

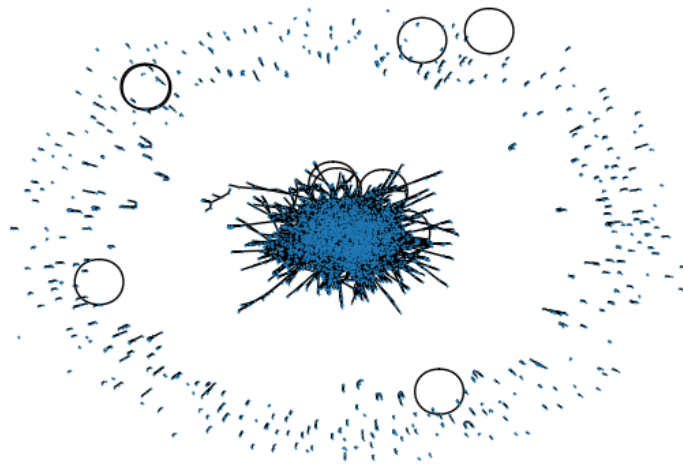
# Project 2 Report

**Student Name:** Jaskeerat Singh

**Student ID:** 47610039

**Student Email:** Jaskeerat.singh@uqconnect.edu.au

This report describes the method devised to sort and rank the future links(edges) in a co-author network. This co-author network is an undirected graph with 11696 edges connecting 5242 nodes. The final output consists of 100 links(edges) that can occur between the existing nodes in the future.



**Figure 1:** Co-Author Graph with 5242 nodes

In this project, we have tried and implemented Neighbourhood-based, Random Walk-based, and Embedding-based Methods algorithms. In each algorithm implementation, the positive and negative validation edges score was calculated and stored in a data frame. The data frame was sorted on the score the algorithm gave, and the top 100 edges were selected based on this score. These 100 edges were then compared with the positive validation edges that act like the ground positive for this project. The algorithm with the highest accuracy was selected to predict the top 100 test edges.

METHOD	ACCURACY %
JACCARD SIMILARITY	72
Adamic-Adar Index	78
Preferential Attachment	22
Resource Allocation Index	78
Sim Rank	65
Shortest Distance (Dijkstra's)	37
Node2Vec	60

**Table 1:** Methods implemented and their respective Accuracy scores

From the above Accuracy scores, a combination of Adamic-Adar and Jaccard similarity edges was preferred, resulting in the highest accuracy of 83%. Both Adamic-Adar and Jaccard similarity belong to the neighbourhood-based methods.

### Jaccard Similarity:

Jaccard Similarity measures the similarity of two nodes in a network. It is defined as the intersection of the set of common neighbours between the two nodes divided by the union of the set of common neighbours. Jaccard Coefficient of nodes  $u$  and  $v$ , where  $\Gamma u$  is the set of neighbours of node  $u$ , is defined as:

$$\frac{|\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u) \cup \Gamma(v)|}$$

The score of each test edge was calculated and then sorted in descending to select the top 100 edges.

### Adamic-Adar Index:

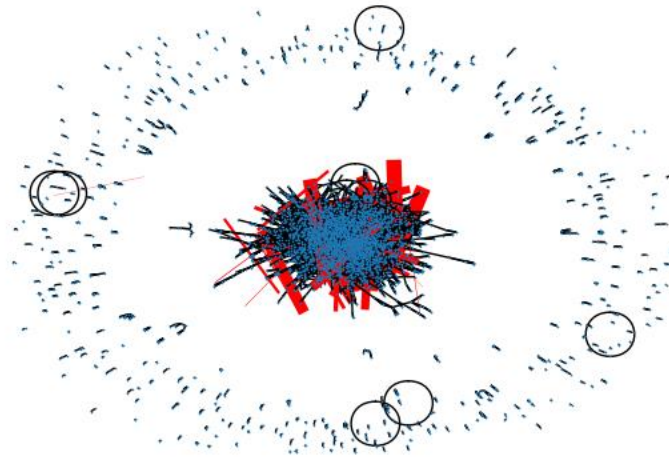
The Adamic-Adar index measures the similarity of two nodes in a network. It is defined as the intersection of common neighbours between the nodes and the importance of these neighbours in the network. It gives more weightage to neighbours with fewer connections.

$$\text{Adamic Adar Index}(X, Y) = \sum_{u \in N(X) \cap N(Y)} 1/\log(|N(u)|)$$

It takes the summation of the inverse of the log of the degree of the common neighbours between two nodes,  $X$  and  $Y$ . The score of each test edge was calculated and then sorted in descending to select the top 100 edges.

### Algorithm Followed –

- Calculate Adamic-Adar Index of each test edge.
- Collect top 100 edges
- Calculate Jaccard Similarity of each test edge.
- Again, collect top 100 edges
- Take Common Edges from the 2 lists
- Take top 10 edges from the remaining predicted values in Adamic-Adar and Jaccard
- Save and publish the results



**Figure 2:** Original Co-Author Graph with predicted 100 edges

## Summary

Through this assignment, we have implemented various algorithms to predict the top 100 edges that can be formed between existing nodes of a graph in the future. After Applying various algorithms, we settled with Jaccard Similarity and Adamic-Adar Index as they gave the highest accuracy. The final output file contains the top 100 edges predicted from the test set. We have used pandas, numpy, node2vec, sklearn, random and networkx Python modules to implement this project.

## Reference

- [1] Joshi, P. (2020, April 21). *A guide to link prediction - How to predict your future connections on Facebook*. Analytics Vidhya. <https://www.analyticsvidhya.com/blog/2020/01/link-prediction-how-to-predict-your-future-connections-on-facebook/>
- [2] *Link prediction - Predict edges in a network using Networkx* - GeeksforGeeks. (2020, May 8). GeeksforGeeks. <https://www.geeksforgeeks.org/link-prediction-predict-edges-in-a-network-using-networkx/>
- [3] *Link prediction — NetworkX 3.1 documentation*. (n.d.). NetworkX — NetworkX documentation. [https://networkx.org/documentation/stable/reference/algorithms/link\\_prediction.html](https://networkx.org/documentation/stable/reference/algorithms/link_prediction.html)