Link Prediction in Complex Networks

Miguel Lourenço Farinha (86798)

Instituto Superior Técnico
Master in Mathematics and Applications
Complex Network Science
1^{st} Semester- 2020/2021

Abstract

In real-world networks, the prediction of future connections can dramatically speed up network data collection and improve network model validation. Many algorithms have been proposed to accurately predict missing links. However, it remains unknown whether a single best prediction algorithm exists, how link predictability varies across methods and networks of different domains and with distinct topologies and how close to optimality current methods are. Here is showed that the performance of individual link prediction algorithms is dependent of the topology of the network and that no individual predictor can be considered best or worse through its application to synthetic networks with known structure and to real-world networks. Furthermore, it was found that the performance of the predictors varied by domain, link prediction in social networks was proved to be more accurate than in economic and biological networks. Finally, the combination of individual predictors into a single algorithm was showed to result in nearly optimal link prediction when applied to both synthetic and real-world networks. Therefore, the state of the art for link prediction comes from ensemble models which may propel future scientific breakthroughs.

1. Introduction

Networks are used to represent the structure of many complex systems. Nevertheless, real-world network data may be corrupted or incomplete. Incomplete networks may contain "missing links" which can influence its structure and properties, ultimately biasing and misleading scientific research and conclusions.

Algorithms capable of predicting which observed pairs of unconnected nodes should, in fact, be connected have broad utility. Such algorithms can be used in multiple tasks, such as extracting missing information or identifying false interactions. For instance, in many biological networks, such as food webs, protein-protein interaction networks and metabolic networks, to determine if a link between two nodes exists must be demonstrated by field and/or laboratorial experiments, which are usually very costly. Accurate prediction of such missing links based on known interactions can reduce experimental costs [18]. Therefore, accurate link prediction models may help understand the organizing principles of complex systems of all kinds.

Most real-world networks are relatively sparse, and the number of unconnected pairs in an observed network — each a potential missing link — grows quadratically, like $O(n^2)$ for a network with n nodes when the number of connected pairs or edges m grows linearly, like $O(n)$. The probability of correctly choosing by chance a missing link is thus only $O(\frac{1}{n})$ which explains why predicting missing links is a statistically hard problem [13].

Several link prediction algorithms have been proposed which belong to three main families: topological methods [17], model-based methods [13] and embedding methods [13]. Here only topological methods were discussed (see Section 5.2.).

All topological predictors appear to work relatively well [17]. However, the influence of the topology of the network on the general accuracy of the different predictors remains unclear. For instance, how does missing link predictability vary across network scales and topology and across different network domains (*e.g.*, social vs. economical)? Is there a single best predictor?

In order to answer these questions individual missing link predictors were evaluated on a variety of networks. Synthetic networks with known structure (topology) were generated (see Section 5.1). Following the literature [13], the topology of these networks varied in its degree distribution's variability, number of communities and fuzziness of the community boundaries. Furthermore, a large corpus of 550 structurally and scientifically diverse real-world networks was also analysed [1, 12].

Here it was concluded that the performance of the predictors was dependent of the topology and characteristics of the network as well as of the scientific domain of the network. For instance, it was found that missing links were easiest to predict in social networks. Predictability was higher with increasing variance of the degree distribution or with dense communities. Furthermore, no method performed best or worst across synthetic or realistic networks.

Following previous work [13], the metalearning approach of model stacking was adopted to the setting of network data (see Section 5.4). Specifically, a standard random forest was utilized to combine all predictors into one single algorithm [13]. Here we showed that this algorithm yielded nearly optimal predictions of missing links when evaluated on the generated synthetic networks and on the diverse corpus of real-world networks. This study was concluded by discussing limitations of the developed work and opportunities for further improvement.

2. Methods and Materials

Here the framework of missing link prediction, the network data and the predictor models were summarized; further details were provided in the Appendix.

Framework of the Problem and Evaluation Metrics. Consider an unobserved network $G(V, E)$, where V is the set of nodes and E is the set of links. According to a given function f, a subset $E' \subset E$ of links is observed. Denote by U the set of all possible links $(V \times V)$. Based on the pattern of the observed links E', a predictor will determine which unconnected pairs $X = U - E'$ are in fact among the set of missing links given by $Y = E - E'$. Specifically, the predictor will assign a score $s_{(i,j)}$ to each pair of unconnected pairs (i, j) in X so that higher scores are more likely to be assigned to missing links [18]. The accuracy of the predictors is usually quantified using the area under the receiver operating characteristic curve (AUC) [11] and the precision [11] metrics. The AUC value can be interpreted as the probability that a randomly chosen missing link $(i, j) \in Y$ (true positive) is given a higher score than a randomly chosen nonexistent link $X - Y$ (true negative) [18].

Network Data. A large and diverse corpus of 550 real-world networks belonging to six different scientific domains was utilized for the simulations [1]. This dataset was drawn from the CommunityFitNet corpus [12] and included networks from the social (23%), biological (32%), economic (23%), technological (12%), information (3%), and transportation (7%) domains. These networks spanned three orders of magnitude in size. Therefore, this dataset enabled the comparison of predictors across various scientific domains and dimensions of data (see Section 3.1).

In order to evaluate the predictability of missing links on networks with known structure, synthetic networks were generated. Firstly, networks with different variance of the degree distribution were studied. Specifically, networks following the Erdös-Renyi [8] and Watts-Strogatz [22] models (Poisson degree distribution) were generated as well as scale-free networks with a power-law degree distribution [3]. Secondly, networks with distinct number of communities or modules, $k \in \{1, 2, 4, 16, 32\}$ were generated. Finally, networks with different fuzziness levels of the community boundaries (ϵ) were generated. The latter two types of synthetic networks were generated utilizing random partition graphs [9]. Therefore, the generated synthetic networks ranged from random networks to scale-free networks by varying the degree distribution, from no modules to many modules (k), and from weakly to strongly modular structure (ϵ) (see Section 5.1).

Topological Predictors. This link prediction methods are simple functions of the observed network topology, *e.g.*, counts of edges, measures of overlapping sets of neighbours, and global measures derived from the network's structure [13]. These methods are classified as global, pairwise or node-based predictors (see Section 5.2). Pairwise predictors have been often used in the literature to directly predict missing links [18]. Following the mentioned approach, five pairwise predictors, which relied on the number of shared neighbours between nodes i and j, were studied in detail. Namely, the Common Neighbours (CN), Preferential Attachment (PA), Jaccard Coefficient (JC), Adamic-Adar Index and the Resource Allocation Index (RA) were evaluated as standalone algorithms (see Section 5.3). These methods were evaluated on the presented synthetic and real-world networks.

Ensemble Methods for Link Prediction. Ensemble methods are a powerful class of estimation algorithms that combine individual predictors into a single, more accurate algorithm which then is used to classify new data points by taking a (weighted or unweighted) vote of their predictions [11]. Ensembles methods generally improve the generalization performance of a set of classifiers on a domain, making them appropriate for hard problems like link prediction [13]. However, the individual classifiers should be accurate and diverse, i.e., they should make distinct errors and explore distinct features of the data. Here the results were obtained utilizing a standard random forest adapted to the setting of network data [13] (see Section 5.4).

3. Results & Discussion

The results and its respective discussion were carried out simultaneously throughout this report.

3.1. Real-World Networks

To evaluate and compare the performance of the different link prediction algorithms in a practical setting, a diverse corpus of networks was analysed. The networks considered for analysis were publicly available and were obtained from the CommunityFitNet corpus [12] which is a dataset containing 550 real-world networks drawn from the Index of Complex Networks [1]. The average degree as a function of the number of nodes for the networks grouped by domain was depicted in Figure 1.

Analysing Figure 1, one can see that the net-

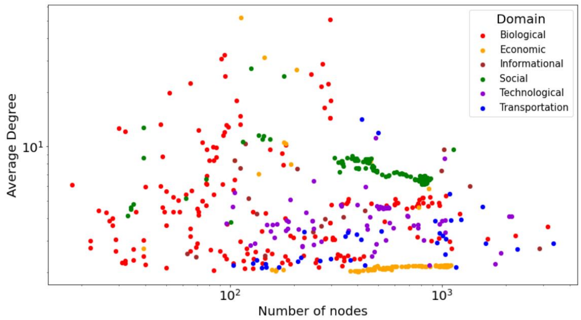


Figure 1: Average degree as a function of the number of nodes for the corpus of 550 real-world networks labelled by domain.

works spanned three orders of magnitude in size. The networks of each scientific domain were scattered across the plot, i.e., the networks had diverse structural global characteristics. This fact was relevant for the generalizability of the prediction results obtained through the evaluation of the various link prediction methods on this corpus of networks. One may infer that the mean degree of the networks was roughly independent of its size. Therefore, the networks with larger number of nodes and lower average degree were rather sparse. As stated previously, link prediction is a statistically hard problem due to this referred sparsity of the graphs. Notice that, in the subsequent analysis, each graph was treated as being simple, meaning that edge weights and directions were ignored.

3.2. Synthetic Networks

Individual predictors and their stacked generalization were also evaluated on a set of synthetic networks with known structure. These networks were generated using different models proposed in the literature. As stated above, the structure of these networks was varied along three dimensions [13]. Here was theoretically hypothesised how each dimension of the network structure would affect predictions of missing links.

Degree distribution. Networks with distinct degree distribution's variability were generated. Firstly, the Erdös-Renyi [8] model (ER) was utilized to generate random networks. The degree distribution of a random network follows a binomial distribution. Additionally, increasing the number of nodes (n) will increase the sparsity of the network (hampering predictability of missing links) and its degree distribution will converge to a Poisson distribution which has low variance [4, 7]. The study of the Erdös-Renyi model was also of interest since the optimal AUC could be derived [13]. In this model links between nodes are generated randomly. Therefore, one expected that the maximum accuracy obtained by any link prediction algorithm would be no better than chance, i.e, no predictor would be able to predict whether an unconnected pair of nodes (i, j) would be a missing link or a nonexistent link. Thus, for networks generated according to the Erdös-Renyi model the expected maximum accuracy was $AUC = \frac{1}{2}$. Several graphs with different number of nodes were created to prove the previous assertions (see Section 5.1).

Secondly, graphs were generated according to the Watts-Strogatz [22] model (WS). These graphs were generated due to their small-world properties. It has been shown that removing an edge from a clustered neighbourhood to make a short cut had, at most, a linear effect on the clustering coefficient [22]. Hence, the clustering coefficient as a function of the probability of rewiring each edge, p, remained practically unchanged for small values of p even though the average path length of the network dropped rapidly [7, 22]. Hence, the study of the impact of decreasing the clustering coefficient on the performance of the missing link predictors was of interest. From the literature it was known that increasing the referred probability resulted in the decrease of the clustering coefficient [7, 22]. On the one hand, for low values of the probability of rewiring each edge regular networks were obtained. Regular networks were not sparse and as a consequence there were few nonexistent links so that predictions were expected to be accurate for such networks. On the other hand, for high values of the probability of rewiring each edge random networks were generated. Random networks are sparse and consequently predictability was expected to be hindered. To corroborate the previous statements, graphs generated under the assumptions of the Watts-Strogatz model were obtained by varying the probability of rewiring each edge of the network (see Section 5.1).

Finally, networks with a power-law degree distribution were generated [3]. These scale-free networks possessed high degree distribution variance and resembled networks found in real-world applications. For networks with a large number of nodes the average degree values were not meaningful as the variance of the mentioned quantity diverged [4]. Therefore, these generated networks were composed of a relatively small number of hubs, nodes with higher degree, as well as large quantities of nodes with lower degree. It has been empirically demonstrated that the degree distribution of most real-world networks follows a power-law distribution with an exponent (γ) that is between 2 and 3 [4, 7]. On the one hand, for values of $\gamma < 2$ the average degree diverges so that such networks cannot be formed. On the other hand, for values of $\gamma > 3$ the generated networks behave as random networks. To that end, it was of interest to study how the power-law exponent of the networks affected the predictability of the missing links (see Section 5.1). Notice, that the previously discussed properties of these scale-free networks are obtained for infinitely large networks. Nevertheless, since the generated networks were finite, according to [21], there was a maximum number of connections a node could have. Therefore, despite the fact that the chosen values of γ were in the range of common values for the parameter for scale-free networks, it would not have been appropriate to classify the generated networks as scale-free. Instead following the classification of graphs proposed by [5] it would be appropriate to state that these networks should be classified as a broad-scale networks.

Number of Communities. Networks with different number of modules were generated (see Section 5.1). The value of the number of communities was chosen to be one of the following $k \in \{1, 2, 4, 16, 32\}$ [13]. These values were considered appropriate for the generalizability of the results as they covered a fairly large spectrum of k values. These networks were generated using random partition graphs [9]. Therefore, the generated networks were expected to have a random structure which may have translated into sparse graphs, thus hampering predictability. In an attempt to mitigate graph sparsity and obtain well-defined and dense communities, the probability of forming links between pairs of nodes of the same community was set to values close to one and the probability of having links between nodes from different communities was set to approximately zero. Thus, approximately k fully connected graphs, i.e., k regular graphs, were obtained for each generated network. This was corroborated by inspecting the average degree of these networks, which was found to converge to k as k successively increased. Theoretically, except for the case when $k = 1$, one would expect nearly optimal performances when evaluating predictors on these networks since approximately all nonexistent links correspond to edges between nodes of different communities which will be assigned a score of zero due to the lack of shared neighbours and/or common properties.

Fuzziness of Community Boundaries. Here only networks with $k \in \{2, 16\}$ number of modules were considered since these values guaranteed the generalizability of the results. Notice that it was not appropriate to study the fuzziness of the community boundaries for the network with a single community $(k = 1)$. Furthermore, it was found that for other values of k similar results were obtained. The fuzziness of community boundaries corresponds to the ratio of edges between the communities to edges inside the communities and is mathematically defined as $\epsilon = \frac{p_{out}}{p_{in}}$ [13]. Three fuzziness levels of community boundaries were considered to assess the predictability of missing links (see Section 5.1). Therefore, the fuzziness of community boundaries was chosen to be one of the following $\epsilon \in \{0.0015\ (low),\ 0.40\ (medium),\ 0.95\ (high)\}$. Specifically, the probability of forming links between pairs of nodes of the same community was held constant $(p_{in} = 0.25)$ and the probability of having links between nodes from different communities was varied according to the desired fuzziness level. These networks were also generated using random partition graphs [9]. As stated, the generated networks were sparse due to its random behaviour. For $\epsilon = low$ graph sparsity increased in comparison to the networks generated in the previous subsection, thus hampering predictability. For $\epsilon = medium$ and $\epsilon = high$, truly sparse random networks were generated. Therefore, it was expected that an increase of the fuzziness level of community boundaries would ultimately lead to less accurate predictions of missing links.

3.3. Individual Predictors on Synthetic Networks

The topology of a network determines the formation of new links [20]. Thus, one might speculate that the performance of individual link predictors is influenced by the topology of the network. Therefore, an experimental model was designed to evaluate, compare and understand the performance of five individual standalone topological predictors (see Section 5.3) on different synthetic networks with known structure (see section 3.2). For each of these generated networks, 30% of the existing edges were removed. Thus, following the previously defined notation, a subset $E' \subset E$ containing 70% of the edges of the network was observed. Thereafter, each individual pairwise predictor was evaluated on the incomplete network $X = V \times V - E'$ with the objective of predicting which unconnected pairs were in fact among the set of missing links given by $Y = E - E'$. To evaluate the predictability of missing links the AUC measure was utilized [11]. To enable the comparison of the results, the subset of removed edges and the training graphs remained identical throughout all simulations for each of the generated networks.

Effect of the Degree Distribution on Performance.

Individual predictors were first applied to the Erdös-Renyi model which was utilized to generate random networks (see Section 5.1). Inspection of the plots of the ROC AUC Curve for all pairwise predictors applied on each of the generated Erdös-Renyi networks corroborated the theoretically expected results (see Section 5.5). In fact, all predictors had an AUC value of approximately 0.50, i.e., predictors behaved as Chance Classifiers when applied on random networks. The performance of the Preferential Attachment method on the generated random networks was presented in Figure 2.

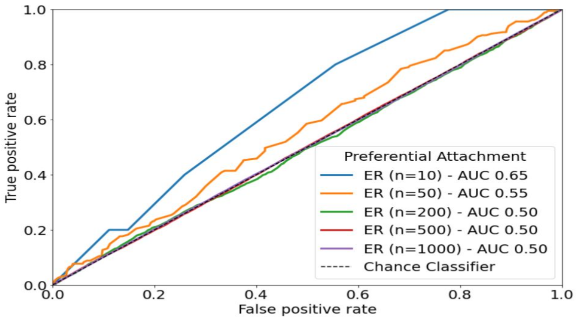


Figure 2: ROC AUC Curve for the Preferential Attachment predictor on random networks with different number of nodes.

Inspection of Figure 2 confirmed the theoretical suppositions. Furthermore, observation of Figure 2 revealed that when the random network had a reduced number of nodes the performance of the algorithm roughly outperformed that of a Chance Classifier. However, due to the reduced number of samples used for prediction the obtained performance was not considered statistically significant. From the analysis of Figure 2, it was verified that all predictors classified links as being missing links or non-existent links at random. Therefore, for increasing values of the number of nodes $(n \to \infty)$ the AUC values will converge to 0.50, according to the law of large numbers. Although the generated networks were finite it was possible to confirm the veracity of the previous statement. According to [4, 7], random networks become fractured when a finite fraction of its links is removed. By removing 30% of the edges of these networks, one may argue that some nodes may have disconnected from the network thus increasing its sparsity. Therefore, the sparsity of the training graphs may provide further insight into the obtained AUC values. Furthermore, due to the fact that the generated networks had a finite number of nodes, it would not have been appropriate to state that the degree distribution of these networks followed a Poisson distribution [7, 20]. Instead the degree distribution of these networks followed a binomial distribution.

Individual predictors were then applied to graphs generated according to the Watts-Strogatz model (see Section 5.1). As expected it was verified that decreasing the clustering coefficient of the network led to the decrease of the AUC values (see Section 5.5). The decrease of the clustering coefficient led to sparsity of the graph which caused an increase of the number of non-existent edges in the graph, which may have been misclassified as missing links. Concretely, increasing the probability of rewiring each edge of the graph led to the decrease of the AUC values, as depicted in Figure 3.

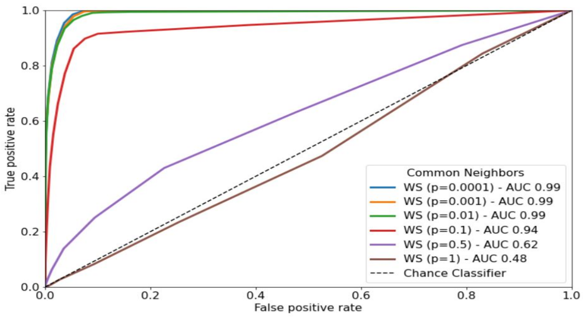


Figure 3: ROC AUC Curve for the Common Neighbours predictor on Watts-Strogatz networks as a function of the probability of rewiring each edge.

The majority of the studied pairwise predictors assigned scores to unconnected pairs (i, j) by quantifying the proportion of common properties/neighbours shared by the two nodes (see Section 5.3). Increasing the sparsity of the graph resulted in a decrease of the referred shared properties between each pair of nodes, thus hampering predictability. The previous statement was corroborated by Figure 3, predictability decreased as the probability of rewiring each edge increased, i.e., as the clustering coefficient decreased. Note that for $p \in \{0.0001, 0.001, 0.01\}$ nearly optimal predictions were obtained due to the reduced number of nonexistent edges in these networks (since the clustering coefficient was approximately one), as observed in Figure 3. When the probability of rewiring each edge of the graph was set to one a random graph was generated [22]. Figure 3 corroborated the previous statement, since the obtained value for the AUC was 0.50. Furthermore, it was found that no algorithm outperformed the others with the exception of the Preferential Attachment algorithm which yielded poorer results than that of a Chance Classifier (see Section 5.5).

Individual predictors were also used to make predictions in synthetic scale-free networks (see Section 5.1), i.e., on networks with a power-law degree distribution [4]. When the exponent of the power-law (γ) is in the interval $(2, 3)$, the average path length (APL) of scale-free networks scales with $\log(\log(n))$, where n is the number of nodes of the network, thus originating a super small-world effect [4, 7]. As a consequence of the scaling pattern of the APL in these networks, nodes will tend to be very close to each other. Therefore, the proportion of common properties/neighbourhoods shared by all nodes will be considerably large. In addition, the majority of the nodes is connected to the hubs of the network, thus neighbourhoods across the network will be locally well-defined. Accordingly, one would expect performances of the pairwise predictors to be nearly optimal. Furthermore, for values of $\gamma > 3$, one expects to obtain a maximum value for the AUC of 0.50. The plot of the ROC AUC Curve for the Adamic-Adar Index predictor evaluated on the generated scale-free networks as a function of the power-law exponent was depicted in Figure 4.

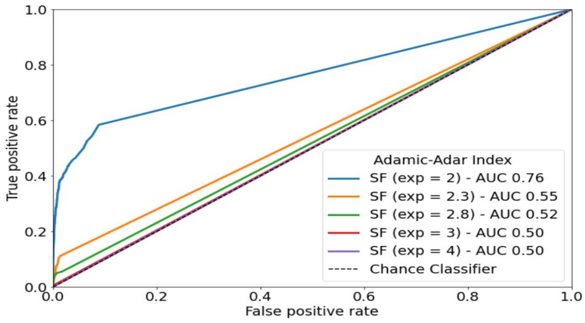


Figure 4: ROC AUC Curve for the Adamic-Adar Index on networks with different power-law exponents (γ).

An analysis of Figure 4 did not corroborate the previous statements. Nonetheless, increasing the value of the power-law exponent (γ) was verified to lead to the decrease of predictability. Moreover, for $\gamma > 3$ random networks were generated, proven by $AUC = 0.50$. As previously explained, the removal of 30% of the edges fractured some regions of the network, thus explaining the AUC obtained for $\gamma > 3$. Here the discussed results were obtained for finite networks [21]. Therefore, the patterns observed in Figure 4 referred, in reality, to broad-scale networks [5] which followed an exponential distribution for large values of the degree (k), due to the fact that the number of nodes with a large degree was reduced (high-degree saturation). Finally, it was shown that predictability was similar across all predictors, except for the Preferential Attachment algorithm which behaved as a Chance Classifier (see Section 5.5).

Effect of the Number of Modules on Performance.

Individual predictor performance was then tested on networks generated with a different number of communities. Graphs with $k \in \{1, 2, 4, 16, 32\}$ modules were generated and analysed (see Section 5.1). As stated previously, each generated graph had approximately k fully connected components. Thus, practically all nonexistent edges $(V \times V - E)$ for these networks connected pairs of nodes (i, j) belonging to distinct communities. These referred edges were assigned scores of approximately zero, since no shared neighbourhoods were found by the predictors for such pairs of unconnected nodes. Additionally, within the same community predictability was nearly optimal, since the majority of the 30% removed edges corresponded to connections between nodes of the same community. The plot of the ROC AUC Curve for the Jaccard Coefficient predictor evaluated on networks with different number of communities was depicted in Figure 5.

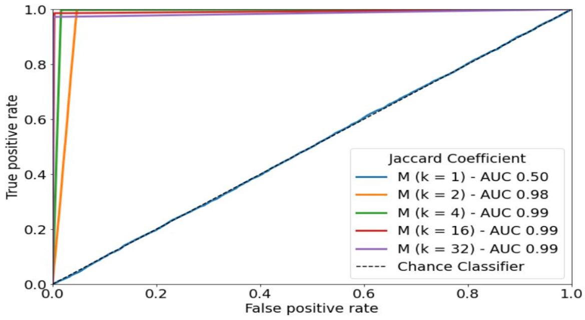


Figure 5: ROC AUC Curve for the Jaccard Coefficient predictor on networks with different number of communities (k).

Observation of Figure 5 confirmed the previous assertions. Firstly, for $k = 1$ (one community) predictions were random as the AUC value was 0.50. For $k = 1$, each node connected roughly with all the remaining nodes of the network in a random fashion. Consequently, the generated neighbourhoods for each of the nodes were practically identical and random, thus hindering predictability. Finally, as depicted in Figure 5, increasing the number of communities yielded nearly optimal predictions as proven by the observed AUC values. Specifically, unconnected pairs (i, j) within the same communities, which corresponded mostly to missing links, were assigned high scores whereas nonexistent links between nodes from different communities were assigned approximately null scores. Furthermore, all predictors had similar performances with the exception of the Preferential Attachment algorithm (see Section 5.6). These results were obtained under specific conditions appropriate to obtain networks with markedly defined and dense communities (see Section 3.2). However, such conditions may not be encountered in real-world data/applications.

Effect of the Fuzziness of Community Boundaries on Performance.

Pairwise predictor performance was evaluated on networks with one of the following fuzziness levels of community boundaries $\epsilon \in \{0.015\ (low),\ 0.40\ (medium),\ 0.95\ (high)\}$ (see Section 5.1). As expected, increasing the fuzziness of the community boundaries led to less accurate missing link predictions. This pattern may have been justified by the increased sparsity of the generated networks as well as the decreased community separability. The plot of the ROC AUC Curve for the Resource Allocation Index evaluated on graphs with different fuzziness of the community boundaries was depicted in Figure 6.

Inspection of Figure 6 corroborated the previous assertions. For low ϵ, the sparsity of the graph did

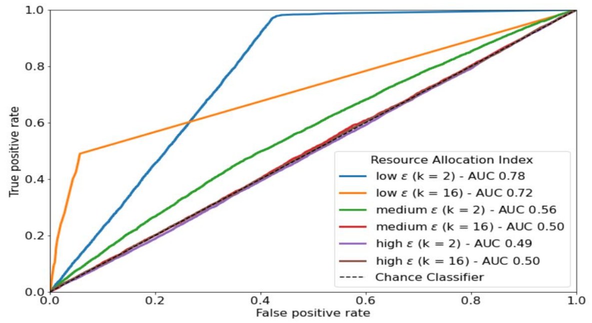


Figure 6: ROC AUC Curve for the Resource Allocation Index on networks with different number of communities (k) and different levels of fuzziness of the community boundaries (ϵ).

in fact hamper predictability as shown by the decrease of the AUC values. For medium and high ϵ, random networks were generated [9], chance predictions were obtained. Predictability was identical across the various pairwise predictors except for the Preferential Attachment predictor (see Section 5.7).

3.4. Individual Predictors on Real-World Networks

The real-world accuracy of the pairwise predictors was evaluated on the corpus of 550 real-world networks (see Section 3.1). The plot of the AUC values for the Adamic-Adar Index as a function of the number of nodes and the plot of the Common Neighbours predictor as a function of the average degree were depicted in Figure 7 and Figure 8, respectively.

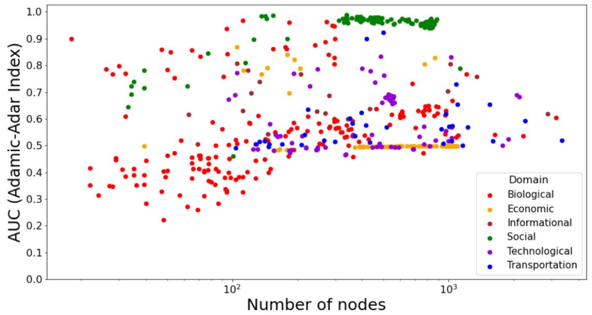


Figure 7: AUC for the Adamic-Adar Index as a function of the number of nodes for the 550 real-world networks.

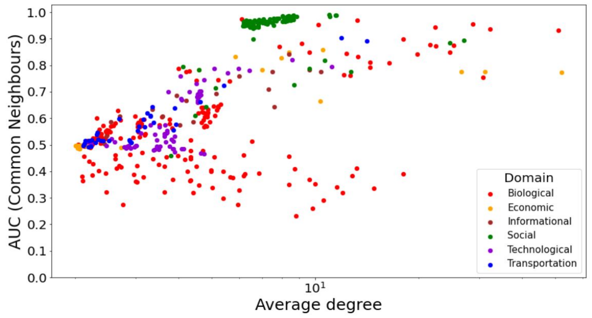


Figure 8: AUC for the Common Neighbours predictor as a function of the average degree for the 550 real-world networks.

From the analysis of Figure 7, it was concluded that predictors tended to be more accurate as the size of the networks increased. Inspecting Figure 8, it was observed that increasing the average degree of the network improved predictability. Therefore, contrarily to what would be expected, individual predictors had more accurate performances where link prediction was expected to be inherently harder. As demonstrated by the green points in Figure 7 and Figure 8, nearly optimal missing link predictions were observed on social networks. For instance, an average AUC of 0.94 was computed for the evaluation of the Adamic-Adar Index on social networks. Contrarily, predictability was harder in nonsocial domains, which may be explained by the fact that the considered predictors were originally designed for social network studies [17, 18]. The economic $(\langle AUC \rangle = 0.52)$ and biological $(\langle AUC \rangle = 0.55)$ domains had the lowest average accuracy values. Finally, it was concluded that no method performed best or worst across the analysed realistic inputs (see Section 5.8).

3.5. Stacking on Synthetic Networks

Here the accuracy of the implemented ensemble method [13] was assessed on the generated synthetic networks. For the entirety of the generated random networks, the stacked model performed no better than chance. Nevertheless, optimal or nearly optimal predictability was observed when the method was evaluated on all the non-random generated networks. For instance, predictability was nearly perfect in networks with high clustering coefficient (Watts-Strogatz networks), a power-law degree distribution (broad-scale networks [5]), many distinct communities (high k) and less fuzzy boundaries (low ϵ). Regardless of the synthetic network's topology it was found that the stacking method outperformed all the pairwise predictors. The performance of the stacked model was significantly closer to optimality than any of the individual predictors. Furthermore, all predictors utilized to assemble the stacked model were equally useful to identify missing links in the synthetic networks. Thus, it was concluded that no method could be considered best or worst across all inputs.

3.6. Stacking on Real-World Networks

To characterize the real-world accuracy of the implemented ensemble method [13], it was applied to the corpus of 550 real-world networks. Across networks and domains, it was concluded that there was wide variation in individual topological predictor importances, i.e., all predictors utilized to assemble the stacked model were equally useful to identify missing links, such that no predictor was best, or worst, on all networks. This variation of importances of the predictors confirmed that nearly optimal link predictions could only be achieved by combining the individual predictors into a single more powerful algorithm. The plot of the average Gini importance (mean decrease in impurity) [11] for predicting missing links for each of the five previously discussed pairwise topological predictors was depicted in Figure 9.

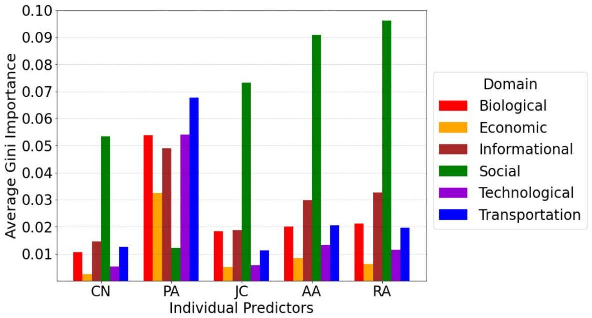


Figure 9: Average Gini importance for predicting missing links in networks within each of the domains for the five standalone pairwise predictors.

The higher the average Gini importance, the more useful a given predictor was for correctly identifying missing links. As observed in Figure 9, predictors exhibited different levels of importance across domains, thus indicating that no predictor was best overall. Furthermore, these topological predictors presented higher average Gini importances for social networks than for the remaining network's domains, as depicted in Figure 9. The same pattern had already been observed in Section 3.4. Social network data was the most common inspiration and application for link prediction methods [13, 18], thus explaining why predictability was best across social networks.

As expected, the implemented ensemble method produced nearly optimal missing link predictions across all the 550 real-world networks (see Section 5.9). Prediction performance (AUC) as a function of the number of nodes of the network was presented in Figure 10.

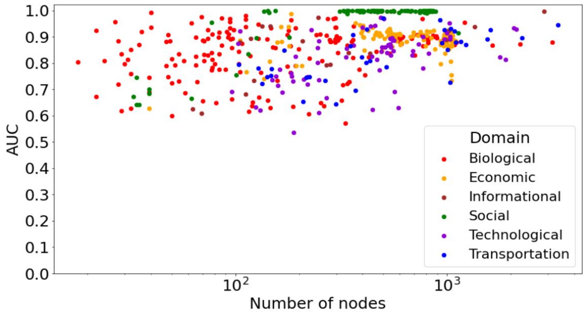


Figure 10: AUC as a function of the number of nodes of the network for the stacked model applied to the 550 real-world networks.

As observed in Figure 10, nearly optimal link prediction was achieved on social networks ($\langle AUC \rangle = 0.97$). In contrast, predictability was lowest on technological ($\langle AUC \rangle = 0.81$) and transportation ($\langle AUC \rangle = 0.82$) networks. In addition, predictions of the stacked model tended to be more accurate as the size of the network increased, as depicted in Figure 10. Comparing these results with the performances obtained by the individual predictors, represented in Figure 7, it was concluded that predictability was improved by combining all individual predictors into a single more powerful algorithm. Furthermore, the performance of the ensemble method on the real-world networks was in alignment with the results obtained on the synthetic networks (see Section 3.5).

4. Conclusion

The analysis of the individual topological predictors on the generated synthetic networks showed that predictability depended of the network's topology. When applied to random networks predictors behaved as Chance Classifiers. The performance of individual predictors was more accurate when networks were characterized by a high clustering coefficient, high variability of the degree distribution and many structurally distinct dense communities. Furthermore, no individual predictor could be considered universally best across all inputs and the Preferential Attachment predictor provided the least information. Predictability was more accurate as the size of the networks increased. The implemented ensemble method produced highly accurate predictions of missing links by combining several topological predictors into a single algorithm. The stacked model outperformed any of the individual predictors. When applied to synthetic networks, predictability was nearly optimal when the networks exhibited a power-law degree distribution (high variance), many dense and markedly separated communities and low fuzziness of the community boundaries. When applied to real-world networks, predictions were nearly perfect across all inputs. Network domain was found to influence predictability both for the individual predictors and for the staked model. Highly accurate predictions were produced in social networks.

This study was limited by the lack of computational power. In future work the size of the generated synthetic networks should be increased. Prediction methods from the model-based and embedding families should be evaluated. Moreover, alternative metalearning algorithms, such as XGBoost [6] and AdaBoost [10], should be studied. Future evaluations of new predictors, should be carried out in the context of metalearning, since they have been shown to achieve nearly optimal performance across a wide spectrum of inputs.

References

[1] Ellen Tucker Aaron Clauset and Matthias Sainz. The colorado index of complex networks (2016). https://icon.colorado.edu/.

[2] Lada Adamic and Eytan Adar. Friends and neighbors on the web. *Social Networks*, 25:211–230, 07 2003.

[3] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[4] A.L. Barabási. *Network Science*. 2016.

[5] L.A.N. Amaral, A. Scala, M. Barthélémy and H. E. Stanley. Classes of small-world networks.

[6] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA, 2016. Association for Computing Machinery.

[7] Francisco Correia dos Santos. Complex network science lecture notes 2020/2021. https://fenix.tecnico.ulisboa.pt/disciplinas/CRC7/2020-2021/1-semestre.

[8] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.

[9] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75–174, Feb 2010.

[10] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119 – 139, 1997.

[11] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[12] Amir Ghasemian, Homa Hosseinmardi, and Aaron Clauset. Evaluating overfit and underfit in models of network community structure. *IEEE Transactions on Knowledge and Data Engineering*, page 1–1, 2019.

[13] Amir Ghasemian, Homa Hosseinmardi, Aram Galstyan, Edoardo M. Airoldi, and Aaron Clauset. Stacking models for nearly optimal link prediction in complex networks. *Proceedings of the National Academy of Sciences*, 117(38):23393–23400, Sep 2020.

[14] Paul Jaccard. Etude de la distribution florale dans une portion des alpes et du jura. *Bulletin de la Societe Vaudoise des Sciences Naturelles*, 37:547–579, 01 1901.

[15] Gueorgi Kossinets. Effects of missing data in social networks. *Social networks*, 28(3):247–268, 2006.

[16] E. A. Leicht, Petter Holme, and M. E. J. Newman. Vertex similarity in networks. *Phys. Rev. E*, 73:026120, Feb 2006.

[17] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[18] Linyuan Lü and Tao Zhou. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, 390(6):1150–1170, Mar 2011.

[19] Mark EJ Newman. Clustering and preferential attachment in growing networks. *Physical review E*, 64(2):025102, 2001.

[20] M.E.J. Newman. *Networks: An Introduction*. Oxford University Press Inc., New York, 2010.

[21] S.N. Dorogovtsev, J.F.F. Mendes, A.N. Samukhin. Size-dependent degree distribution of a scale-free growing network. *Physical Review E*, (63), 2001.

[22] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[23] Tao Zhou, Linyuan Lü, and Yi-Cheng Zhang. Predicting missing links via local information. *The European Physical Journal B*, 71(4):623–630, Oct 2009.

5. Supporting Information

This section contains supporting information for the report. Detailed descriptions of the generated networks, predictors, ensemble model and complementary plots for Section 3 were presented below.

5.1. Generating Synthetic Networks

Here was described the generation processes for the synthetic networks. Under each subsection were described the chosen parameters and relevant auxiliary functions for each network model.

Generating Erdös-Renyi Networks. Random networks were generated using the Erdös-Renyi [8] model. The implemented method used the function erdos_renyi_graph implemented in NetworkX as an auxiliary function. The input parameters for this method were:

- number of nodes, n,
- probability of connecting each pair of nodes independently, p.

In this report's simulations, the parameter p was set to 0.50 which corresponded to the case where all $2^{\binom{n}{2}}$ graphs on n nodes were chosen with equal probability. The number of nodes (n) spanned over the following values $n \in \{10,\ 50,\ 200,\ 500,\ 1000,\ 5000,\ 7500\}$. Due to the lack of computational power it was not possible to analyse the networks with $n = 5000$ and $n = 7500$. Therefore, predictability was not assessed on these networks.

Generating Watts-Strogatz Networks. Graphs were generated according to the Watts-Strogatz [22] model. The implemented method used the function watts_strogatz_graph implemented in NetworkX as an auxiliary function. The method received as input parameters:

- number of nodes, n,
- number of nearest neighbours to which each node connected, k,
- probability of rewiring each edge, p.

Here, the number of nodes (n) was set to 250 since it was considered an appropriate value to evaluate predictability on the Watts-Strogatz model without leading to extremely computationally complex operations. The number of nearest neighbours to which each node connected to (k) was held at the constant value of 30. The probability of rewiring each edge (p) spanned over the following values $p \in \{0.0001,\ 0.001,\ 0.01,\ 0.1,\ 0.5,\ 1\}$. Thus, a regular network was generated for low p values whereas a random network was generated for high p values [7, 22].

Generating Scale-free Networks. Scale-free networks [3] were generated by implementing a function resorting to NetworkX that received the following input parameters:

- number of nodes, n,
- exponent of the power-law degree distribution, γ.

Here, the number of nodes (n) was set to 600. The choice of this value implicated that for large degree values k, the distribution failed to follow a power-law distribution. Since these network were finite, according to [21], there was a maximum number of connections each node could have. Therefore, it would not have been appropriate to classify the generated networks as scale-free. Instead, these networks should be classified as broad-scale networks as proposed by [5]. Nevertheless, due to the lack of computational power a value of 600 nodes was considered appropriate and sufficient to attempt to explain predictability on these networks. The exponent of the power-law degree distribution (γ) was defined to be one of the following values $\gamma \in \{2,\ 2.3,\ 2.8,\ 3,\ 4\}$.

Generating Networks with k Modules. Graphs with different number of communities were generated [9]. The implemented method used the function random_partition_graph implemented in NetworkX as an auxiliary function. The input parameters for this method were:

- number of communities (k) and size of each community, s,

- probability of node connections within the same community, p_{in},
- probability of node connections from different communities, p_{out}.

Here the number of modules used to generate the graphs was $k = \{1,\ 2,\ 4,\ 16,\ 32\}$. Hence, the number of nodes of each graph was fixed and only the number of communities was varied. The number of nodes (n) was set to 512 yielding:

- $k = 1$: 1 community with 512 nodes,
- $k = 2$: 2 communities with 256 nodes,
- $k = 4$: 4 communities with 128 nodes,
- $k = 16$: 16 communities with 32 nodes,
- $k = 32$: 32 communities with 16 nodes,

For these generated networks, the probability of nodes being connected within the same group (p_{in}) was set to 0.95 and the probability of having nodes from different communities connected (p_{out}) was set to 0.001. The chosen values aimed at creating very well defined and dense communities (the fuzziness of the community boundaries was very low).

Generating Networks with Different Fuzziness of Community Boundaries. Graphs with a number of communities $k \in \{2,\ 16\}$ were generated [9]. Furthermore, the fuzziness of the community boundaries $(\epsilon = p_{out}/p_{in})$, i.e., the ratio of links between the communities to links inside the communities was defined to three different levels, $\epsilon \in \{low,\ medium,\ high\}$. The implemented method used the function random_partition_graph implemented in NetworkX as an auxiliary function. The method received as input parameters:

- number of communities (k) and the size of each community, s,
- probability of node connections within the same community, p_{in},
- probability of node connections from different communities, p_{out},
- level of fuzziness of community boundaries, $level$.

Here, for these networks, the probability of node connections within the same community (p_{in}) was set to a constant value of 0.25. The value of the probability of node connections between nodes of different communities (p_{out}) was tuned so that the three different levels of fuzziness were obtained. Therefore, the following networks were generated:

- $k = 2$ (2 communities with 256 nodes) with $\epsilon = 0.0015\ (low)$ obtained by setting $p_{out} = 0.00375$,
- $k = 2$ (2 communities with 256 nodes) with $\epsilon = 0.40\ (medium)$ obtained by setting $p_{out} = 0.1$,
- $k = 2$ (2 communities with 256 nodes) with $\epsilon = 0.95\ (high)$ obtained by setting $p_{out} = 0.2375$,
- $k = 16$ (16 communities with 32 nodes) with $\epsilon = 0.0015\ (low)$ obtained by setting $p_{out} = 0.00375$,
- $k = 16$ (16 communities with 32 nodes) with $\epsilon = 0.40\ (medium)$ obtained by setting $p_{out} = 0.1$,
- $k = 16$ (16 communities with 32 nodes) with $\epsilon = 0.95\ (high)$ obtained by setting $p_{out} = 0.2375$.

5.2. Topological Predictors

In this study only topological predictors were analysed [17, 18]. These predictors belonged to one of the following three categories:

Global predictors. These predictors provided an overview of the global structure and topology of the network by computing a series of relevant global statistics. Global predictors were not accurate predictors of missing links. Nevertheless, they provided useful information when applied to supervised models (ensemble methods). Here were considered 8 global predictors, the number of nodes (N), number of observed edges (OE), average degree (AD), variance of the degree distribution (VD), network diameter (ND), degree assortativity of graph (DA), network transitivity or clustering coefficient (NT), and average (local) clustering coefficient (ACC) [20].

Node-based predictors. These predictors were functions of the independent topological properties of the individual nodes i and j, and thus produced a pair of predictor values [13]. These predictors could not be used to predict missing links, since they did not assign scores to the likelihood that nodes i and j formed a missing link. Instead, the particular function that converted the pair of node-based predictors into a score was learned within the supervised framework. The 20 node-based predictors were two instances each of the local clustering coefficient (LCC), average neighbour degree (AND), shortest-path betweenness centrality (SPBC), closeness centrality (CC), degree centrality (DC), eigenvector centrality (EC), Katz centrality (KC), local number of triangles (LNT), Page rank (PR), and load centrality (LC) [13, 20].

Pairwise predictors. These predictors were functions of the joint topological properties of the pair of nodes i, j being considered [13]. They may be referred to as similarity-based algorithms since these predictors assigned a score $s_{(i,j)}$ to each pair of nodes i, j, which was defined as the similarity or proximity of nodes i and j. All non-observed links were ranked according to their scores, and the links connecting more similar nodes were supposed to be of higher existence likelihoods [18]. Here 14 of such predictors were studied, the number of common neighbours of i, j (CN), shortest path between i, j (SP), Leicht-Holme-Newman index of neighbor sets of i, j (LHN) [16], personalized page rank (PPR), preferential attachment or degree product of i, j (PA), Jaccard coefficient of the neighbor sets of i, j (JC) [14], Adamic-Adar index of i, j (AA) [2], resource allocation index of i, j (RA) [23], the entry i, j in a low rank approximation (LRA) via a singular value decomposition (SVD) (LRA), the dot product of the i, j columns in the LRA via SVD (dLRA), the mean of entries i and j neighbours in the LRA (mLRA), and simple approximations of the latter three predictors (LRA-approx, dLRA-approx, mLRA-approx).

5.3. Standalone Predictors

Here a more thorough study of 5 pairwise predictors was conducted following their usage as standalone algorithms to predict missing links [18]. The studied predictors were the Common Neighbours (CN), the Preferential Attachment (degree product) (PA), the Jaccard Coefficient (JC), the Adamic-Adar Index (AA) and the Resource Allocation Index (RA). To understand the definition of each predictor let $\Gamma(i)$ define the set of neighbours of node i, i.e., $|\Gamma(i)|$ represents the degree of node i, (k_i). The definition of the majority of these predictors was based on the idea that two nodes i and j were more likely to form a link if $\Gamma(i)$ and $\Gamma(j)$ had a large overlap.

Common Neighbours (CN). In common sense, two nodes, i and j, are more likely to have a link if they have many common neighbours which may be an indication that they share common attributes. The simplest measure of this neighbourhood overlap is the directed count, namely

$$score_{(i,j)}^{CN} = |\Gamma(i) \cap \Gamma(j)|$$

This measure can be obtained from the adjacency matrix (A) of the network. Namely, $score_{(i,j)}^{CN} = (A^2)_{(i,j)}$. This algorithm was appropriate to analyse and predict new links in the context of social networks. In fact, this quantity has been used in the study of collaboration networks, showing a positive correlation between the number of common neighbours and the probability that two scientists would collaborate in the future [19]. In the context of large-scale social networks, it has been suggested that two students having many mutual friends were very probable to become friends in the future [15].

Preferential Attachment (PA). The preferential attachment method mirrors the “rich get richer” effect, i.e., nodes with more connections will be the ones to be more likely to get future connections. The mechanism of preferential attachment was used to generate evolving scale-free networks, where the probability that a new link was connected to node i was proportional to $\Gamma(i)$ [3]. Motivated by this mechanism, the corresponding similarity index can be defined as

$$score^{PA}_{(i,j)} = |\Gamma(i)| \cdot |\Gamma(j)| = k_i \times k_j$$

where k_i and k_j are the degrees of node i and j, respectively. This index does not require information about the neighbourhood of each node, as a consequence, it is the least computationally expensive algorithm. It also provides the lowest amount of information among all considered individual predictors [18].

Jaccard Coefficient (JC). The Jaccard coefficient, commonly used in information retrieval, measures the number of features that both nodes i and j have compared to the number of features that either i or j has, i.e., this metric measures the proportion of neighbours the pair of nodes (i, j) shares. It is defined as:

$$score^{JC}_{(i,j)} = \frac{|\Gamma(i) \cap \Gamma(j)|}{|\Gamma(i) \cup \Gamma(j)|}$$

Adamic-Adar Index (AA). The Adamic-Adar index refines the simple counting of common neighbours by assigning to the less-connected neighbours more weight, in other words rarer features that are common to both nodes will receive an heavier weight. This measure is defined as:

$$score^{AA}_{(i,j)} = \sum_{z \in \Gamma(i) \cap \Gamma(j)} \frac{1}{\log(|\Gamma(z)|)}$$

Resource Allocation Index (RA). The Resource Allocation index was motivated by the resource allocation dynamics on complex networks. Consider a pair of nodes, i and j, which are not directly connected. Node i can send some resource or information to node j, with the common neighbours of both nodes playing the role of transmitters. The similarity between nodes i and j can be defined as the amount of resource j received from i, which is defined as:

$$score^{RA}_{(i,j)} = \sum_{z \in \Gamma(i) \cap \Gamma(j)} \frac{1}{|\Gamma(z)|}$$

The Adamic-Adar and Resource Allocation indexes have similar expressions. Both measures reduce the contribution of the common neighbours of nodes i and j with high degree. The difference between the two measures is insignificant when the degree, $k_z = |\Gamma(z)|$, is small whereas it becomes considerable for large degree values. In other words, the RA index punishes the high degree of common neighbours more heavily than the AA index [18].

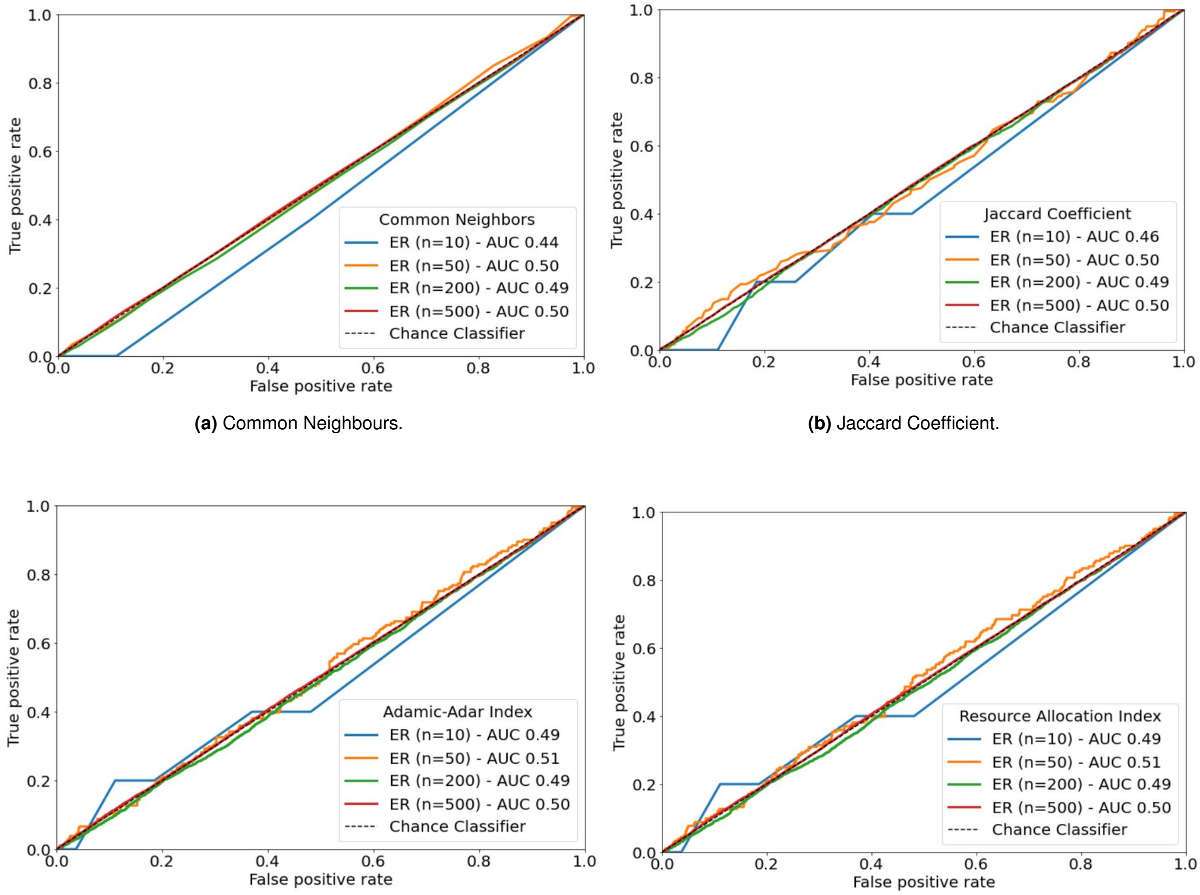
5.4. Stacked Generalization

Ensemble techniques are estimation algorithms that construct a set of classifiers and then classify new data points by taking a (weighted or unweighted) vote of their predictions [11]. Combining predictions of an ensemble is often more accurate than the individual classifiers that make them up. However, the classifiers should be accurate and diverse. By treating the output of individual prediction algorithms as features of the input instances themselves, a supervised meta-learning algorithm can construct a correlation function that relates which individual algorithm is most accurate on which subset of inputs [13]. Of the several approaches to meta-learning, we focus on the approach of stacked generalization or model “stacking”. Stacking aims to minimize the generalization error of a set of component learners.

In this study, the evaluations of the ensemble model assumed a missingness function f that sampled edges uniformly at random from the set of edges (E) of the network, so that each edge $(i, j) \in E$ was observed with probability α. Then for a given network $G = (V, E)$, the uniformly observed edges E' constructed the observed network $G' = (V, E')$, where $|E'| = \alpha|E|$ (α was set to 0.80). The removed edges $E - E'$ were considered as the held-out data in the link prediction task. Then, in order to train a model, $1 - \alpha'$ of the edges were removed (positive examples $E'' \subset E'$) according to the same missingness model f, and all non-edges U $(V \times V) - E'$ in the observed network G' were taken as negative examples. Following [13], here was implemented a standard random forest model with parameters chosen through 5-fold cross validation to maximize the F-measure on the training set. The results were reported on the holdout test set. The threshold for reported precision and recall is chosen to maximize the F-measure on holdout test set.

5.5. Effect of the Degree Distribution on the Performance of Individual Predictors

Erdös-Renyi Model. Each individual predictor was used to predict missing links on the random networks generated according to the Erdös-Renyi model (ER). The plots of the ROC AUC Curve for each predictor as a function of the number of nodes of the network were presented below:



(a) Common Neighbours.

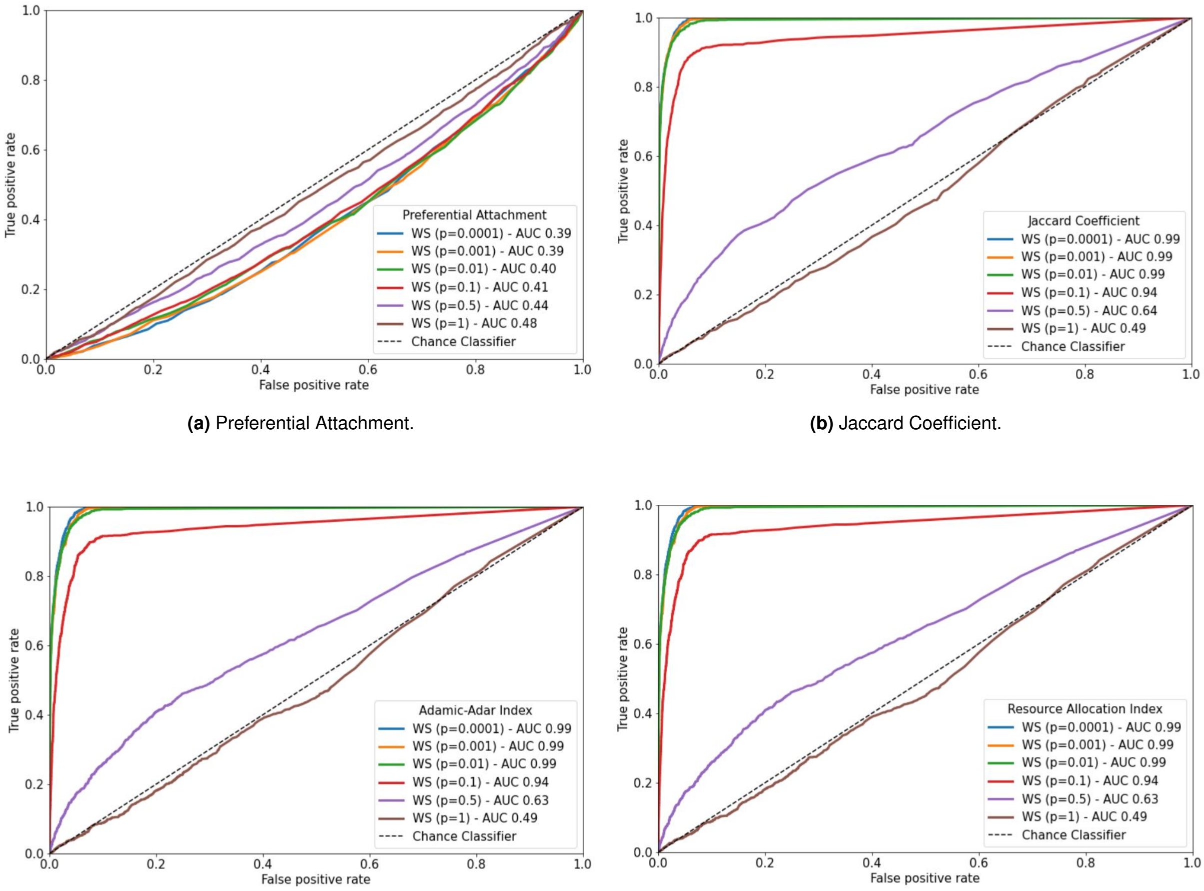
(b) Jaccard Coefficient.

(c) Adamic-Adar Index.

(d) Resource Allocation Index.

Figure 11: ROC AUC Curve for each predictor as a function of the number of nodes of the ER random network.

Watts-Strogatz Model. Each individual predictor was used to predict missing links on the networks generated according to the Watts-Strogatz model (WS). The plots of the ROC AUC Curve for each predictor as a function of the probability of rewiring each edge were depicted below:



(a) Preferential Attachment.

(b) Jaccard Coefficient.

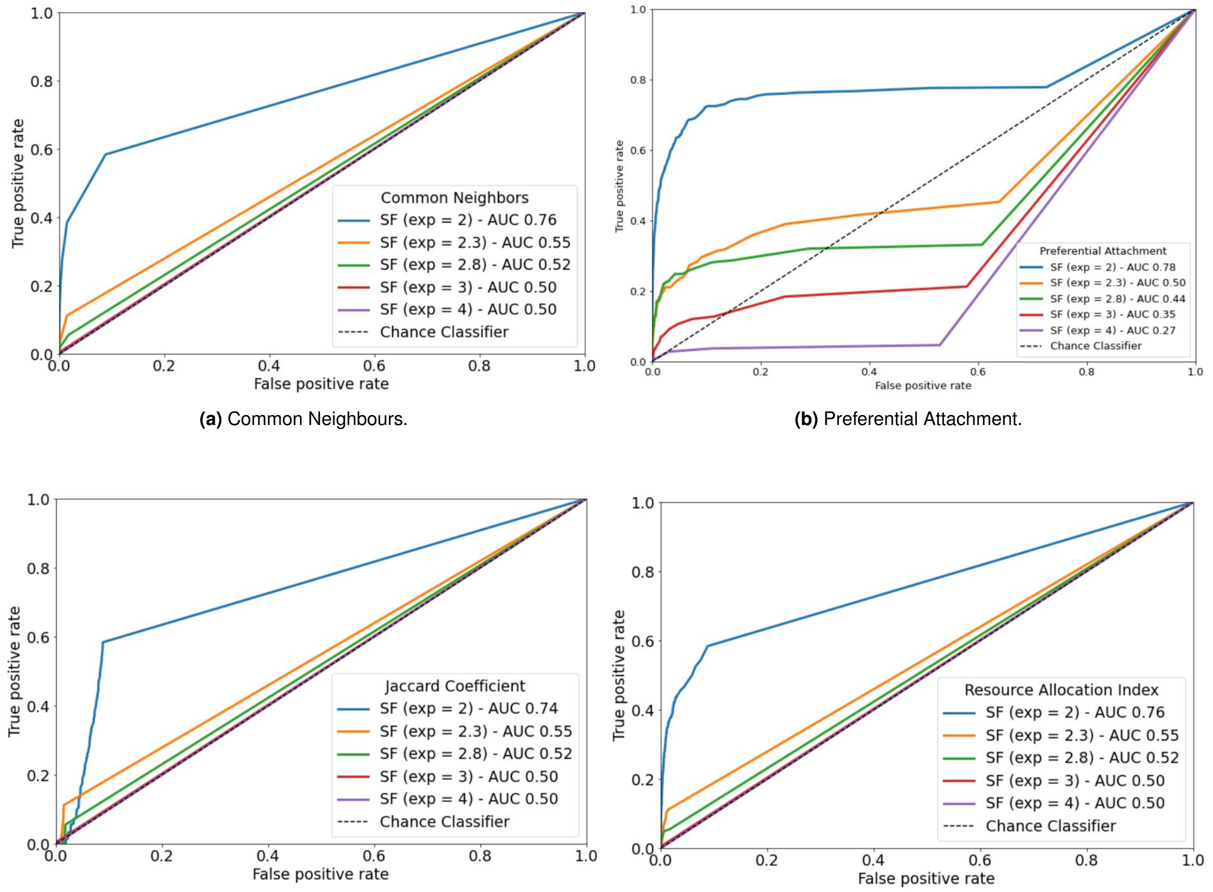
(c) Adamic-Adar Index.

(d) Resource Allocation Index.

Figure 12: ROC AUC Curve for each predictor as a function of the probability of rewiring each edge of the WS network.

As stated previously, all individual predictors had similar predictability with the exception of the Preferential Attachment algorithm. Preferential Attachment only took into account the degree of the two nodes for which the score was being computed. For the graphs generated according to the Watts-Strogatz model, its clustering coefficient presented a very high robustness to variations of the probability of rewiring each edge, i.e., the clustering coefficient remained approximately constant at a value of one, until a certain threshold probability was reached from which the clustering coefficient of the network began to decay [7, 22]. The high cliquishness of the local neighbourhood for all nodes implied that all nodes had similar degree values, which were rather high since the clustering coefficient of the network was high, i.e., the generated networks were dense. Consequently, most of the scores obtained by the Preferential Attachment predictor for each possible pair of unconnected nodes i, j were identical. Specifically, the algorithm could not capture any distinctive properties or information about the neighbours of the nodes of the network. This reasoning might explain the low accuracy values obtained when evaluating the Preferential Attachment predictor as observed in Figure 12 (Preferential Attachment). Therefore, it was concluded that the usage of the Preferential Attachment predictor should be avoided when the degree distribution of the graph was roughly homogeneous, i.e., when the variance of the degree distribution was low (which was the case for the Binomial/Poisson distribution).

Scale-free Network. Each individual predictor was used to predict missing links on networks generated with a power-law degree distribution. The plots of the ROC AUC Curve for each predictor as a function of the power-law degree distribution exponent were presented below:



(a) Common Neighbours.

(b) Preferential Attachment.

(c) Jaccard Coefficient.

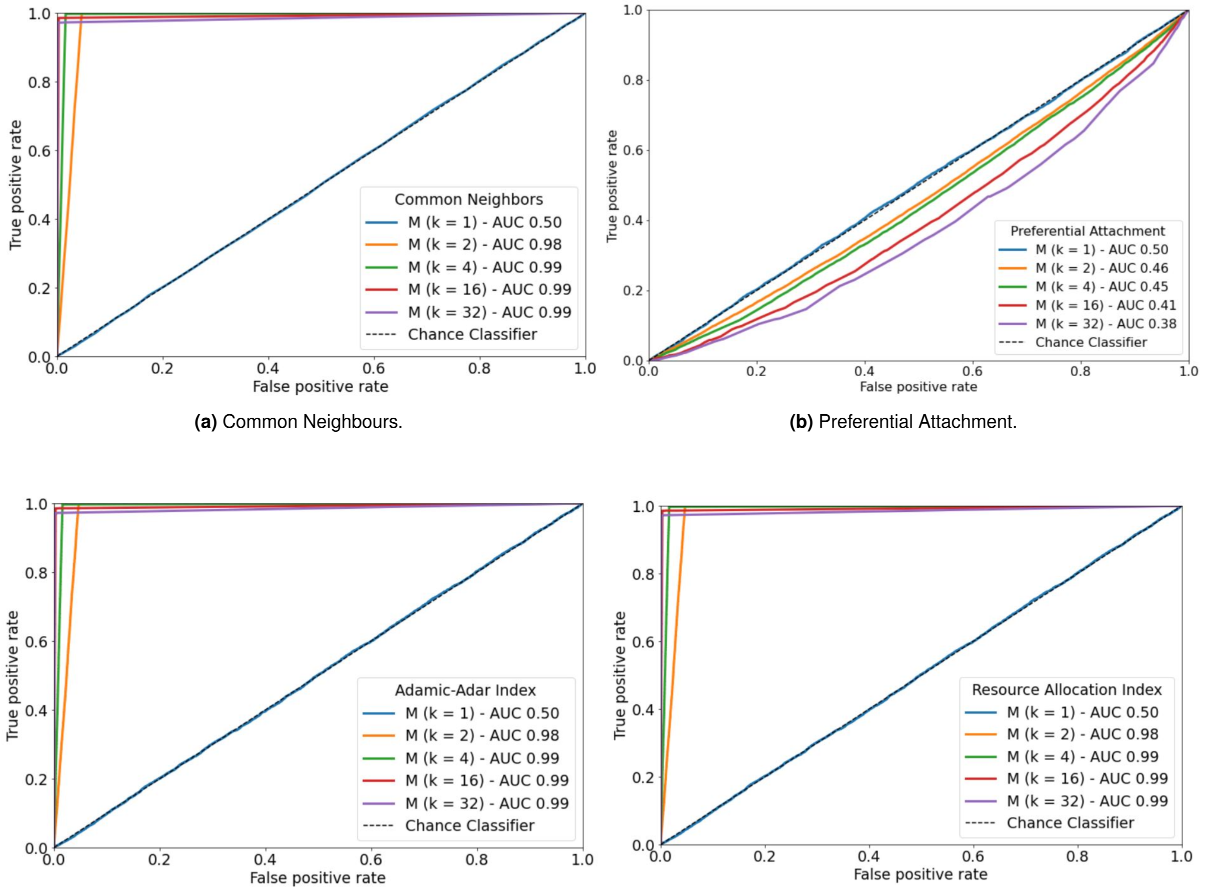
(d) Resource Allocation Index.

Figure 13: ROC AUC Curve for each predictor as a function of the power-law degree distribution exponent of the scale-free networks.

It has been theoretically shown that scale-free networks were robust to random failures. In other words, to break a scale-free network apart one would have to remove all nodes of the network [4]. Such robustness was rooted on the fact that random failures affected mainly the numerous small nodes, which played only a limited role in maintaining the network's integrity [4]. Such robust behaviour was also encountered when links were removed. Considering the designed experiment, in order to train the predictors some links of the generated networks were removed. Based on the previously explained theoretical results, one could infer that the removal of links from the generated scale-free networks did not alter the structure of the network. The process of edges removal, according to [20], affected roughly only nodes with low degree, which might have become disconnected from the network. Therefore, the topology of the networks did not suffer significant modifications and, as a consequence, the neighbourhoods for practically every node remained unaltered. As a result, the designed experiment would not interfere with the variance of the degree distribution of these networks, which would continue to follow a power-law distribution. Nevertheless, as noted, according to [5], the generated networks were, in fact, broad-scale networks. The tail of the distribution of these broad-scale networks as been shown to follow an exponential distribution [5, 7], thus explaining the performances observed in Figure 13.

5.6. Effect of the Number of Communities on the Performance of Individual Predictors

Each individual predictor was used to predict missing links on networks generated according to random partition graphs [9]. The number of communities of the generated networks was set to one of the following values $k \in \{1,\ 2,\ 4,\ 16,\ 32\}$. The plots of the ROC AUC Curve for each predictor as a function of the number of communities (k) were depicted below:



(a) Common Neighbours.

(b) Preferential Attachment.

(c) Adamic-Adar Index.

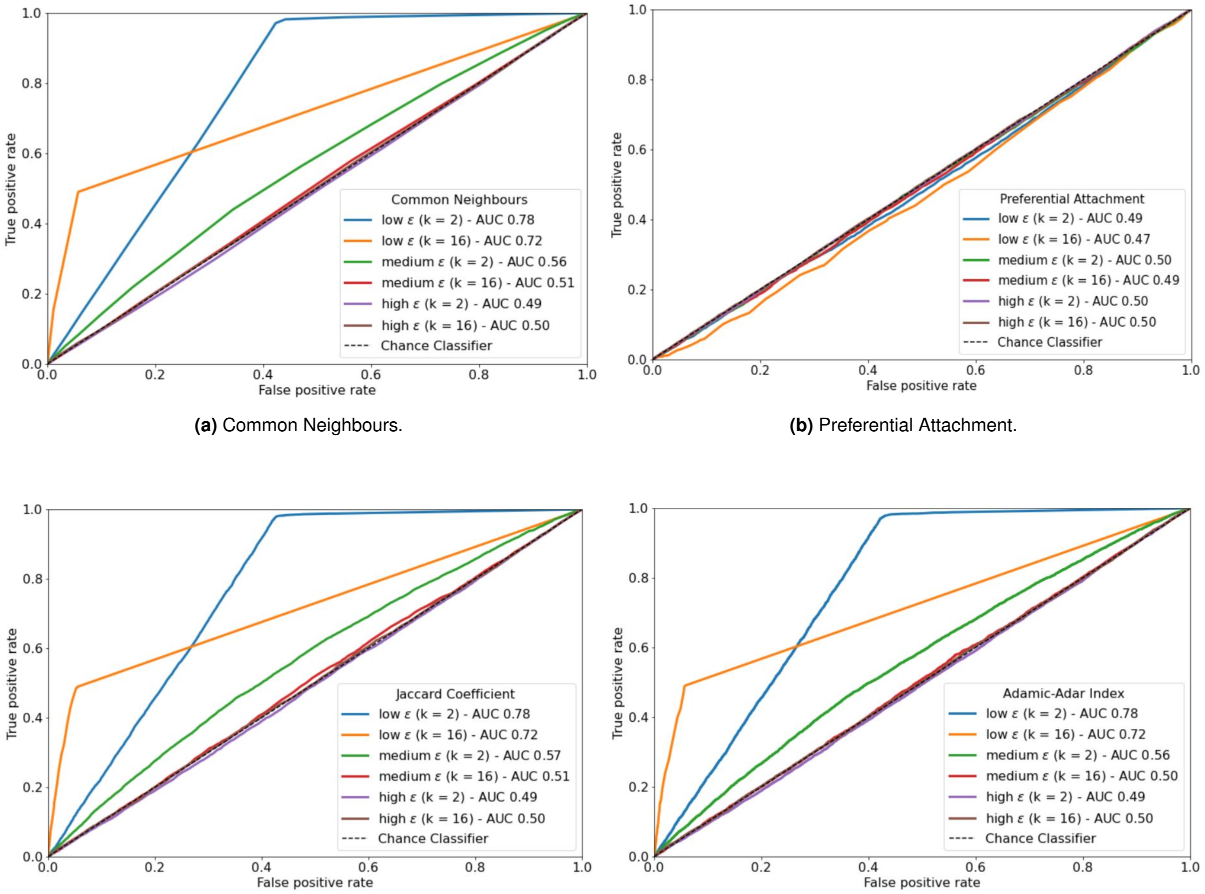
(d) Resource Allocation Index.

Figure 14: ROC AUC Curve of each predictor as a function of the number of modules of each generated network.

As observed in Figure 14 (Preferential Attachment), the accuracy of the predictions of the Preferential Attachment predictor was worse than that of a Chance Classifier. It was verified that increasing the number of communities resulted in the decrease of the average degree of the network. Specifically, the value of the average degree converged approximately to the number of nodes within each module of the network. For instance, for the graph with $k = 32$ (32 modules) and 16 nodes per community, the computed average degree was 14.74. Therefore, for each of the 32 communities of the network, a given node i was connected to practically all the remaining 15 nodes of the same community and there were roughly no connections to nodes from distinct communities. Moreover, since communities were dense and markedly separated the degree of every node was approximately constant throughout the entire network. Hence, the observed predictability for the Preferential Attachment method.

5.7. Effect of the Fuzziness of Community Boundaries on the Performance of Individual Predictors

Each individual predictor was used to predict missing links on networks generated with a total of $k \in \{2,\ 16\}$ communities and one of the following values of community boundaries fuzziness $\epsilon \in \{0.015\ (low),\ 0.40\ (medium),\ 0.95\ (high)\}$. The plots of the ROC AUC Curve for each predictor as a function of the fuzziness of the community boundaries of the network were presented below:



(a) Common Neighbours.

(b) Preferential Attachment.

(c) Jaccard Coefficient.

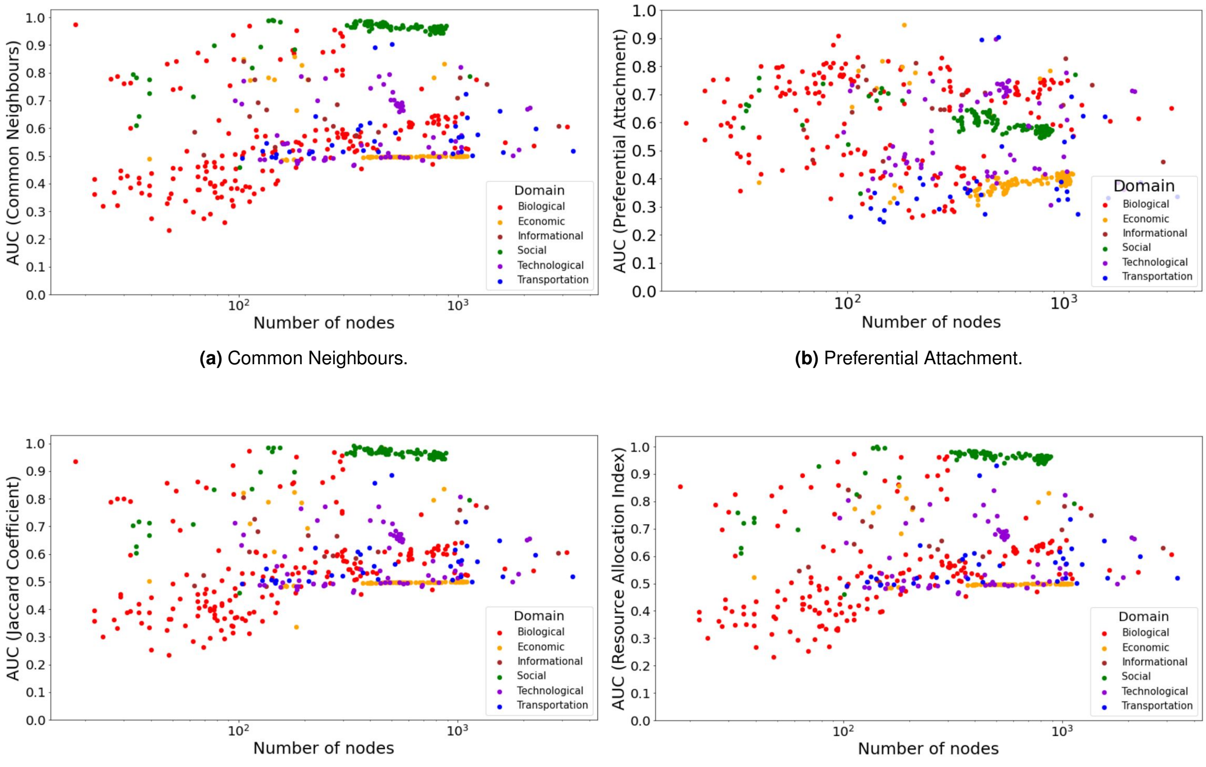
(d) Adamic-Adar Index.

Figure 15: ROC AUC Curve of each predictor as a function of the fuzziness of the community boundaries.

As observed in Figure 15, for low ϵ values, lower accuracies were obtained in comparison to the results obtained in Figure 14. The reduction of predictability resulted from the fact that less dense communities were generated, i.e., these networks were sparser, thus hindering predictions. For the medium ϵ and high ϵ values predictions were made at chance, as observed in Figure 15. Chance predictions were obtained since these generated networks were sparse and behaved as random networks.

5.8. Performance of Individual Predictors on Real-World Networks

Each individual predictor was evaluated on the corpus of 550 real-world networks. The plots of the AUC for each predictor as a function of the number of nodes of the 550 real-world networks were depicted below:



(a) Common Neighbours.

(b) Preferential Attachment.

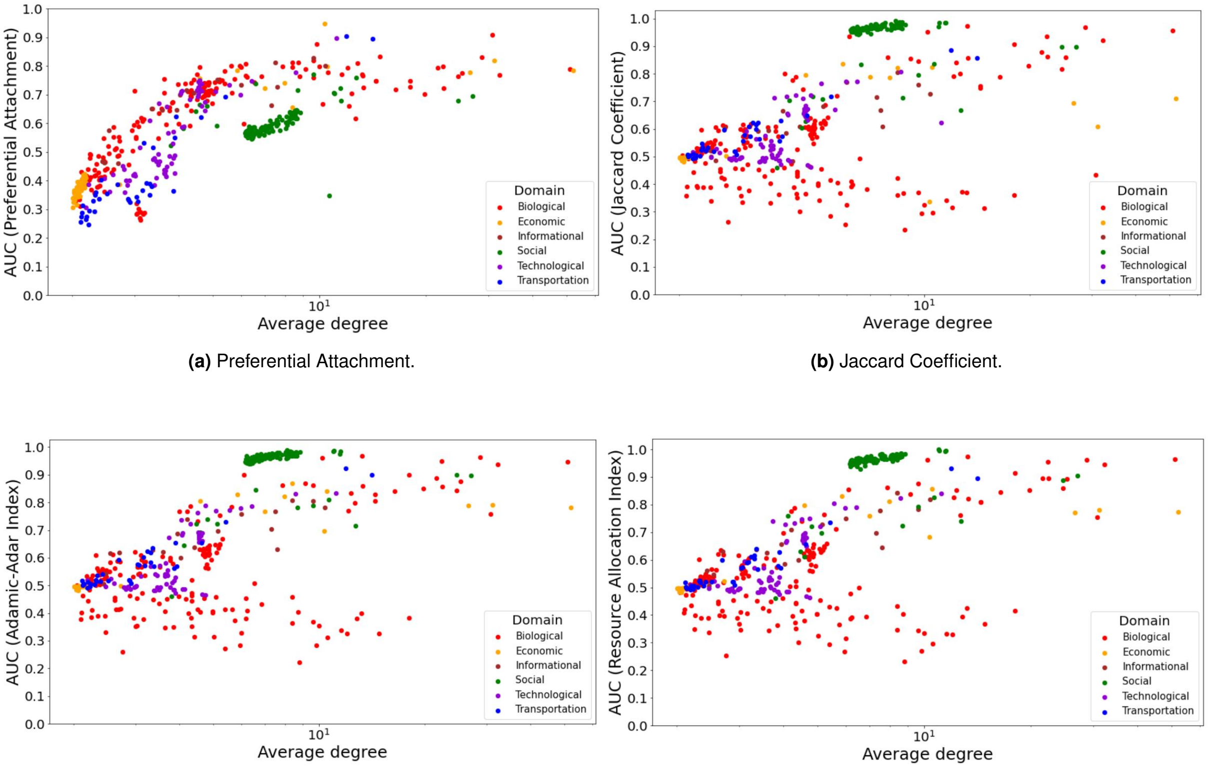
(c) Jaccard Coefficient.

(d) Resource Allocation Index.

Figure 16: AUC of each predictor as a function of the number of nodes of the 550 real-world networks.

As described in Section 3.4, individual link predictors were more accurate when evaluated on networks with a larger number of nodes. All predictors presented similar performance, as observed in Figure 16. The Preferential Attachment predictor had the worst performance. Nonetheless, this predictor was of interest since it required the least information and was the least computationally expensive method. Notice, for instance, as depicted in Figure 16, that the predictability of the Preferential Attachment predictor on economical networks was worse than that of a Chance Classifier.

The plots of the AUC for each predictor as a function of the average degree of the 550 real-world networks were presented below:



(a) Preferential Attachment.

(b) Jaccard Coefficient.

(c) Adamic-Adar Index

(d) Resource Allocation Index.

Figure 17: AUC of each predictor as a function of the average degree of the real-world networks.

As described in Section 3.4, predictors were more accurate when applied to networks with larger average degree. From the inspection of Figure 17 it was concluded that all predictions were approximately identical across the various predictors.

5.9. Ensemble Method on Real-World Networks

Prediction performance (AUC) as a function of the number of edges of each of the 550 real-world networks was presented in Figure 18.

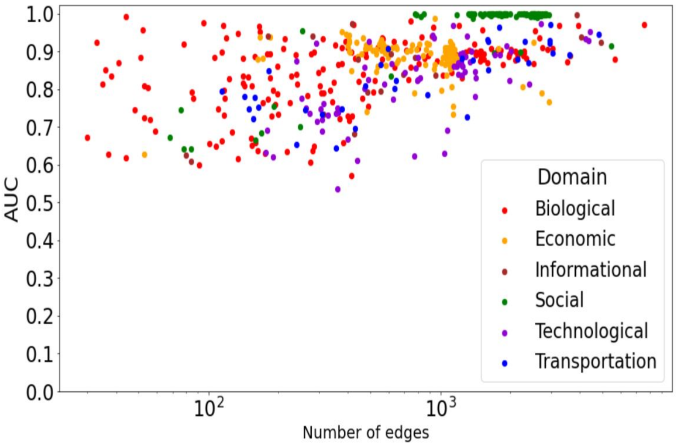
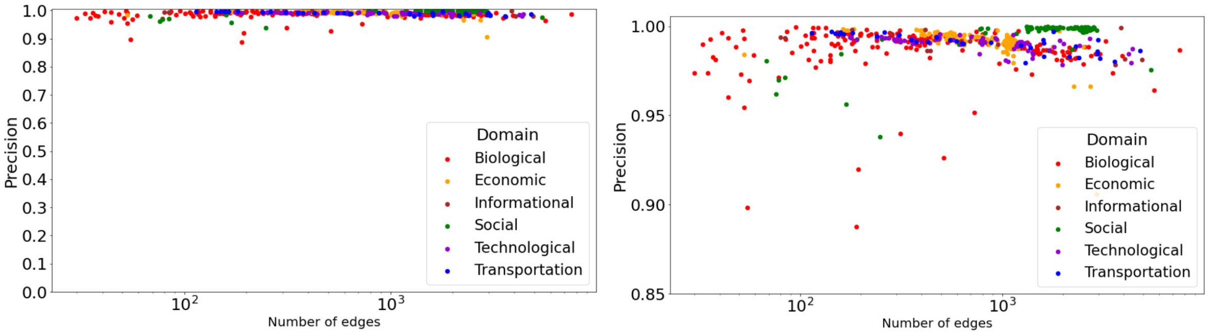


Figure 18: AUC as a function of the number of edges of the network for the stacked model applied to the 550 real-world networks.

Observation of Figure 18 corroborated the assumptions stated in Section 3.6. Predictability increased as the size of the networks increased. Furthermore, predictions were more accurate on social networks in comparison with any of the remaining domains.

Precision as a function of the number of edges of each of the 550 real-world networks was presented in Figure 19. Higher precision was equivalent to higher prediction accuracy.



(a) Precision values.

(b) Zoom of the plot on the left.

Figure 19: Precision as a function of the number of edges of the network for the stacked model applied to the 550 real-world networks.

As observed in Figure 19 (a), higher precision values were obtained when the stacked model was applied to the social networks. Precision also increased as the size of the network increased. Figure 19 (b) corresponded to the zoom of Figure 19 (a) for the region more densely populated.