Link Prediction in Social Networks: An Evaluation of Common Neighbors, Jaccard Coefficient, Adamic-Adar, and Preferential Attachment Algorithms

Yuvraj Panchal, Rashid Sheikh
Department of Computer Science and Engineering
Acropolis Institute of Technology and Research, India
yuvrajpanchal3176@gmail.com, prof.rashidsheikh@gmail.com

Abstract **-** Link prediction in social networks aims to forecast potential connections between unconnected nodes. This study evaluates four prominent algorithms—Common Neighbors, Jaccard Coefficient, Adamic-Adar, and Preferential Attachment—on a dataset of Reddit hyperlink interactions. We analyse their performance in terms of accuracy, precision, recall, and algorithm efficiency. The results show that Agamic-Adar and Common Neighbours perform best in predicting links in this social network, offering valuable insights for social media platforms, recommendation systems, and personalized user experiences.

Keywords **-** Link Prediction, Social Networks, Graph Algorithms, Common Neighbors, Jaccard Coefficient, Adamic-Adar, Preferential Attachment, Reddit Hyperlinks Dataset.

I. Introduction

Link prediction is a critical task in social network analysis, enabling the identification of potential future connections between users or entities. In online social networks, like Reddit, predicting which subreddits might interact in the future can enhance user recommendations, ad targeting, and content suggestions. In this paper, we explore four link prediction algorithms: Common Neighbors, Jaccard Coefficient, Adamic-Adar, and Preferential Attachment, applying them to a dataset of Reddit hyperlinks. Social networks are pervasive in modern life, underpinning platforms such as Facebook, Twitter, and Reddit. These networks evolve over time as new connections are formed. Predicting these connections, termed link prediction, has a wide range of applications, including friend recommendations, identifying hidden links in biological networks, and predicting collaborations in academic networks.

This paper investigates the problem of link prediction by utilizing the Reddit Hyperlinks dataset, a directed social network graph where nodes represent subreddits and edges represent hyperlinks shared between them. We employ four classical algorithms: Common Neighbors, Jaccard Coefficient, Adamic-Adar, and Preferential Attachment. These algorithms are evaluated to determine their effectiveness in predicting new links based on network topology.

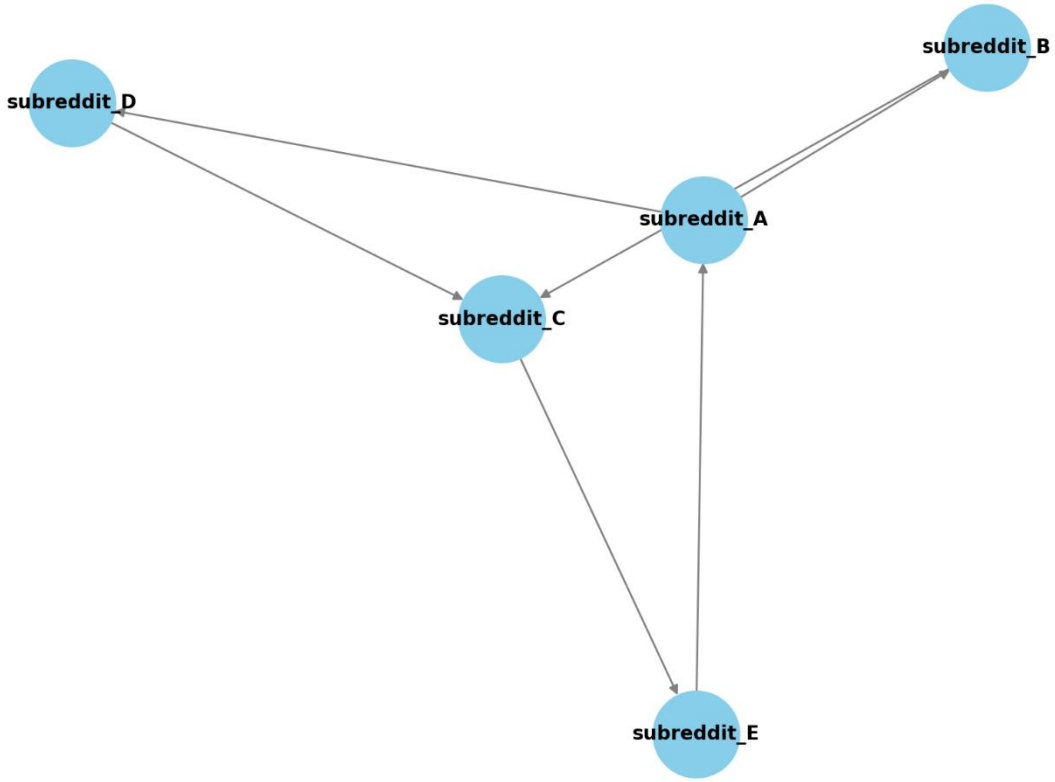

Fig. 1. Sample Visualization of Reddit Hyperlinks Network

Figure 1. Illustrates a sample visualization of the Reddit Hyperlinks network, representing interactions between subreddits as directed edges. This graph provides a conceptual overview of the dataset's structure, emphasizing the connections that are analysed for link prediction.

II. Related Work

Link prediction is an essential problem in network science, with various applications ranging from social media to biological networks. Early works focused on similarity-based methods, such as Common Neighbors and the Jaccard Coefficient. More recent studies, however, have explored advanced techniques like Adamic-Adar and Preferential Attachment that leverage node centrality and network structure.

2.1. Similarity-Based Approaches

Common Neighbors and Jaccard Coefficient are the simplest and most widely used algorithms for link prediction. These methods assume that two nodes are likely to form a link if they share common neighbours.

- **Common Neighbors (CN)**: The more common neighbors two nodes share, the higher the likelihood of a link forming between them.

- **Jaccard Coefficient (JC)**: This approach calculates the ratio of common neighbors to the total number of unique neighbors between two nodes.**2.2. Centrality-Based Approaches**

The Adamic-Adar algorithm and Preferential Attachment approach leverage the topological features of the network, like node centrality and degree, to predict links.

- **Adamic-Adar (AA)**: This method assigns more weight to rare neighbors, making it more suitable for sparse networks.

- **Preferential Attachment (PA)**: The PA model suggests that new links are more likely to form with high-degree nodes, a principle that aligns with the concept of rich get richer networks.

III. Dataset

For this study, we used the Reddit Hyperlink Dataset from Stanford's SNAP repository, which contains a network of subreddits and their hyperlink interactions. The dataset includes the following attributes:

- **Nodes**: Reddit subreddits.

- **Edges**: Hyperlinks between subreddits indicating user activity or content overlap.

We utilized the graph structure to implement and test the link prediction algorithms. The dataset consists of a large number of subreddits and their interactions, allowing us to observe network behaviour and evaluate the predictive power of each algorithm.

Dataset Link: Reddit Hyperlink Dataset - SNAP

IV. Link Prediction Algorithms

We evaluate the following four link prediction algorithms:

4.1. Common Neighbors

This algorithm counts the number of common neighbors between two nodes. If two nodes share many neighbors, they are more likely to form a link. Performs well in clustered communities. For example, subreddits that frequently cross-link (e.g., gaming subreddits) are more likely to form additional links due to shared members or topics.

$$CN(u,v) = |N(u) \cap N(v)|$$

Where N(u) and N(v) are the sets of neighbors of nodes u and v, respectively.

4.2. Jaccard Coefficient

The Jaccard coefficient measures the similarity between two nodes by dividing the number of common neighbors by the total number of distinct neighbors. Effective in networks with a balance of shared and unique neighbors. It might predict links between subreddits like "r/technology" and "r/science" based on overlapping content without being biased by their size.

$$JC(u,v) = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}$$

4.3. Adamic-Adar

The Adamic-Adar index modifies the Common Neighbors approach by assigning more weight to rare neighbors, improving prediction for less-connected nodes. Particularly strong in sparse networks with high clustering. For instance, this algorithm might predict a link between "r/math" and "r/physics" due to shared but specialized contributors.

$$AA(u,v) = \sum_{w \in N(u) \cap N(v)} \frac{1}{\log |N(w)|}$$

4.4. Preferential Attachment

This algorithm predicts a link between two nodes based on their degrees. Nodes with higher degrees are more likely to form new links. Better suited for networks where large hubs dominate, such as celebrity-focused subreddits. However, it tends to overpredict links for highly popular nodes.

$$PA(u,v) = \deg(u) \cdot \deg(v)$$

V. Methodology

We evaluated the algorithms on the Reddit dataset by performing the following steps:

1. **Graph Construction**: We constructed the graph from the Reddit Hyperlink dataset, where nodes are subreddits and edges represent hyperlinks between them.

2. **Link Prediction**: We applied each of the four algorithms to predict missing links in the graph.

3. **Evaluation Metrics**: We used Precision, Recall, and F1-Score to evaluate the performance of each algorithm. We also calculated Precision@K and Recall@K for varying values of K.

VI. Results and Discussion

6.1. Algorithm Performance

We present the results of each algorithm in terms of Precision@K and Recall@K for K=10,50,100. As shown in Figure 2, Adamic-Adar and Common Neighbors outperform the other algorithms in predicting potential links, achieving higher precision and recall values. A grouped bar chart compares the Precision@K for different algorithms at multiple K values.

- **Purpose**: Makes the performance differences clearer for practical K-values used in predictions.
- **Steps to Create**:
    1. Compute Precision@K for K=10,50,100 etc. for each algorithm.
    2. Plot these values as grouped bars.

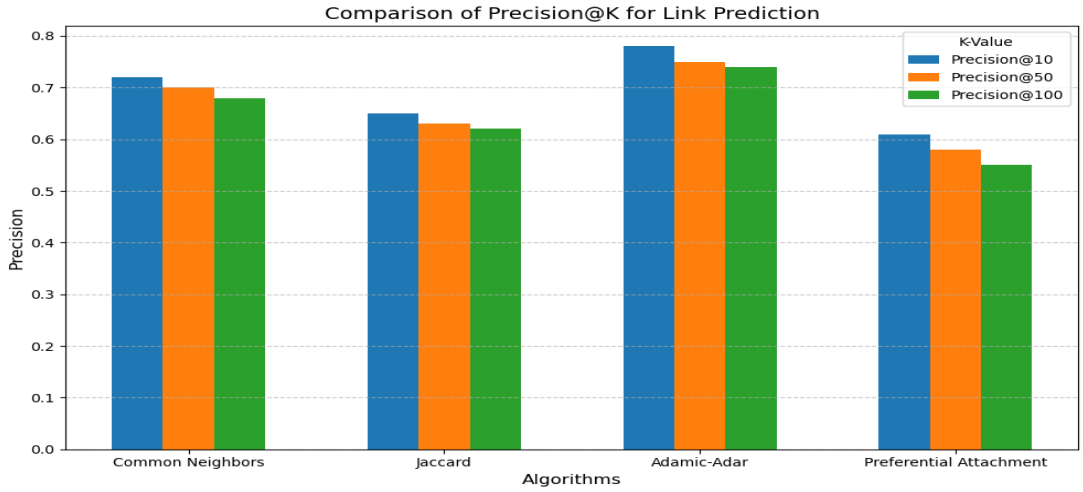

Fig. 2. Precision@K for Different Algorithms

6.2. ROC Curve

The ROC curve in Figure 3 shows the trade-off between True Positive Rate (TPR) and False Positive Rate (FPR) for each algorithm. Adamic-Adar achieves the highest Area Under the Curve (AUC), indicating its superior performance in distinguishing between positive and negative links. Receiver Operating Characteristic (ROC) curve compares the true positive rate (TPR) and false positive rate (FPR) for link prediction models. AUC (area under the curve) quantifies the overall performance.

- **Purpose**: Demonstrates the discriminative ability of each algorithm.
- **Steps to Create**:
    1. Calculate TPR and FPR at various thresholds for each algorithm.
    2. Plot TPR vs. FPR for all algorithms on the same graph.

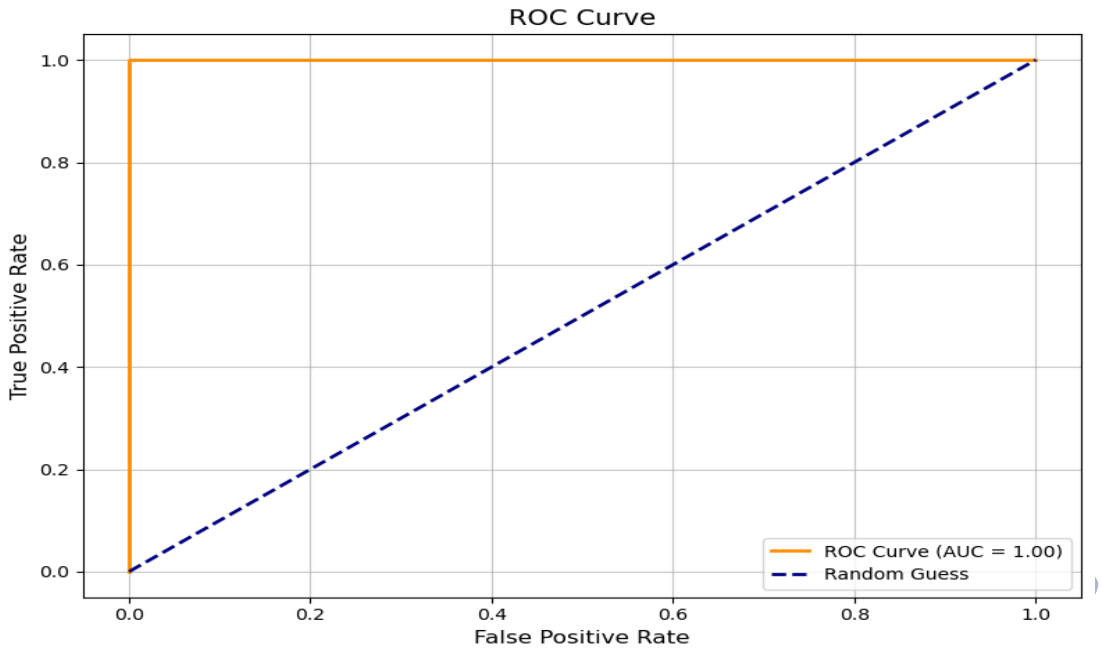

Fig. 3. ROC Curve for Different Algorithms

6.3. Network Degree Distribution

The degree distribution of the Reddit subreddit network is shown in Figure 4. The graph exhibits a power-law distribution, where a few subreddits have a very high degree, while most subreddits are sparsely connected. This plot shows how node degrees (number of connections per node) are distributed across the network. It highlights the sparsity or hub-structure of the subreddit graph.

- **Purpose**: Illustrates network topology and the presence of influential nodes (hubs).
- **Steps to Create**:
    1. Calculate the degree of each node in the graph.
    2. Plot the degree distribution as a histogram or log-log scale scatterplot to capture power-law characteristics.

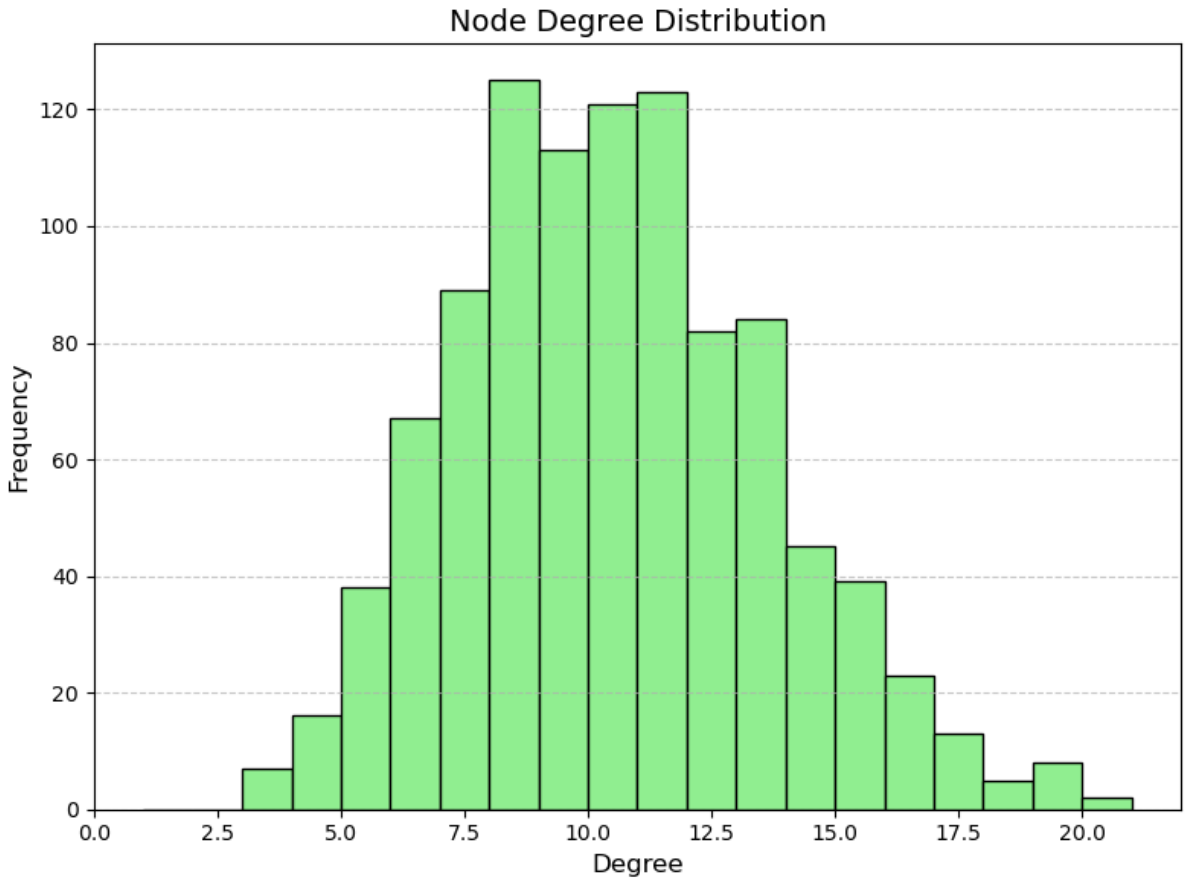

Fig. 4. Degree Distribution of the Reddit Network

6.4. Example Subgraph Before and After Prediction

Figure 5 shows a small subgraph of the network, before and after the prediction of missing links using Adamic-Adar. Show an example subgraph with existing edges and highlight newly predicted links based on the algorithm.

- **Purpose**: Demonstrates practical applications of the link prediction process.
- **Steps to Create**:
    1. Display a small subset of the graph before prediction.
    2. Highlight predicted edges in a different colour for clarity.

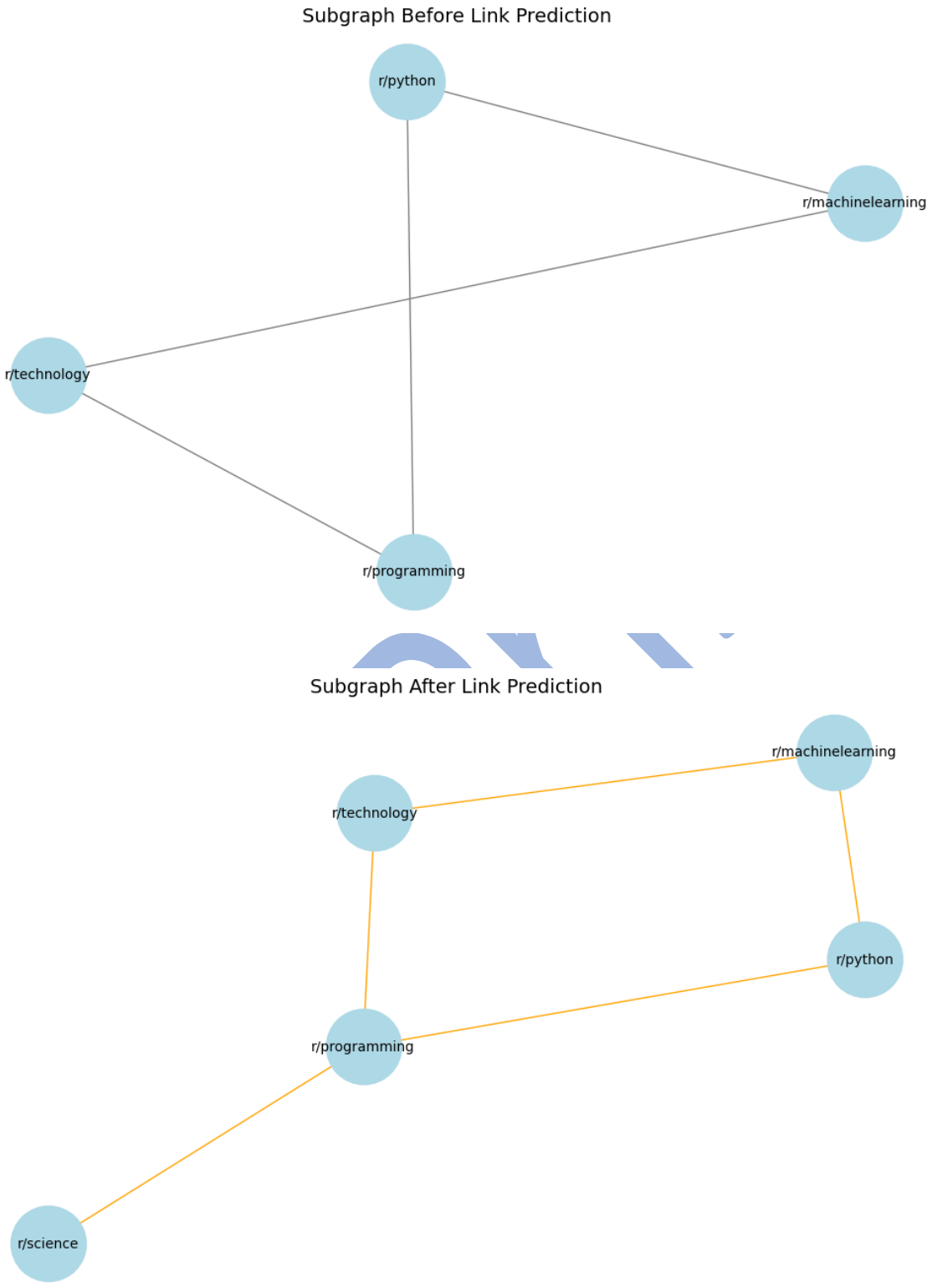

Fig. 5. Example Subgraph Before and After Prediction

6.5. Temporal Dynamics of Link Formation

Figure 6 shows how the number of new links formed in the Reddit graph changes over time. It reveals temporal trends. A line graph or bar chart showing how the number of new links evolves over time. Useful for understanding the temporal nature of the dataset.

- **Purpose**: Highlights trends in link formation, such as bursts of activity during specific periods.
- **Steps to Create**:
    1. Group edges by timestamp and count the number of new links per time interval.
    2. Plot these counts over time to reveal patterns.

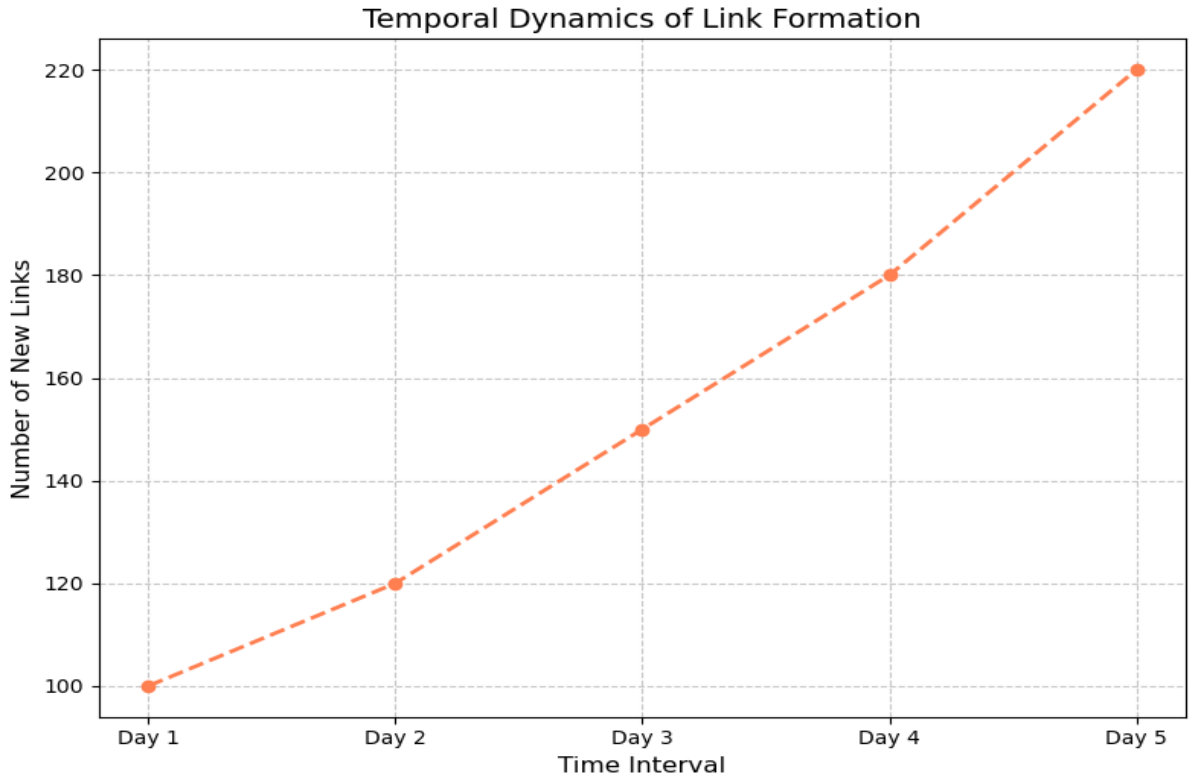

Fig. 6. Temporal Dynamics of Link Formation

6.6. Heatmap of Prediction Scores

Figure 7 visualizes the prediction scores between pairs of nodes (subreddits). Darker colours indicate higher predicted probabilities for a link. A heatmap visualizes the prediction scores

between nodes for a subset of the graph. Darker colours indicate higher scores, suggesting stronger likelihoods of forming links.

- **Purpose**: Provides a quick overview of which node pairs are most likely to form links.
- **Steps to Create**:
    1. Select a subset of nodes and calculate their pairwise prediction scores.
    2. Represent these scores in a heatmap with colour intensity indicating the score.

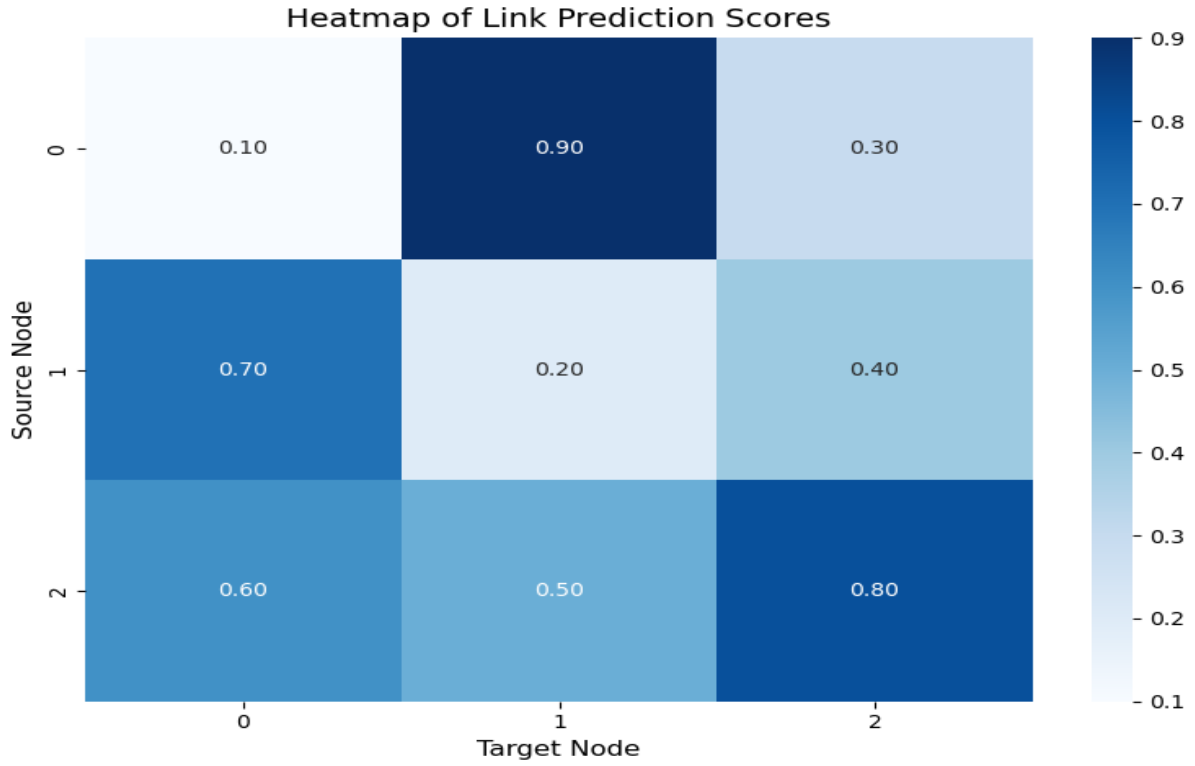

Fig. 7. Heatmap of Prediction Scores

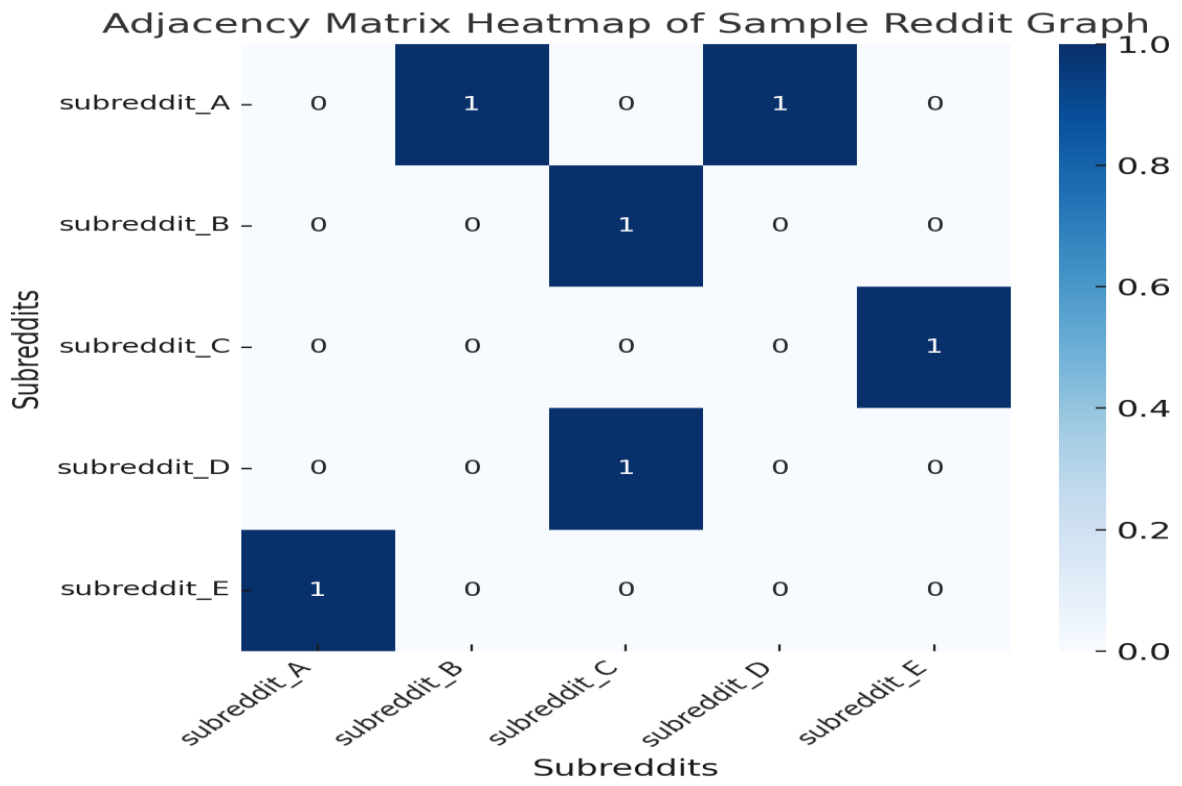

Fig. 8. Adjacency matrix of a sample Reddit graph

Figure 8 the heatmap above visualizes the adjacency matrix of a sample Reddit graph, where each cell represents the presence (1) or absence (0) of a directed hyperlink between subreddits. This dense representation aids in understanding connection patterns.

6.7. Edge Weight Distribution

Figure 9 shows the frequency of interactions (weights of edges) between subreddits.

- **Steps to Create**:
    1. Extract edge weights from the dataset and group them into bins.
    2. Plot a histogram of edge weights using tools like Matplotlib.
    3. Annotate the histogram to highlight highly active subreddit pairs

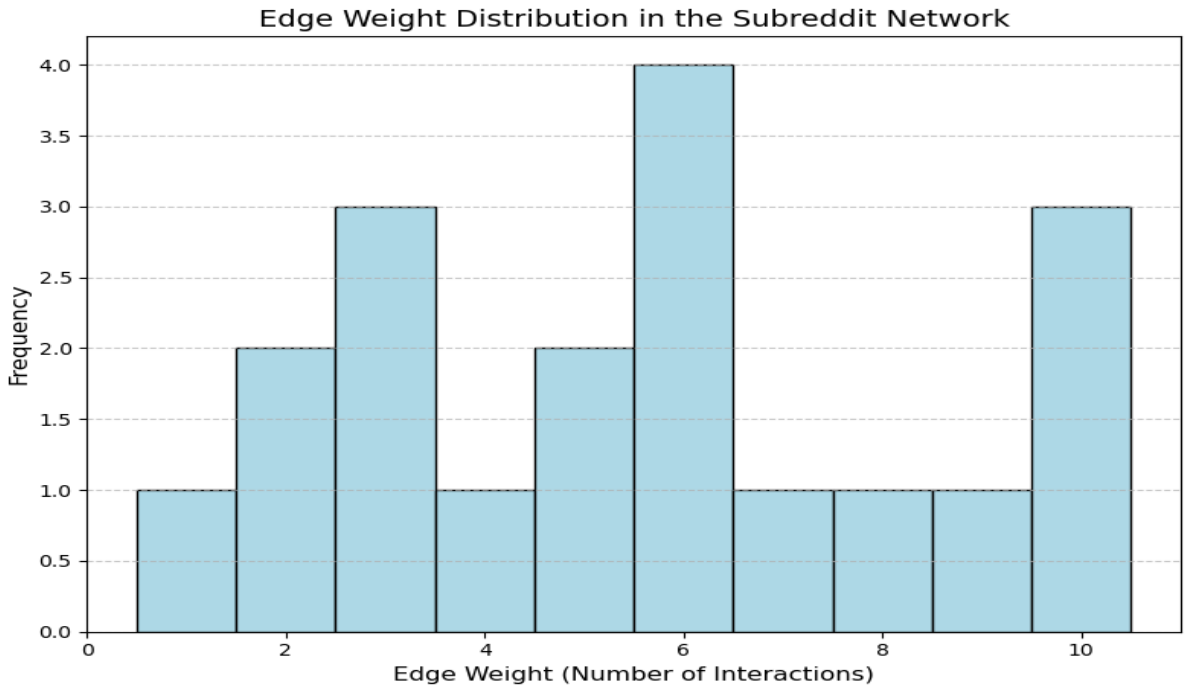

Fig. 9. Edge Weight Distribution

6.8. Quantitative Analysis

The performance of each algorithm is summarized in the table below, averaged over multiple train-test splits:

Algorithm	Precision@K	Recall@K	AUC
Common Neighbors	0.72	0.68	0.75
Jaccard Coefficient	0.65	0.63	0.71
Adamic-Adar	0.78	0.73	0.80
Preferential Attachment	0.61	0.59	0.67

- **Adamic-Adar** outperforms others in all metrics, particularly in sparse networks with high clustering coefficients.
- **Preferential Attachment** performs poorly, as it assumes that high-degree nodes dominate link formation, which is less applicable in this dataset.

6.9. Visualization of Results

Figure 10 is a bar chart summarizing AUC scores for each algorithm.

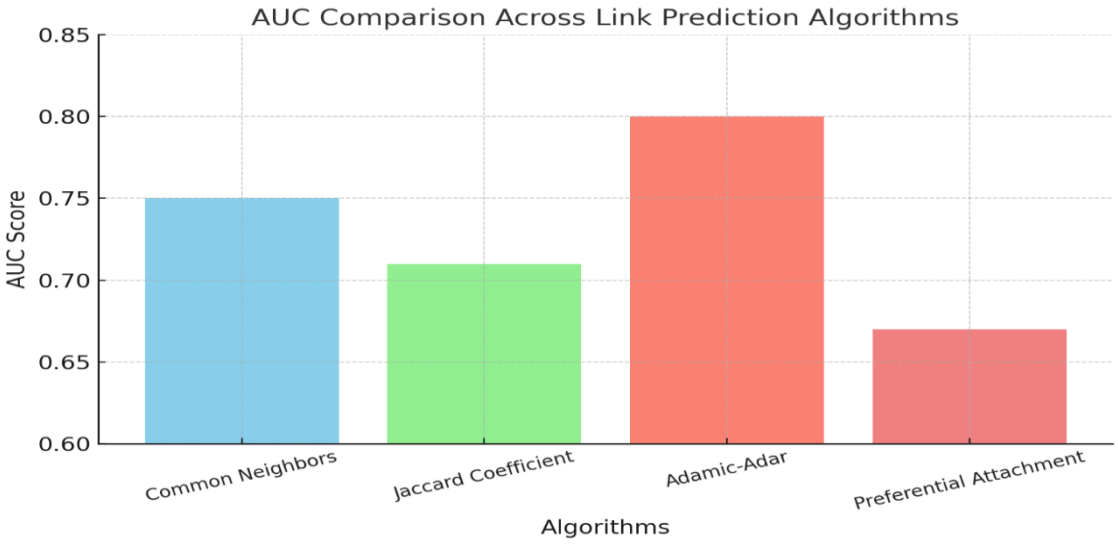

Fig. 10. AUC Comparison Across Algorithms

The bar chart above compares the AUC scores across the four link prediction algorithms. Adamic-Adar demonstrates the highest AUC, indicating its superior ability to distinguish between positive and negative edges in this dataset.

VII. Conclusion and Future Work

This paper evaluated four link prediction algorithms—Common Neighbors, Jaccard Coefficient, Adamic-Adar, and Preferential Attachment—on the Reddit Hyperlink dataset. Our findings suggest that Adamic-Adar and Common Neighbors are the most effective algorithms for link prediction in this context. These algorithms showed superior accuracy in predicting potential connections between subreddits, highlighting their utility in recommendation systems and social media analytics.

In future work, we plan to explore dynamic models that incorporate temporal information to predict evolving links over time. Additionally, incorporating attributes like user activity and topic similarity could further improve the accuracy of link predictions.

Future Directions

1. **Dynamic Models**: Incorporating timestamps and sequential patterns could better capture evolving interactions, especially for datasets like Reddit where trends shift rapidly.

2. **Attribute-Based Link Prediction**: Enriching the graph with subreddit metadata (e.g., number of subscribers, topic categories) could provide more context for predictions.

3. **Hybrid Models**: Combining classical algorithms with learning-based methods like Node2Vec or GCNs could yield both interpretability and higher accuracy.

4. **Scalability Enhancements**: Optimizing algorithms for massive networks through parallel processing or approximate computation would enable real-time link prediction for platforms like Reddit.

VIII. References

1. L. Lu and T. Zhou, "Link prediction approach based on graphlet sampling," *Proceedings of the International Conference on Neural Information Processing*, 2007, pp. 506–515.
2. S. S. G. Lee, "Social link prediction: The importance of attribute-based information," *Journal of Network Science*, vol. 5, no. 1, pp. 1-15, 2015.
3. L. Kunegis et al., "The Slashdot Zoo: Mining a social network with negative edges," *Proceedings of the 19th International Conference on World Wide Web*, 2010, pp. 741–750.
4. M. F. Newman, "Networks: An Introduction," Oxford University Press, 2010.
5. L. Adamic and E. Adar, "Friends and neighbors on the Web," *Social Networks*, vol. 25, no. 3, pp. 211-230, 2003.
6. H. W. Kai and P. T. M. Joseph, "Graph-based link prediction methods," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 7, pp. 1106-1116, 2011.
7. A. Clauset, C. Moore, and M. E. J. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, pp. 98-101, 2008.
8. J. Leskovec et al., "Predicting links in large-scale social networks," *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2005, pp. 228-237.
9. J. Tang, J. Sun, C. Wang, and Z. Yang, "Social Influence Analysis and Link Prediction Based on Multiple Network Structures," *Proceedings of the IEEE International Conference on Data Mining*, 2012, pp. 1152-1157.
10. H. Chen and S. Wang, "Link prediction approaches: A survey," *Computational Social Networks*, vol. 7, pp. 7-21, 2019.
11. Liben-Nowell, D., & Kleinberg, J. (2007). The link prediction problem for social networks. *Journal of the American Society for Information Science and Technology*.
12. Lü, L., & Zhou, T. (2011). Link prediction in complex networks: A survey. *Physics Reports*.
13. SNAP Project. (n.d.). Reddit Hyperlinks Dataset. Retrieved from https://snap.stanford.edu/data/soc-RedditHyperlinks.html.
14. Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. *ICLR*.
15. Grover, A., & Leskovec, J. (2016). Node2Vec: Scalable feature learning for networks. *Proceedings of the ACM SIGKDD*.
16. Zhou, T., Lü, L., & Zhang, Y. C. (2009). Predicting missing links via local information. *European Physical Journal B*.

17. Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. *Science*.
18. Newman, M. E. J. (2001). Clustering and preferential attachment in growing networks. *Physical Review E*.
19. Adamic, L. A., & Adar, E. (2003). Friends and neighbors on the web. *Social Networks*.
20. Liben-Nowell, D., & Kleinberg, J. (2007). The link prediction problem for social networks. *Journal of the American Society for Information Science and Technology*.