We certify that:

- The work contained in this report is original and has been done by us our self and in the general supervision of our supervisor.
- The work has not been submitted for any project.
- Whenever we have used materials (data, theoretical analysis, results) from other sources, we have given due credit to them by citing them in the text of the thesis and giving their details in the references.
- Whenever we have quoted written materials from other sources, we have put them under quotation marks and given due credit to the sources by citing them and giving required details in the references.

Place: IIT BHU(Varanasi)

Date: 4 May 2021

**Md Zuhair (IDD)**

**Shaan Kumar (IDD)**

**Naresh Kumar (B.Tech.)**

Department of Computer Science and Engineering,  
Indian Institute of Technology (BHU) Varanasi,  
Varanasi, INDIA 221005.

## **Certificate**

This is to certify that the work contained in this report entitled “**Link Prediction in Complex Networks**” being submitted by **Shaan Kumar** (Roll No. 19074015), **Md Zuhair** (Roll No. 19074008) and **Naresh Kumar** (Roll No. 19075047), carried out in the Department of Computer Science and Engineering, Indian Institute of Technology (BHU) Varanasi, is a bona fide work of our supervision.

Date: 4 May 2021

Place: IIT BHU(Varanasi)

**Dr. Bhaskar Biswas**

Department of Computer Science and Engineering,  
Indian Institute of Technology (BHU) Varanasi,  
Varanasi, INDIA 221005.

## **Acknowledgements**

We would like to express our special thanks of gratitude to our **Prof. Dr. Bhaskar Biswas** who gave us the golden opportunity to do this wonderful project on the **Link prediction of complex networks** which helped us in doing a lot of research and we came to know about so many new things about the day to day use of link prediction in our daily life.

We thank our parents for their support and attention. At-last but not the least we would like to thank our friends who encouraged us and helped us out in finalizing the project.

Date: 4<sup>th</sup> May 2021

Place: IIT BHU(Varanasi)

**Md Zuhair  
Shaan Kumar  
Naresh Kumar**

## ABSTRACT

In today's world, Link prediction has gained increased attention from researchers and social network developers. Link Prediction is basically a problem in which we are given a graph  $G$  at a current moment of time and we need to predict the possibility of a future connection between two nodes of a graph at a later time through analyzing various similarities among the nodes. Link Prediction is a very important topic which is used in Social Networking Sites, for e.g. Facebook, Instagram.

It has wide range of applications like: -

- Suggesting relevant product to users in E-Commerce Industries, for e.g. Flipkart, Amazon, etc.
- Suggesting relevant videos of similar tags and topics in YouTube, and other streaming websites.
- Friend suggestion in social networking sites like Facebook
- Protein interaction networks (PINs) are often used to “learn” new **biological** functions from their topology.

Some of the algorithms that we have explored with the help of this project are:

- Adamic Adar Index
- Preferential Attachment Index
- Jaccard Coefficient Index
- Common Neighbours Index
- Cosine Similarity Index
- Resource Allocation Index
- Sorensen Index

We also tried to implement an algorithm using our understanding of the problem.

# Table of Contents

## Introduction

|                        |   |
|------------------------|---|
| Overview               | 8 |
| Research Motivation    | 8 |
| Organization of Report | 9 |

## Project Work

|   |    |
|---|----|
| Problem Definition                                    | 9  |
| Dataset Preparation                                   | 10 |
| Dataset Selection                                     | 10 |
| Data Preparation                                      | 10 |
| Splitting the Dataset                                 | 10 |
| Link Prediction Algorithms                            | 11 |
| Adamic Adar (AA)                                      | 12 |
| Common Neighbour (CN)                                 | 12 |
| Cosine Similarity (CS)                                | 12 |
| Jaccard Coefficient (JC):                             | 12 |
| Preferential Attachment (PA)                          | 13 |
| Resource Allocation Index (RA)                        | 13 |
| Sorensen Index (SI)                                   | 13 |
| Self-Created Index                                    | 13 |
| Performance Metrics                                   |    |
| Area Under the ROC Curve (AUC):                       | 14 |
| Precision   | 14 |
| Recall  | 14 |
| Results   |    |
| Protein-Protein interaction network in building Yeast | 15 |
| Netscience Collaboration network                      | 16 |
| American College Football                             | 17 |
| Jazz  | 18 |
| Karate Dataset  | 19 |

|            |    |
|------------|----|
| Conclusion | 20 |
|------------|----|

|              |    |
|--------------|----|
| Bibliography | 21 |
|--------------|----|

# Introduction

---

## Overview

Link prediction is one of the most important and useful research topics in recent times. The main objective behind the link prediction is to identify whether there will be a connection between a pair of nodes in future or not.

Now you must be thinking that what is its significance in real life. There are a wide range of places where various industries use link prediction to get things easy and efficient for their users.

Like we see the recommendation of products in E-Commerce websites like Amazon and Flipkart. Another example is how the add recommendation system works for people.

It also has a wide use in many social media platforms like Facebook, Instagram, Twitter, LinkedIn etc. Using link prediction these social media suggests us friends which makes our experience better. Another example is the recommendation of songs in various music apps. Matrimony sites also use link prediction to find the best match for a person.

For instance, if we are talking about Facebook, each person is considered to be a node and every connection between two persons is considered to be an edge in this case. These edges contain information about what those two nodes have in common and what not. With the help of Link prediction algorithms, we predict the probability of connection between two people who are unconnected.

## Research Motivation

In the field of Link Prediction, various attempts and tries have been made to find new Link prediction algorithms that could give better results and better accuracy. But however, mostly the link prediction algorithms improve accuracy over a single dataset and can't generalize it for any type of connection. This is an attempt to generalize and compare the results of various link prediction algorithms like Adamic Adar, Jaccard Coefficient, etc. On different datasets, when we study how they change when the edges and nodes are huge in number or small. Through these comparisons, researchers might find a pattern for accuracy over certain datasets which might help them to improve the present algorithms and can work on to create a better algorithm by taking some weighted mean on the score of the present algorithm.



## Organization of Report

In this report we cover the Theory for Link Prediction, first defining the problem. And then in the next section, we describe the process of Dataset preparation and information of Link Prediction Algorithms.

In the next section, we provide the results comparing the various algorithms and conclusions in the end to finally conclude the report.

## Project Work

---

### Problem Definition

Social networks are represented as an unweighted, undirected graph  $G \{V, E\}$  where each node  $\in V$  represents a person/product etc. and each edge  $(U, V) \in E$  indicates an interaction between the respective nodes  $U$  and  $V$ . Now as due to the dynamic nature of social networks i.e. further connections can be made between any two nodes, Let's take two times,  $t$  and  $t'$  where  $t' > t$ . Let the state of the network at  $t$  and  $t'$  be  $G(t)$  and  $G(t')$  respectively and  $G[t, t']$  be all the edges that had been formed from time  $t = t$  to  $t = t'$ .

### The Link Prediction Task is:

Given time points  $t$ ,  $t'$  and  $t''$ , and network  $G[t, t']$ , We are required to find the set of edges that are predicted to appear in  $G[t', t'']$ , here  $[t, t']$  is referred to as **training interval** and  $[t', t'']$  as **testing interval**.

This set is generated by the Link Prediction Algorithms by assigning a Similarity score  $S_{xy}$ , which represents the similarity between nodes  $x$  and  $y$ . We compute the similarity score  $S_{xy}$  with different Similarity Index Algorithms like **Adamic Adar**, **Jaccard Coefficient**, etc. All predicted edges are then sorted in a non-increasing order by their scores and the edges with the highest scores are most likely to exist [1].

## Dataset Preparation

- Dataset Selection

We have used various real-life datasets for our project from different sources.

| Dataset Name  | Number of nodes | Number of edges | Source                              |
|---|-----------------|-----------------|-------------------------------------|
| <a href="#">Protein-Protein interaction network in building yeast</a> [5] | 2361            | 6646            | <a href="#">Pajek Datasets</a>      |
| <a href="#">Netscience Collaboration network</a> [7]                      | 1589            | 2742            | <a href="#">Pajek Datasets</a>      |
| <a href="#">Jazz</a> [8]  | 198             | 2742            | <a href="#">Alex Arenas Website</a> |
| <a href="#">American College Football</a> [6]                             | 115             | 613             | <a href="#">Network Data</a>        |
| <a href="#">Zachary's karate club</a> [9]                                 | 34              | 78              | <a href="#">Network Data</a>        |

- Data Preparation

Firstly, we have collected various datasets from the above-mentioned sources. We haven't trimmed any dataset in order to maintain a high precision for our models.

- Splitting the Dataset

Given the snapshot of current state of the network, we split the dataset into two parts:

- Training Dataset
- Testing Dataset

where the Training Dataset represents the subgraph  $G[t, t']$  as defined in the project definition and the testing dataset is the subgraph  $G[t', t'']$  which contains the newly formed links.

Splitting of the dataset serves the purpose of the project as we aim to predict links to be created between the present nodes. We will not be adding any extra nodes for the testing dataset, and so it is necessary to process between the same set of nodes for both sets.

For our exploratory project, we have given only **10%** of the dataset to the testing dataset and the rest is our training dataset. Our formation of testing dataset is by selecting completely random edges from the graph.

## Link Prediction Algorithms

In link prediction we find the Similarity index  $S_{xy}$  for each  $x, y \in V$ . The similarity index score gives us an estimate as to how much there can be a link at a later time.

There are certain methods to calculate the Similarity index  $S_{xy}$  between two nodes  $x$  and  $y$ . Let's discuss the methods of Local Similarity index in the next section.

Now, we calculate the local similarity score  $S_{xy}$  of every pair of nodes in the graph. We have calculated it using different algorithms, namely: -

- Adamic Adar (AA)
- Common Neighbours (CN)
- Cosine Similarity (CS)
- Jaccard Coefficient (JC)
- Preferential attachment (PA)
- Resource Allocation (RA)
- Sorensen Index (SI)
- Self-created Index (Self)

Some of the notations used below for explaining each algorithm are: -

- $\Gamma(x)$  → Set of neighbours of node  $x$
- $N$  → Number of nodes in the graph
- $S_{xy}^A$  → Similarity score between node  $x$  and  $y$ , when **A** Algorithm is used.
- $k_x$  → Degree of node  $x$
- $A_{x,y}^l$  → Number of paths between node  $x$  and node  $y$  with length  $l$
- $\beta$  → Katz parameter,  $\beta \in (0,1)$
- $U$  → Total possible links
- $E$  → Existent links
- $U - E$  → Non-existent links
- $E^T$  → Observed links
- $U - E^T$  → Non-observed links
- $E^P$  → Missing Links

## Adamic Adar (AA)

In this algorithm, more number of neighbours of a node  $z$  (Common neighbour of the two nodes for which we are calculating the score) have, less score is given to that node. Thus, the common neighbours of a pair of nodes with few neighbors contributes more to the Adamic Adar score.

In real life we can understand this algorithm as if a common friend (mutual friend) of two people has more number of friends, then it is less likely that he will introduce the two people with each other, rather than in the case when he will have less number of friends.[2]

$$S_{xy}^{AA} = \sum \frac{1}{\log k_z} \forall Z \in \Gamma(x) \cap \Gamma(y)$$

## Common Neighbour (CN)

This similarity index is based on the assumption that two nodes with a greater number of common neighbours will have a high probability of getting connected in future. This algorithm is considered as the most basic algorithm for link prediction. It is mainly usually used as a baseline to judge the performance of the other methods.[2]

$$S_{xy}^{CN} = |\Gamma(x) \cap \Gamma(y)|$$

## Cosine Similarity (CS)

The main idea behind this algorithm is based on the dot product of two vectors. In this method we will do a vector dot product for each pair of nodes having common neighbours in order to calculate the Cosine Similarity index for predicting the link between those two pairs of nodes.[2]

$$S_{xy}^{CS} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\sqrt{k_x \cdot k_y}}$$

## Jaccard Coefficient (JC):

Jaccard Coefficient (which is also known as Jaccard Similarity Index) compares two sets, namely the neighbours of  $x$  and neighbours of  $y$  where  $x, y \in V$ . This is more or less based on the theory of Common Neighbours, the modification being, it is divided with  $\Gamma(x) \cup \Gamma(y)$ . [2] The similarity index for Jaccard Coefficient is defined as

$$S_{xy}^{JC} = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

## Preferential Attachment (PA)

The concept used behind preferential attachment is something similar to the term as saying, “rich getting richer”, or the node having denser network is likely to create another edge with an unconnected node. So, we can see directly that the similarity index will directly depend upon the degree of the nodes ( $k_x$  = degree of node ‘x’)

$$S_{xy}^{PA} = k_x * k_y$$

Unlike CN, JC, AA where the score is clustered around the node and its neighbours, the score here is evenly distributed over all pairs of nodes. [2]

## Resource Allocation Index (RA)

The intuition behind the RA is that if A and B have a huge number of mutual friends, it’s likely that they are also friends.

So, two nodes x, y with huge number of common neighbours will have a greater value of RA. [4]

$$S_{xy}^{RA} = \sum \frac{1}{|k_u|} \forall u \in \Gamma(x) \cap \Gamma(y)$$

## Sorensen Index (SI)

Developed by botanist Thorvald Sorensen, the Sorensen Index (aka Sorensen's Similarity Coefficient) is a measure of similarity between two samples. It is similar to JI. It gives more weightage to the common members among the samples than the ones from a single sample. [2]

$$S_{xy}^{SI} = \sum \frac{2|\Gamma(x) \cap \Gamma(y)|}{(k_x + k_y)}$$

## Self-Created Index

We have created an index during our study to see if we combine the known indices and take various types of mean, which of them will give us the best result. We tried taking various weighted geometric arithmetic and harmonic means after reading the accuracy of the result, then we reached to an unproved conclusion (according to our result) that if we take **Adamic Adar**, **Common Neighbour** and **Cosine Similarity**, and take their unweighted arithmetic mean, it gave the best result among different combination of all the known indices.

## Performance Metrics

To measure the accuracy of our Link Prediction Algorithms, we have used the following three metrics:

- Area Under the ROC Curve (AUC):

Given a ranking of Non-observed links, the term AUC is estimated as the likelihood that a chosen missing link is given a higher score than a randomly chosen unconnected link. Each time we pick one edge randomly from both missing and non-existent links set and compare their scores and define AUC as

$$AUC = \frac{n_1 + 0.5n_2}{n}$$

where,  $n$  is total independent comparisons,  $n_1$  is the number of times the missing link has a higher score and  $n_2$  is the number of times they have the same score. [2][3]

- Precision

After calculating the similarity score of all possible links for each node, we select the some top links and for that node, precision is the ratio of number of missing links in the selected top links to the total number of links selected.

$$precision_x = \frac{L_R}{L_X}$$

where  $L_X$  is the number of selected links for node  $X$  and  $L_R$  is the number of missing links in that set of selected links.

Total precision is taken as the average of precision for all nodes.[2]

$$TotalPrecision = \frac{\sum precision_x}{|V|} \forall x \in V$$

- Recall

After calculating the similarity score of all possible links for each node, we select the some top links and for that node, recall is the ratio of number of missing links in this set of links to the total number of missing links. [2] i.e.

$$recall_x = \frac{L_R}{|E_P|}$$

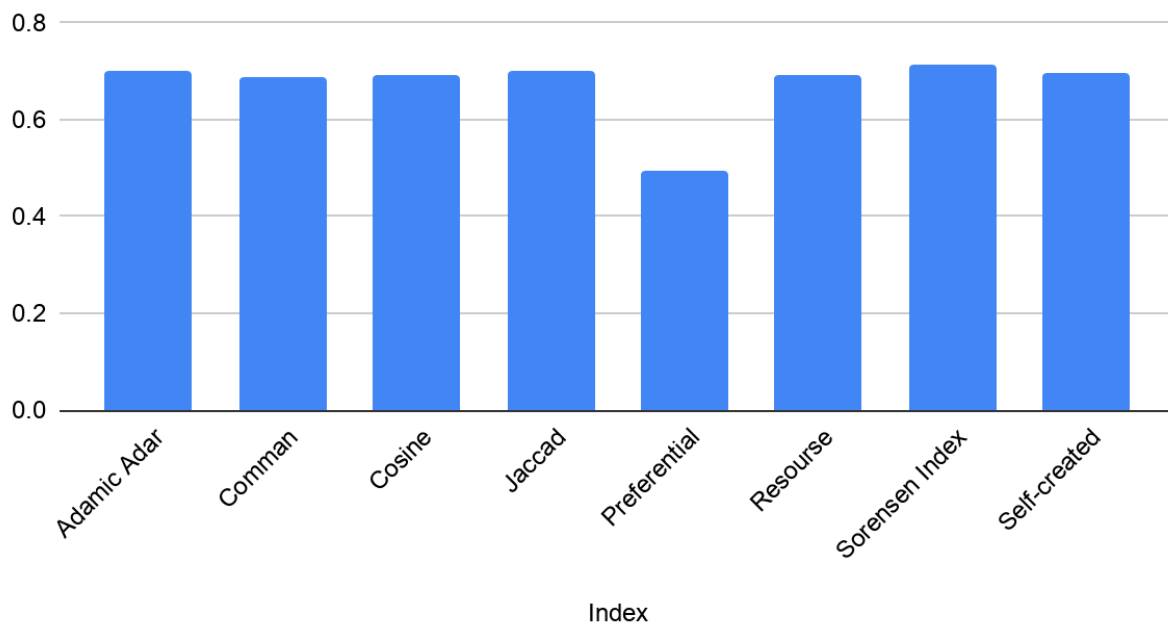
$$TotalRecall = \frac{\sum recall_x}{|V|} \forall x \in V$$

# Results

## Protein-Protein interaction network in building Yeast

| Index                   | AUC (Average) | Precision | Recall | Efficiency(secs) |
|-------------------------|---------------|-----------|--------|------------------|
| Adamic Adar             | 0.6995        | 0.0003    | 0.0006 | 2914.465         |
| Common Neighbour        | 0.6894        | 0.0003    | 0.0005 | 2206.791         |
| Cosine Similarity       | 0.6929        | 0.0003    | 0.0006 | 2912.038         |
| Jaccard Coefficient     | 0.7003        | 0.0003    | 0.0005 | 4240.578         |
| Preferential attachment | 0.4963        | 0.0003    | 0.0006 | 1252.636         |
| Resource Allocation     | 0.6926        | 0.0003    | 0.0005 | 2296.624         |
| Sorensen Index          | 0.7126        | 0.0003    | 0.0006 | 2823.942         |
| Self-created            | 0.6963        | 0.0003    | 0.0006 | 2896.121         |

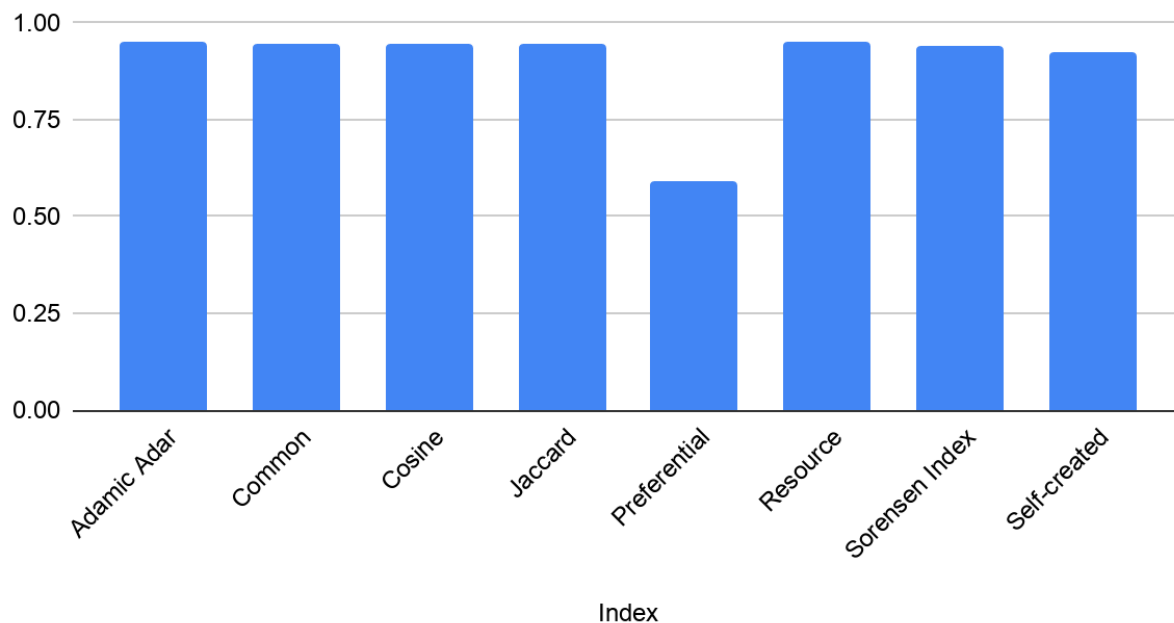
AUC (Average)



## Netscience Collaboration network

| Index                   | AUC (Average) | Precision | Recall | Efficiency(secs) |
|-------------------------|---------------|-----------|--------|------------------|
| Adamic Adar             | 0.9504        | 0.0002    | 0.0006 | 383.055          |
| Common Neighbour        | 0.9484        | 0.0002    | 0.0006 | 396.192          |
| Cosine Similarity       | 0.9466        | 0.0002    | 0.0007 | 567.349          |
| Jaccard Coefficient     | 0.9466        | 0.0002    | 0.0007 | 993.972          |
| Preferential attachment | 0.5915        | 0.0002    | 0.0006 | 222.054          |
| Resource Allocation     | 0.9505        | 0.0002    | 0.0007 | 451.996          |
| Sorensen Index          | 0.9411        | 0.0002    | 0.0007 | 451.031          |
| Self-created            | 0.9248        | 0.0002    | 0.0007 | 572.629          |

AUC (Average)

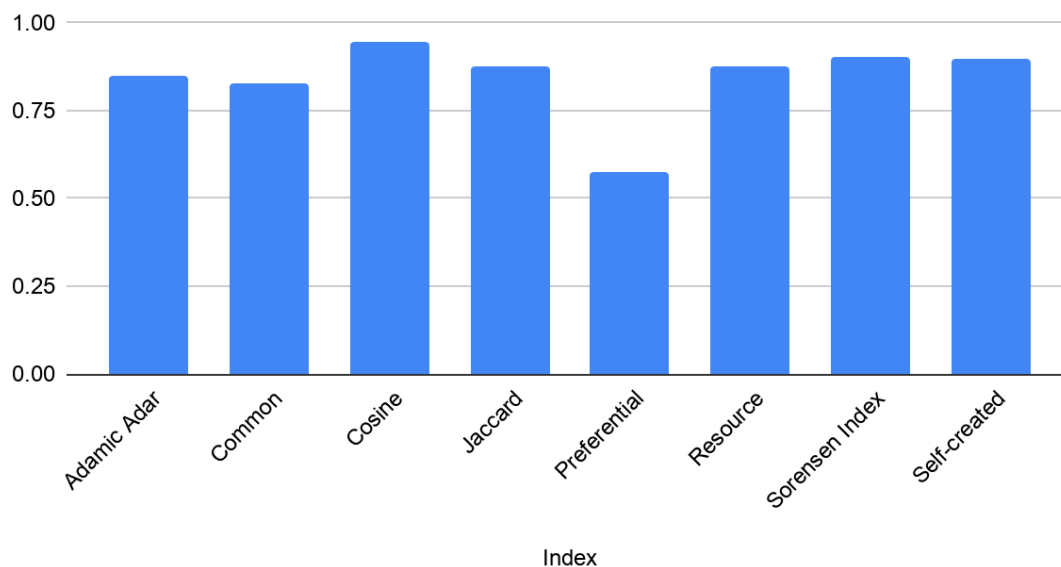




## American College Football

| Index                   | AUC (Average) | Precision | Recall | Efficiency(secs) |
|-------------------------|---------------|-----------|--------|------------------|
| Adamic Adar             | 0.8500        | 0.0104    | 0.0091 | 0.22080          |
| Common Neighbour        | 0.8295        | 0.0101    | 0.0088 | 0.21781          |
| Cosine Similarity       | 0.9466        | 0.0002    | 0.0007 | 0.43042          |
| Jaccard Coefficient     | 0.8752        | 0.0094    | 0.0083 | 0.77849          |
| Preferential attachment | 0.5775        | 0.0096    | 0.0084 | 0.28931          |
| Resource Allocation     | 0.8778        | 0.0114    | 0.0099 | 0.44514          |
| Sorensen Index          | 0.9026        | 0.0106    | 0.0093 | 0.47087          |
| Self-created            | 0.8987        | 0.0098    | 0.0085 | 0.40964          |

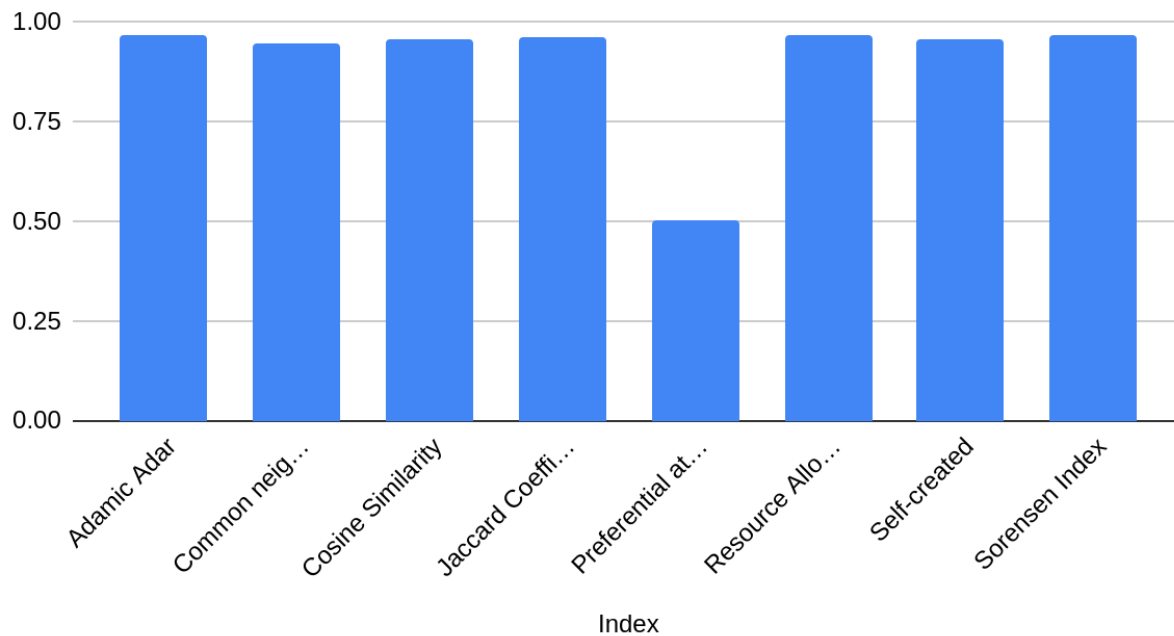
AUC (Average)



## Jazz

| Index                   | AUC (Average) | Precision | Recall  | Efficiency(secs) |
|-------------------------|---------------|-----------|---------|------------------|
| Adamic Adar             | 0.967         | 0.0155    | 0.00466 | 2.1340           |
| Common Neighbour        | 0.9484        | 0.0002    | 0.0006  | 2.0788           |
| Cosine Similarity       | 0.9557        | 0.0149    | 0.00449 | 3.9650           |
| Jaccard Coefficient     | 0.9646        | 0.015     | 0.0044  | 7.5964           |
| Preferential attachment | 0.5062        | 0.014     | 0.00412 | 1.8913           |
| Resource Allocation     | 0.9672        | 0.0158    | 0.0046  | 3.0006           |
| Self-created            | 0.9599        | 0.015     | 0.0044  | 4.0071           |
| Sorensen Index          | 0.966         | 0.0154    | 0.0045  | 4.8451           |

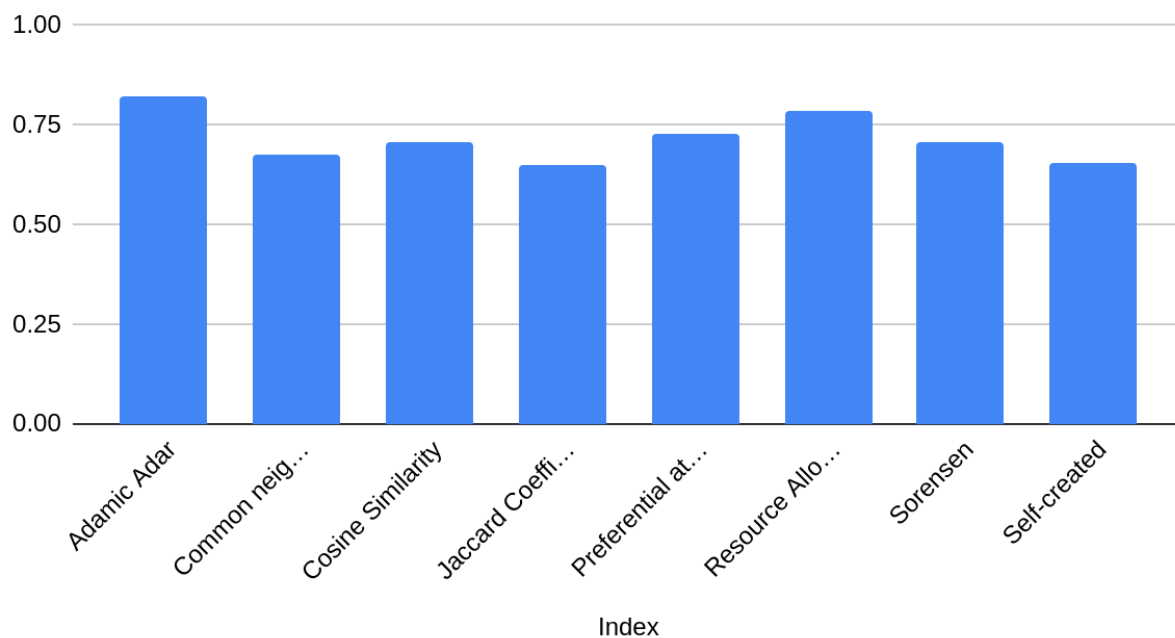
AUC (Average), Precision and Recall



## Karate Dataset

| Index                   | AUC (Average) | Precision | Recall  | Efficiency(secs) |
|-------------------------|---------------|-----------|---------|------------------|
| Adamic Adar             | 0.8243        | 0.0193    | 0.0378  | 0.00526          |
| Common Neighbour        | 0.675         | 0.00789   | 0.0168  | 0.02938          |
| Cosine Similarity       | 0.7086        | 0.0206    | 0.04201 | 0.01532          |
| Jaccard Coefficient     | 0.648         | 0.01298   | 0.02521 | 0.03631          |
| Preferential attachment | 0.728         | 0.02269   | 0.04621 | 0.72818          |
| Resource Allocation     | 0.784         | 0.02265   | 0.04621 | 0.14748          |
| Sorensen                | 0.7074        | 0.0217    | 0.04201 | 0.20930          |
| Self-created            | 0.6533        | 0.0136    | 0.0252  | 0.01479          |

AUC (Average), Precision and Recall



# Conclusion

---

After analyzing different kind of indexes on different types of datasets we can rank algorithms on the basis of the AUC value as:

| Datasets   | AA           | CN    | CS    | JC    | PA    | RA    | SI    | Screated |
|------------|--------------|-------|-------|-------|-------|-------|-------|----------|
| Karate     | 0.824        | 0.675 | 0.708 | 0.648 | 0.728 | 0.784 | 0.707 | 0.653    |
| Jazz       | 0.967        | 0.948 | 0.955 | 0.964 | 0.506 | 0.967 | 0.966 | 0.959    |
| Football   | 0.850        | 0.829 | 0.946 | 0.875 | 0.577 | 0.877 | 0.902 | 0.898    |
| Netscience | 0.950        | 0.948 | 0.946 | 0.946 | 0.591 | 0.950 | 0.941 | 0.924    |
| Yeast      | 0.699        | 0.689 | 0.692 | 0.700 | 0.496 | 0.692 | 0.712 | 0.696    |
| Average    | <b>0.858</b> | 0.817 | 0.849 | 0.827 | 0.580 | 0.854 | 0.846 | 0.826    |

**Adamic/Adar > Resource Allocation > Cosine Similarity > Sorensen Index > Jaccard's Coefficient > Common Neighbour > Preferential Attachment**

In this project, we studied the working of prediction algorithms for finding similarity scores and also compared those prediction algorithms on the basis of performance metrics on different datasets.

Apart from that, we also came to know of different kinds of factors which can possibly enhance the possibility of a link between nodes. A link between any two nodes also depends on the neighbour nodes as well. If two nodes share a common neighbour it improves the possibility of link between them in the near future. Distance between nodes can also affect the possibility of a link in between them. A node is more likely to create a link with a node which has less distance to it rather than a node having a large distance. This can be understood in a real-life scenario as well. For example, you will more likely want to play with your cousin rather than a stranger because your cousin has a link to you through your uncle/aunt.

The major problem that comes in Link Prediction is huge datasets. Because of the large number of nodes present in the graph it becomes so time taking to predict similarity scores for two nodes and to improve the time complexity we sometimes need to compromise with accuracy.

# Bibliography

---

- [1] Link Prediction in Complex Networks by Multi Degree Preferential-Attach Indices - Ke Hu, Ju Xiang
- [2] Fei Gao, Katarzyna Musial, Colin Cooper, Sophia Tsoka, "Link Prediction Methods and Their Accuracy for Different Social Networks and Network Metrics", *Scientific Programming*, vol. 2015, Article ID 172879, 13 pages, 2015. <https://doi.org/10.1155/2015/172879>
- [3] Hanley JA, McNeil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*. 1982 Apr;143(1):29-36. doi: 10.1148/radiology.143.1.7063747. PMID: 7063747.
- [4] Q. Ou, Y.-D. Jin, T. Zhou, B.-H. Wang, B.-Q. Yin, Power-law strength-degree correlation from resource-allocation dynamics on weighted networks, *Phys. Rev. E* 75 (2) (2007) <http://dx.doi.org/10.1103/physreve.75.021102>.
- [5] V. Batagelj, 25 July 2003, Pajek Datasets, available at <http://vlado.fmf.uni-lj.si/pub/networks/data/bio/Yeast/Yeast.htm>
- [6] M. Girvan and M. E. J. Newman, *Proc. Natl. Acad. Sci. USA* 99, 7821-7826 (2002), American College Football Dataset available at <http://www-personal.umich.edu/~mejn/netdata/>
- [7] V. Batagelj, 1 March 2007, Pajek Datasets, available at <http://vlado.fmf.uni-lj.si/pub/networks/data/collab/netscience.htm>
- [8] P. Gleiser and L. Danon, *Adv. Complex Syst.* 6, 565 (2003), Jazz Musicians Dataset available at <https://deim.urv.cat/~alexandre.arenas/data/welcome.htm>
- [9] W. W. Zachary, An information flow model for conflict and fission in small groups, *Journal of Anthropological Research* 33, 452-473 (1977), Zachary's karate club data available at <http://www-personal.umich.edu/~mejn/netdata/>