Universiteit Leiden

Master Computer Science

Topology-aware Network Feature Selection in Link Prediction

Name:	Philippe P. Bors
Student ID:	s1773585
Date:	16/08/2022
Specialisation:	Data Science
1st supervisor:	Dr. E.S.C. Mattsson
2nd supervisor:	Dr. F.W. Takes

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

Across the different scientific domains, real-world complex networks have shown to differ vastly in structure and distribution of topological measures. These measures, aiding as the input variables for topological link prediction models, help a machine-learning model in its task of deciding whether a link exists or not in a network. It is not always immediately clear which set of measures should be chosen to represent the network.

Coming to aid are the scientific domains of origin. They help generalizing the task of selecting an appropriate set of measures, under the assumption that certain measures perform better on certain network domains. In this research, we investigate the individual importances of network measures in link prediction models and how these importances are distributed among the network domains.

In our experiments, we show that grouping networks by domain is only partially effective as a method for feature selection. Namely, we find that only social networks can be accurately distinguished from other domains. Most importantly, we introduce a new topological measure that we use to show that grouping networks by their global topological measure distributions is a better and more effective approach to feature selection.

Acknowledgements

I would like to foremost thank my first supervisor from Leiden University, Carolina Mattsson, for her day-to-day guidance of this research. The numerous meetings we had were always insightful, where her expertise helped me shape my project, and her contagious enthusiasm for and genuine interest in the subject always helped keeping me motivated.

My special thanks go to my second supervisor, Frank Takes, for his suggestions and constructive feedback that helped making this research more a more complete and coherent work. It is in his classes, that I developed a fascination for network science that I will always carry with me in life.

Furthermore, I would like to thank the entire CNS group from LIACS for their feedback and help during my intermediate presentation. The event helped me a great deal and the thesis would have looked different without it.

I would like to express my sincere gratitude to my in-laws, Cora, Matthijs and Lisan. Your love, care and precious support are something I would never be able to miss. As well as my mother and father, who helped me to pursue my studies.

Finally, I would like to thank my girlfriend Marieke for her continuous support. This would not have been possible in any way without you.

Contents

List of Figures

List of Tables

1 Introduction

Network science is a field of research that focuses on the connections and structures in networks, which originally emerged from real-world phenomena. Euler can be recognized as the first researcher in the field with his immensely popular paper nicknamed "The Seven Bridges of Königsberg" [Eul36] that was published in 1736, where a real-life walking cycle across different bridges is translated to what can be seen as the first actual problem in graph theory. Since then, innumerous advancements and contributions have been made by thousands of researchers that define our view on network science today.

Real-world networks originate from daily life and are maintained or extracted by companies, the government or researchers. An example are social networks, such as Facebook [Fac], that serve as an apparatus to stay connected with others, and global pandemic spread models [BFM+20], that help modelling a virus in case of a pandemic outbreak by simulating direct interactions between humans.

A graph is a mathematical construct that consists of two sets of elements: *nodes* and *edges*. Nodes can be connected by an edge, where the latter represents the link between two nodes. Serving as an example are social networks, where nodes can represent individual accounts, and links a friendship between two accounts. The conceptual representation of a network as a graph allows for deeper understanding of the complex topology that the network possesses, because these properties and their distributions can now be measured using an algorithmic approach as a function over the graph. These topological *network measures* are indicators of the structural properties they define. Many measures of the connectivity in a network exist, and each tell something different about the structure of the network in question.

Within this network structure, entities and their properties can be classified or predicted by a machine-learning model to serve different purposes. Some examples of these tasks are node classification on a publication network to classify the scientific domain of papers [KCKY20], anomaly detection on text data to find spelling errors [FBM21], and link prediction on an online social network to recommend new connections [KFA+16]. The *features* used in machine-learning on networks can be defined using topological, embedding, or model-based approaches [GHG+20]. Measures operate directly on the visible network structure, and the features based on these measures are therefore interpretable and explainable. Embedding methods use an embedding algorithm such as `node2vec` [GL16] that can serve as direct input to embed the network's vertices into a high-dimensional space to a machine-learning algorithm. Model-based approaches such as the Stochastic Block Model [HLL83] divide the nodes of the network in different communities and try to induce link probabilities from the community structure.

In this research, we focus on the topological approach to link prediction. This approach estimates the probability that a missing link exists or not, using the scores from one or multiple measure vectors. These are combinations of one or multiple computed measures over the network. In this context, the decision whether a link exists or not can be seen as a binary classification task. This task is to be performed by a *machine-learning model*. Statistical and machine-learning approaches have opened up new possibilities for classification by means of object classifiers. Examples of these decision-making algorithms are Naïve Bayes [Ris01], Quadratic Discriminant Analysis [GC] and Linear Regression [KY18].

All of these models require some kind of processable input. *Feature extraction* is a common step in machine-learning in which the actual observations are transformed to a latent or observable feature space. Topological measures capture connectivity in a network on a local scale and therefore help a machine-learning model by providing processable input. However, choosing the correct set of measures can prove difficult. Measure performance can be dependent on the type, community structure or even size of the network. Bias could be introduced to a model if the measure captures the wrong information due to the structure of the network. Moreover, the dimensionality of the algorithm will only increase (and efficiency decreases) if more and more measures are added as features. All of these issues together raise the question of how to select the correct feature set for a certain type of network.

A machine-learning model is heavily dependent on its training input data and how features correlate. At first glance, grouping networks by *network domain* seems a logical method to inherit and interpret feature importances. This method has also been suggested by prior work from Ghasemian et al. [GHG+20]. Real-world networks can span many different originating domains, but are roughly divided into six overarching domains: social, biological, economic, technological, informational, and transportation. This way of grouping networks has several advantages, such as not having to perform any prior calculations, using network meta information as a grouping factor.

However, there exists no guarantee that measures behave the same on different network domains. Although a large set of commonly used network measures and algorithms were specifically designed for social networks [KGZ15], suggesting their similar network structure, the structural differences within a domain can be large [GHG+20]. This serves as a lead that the performance of a measure is dependent on the structure of the network, rather than the originating domain. We could therefore argue that grouping by domain is too simple; this would especially be the case when the distributions of network measures within the domain itself would be large.

We aim to identify network groupings that effectively generalize the process of selecting good measures as features in link prediction. If we first observe a network *before* it is passed to the model, can we extract sufficient global topological information from it for us to be able to say something about using certain measures as features in general? In short, the three research questions to be answered are:

RQ1: Is grouping by domain an effective generalization method for selecting good measures as features for link prediction?

RQ2: How can we identify alternative groupings based on the network topology alone?

RQ3: What network measures are suitable for the task of link prediction on particular kinds of networks?

The structure of this thesis can be briefly explained as follows. In Section 2 we discuss recent developments and prior methods for feature selection and link prediction. Section 3 introduces our network dataset, mathematical fundamentals, approach to the problem and network features, as well as the machine-learning model for link prediction and its evaluation criteria. Section 4 contains our link prediction, topology-based grouping and feature selection experiments and results, after we will give a conclusion in the final section.

This page intentionally left blank.

2 Background and Related Work

Model stacking is a broad concept to which many ensemble type of models belong to. These models are best described by the idea of combining the output of multiple other models and passing them to another machine-learning model. This concept is also referred to as *meta learning* [HAMS20]. Popular approaches are bagging [KTP05] and Random-Forest models [Ho95] that average predictions from a collection of decision trees. In the context of complex networks and topological measures, the features for stacking are given by the topological measures that compute a function over the network. The stacking model itself can be any stacked machine-learning model that allows vector input.

Ghasemian et al. [GHG+20] developed such a stacking model for link prediction by including topological, embedded and model-based features classes into a single Random-Forest classifier. Their model performs near-optimal when error measures are compared to theoretical computed maxima. Another interesting outcome of this work lies in the contribution of each of the classes. Topological measures achieve considerable performance near-equal to the performance of all three classes combined, which exposes the apparent redundancy of embedded and model-based methods for link prediction in combination with topological measures.

Traditional social networks and their analysis has had large impact on algorithmic approaches now available for complex networks. Focusing mainly on clustering, neighbour similarity and the triadic closure, most measures available were specifically developed for social structure. Examples are the Adamic-Adar index [AA03] and the clustering coefficient [HL71]. These methods try to link similar nodes together and assume that a degree distribution is present in the network that follows a certain power law. Nodes in social networks are likely to be connected if they have relatively more common neighbours, facilitating triadic closure and shaping the network by the principle of homophily. This concept can best be described by the sentence: "A friend of a friend is also my friend." This triangular relationship between humans is most common in social networks.

Analysis by Mattsson et al. [MTH+21] gives insight into the opposite type of structure that emerges from production networks. So called functional structure shows tetradic closure and contains much less triangles compared to the structure of social networks. In functional structure, the network is shaped by the principle of complementarity: nodes are likely to bind if they are similar to the other's other neighbours. This is a property that is commonly found in protein-protein interaction networks [HZW+19]. The contrast in local connectivity with social networks is large, and the assumptions made in measures based on triadic closure are unlikely to hold for functional networks.

Kovács et al. [KLS+19] even speak of the *triadic closure principle paradox* – their research on protein-protein-interaction networks shows that the expected probability that two nodes interact decreases as the Jaccard coefficient (a measure for triadic closure) increases. This trait is completely opposite in social networks. Where common neighbours measures the number of unique length-2 paths between two nodes, they propose the number of length three paths as a feasible solution for measuring link probability in functional structure. Their experiments show that this measure outperforms common neighbours on protein-protein-interaction networks and the number of length three paths is a more stable predictor for networks falling in the functional spectrum.

This page intentionally left blank.

3 Methodology and Fundamentals

In this section, we describe in detail the dataset, approach to answering our research questions, topological features and machine-learning model that we use in the experiments section for link prediction. The features and the principles behind them will be used extensively throughout this thesis.

In line with answering our research questions, we put forward a method for deciding which network measures work well on which network groups. The first step in doing so, is by creating a machine learning model for link prediction. We apply this model to each of the 550 networks, which starts by sampling the edges into a *training network* and *test network*. For both sets of samples, a fixed set of topological measures are computed. These make up the actual training and testing data used in the machine-learning model, and are referred to as the training and test set. First, a model will be trained on the training set, where it will select the best hyper-parameters using cross-validation. At the end, we optimize the models by applying a hyper-parameter search using a parameter grid. The details of the machine-learning model are discussed in Section 3.3.

Our research questions are fixated towards the goal of finding out whether a grouping is an effective method for finding good link prediction features. Therefore, if there exists a lot of variance in model performance within the domains, we can conclude the grouping is ineffective. This effectiveness stands in a direct relation to the variance in structure of the network grouping. As discussed in Section 2, the differences in network structure make choosing correct measures difficult. If we find a grouping where networks are placed in groups that are highly similar in structure, we can expect them to behave equally for network measures, and therefore link prediction. To grasp potential differences in structure within the groupings that is important to our link prediction model, Section 3.4 looks at three different aspects of how the model behaves on the grouping, by analyzing *performance variability*, *global measures* and *feature importances*.

3.1 Network data

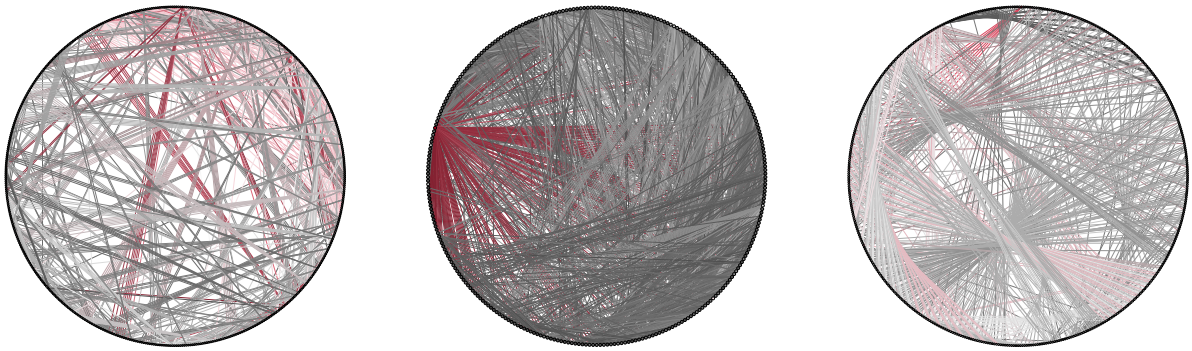
The dataset that we use is an expanded and revised version of the CommunityFitNet corpus [GHC20]. The exact same dataset is also used in the model stacking paper by Ghasemian et al. [GHG+20] and consists of 550 diverse real-world networks of different size, structure and domain. The data was is available via ICON [ATS], an open-source project containing the index of references to real-world complex networks. In 2017, ICON claimed to be several magnitudes larger than the second-largest network repository available for researchers.

Data specification The data we use contains basic information about the networks including the edgelist, network name and domain. Our dataset contains 124 (23%) social, 179 (32%) biological, 124 (23%) economic, 70 (12%) technological, 18 (3%) information and 35 (7%) transportation networks. The smallest network contains only 18 nodes and 30 edges, and the largest 3353 nodes and 7562 edges. On average, a network contains 510 nodes and 1155 edges. In this work, edge types are disregarded and all networks are therefore interpreted as undirected graphs. Three networks from the dataset are depicted in Figure 1.

3.2 Topological measures

In this section, we explain the mathematical foundations of the measures that are referenced in the remaining sections. These are the measures that are used in our link prediction model. Alongside already existing measures, we propose a new network measure for identifying square closure with the intent to capture new and additional topological information for link prediction.

In general, we observe an undirected network $G = \langle V, E \rangle$ with $n = |V|$ vertices and $m = |E|$ pairwise links, where two vertices u and v are connected iff $\{u, v\} \in E$. We also say that a link e_{ij} creates a connection between two nodes v_i and v_j. The direct neighbours (neighbourhood) of a node u are defined as $N(u) = \{v \mid \{u, v\} \in E\}$. The number of neighbours of u is referred to as the degree of u, $d(u) = |N(u)|$. In the context of link prediction, we observe *only* a subset $E' \subset E$ of links among the set of vertices V. The missing links $\bar{E} = (E \times E) - E'$ are to be predicted by a function (measure) $\delta : \{u, v\} \to \mathbb{R}$ that inputs the observed and incomplete network's missing links and outputs a score for



(a) Social.
Norwegian board of directors
$n = 520, m = 1814$

(b) Biological.
Roundworm metabolic rate
$n = 297, m = 2148$

(c) Technological.
ISCA89 circuit benchmark
$n = 491, m = 704$

Figure 1: Three networks randomly sampled from 3 domains, including meta information and in circular layout using variable diameters. Links are color-scaled to indicate high-degree nodes in red.

each of such a missing link $\{u, v\} \in \bar{E}$, based on the topological measures of the observed network. The higher the score, the higher the fitted probability that the link actually exists in the complete network. We use these formal notations throughout the definition of the topological features in this section.

We also distinguish between local and pairwise measures.

- Local measures are defined for pairs of nodes $\{u, v\}$ and typically return a scalar $f(u, v) = y$ for a subset of links in the network.
- Pairwise measures are constructed from node-specific measures. For example node degree can be defined for pairs of nodes $\{u, v\}$, such that the output of the function $f(u, v) = (y_1, y_2)$ is a tuple, with y_1 being the degree of the first node and y_2 of the second node. This is particularly useful in link prediction, where we can now capture node-level information on both ends of the link.

3.2.1 Local measures

Common neighbours Nodes with ties to the same neighbours are often said to be similar. The simplest measure of similarity between two nodes is the number of common neighbours that they share. Common neighbours is defined as the size of the set intersection of the neighbours of two nodes.

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

Jaccard coefficient Common neighbours requires scaling because the degree of a node may play a role. Nodes with higher degree have proportionally more ties and possibly more common neighbours. Therefore, we divide this number with the union of neighbors of both nodes. Respecting only the direct neighbours of the nodes, the Jaccard coefficient is defined as

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

Adamic-Adar index Based on the theory that a link of a node connected to a large hub is less meaningful in predicting links than a link from a node that is more isolated and therefore more unique, Adamic and Adar proposed inversely counting the common neighbours [AA03] (which is also inversely respective to the degree of a node). The Adamic-Adar index for node pair u, v is defined as

$$A(u, v) = \sum_{x \in N(u) \cap N(v)} \frac{1}{\log d(x)}$$

L3-score Expanding on the functional structure of networks as described in Section 2, Kovács et al. [KLS+19] define a new measure for link prediction on protein-protein-interaction networks. They do this based on the principle that proteins do not necessarily interact when they are deemed similar, but when their neighbours are. The measure is based on paths of length 3 and defined for two nodes u and v, including degree normalization, as

$$L_3(u,v) = \sum_{x,y \in V} \frac{a_{ux} a_{xy} a_{yv}}{\sqrt{d(x)d(y)}},$$

where $a_{ij} = 1$ if $\{i,j\} \in E$, and else $a_{ij} = 0$. The scoring in the numerator is corrected by the square root of the multiplied degrees of the iteration variables. Therefore, L3-score for pairs where either one of the nodes is an isolate is undefined. Kovács et al. showed that L3-score exceeds conventional similarity measures, such as common neighbours, in predicting links for protein-protein-interaction networks.

Shortest paths The shortest path between two nodes u, v is a sequence of nodes $(u, x_0, \ldots, x_\alpha, v)$ such that the length between the two nodes of such a sequence $\text{dist}(u,v) = \alpha$ is minimal. This number is also called the distance between u and v. Shortest paths are applied in many different network types, such as transportation networks (finding optimal routes), social networks (finding similar peers) or technical networks (internet routing). The most popular algorithm for finding shortest paths was developed by Dijkstra [Dij59] and runs in $\mathcal{O}(V^2)$ time, polynomial in the number of nodes in the network.

3.2.2 Pairwise measures

Node degree The degree of a node is equal to the number of neighbours that the node has. Intuitively, the maximum degree $d(u)$ of a node u is $d(u) \leq n-1$ when the network consists of n nodes. Nodes with degree 0 have no connection to the network and are called isolates. Nodes with degree 1 are situated at the outskirts of the vertex structure and are called leafs.

Average nearest neighbour degree A well known and widely used measure for capturing degree dependencies and relations between nodes is ANND. The ANND of a node is equal to

$$d_{nn}(u) = \frac{1}{d(u)} \sum_{v \in N(u)} d(v)$$

Dong et al. [YvdHL18] found that ANND scales with the size of the network, and could therefore not be used to directly compare networks of different sizes.

Number of triangles and squares The number of triangles $\lambda(u)$ is equal to the number of unique triangles a node participates in. The number of squares $\zeta(u)$ is defined analogously.

Clustering coefficient To measure clustering in networks, specifically social and real-world networks, the clustering coefficient [HL71] is used. Social networks often have knits of nodes that are densely packed together by ties. Near and between these so-called clusters, the likelihood of links existing increases above the level of the the average link probability in a graph. The local clustering coefficient [WS98] takes advantage of this property by measuring the extent to which these nodes cluster together; this means calculating the proportion of the links present in a node's neighbourhood. The local clustering coefficient

for a node u is the number of links that exist in the node's neighbourhood divided by the total number of possible links that could have existed (first expression).

$$C_2(u) = \frac{2 \cdot |\{j,k \ : \ v_j, v_k \in N(u), \{j,k\} \in E\}|}{d(u)(d(u)-1)}$$

Squares clustering coefficient The local squares clustering coefficient $C_3(u)$ of a node u is defined as the fraction of possible squares that exist at node u.

$$C_3(u) = \frac{\sum_{x=1}^{d(u)} \sum_{y=x+1}^{d(u)} CN(x,y)}{\sum_{x=1}^{d(u)} \sum_{y=x+1}^{d(u)} [a_u(x,y) + CN(x,y)]},$$

where $d(u)$ denotes the degree at node u and x and y are node labels, $CN(x,y)$ the number of common neighbors of x and y, $a_i(x,y) = (k_x - \eta_u(x,y))(k_y - \eta_u(x,y))$ with $\eta_u(x,y) = 1 + CN(x,y) + \theta_{xy}$ with $\theta_{xy} = 1$ if x and y are connected and 0 otherwise [LGH05]. Considering that conventional triangle clustering (as explained above) calculates the probability that two neighbours of v are connected, square clustering could by explained by the sentence: "The probability that two neighbouring nodes of v share another neighbour $w \neq v$."

Closure coefficient In [YBL19], Yin et al. introduce the local closure coefficient as a measure that quantifies head-node-based link clustering. Instead of the "two of my friends are also share a friendship" used in local clustering (Figure 2-A), local closure sheds new light on triadic closure. So far, we discussed clustering techniques where a local value can be computed for a node u residing in the center of its neighbours. A different way of reasonable thinking in this aspect, especially for social networks, would be the principle of "a friend of a friend is also my friend" (Figure 2-B).

As an alternative to the center-based approach, the local closure coefficient is defined for a head node u. Firstly, a 2-wedge W_2 is an ordered pair of edges $W_2 = (e_1, e_2)$ that share exactly 1 common node, also called the center of the wedge. The head of a wedge u is defined as the other endpoint at the beginning of the wedge. A wedge is called closed, if the head and the tail of the wedge are directly connected. For example, in Figure 2-B, (u, v, w) is an open length-2 wedge.

The local closure coefficient $H_2(u)$ for a node u is defined as the fraction of closed wedges for which u is the head node.

$$H_2(u) = \frac{2\lambda(u)}{W_2^{(h)}(u)},$$

where $W_2^{(h)}(u)$ is the number of 2-wedges where u is the head. Each triangle consists of two closed wedges, hence the 2 in the numerator. Yin et al. found that including the local closure coefficient as a covariate in machine-learning improves link prediction done on social networks. Including this feature in a model for multiple domains will show if the local clustering coefficient can prove useful in a more broad application.

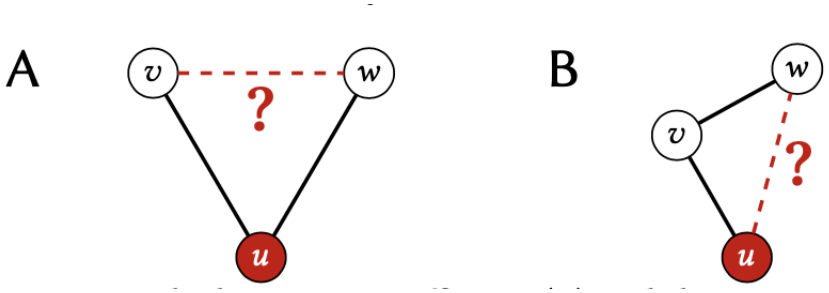


Figure 2: The local clustering coefficient (A) and the local closure coefficient (B) with center- and head node u. Figure taken from the paper from Yin et al [YBL19].

3.2.3 Novel measure

Squares closure coefficient We extend the notion of closure to tetradic (and afterwards to arbitrary) form by defining the local squares closure coefficient as the fraction of length-3 wedges that induce an length-4 cycle (a path of unique vertices of which only the first and last vertex are the same).

$$H_3(u) = \frac{2R_4(u)}{W_3^{(h)}(u)},$$

where $R_4(u) = \zeta(u)$ is the number of unique length-4 cycles trough node u and $W_3^{(h)}(u)$ the number of length-3 wedges headed at node u. The basic principle for this type of local closure coefficient H_ℓ for wedges of length $\ell > 2$ can be analogously defined as:

$$H_\ell(u) = \frac{2R_{\ell+1}(u)}{W_\ell^{(h)}(u)}.$$

Now, $R_\ell(u)$ is the number of unique length-$(\ell+1)$ cycles trough node u and $W_\ell^{(h)}(u)$ the number of length-ℓ wedges headed at node u. A depiction of the idea behind square closure and clustering can be found in Figure 3. Whereas squares clustering coefficient measures the proportion of closed squares around a node u, the squares closure coefficient in Figure 3b measures this proportion only for tetradic closure where u is the head node.

(a) Squares clustering, center-based.

(b) 3$^{\text{th}}$-order closure, head-based (proposed).

Figure 3: Squares clustering coefficient and the newly introduced closure coefficient.

3.2.4 Overview of features

In concreto, the measures explained in Section 3.2 that are used in link prediction model are summarized in Table 1 above, including their types and notation. All of these methods define a score for a pair of nodes (local) or for both nodes individually (pairwise).

	Description	Notation	Type
Link-based	Common neighbours	CN	Local
	Jaccard coeff.	J	Local
	Adamic-Adar index	A	Local
	L3-score	L_3	Local
Quasi-link-based	Shortest path	P	Local
Node-based	Degree	d	Pairwise
	Triangle count	λ	Pairwise
	Square count	ζ	Pairwise
	Clustering	C_2	Pairwise
	Square clustering	C_3	Pairwise
	Closure	H_2	Pairwise
	Square closure	H_3	Pairwise
Quasi-node-based	Neighbour degree	d_{nn}	Pairwise

Table 1: All network measures used as topological predictors in this research.

3.3 Link prediction model

In this subsection, we explain our choices made in the set-up and implementation of our machine-learning model for link prediction. We will be sampling the training set and creating the corresponding features. In our case, the topological features can be used as input for the link prediction task. These features carry topological meaning with regard to the graph that is analyzed and represent its structural properties on a local (node/link-based) level. All the local features together make up the feature matrix of a graph, where the columns represent local entities of the graph such as nodes or links, and the rows represent the different network measures. Machine-learning models rely on structured data of invariable length, which is why a fixed set of features is chosen as representatives of the graph data.

3.3.1 Sampling edges

Creating a representative training and test set for our model is an important step in the process of preventing model bias. In machine-learning, it is common practise to split the set of individual examples into two sets: one for training and one for testing. In Figure 4, we observe the structure of our machine-learning model. As its name reveals, the training set is used to train the model, or more specifically, find the best fit for its parameters. In turn, the test set is used to evaluate the model after training. The combination of samples in this dataset have not been seen by the model during training and are therefore of evaluative importance. Both datasets should contain both positive and negative examples of missing edges, ideally with an even distribution, so there exists no class imbalance. The first step in the process of creating training/test data is creating a representative sample of edges, and is divided into two steps: sampling the original network and sampling true and false edges.

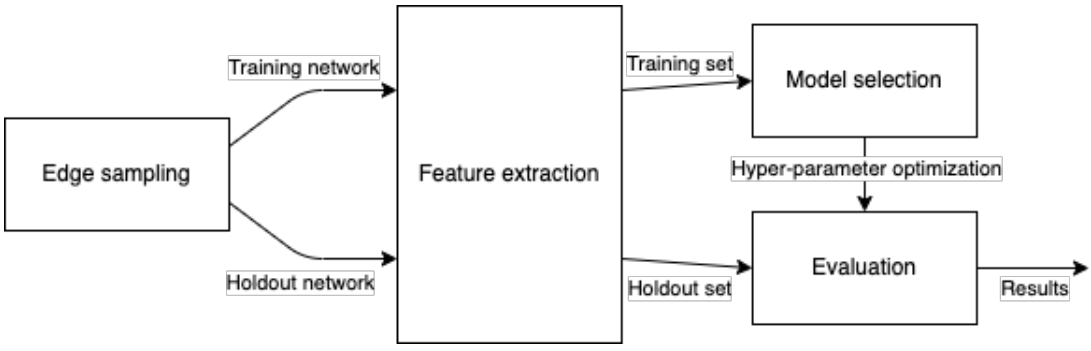
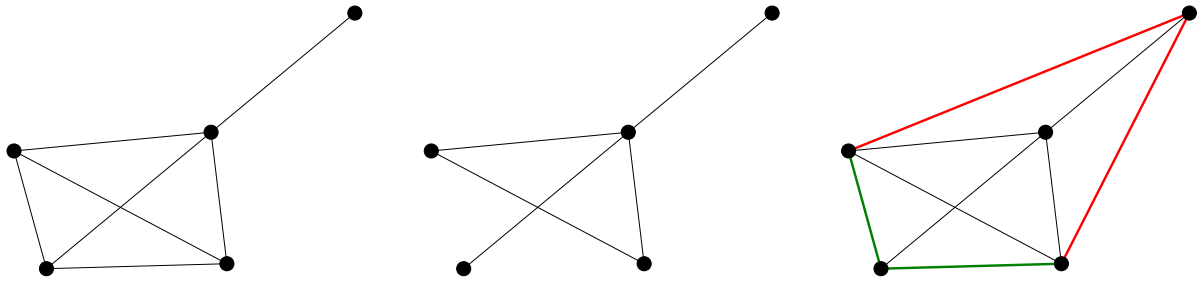


Figure 4: The flow of processes in our machine-learning model.

Sample the original network To be able to predict missing links, a uniform sample of the original network is drawn. Such a sample is what we consider to be the observed network. From a depiction found in Figure 5, we see a network in its original state (a) as found in the corpus, an observed network (b) where two edges have been removed by the sampling method, and examples of true and false missing edges (c) in green and red. The sample in (b) is drawn two times individually, for the training and test set. It is the task of the machine-learning model to predict the missing links, only taking into consideration the structure that exists in this observed network. The sampling method that we use in this step is a standard binomial (discrete) distribution.

$$P(N) = \binom{n}{N} p^N (1-p)^{n-N},$$

where we choose probability parameter $p = 0.8$, and N is the number of edges in the network. This method creates a uniform distribution for our edges, with a probability for every edge to be in the observed network of 80%. The edges that are not chosen are defined as missing edges. This process is repeated two times to create two observed networks, one for training and one for testing using the same sampling rate p. As previously mentioned, we call these networks the *training network* and *test network* respectively.



(a) Original network. (b) Observed network. (c) True/false edges.

Figure 5: Depiction of the sampling procedure.

Sample true and false edges By comparing the observed and original network, we create a set of positive and negative samples. These are respectively the true (existing) and false (missing) edges of the observed network. For both types of edges, we draw the samples from a discrete uniform distribution. For practical reasons and preventing class imbalance, the sample size N is fixed to a common constant for both true and false candidates.

Note that for networks with smaller amount of links, this will intentionally lead to many duplicate samples. The uniform distribution that we chose to select our samples does not take into account that there exist much less links than "gaps" (no links) in the network. Therefore, especially in smaller networks, links are massively over-sampled while gaps are not, resulting in a model that is trained with more representative data for gaps than for links. It is because of this fact that the model is likely to perform much better for detecting gaps than it is in detecting links. Larger quantities of data are, however, required for most machine-learning models, as they are found to be "data hungry" [Lee75]. We therefore knowingly anticipate that our model performs better on gaps than on missing links.

3.3.2 Supervised machine-learning

Supervised machine-learning is best described as a learning task where the correct output of a sample is known and used to punish or reward the model when it acts upon it. In the context of link prediction, the task at hand is the binary decision problem `link (1) / no link (0)` where a model is trained by feeding it samples of links and their topological context. The model is subsequently updated using the values of the labels. The samples in this context are provided by a pair of nodes (u, v) and their calculated features, together representing a potential link and its context.

```
    u   v   CN  J   A   L_3 P'  T-u T-v S-u S-v ...
-------------------------------------------------------
1 | 3   7   2   .5  .3  .8  4   0   0   1   1
2 | 1   4   3   .1  .2  .1  1   1   0   1   0
3 | ...
```

Figure 6: Two rows of example training data.

In Figure 6, two dummy samples are shown. The observations (rows) include a pair of nodes and their respective features. Because node-based measures are not directly defined for node pairs like link-based measures, these measures will add both the feature value for u and v to the context of node pair (u, v) and are called pairwise methods. The method for sampling the edges (node pairs) is described in Section 3.3.1.

Because the main objective in this work is to find feature importances of the network features, the machine-learning model that will be used as a classifier for link prediction will be the RandomForest algorithm. RandomForest is popular model that allows for easy interpretation: the model builds a collection of decision trees and applies majority voting to decide the outcome value, as is shown in Figure 7. This way, the model never performs "worse" than selecting the best measure. The major asset this model has, is that its feature importances are easily computed since they follow directly from the splitting of trees.

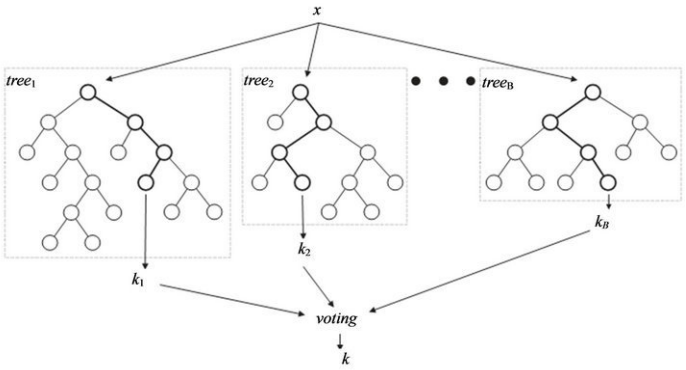


Figure 7: Internal structure of the RandomForest classifier [ZCR18].

3.3.3 Evaluation metrics and criteria

To measure the performance of the model, we use three evaluation metrics, namely *precision*, *recall* and *Area Under the Curve* (AUC). Precision measures the proportion of correctly guessed links (or nonexistence of a link – called a gap) and recall measures how many of the total links existing in the original network were actually found.

TP: Correctly predicted links.	FP: Wrongly predicted links.
TN: Correctly predicted gaps.	FN: Wrongly predicted gaps.

Using the collections of true/false positives/negatives, we can calculate precision and recall accordingly by dividing the correctly predicted links.

$$Pr = \frac{TP}{TP + FP} \qquad Rc = \frac{TP}{TP + FN}$$

The values of precision and recall are in the range $[0, 1]$; higher values indicate a better performing model. For example, models with $Pr \approx 1$ and $Rc \approx 1$ indicate a near-perfect model and in a binary decision problem, a precision value of $Pr = 0.5$ would indicate a random guessing model.

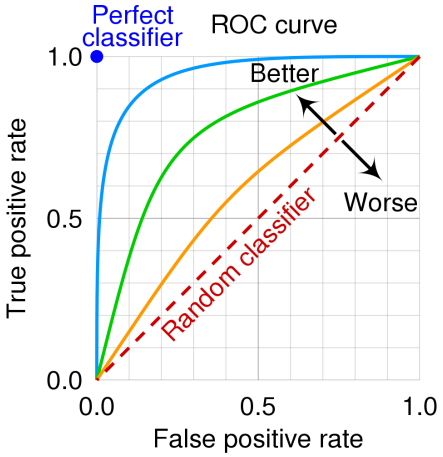


Figure 8: Three ROC curves of different models [Tho].

Receiver Operating Characteristics (ROC) are common practise for evaluating model performance. These type of curves plot the true positive rate (recall) against the false positive rate, as seen in Figure 8. The area under this curve is considered to measure how well the model discriminates between two classes and is a summarization method for the ROC curve. When AUC converges to 1, the model is perfect in discriminating classes.

AUC should, however, always be accessed in combination with other evaluation metrics. Especially if the originating dataset is imbalanced as in our case, AUC can give misleading results as a metric in classification problems. Because the ROC curve shows the trade-off between the false and true positive rate, if our large false positive rate is high but our true positive rate is not, AUC can still evaluate to high values. The expected high performance on gaps and therefore high AUC must therefore be analyzed with the lower performance on links in mind.

3.3.4 Model selection

A RandomForest classifier has many hyper-parameters that can be tuned. We choose two important hyper-parameters that need tuning, namely the number of trees in the forest and the maximum depth of a tree. We optimize the parameters by means of a grid search. To try and generalize the model, we perform fivefold cross validation during training. This means randomly splitting up the training network in 5 parts. For five iterations, the model is trained on 4 different parts of the data and evaluated on 1. After this process, the statistics (precision, recall, AUC) are averaged. Using cross-validation can prevent overfitting and selection bias. The parameters of the model with the best statistics are saved. This is what we call the best model. Finally, the best model is tested against the test network; these test statistics serve as the conclusive verdict for the model.

3.4 Group evaluation

In order to access whether a grouping of networks is successful in generalizing networks with the aim of selecting features for link prediction, we look at three different aspects of our networks in relation to the model.

Model performance We analyze how groups of networks perform in link prediction using histograms. This performance distribution tells something about the structures of the networks within and between groups. Ideally, each group has a small variance in the group itself, meaning that all networks in that group have near-equal performance.

Global measures We observe the global network measure averages using tables to find out how the global measures in a group are distributed. This global structural information is a factor in deciding if networks in the same groups are similar to each other.

Feature importances We report the importances of the network measures as features of our model per group to find out which measures work well as link prediction features for particular groups of networks.

These three aspects combined are our approach to answering the first two research questions in the Experiments section and to creating a good topology-based grouping of networks. For answering our final research question, we look at the feature importances per group (last aspect) to find out which measures work well as features on which groups.

.

3.4.1 Performance variability

The performance of the link prediction model on particular groups of networks tells something about the similarity of these groups seen from the model's perspective. In order to find these differences in model performance, we look at the evaluation metrics and criteria of the link prediction model as discussed in Section 3.3.3, namely AUC, precision and recall, as well as the distribution of these metrics among the different network groups.

3.4.2 Global measures

Global measures are those that are defined on a network scale and typically return a single scalar value $f(G) = y$ that relates to the network G as a whole, such as average network measures. Unlike local and pairwise measures, this type of measure is not used as a feature for link prediction, but as a tool for helping better generalize grouping networks in the Experiments section. We use tables to report the global measure values and variance in order to find out if the grouping resulted in groups with small variance, indicating their structural similarity on a global level. The measures used as grouping factors are listed below.

Average degree and degree variance An often used concept to gain a better understanding of a networks' structure is the distribution of the degree over the nodes in a network. For a node degree d and n_d nodes in the network G that have degree d, we define the degree distribution as $P_{\mathcal{G}}(d) = \frac{n_d}{n}$. In real-world (that is, non-artificially created) networks, we often observe the degree distribution to be right-skewed. This is explained by the observation that a higher number of nodes has a lower degree, and a small amount of nodes has a high degree. For in example social networks, such as social media platforms where a relatively smaller number of popular individuals are followed by a large audience, this is especially the case. The degree distributions of these kind of networks are sometimes said to follow a certain power-law, where $P_{\mathcal{G}}(d) \sim d^{-\gamma}$ and $\gamma \in \mathbb{R}$. The average degree of a network is defined as $\bar{d} = \frac{1}{n}\sum_{i=0}^{n} d(i)$ where $d(i)$ is the node degree of node i. The variance of the degree distribution d_{σ^2} is analogously defined as $d_{\sigma^2} = \frac{1}{n}\sum_{i=0}^{n}(d(u) - \bar{d})^2$.

Network density The density D of a graph is a normalized value $0 \leq D \leq 1$ and represents the percentage of possible links in the network. A dense graph has a density closer to the value 1 and the same graph with fewer edges will have a lower density value. The density of an undirected graph is defined as $D = \frac{2m}{n(n-1)}$. The denominator is equal to the total possible number of links and the numerator represents the number of edges currently present. Because we analyze undirected graphs, the number of theoretical pairwise links is two times larger than the number of undirected links.

Network diameter Network diameter R is defined as the longest of all shortest paths in a network $R = \max(\{\text{dist}(u, v) \mid (u, v) \in V \times V\})$. Network diameter tells something about the size and the connectivity in the network.

Network assortativity The degree and degree distribution of a network provides insight in how the structure of a network may look, but also requires further analysis,

because these measures are, after all, simple frequencies of the local entities of a graph. We increase our top level graph analysis by including another feature (a coefficient), that can prove interesting for our research. The degree assortativity of a network [New03] is a value $-1 \leq \rho \leq 1$ that measures in what way high and low degree nodes are connected to each other. An assortative network is a network where on average high degree nodes are connected to other high degree nodes, and low degree nodes are mostly connected to other low degree nodes. The network is said to be disassortative when the opposite is true: high degree nodes are connected to low degree nodes and vice versa. Negative values of ρ therefore indicate that a network contains a decent amount of connections between nodes of different degree. Network assortativity between pairs of linked nodes is defined as

$$r = \frac{\sum_{u,v \in V} uv \cdot (e_{uv} - q_u q_v)}{\sigma_q^2},$$

where e_{jk} is the excess degree distribution of the two nodes together, $q_v = \frac{(v+1) \cdot P(v+1)}{\sum_{u \geq 1} uP(u)}$ where, and σ_q^2 is the standard deviation of q_v. The numerator represents the variance of our variable (the degree) and the denominator the deviation. Therefore, this fraction is simply the Pearsons correlation coefficient between the degrees of linked node pairs.

Spectral bipartivity We refer to the paper of Mattsson et al. [MTH+21], where a distinctive type of network is identified that has an unusual structure. The so-called functional networks are those that follow a rather different interpretation of the meta concept of a tie discussed so far, and have an over-representation of square ties instead of triangle ones. Such structures are usually found in large production networks, biological networks or any network that represents a complex (often biological) function. In the biological field, modeling these interactions results in a mapping of proteins and their interactomes that are interpreted as a complex network and have important applications in the discovery of diseases and new drugs.

The presence of a relatively large amount of closed squares, makes that the process of identifying the structure of functional networks is inherently connected to bipartiteness. A bipartite network is defined as a network whose vertices can be divided in two sets, such that there exist only links between these two sets and not between elements of sets itself. An important characterization and result of this property, is that the graph does not contain any odd-length cycles. Naturally, graphs that come close to being bipartite contain more cycles of even-length and vice versa. In search for a quantifying method in functional structure, Mattsson et al. try to identify this structure using spectral bipartivity [ERV05]. The spectral bipartivity b_s of a graph with adjacency matrix A is defined as

$$b_s = \frac{tr\cosh(A) - tr\sinh(\mathcal{A})}{tr\cosh(\mathcal{A}) + tr\sinh(A)} = \frac{tr\exp(-\mathcal{A})}{tr\exp(\mathcal{A})} = \frac{\sum_{u \in V} e^{-\lambda u}}{\sum_{u \in V} e^{\lambda u}},$$

where tr denotes the trace of the matrix and $\lambda_1 \leq \ldots \leq \lambda_n$ are the eigenvalues of the adjacency matrix A. To normalize the value of b_s for all graphs, a logistic transformation is applied to bind spectral bipartivity so that $0 \leq b_s \leq 1$. Under this circumstance, a value $b_s = 1$ would indicate a fully bipartite graph, and a value $b_s = 0$ a complete graph (a graph with maximal linkage and cycles).

Node-based averages For degree and both clustering and closure techniques, we also incorporate network averages to find out if any of these values can help in selecting features. Averages are derived by summing up and dividing by the number of total nodes in the network. For a measure X taking a certain node u as argument, the following holds:

$$\bar{X} = \frac{1}{|V|} \sum_{u \in V} X(u)$$

	Description	Notation
Meta information	Number of nodes	n
	Number of edges	m
Averages	Average degree	$\bar{d}$
	Clustering	$\bar{C}_2$
	Square clustering	$\bar{C}_3$
	Closure	$\bar{H}_2$
	Square closure	$\bar{H}_3$
Statistics	Degree variance	d_{σ^2}
	Assortativity	r
	Transitivity	C
	Diameter	R
	Spectral bipartivity	b_s
	Density	D

Table 2: All global measures used in the topology-based grouping of networks.

3.4.3 Feature importances

Determining if a feature is useful or not in a collection of decision trees can be measured by the Gini-importance [NKW18]. This metric measures per variable how well it splits nodes with yet undetermined labels into pure single class nodes. For example, a feature that is responsible for many conclusive splits in the decision tree will have high Gini-importance and can be described as an important feature for the model. A feature that accounts for only a few splits has little deciding power in the model, and has a low value for Gini-importance. As shown in Figure 6, a pairwise network measure results in two separate features in the machine-learning model. For better interpretation, we add the feature importances of nodes u and v to create the joint feature importance of a pairwise measure. All feature importances are averaged per domain for improving the interpretability of the results.

This page intentionally left blank.

4 Experiments and Results

In this section, we describe the set-up of our link prediction experiments and show results. Section 4.2 describes how we construct a baseline model for link prediction, with the aim of studying whether grouping networks can be an effective generalization method for selecting good measures as features for link prediction. In a second experiment, we analyze the relations between global measures of the network to find out whether a topology-based grouping is a better solution for feature selection in link prediction.

4.1 Experimental setup

For all experiments, we used a fixed sample size $N = 10^4$ for the edges. The grids for our grid searches on the depth and number of trees are respectively $[5, 10, \ldots, 95, 100]$ and $[20, 40, \ldots, 180, 200]$. This gives a total of 200 potential combinations of parameters. Furthermore, the experiments were implemented using Python v3.9.6 and performed with the following setup:

CPU: Intel Core i7 6-core @2.6GHz

RAM: 16GB DDR4 @2667 MHz

OS: macOS Monterey 12.5

In this series of experiments, a RandomForest model for link prediction is fitted and tested on each network in the dataset. We follow the approach from Ghasemian et al., in which every network is fitted individually on a model, creating a total of 550 models with the same feature set. The networks are sampled for their edges, and the corresponding topological features are calculated. We follow a similar extensive hyper-parameter grid search, applying 5-fold cross-validation to find the best number of estimators and tree depths. We apply this optimization procedure for each of the 550 models.

Concretely, this means that we first apply a model selection phase in which the model is fitted to each specific instance of the parameters in the searching grid. The parameters of the model with the highest average performance are kept. A second holdout performance phase serves as a final test. The model is fitted one more time, using the best hyper-parameters found in the previous phase. The feature importances are calculated from this final model, and the final report statistics are found in applying this model to the test network.

4.2 Group by domain

The results in this experiment are analyzed per domain. As discussed in Section 3, we follow our approach by first analyzing the variability of model performance. After this, we look at the global measures in order to determine whether this way of discriminating complex networks is sufficient enough for distinguishing between important and non-important topological measures for link prediction. Finally, we decide (if possible), based on the feature importances of the model, which measures work well on which network domains as link prediction features.

4.2.1 Model performance

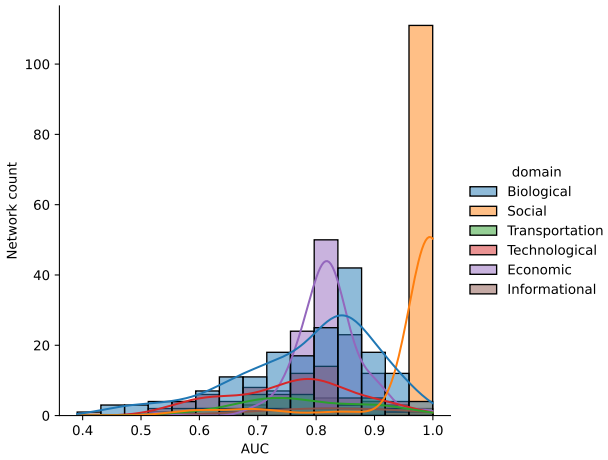
In Table 3 the performance metrics are given per domain, where the precision and recall values are given for the `link` class label. Social networks score the highest; on average, the model is correct 89% of the time a link is predicted and is able to retrieve 79% of the total missing links. Since we report AUC = 0.96 for this domain, the model seemed to learn the task very well for social networks. Precision and recall values for the other domains are substantially lower. Relatively low precision and high recall can be found in the economic domain, indicating that this model is not very accurate in classifying missing links, but can retrieve around 49% of the total missing links. Note that the AUC scores are not in line with the precision and recall values of *only* the `link` class label. This is expected behavior, which is discussed in Section 3.3.3.

If we compare our findings with those of Ghasemian et al. on all domains (Pr: 0.31 ± 0.33, Rc: 0.35 ± 0.29), we observe that we report both better and more stable precision (+4% percentage points) and recall (+5%) metrics. The added context of the new features seem to capture additional information about network structures that favors the link prediction model. However, we report a decrease in AUC (−5%) due to our lower performance on the `no link` class. We are therefore unable to state that our model discriminates better between both classes. Nevertheless, our model makes an improvement towards the retrieval of missing links and the performance of precisely selecting missing links.

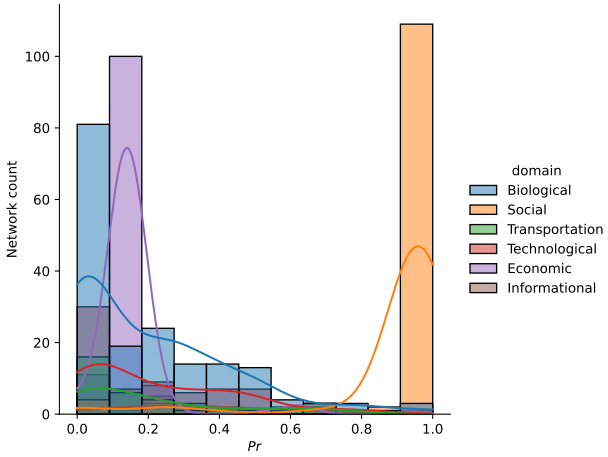
Domain	AUC	Pr	Rc
Social	0.96 ± 0.09	0.89 ± 0.22	0.79 ± 0.28
Biological	0.79 ± 0.12	0.21 ± 0.23	0.17 ± 0.18
Economic	0.82 ± 0.06	0.16 ± 0.12	0.49 ± 0.16
Technological	0.76 ± 0.11	0.22 ± 0.21	0.20 ± 0.18
Informational	0.81 ± 0.14	0.34 ± 0.26	0.23 ± 0.26
Transportation	0.78 ± 0.10	0.20 ± 0.23	0.35 ± 0.19
All domains	0.83 ± 0.12	0.35 ± 0.36	0.40 ± 0.32

Table 3: Performance metrics (mean ± std). Precision and recall for the `link` class.

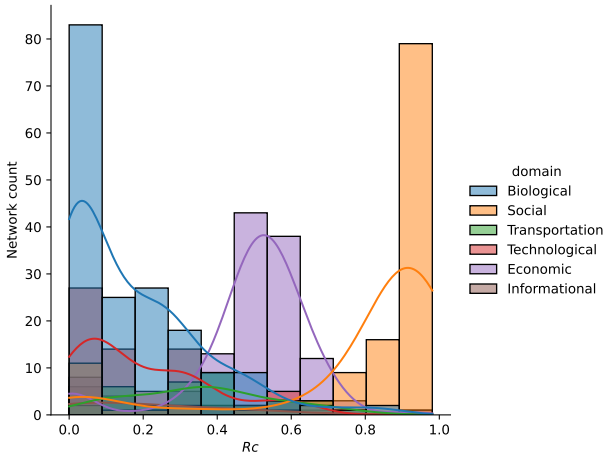
In Figure 9, we observe the distribution plots of our performance metrics among the six different domains. A first observation reveals that there exists a somewhat clear unimodal distribution with one peak for the social domain for all performance measures. This indicates that the diversity within these networks in terms of performance is not large. Moreover, this peak is always near the right side, indicating that the links of almost all social networks are easy to predict. For economic networks, we also observe a clear unimodal distribution. For recall, this if found near the center, which indicates that a moderate amount of the missing links is still found. Precision is, like for the other remaining domains, very low. Biological and technological networks have a long distribution span, which indicates that within the domain itself, large performance differences can exist for networks of these domains.



(a) AUC



(b) Precision



(c) Recall

Figure 9: Distribution plots with kernel density estimations for performance metrics.

4.2.2 Global measures

In the previous experiment, we have shown that the link prediction performances of networks grouped by domain differ vastly, and the variance of the performance within domains can be large as well. To further investigate the differences between network domains, we observe the global measure averages in Table 4. If we observe these values by means of a variation coefficient $CV = \frac{\sigma}{\mu}$, it becomes clear that the variance within domains is very high. For almost all domains and measures, it holds that $CV > 1$; with the one exception being social domains. For all global measures in the social domain except D, it holds that $CV > 1$, and therefore the social domain is the most stable one. Other domains show high variance coefficients for either the proportional network values, measures or both.

So far, this explains the uniform performance of our link prediction model on social networks found in Section 4.2. The grouping of social networks within a single domain as a separate category of networks is effective: the variance *within the domain* is low and, especially clustering and closure measures, take on distinctive values which cause this network domain to be different *between domains*. The increased connectivity in social networks causes the closure and clustering coefficients to be larger than in other domains. We observe higher triangle closure and clustering for social networks when compared to square-based methods. This trait is reversed in biological and economic networks; 3-wedges more often close into a square than 2-wedges close into a triangle.

With the exception of social networks, we observe high variance for all network domains. This further validates our assumption that network domains of origin are *not* a feasible generalization method for feature selection.

	Social	Bio.	Eco.	Tech.	Info.	Transport.
	124 (22.5%)	179 (32.5%)	124 (22.5%)	70 (12.7%)	18 (3.27%)	35 (6.36%)
n	559 ± 260	294 ± 392	702 ± 303	533 ± 436	494 ± 703	721 ± 693
m	$1,988 \pm 800$	$780 \pm 1,033$	866 ± 460	$1,061 \pm 896$	$1,266 \pm 1605$	$1,274 \pm 1216$
λ	$4k \pm 1k$	$2k \pm 9k$	$1k \pm 4k$	$1k \pm 1k$	$1k \pm 2k$	$1k \pm 4k$
η	$26k \pm 24k$	$54k \pm 420k$	$21k \pm 147k$	$13k \pm 45k$	$19k \pm 46k$	$29k \pm 122k$
$\bar{d}$	8 ± 3	6 ± 7	3 ± 6	4 ± 1	5 ± 3	3 ± 3
$\bar{C}_2$	0.84 ± 0.16	0.14 ± 0.11	0.04 ± 0.14	0.12 ± 0.14	0.22 ± 0.12	0.10 ± 0.13
$\bar{C}_3$	0.50 ± 0.14	0.15 ± 0.12	0.04 ± 0.05	0.05 ± 0.05	0.06 ± 0.04	0.04 ± 0.04
$\bar{H}_2$	0.59 ± 0.12	0.09 ± 0.15	0.02 ± 0.09	0.04 ± 0.05	0.08 ± 0.06	0.06 ± 0.06
$\bar{H}_3$	0.15 ± 0.03	0.06 ± 0.06	0.02 ± 0.03	0.02 ± 0.02	0.03 ± 0.02	0.02 ± 0.02
d_{σ^2}	5 ± 1	6 ± 4	3 ± 4	5 ± 5	8 ± 4	3 ± 5
r	0.23 ± 0.22	-0.23 ± 0.25	-0.55 ± 0.22	-0.12 ± 0.14	-0.25 ± 0.23	0.05 ± 0.15
C	0.66 ± 0.14	0.11 ± 0.18	0.03 ± 0.12	0.07 ± 0.07	0.13 ± 0.09	0.10 ± 0.08
R	14 ± 4	11 ± 9	28 ± 8	14 ± 18	7 ± 3	34 ± 18
b_s	0.51 ± 0.05	0.78 ± 0.21	0.96 ± 0.14	0.76 ± 0.22	0.59 ± 0.14	0.88 ± 0.15
D	0.03 ± 0.05	0.07 ± 0.08	0.01 ± 0.05	0.01 ± 0.01	0.03 ± 0.03	0.01 ± 0.01

Table 4: Global network measures over the six different network domains in the dataset (mean $\pm$ std). $k = 1000$.

4.2.3 Feature importances

In Figure 10, we report the feature importances for the features used in the machine-learning model for link prediction. We observe that (quasi) link-based measures work best for social networks, likely because of their relation with triadic closure. However, the shortest path is in the lead. L3-score is not expected to make any contribution here, because the measure was designed for functional networks. The other features play little part for social networks. Methods like the clustering and closure coefficient would theoretically work well on social networks. Apparently, providing more node-based structural triadic closure information in addition to methods based on triadic closure, like common neighbours, has little benefits for the model. This could also be an explanation for the fact that square/node-based measures are slightly more important than triangle ones for this domain.

Other networks seem to rely most on node-based measures, such as the local degree and average neighbour degree of nodes, where the link-based measures are of very little importance for these domains. This is easily explained by the fact that these models are built with triadic closure as a base principle in mind. As a quasi link-based method, shortest paths still works decently on technological and transportation networks. For the other node-based measures, we observe that square-based methods work better than triangle-based ones, especially for biological, technological and informational networks. The difference, however, is minor, and not as high as expected. The L3-score was expected to work (very) well on functional networks, but shows very little importance for almost all domains. This is another indication that the domain grouping is ineffective in separating networks based on their structure.

4.2.4 Feature recommendations

Only for social networks, we found that the variance within the domain is small. This means that we can only come up with a feature recommendation for social networks versus all other networks in Table 5:

Domain	Recommended features for link prediction
Social	Link-based measures: P, CN, J, A
Other	Degree-based measures + P: d, d_{nn}, P

Table 5: Feature recommendations for link prediction on the six network domains.

We see, that for social networks, the link prediction model largely relies on link-based measures, especially ones based on triadic closure. Node-based measures add very little structural information in addition to important measures for social networks, such as common neighbours. Shortest paths is found to be the most important measure. Other networks rely heavily on the local degree and average neighbour degree. Link-based measures are not important at all, including the L3-score which was expected to provide additional structural information about functional networks. We see that shortest paths also works decently on other networks than social ones.

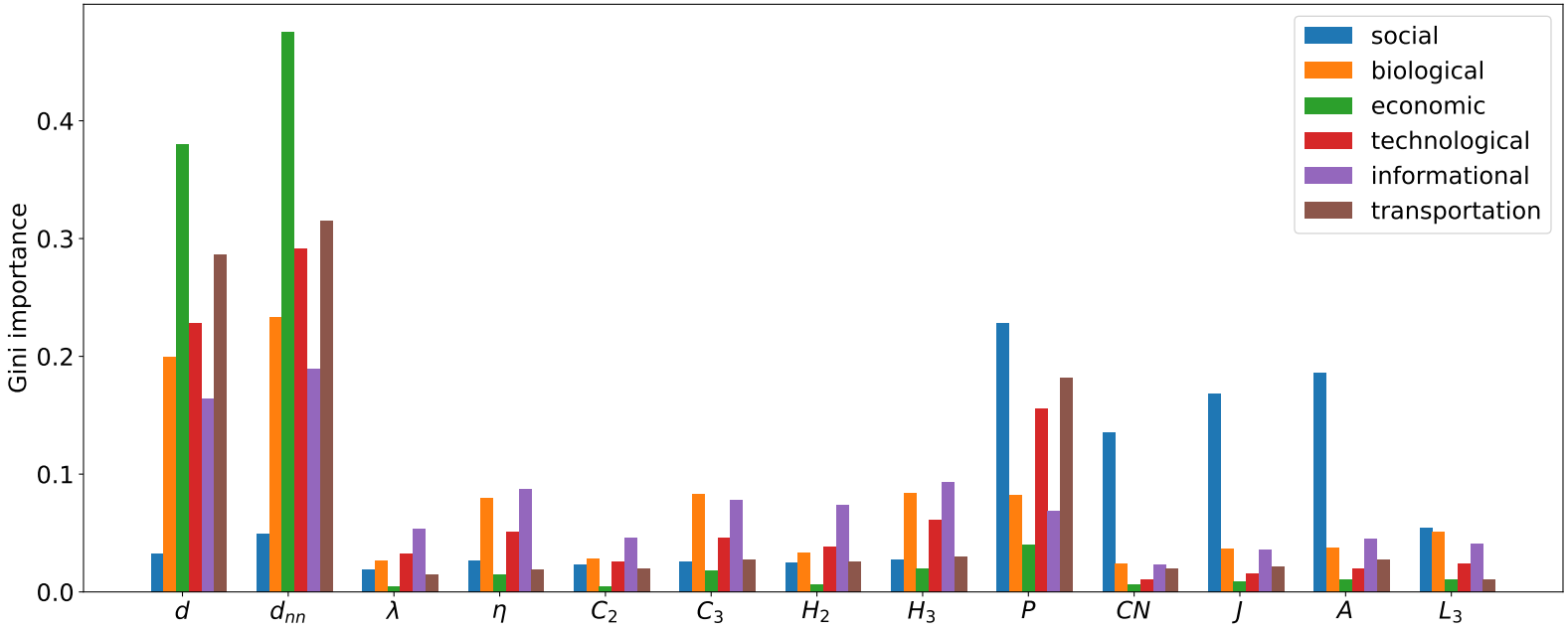


Figure 10: Feature importances for the 13 features used in the link prediction model.

4.3 Finding a topology-based grouping

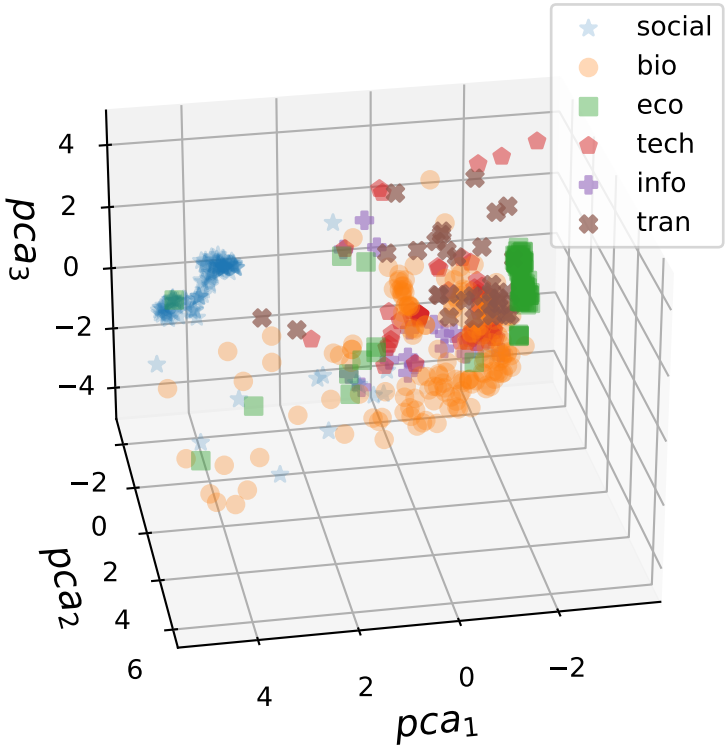
So far, we have found that domains are not a good generalization method for the structure of networks, and therefore also not a good method for recommending features in link prediction. In this experiment, we try to find relations between the global measures of networks to create a topology-based grouping that is potentially better suited for generalizing network structure. Until now, we have experimented with using the domain of origin of networks as being the generalization method for selecting measures as features in a link prediction model. We discovered that the large variance present in most network domains prevents an accurate mapping of effective features against domains as network groups. In this experiment, we will discover whether we can effectively lessen this variance. The distributions of the global network measures accountable for most of the variance possibly provide an accurate categorization of networks that will help us select good features for link prediction. Firstly, we will use a principal component analysis to find the global measures accountable for the most variance in the networks. Secondly, we create a grouping based on the distribution of these measures and define these groups.

4.3.1 Principal component analysis

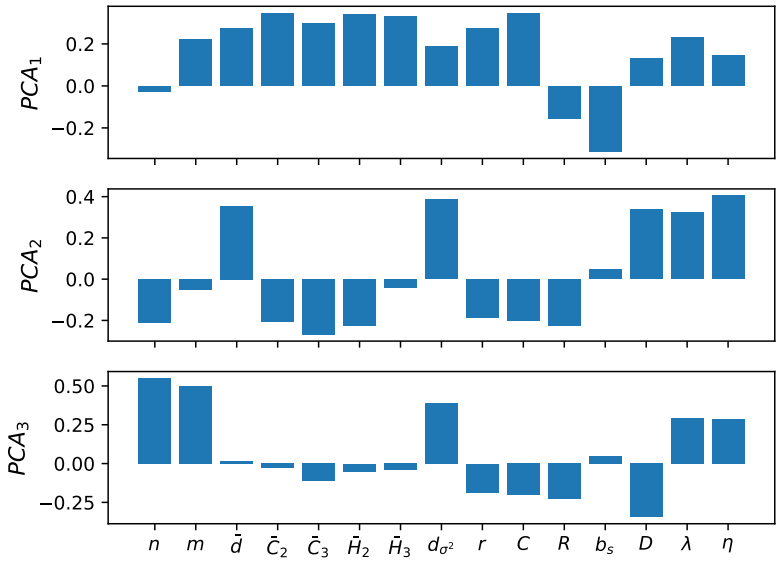
The high variance of global measures within domains that was found so far can have different origins. We use a linear decomposition method to describe and begin to explain this variance. Principal component analysis [BP87] reduces a feature matrix of arbitrary dimensions into a fixed dimension size by creating a new set of uncorrelated variables that represent the original matrix, while retaining the maximum possible amount of information. The axes of this decomposition, effectively linear combinations of the original network measures, are also called principal components and all explain a percentage of the variance in the original data. By examining the linear coefficients (component loadings) between the principal components and the original measures, we can determine which of the measures explain the most variance.

In Figure 11, we observe the results for PCA on the global measures of the entire corpus. The components are accountable for respectively 47%, 18% and 14% of all the variance among the measures. The first axis of the linear transformation captures only about half of the explained variance, and is on itself not sufficient for a good representation of the data. The three components together explain 79% of the variance among the variables. The results of the object coordinates in Figure 11a show that PCA allows us to reasonably identify the social networks and economic networks as clusters. This is in line with our previous experiments. Other networks tend to be more spread in the principal component plane. Especially biological networks are found to be very diverse.

To find out which measures are a good candidate for grouping networks, we observe the component loadings of this instance of PCA in Figure 11b. In the first and most important component, the largest linear correlations correspond to measures of local closure based on either triangles or squares. Biological networks are widely spread on this axis in Figure 11a, indicating that there exists a lot of variance within this domain with respect to triangles- and square-favored networks. The other two principal components mainly account for degree-based measures (PCA_2) and the size of the network (PCA_3).



(a) PCA coordinates.



(b) PCA component loadings.

Figure 11: PCA using global network measures as features.

4.3.2 Topology-based groups

In order to determine whether there exist obvious other groups of networks in our dataset, we go one step back in the process of generalization by computing global pairwise network measures for all 550 networks. This time, we do not rely on the assumption that network domains categorize networks well in terms of structure. Therefore, we visualize all combinations of axes associated to different global measures that are possible. The result pairplot can be found in Figure 12. Firstly, it becomes clear again that social networks have a very distinctive distribution in almost all global measures. The unity of social networks is most visible in plots associated to triangle and square-based measures (clustering, closure and network transitivity).

The long tail that is visible for biological networks in many global measure distributions is very interesting. It could be that the answer to this generalization method could be related to these networks. In the previous experiment, we found that the largest explained variance can be traced back to the measures $\bar{C}_2$, $\bar{C}_3$, $\bar{H}_2$, $\bar{H}_3$. These are all measures that measure square and triadic closure or clustering. Observing the plots for these measures, for example $\bar{H}_2/\bar{H}_3$, brings to our attention that there exists a group of bipartite networks (following the y-axis). Nearly bipartite, but not completely, we find the supposed functional networks. Other than the social cluster, which was found to be a very distinctive group in the past experiments, other distinctive groups of networks in this plot are the networks clustered at the origin (small amount of squares and triangles) and a group of networks following the diagonal. We could argue that this last group of networks favor squares as much as triangles, regardless of the size of the network.

The networks associated with these 5 groups discussed above (social cluster, bipartite, functional, origin and diagonal) can be found in similar or other groups in many of these plots, indicating their independence as a group from other networks. We define them for $\bar{H}_2$ and $\bar{H}_3$ in Table 6 as our new topology-based groups. In the following experiments, we will find out whether this categorization is a more effective method for selecting the best features for link prediction.

Group	Description	Total networks
Social	All networks from the social domain	124 (22.5%)
Origin	Networks with low clustering/closure values	286 (52.0%)
Bipartite	Networks containing no triangles	25 (4.5%)
Functional	Networks with some triangles and many squares	36 (6.5%)
Diagonal	Networks with high clustering/closure values	85 (15.5%)

Table 6: Topology-based groups of networks.

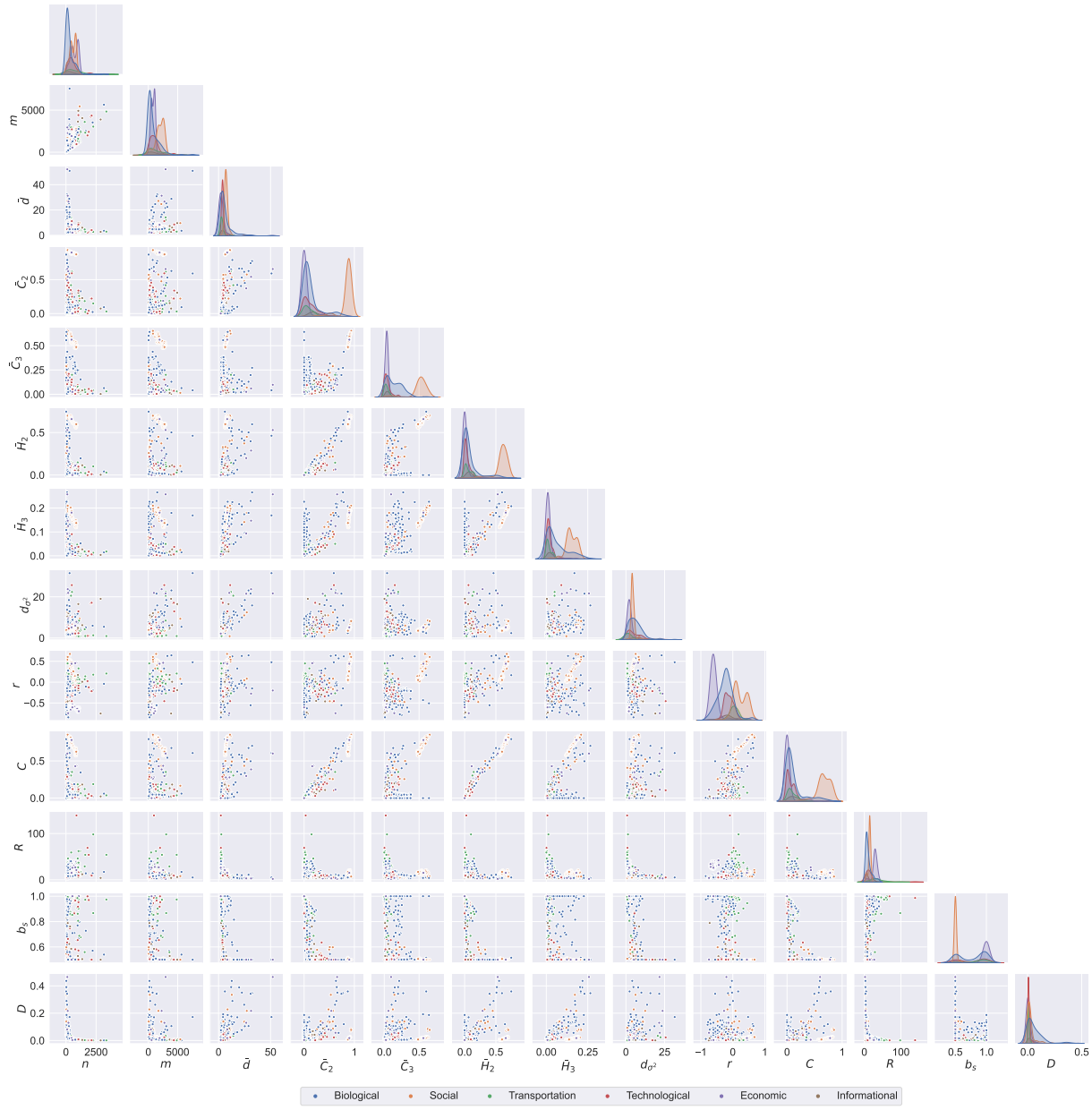


Figure 12: Network distributions under all combinations of global network measures.

4.4 Group by global measures

In this experiment, we compare the topology-based grouping of networks from the previous experiment to network domains as a grouping. In Section 4.3.2, we came up with different topology-based groups that separate networks based on global network measures accountable for high variance. This resulted in five groups: social, origin, bipartite, functional and diagonal.

To find out if this grouping is better suited as a generalization method for finding good link prediction features, we perform the same series of experiments as we did in Section 4.2. This includes a model performance and global measure analysis, as well as giving a recommendation of link prediction features.

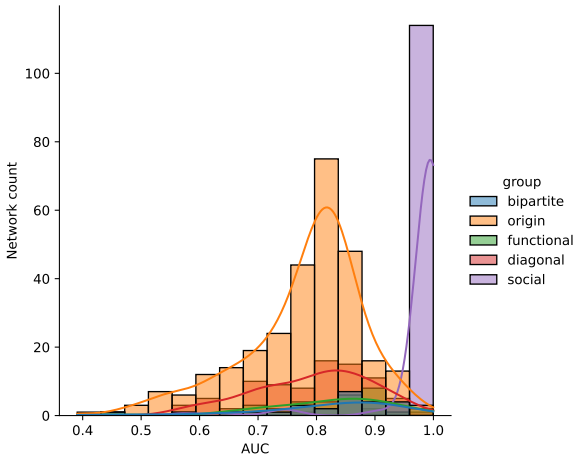
4.4.1 Model performance

In Table 7, we observe the same results for our link prediction models as in Section 4.2, but now grouped using the topology-based grouping. Comparing the results with those in Table 3, we do not find any big deviations in variance for any of the groups. However, we do find that precision and recall are very low for both bipartite and functional networks, potentially indicating these two groups have a different structure from others. Again, only social networks are found to perform well on predicting links.

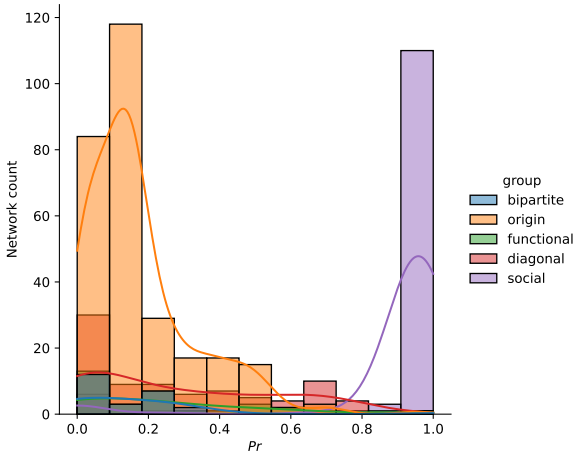
Group	AUC	Pr	Rc
Social	0.96 ± 0.09	0.89 ± 0.22	0.79 ± 0.28
Origin	0.78 ± 0.10	0.17 ± 0.15	0.36 ± 0.21
Bipartite	0.82 ± 0.12	0.15 ± 0.21	0.08 ± 0.13
Functional	0.82 ± 0.11	0.24 ± 0.26	0.08 ± 0.13
Diagonal	0.80 ± 0.10	0.28 ± 0.27	0.16 ± 0.20
All groups	0.83 ± 0.12	0.35 ± 0.36	0.40 ± 0.32

Table 7: Performance metrics (mean ± std). Precision and recall for the `link` class.

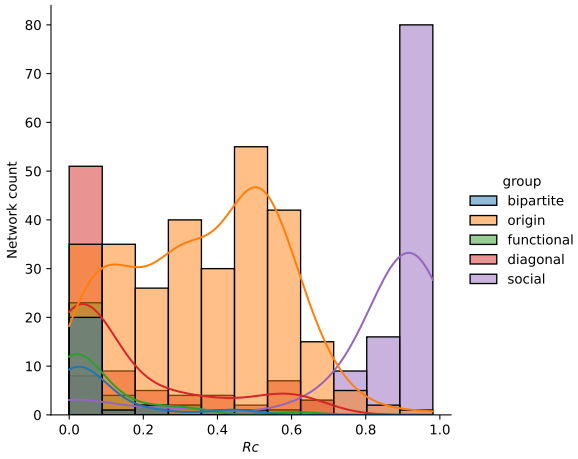
In Figure 13, we observe the performance distributions for the topology-based groups. Again, for the same model instances used in Section 4.2, but plotted with a different grouping. Bipartite, functional and diagonal networks effectively differentiate networks with low and high recall. The origin group of networks is very diverse in performance, indicating that this group is not well specified. Also, the middle peak for recall in Figure 13c for the origin group corresponds to that of the same peak in Figure 9c, where this peak is from the economic domain. This indicates that this domain should probably be part of our grouping and could help explaining the origin group, although it was not chosen as a group in the previous experiment.



(a) AUC



(b) Precision



(c) Recall

Figure 13: Distribution plots with kernel density estimations for performance metrics (topology-based groups).

4.4.2 Global measures

In this experiment, we analyze the variances and values of the global network measures one more time, but this time for the topology-based grouping. If we compare the values to those of Table 4, we find that this topology-based grouping greatly reduces the variance of the global measures. We observe that bipartite and functional networks have very little variance on almost all network measures. Bipartite networks have a high variance for $\bar{H}_3$, which indicates that the proportion of square clustering differs. The origin and diagonal group also contains small variations with respect to the means, although not as small as the bipartite and functional groups. Where for most network domains in Table 4, we obtain CV > 1, with the topology-based grouping we see the opposite trend: for most variation coefficients it holds that CV ≤ 1. The uniformity of the bipartite and functional groups indicates that we have successfully extracted at least two groups that can be used for selecting link prediction features. Most of the variance in size has gone to the diagonal group. This is to be expected, as the remainder of the networks not belonging in any of the groups was assigned to this group.

	Social	Origin.	Bipar.	Func.	Diag.
	124 (22.5%)	286 (52.0%)	25 (4.5%)	36 (6.5%)	85 (15.5%)
n	559 ± 260	616 ± 468	57 ± 32	147 ± 179	345 ± 405
m	$1,988 \pm 800$	905 ± 805	133 ± 88	522 ± 533	$1,260 \pm 1400$
λ	$4k \pm 1k$	205 ± 581	0 ± 0	$365 \pm 1,196$	$3.6k \pm 12k$
η	$26k \pm 24k$	$4k \pm 15k$	$823 \pm 1,025$	$24k \pm 64k$	$113k \pm 593k$
$\bar{d}$	8 ± 3	3 ± 1	4 ± 1	8 ± 4	9 ± 8
$\bar{C}_2$	0.84 ± 0.16	0.04 ± 0.06	0.00 ± 0.00	0.10 ± 0.14	0.33 ± 0.15
$\bar{C}_3$	0.50 ± 0.14	0.06 ± 0.08	0.04 ± 0.24	0.18 ± 0.05	0.09 ± 0.07
$\bar{H}_2$	0.59 ± 0.12	0.01 ± 0.02	0.00 ± 0.00	0.03 ± 0.02	0.20 ± 0.10
$\bar{H}_3$	0.15 ± 0.03	0.01 ± 0.01	0.10 ± 0.04	0.11 ± 0.05	0.06 ± 0.05
d_{σ^2}	5 ± 1	4 ± 3	4 ± 2	8 ± 5	7 ± 6
r	0.23 ± 0.22	-0.33 ± 0.27	-0.51 ± 0.14	-0.33 ± 0.15	-0.04 ± 0.22
C	0.66 ± 0.14	0.03 ± 0.04	0.00 ± 0.00	0.05 ± 0.04	0.27 ± 0.13
R	14 ± 4	22 ± 14	5 ± 1	7 ± 5	14 ± 14
b_s	0.51 ± 0.05	0.78 ± 0.21	1.00 ± 0.00	0.76 ± 0.22	0.59 ± 0.14
D	0.03 ± 0.05	0.07 ± 0.08	0.01 ± 0.05	0.01 ± 0.01	0.03 ± 0.03

Table 8: Global network measures over the topology-based grouping (mean $\pm$ std). $k = 1000$.

4.4.3 Feature importances

In the previous experiments, we discovered 5 categories of networks that share similar distributions for global network measures. Using these 5 categories as our new generalization method for selecting features in link prediction models, we are interested whether this approach is more effective in a broad sense. The previous experiment on feature importances in Section 4.2.3 showed that several network measures that were expected to work well on certain domains of networks failed to show their importance, causing the problem

that we were unable to give a recommendation for other network domains beyond the social domain. Using the same feature importances from this experiment, we regroup the networks according to the new network groups in Table 6.

A first observation shows that we still observe high feature importances for conventional link-based social network measures on social networks. Feature importances for shortest paths, common neighbours, Jaccard index and Adamic-Adar index have not changed. This is to be expected, as we extracted the social networks from the cluster they were in and re-categorized them. We also find that all other network categories are still very dependent on degree-based measures. Networks lying in the origin of all the global network measures can best be approached in link prediction using these measures. This has to do with the fact that these networks have an absence of triangles and squares, which is why other measures (except shortest paths) that all based on the concept of triangle or squares are not effective here. The measure that we expected to work well on biological networks, L3-score, now shows it importance on functional networks. This measure does therefore not apply to all biological networks, and should only be used on functional, near-bipartite networks. Fully bipartite networks, on the other hand, gain no information from triangle-based node measures. This is to expected, as these measures all evaluate to zero in a bipartite network context. Square-based measures, such as square closure, are found to work good on functional and bipartite networks. Triangle-node-based measures, such as triangle closure, are found to work poorly on almost all network categories.

4.4.4 Feature recommendations

We conclude that node-based measures only effective for bipartite and functional networks. L_3 shows that it can be important on functional networks. Social networks are bound to the conventional link prediction algorithms that were specifically designed for this network domain. Networks lying in the origin of the global measure space have no association to triangles or squares. Degree-based methods are most appropriate for this network category. Diagonal networks show spread feature importance, which indicates that more work is required to break down this category in others. We summarize our findings in Table 9.

Category	Recommended features for link prediction
Social	Link-based methods: P, CN, J, A
Origin	Degree-based methods: d, d_{nn}
Bipartite	L3-score, degree and square-based methods
Functional	Degree and square-based methods
Diagonal	Further analysis required

Table 9: Recommendations for link prediction on our five network categories.

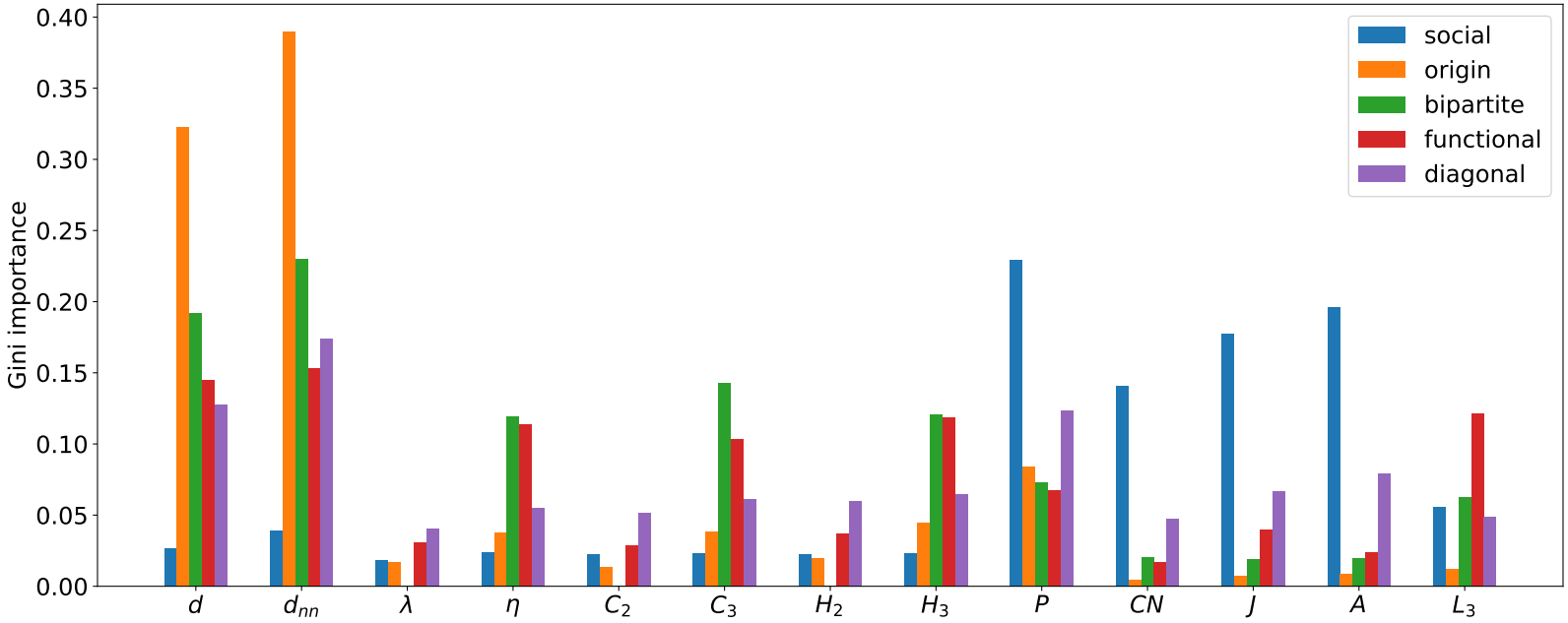


Figure 14: Feature importances grouped under our newly created topology-based grouping.

This page intentionally left blank.

5 Conclusion

In this work, we analyzed the effectiveness of using the networks originating domain as a grouping factor for determining which links work well on which groups of networks. As turned out, network domains are not a good generalization method for the structure of the network, and therefore also not a good method for globalization for the importance of individual network measures as link prediction features. The only network domain that is generalized well by structure is the social domain. This is the easiest domain of networks to classify links in. Well known link-based measures, such as common neighbours, have shown to be good features in a topological link prediction model for this domain of networks. The pairwise similarity between nodes that is measured by for example the Jaccard index, in combination with the shortest path distance between two nodes are the best features for such a model. The large variance that exists within the other domains prevents any meaningful recommendation, other than that the measures based on degree are good predictors for these domains. Shortest path distance can also play a role here, but networks that are not social are not generalized well by the concept of domains and are much harder to predict links for on average.

Our second research question suggested creating a new topology-based grouping using global network measures. We discovered that most of the variance present in the global measures was due to clustering and closure measures, including our newly defined squares closure coefficient. The large standard deviations and distributions of our evaluation distribution plots suggested that the results in performance vary heavily within domains, indicating that very different structures do exist within a single domain. We left the uniformity of social networks alone, and created 4 new groups alongside it, using the squares and conventional closure coefficient as a grouping factor. We observed that this grouping reduces the global measure variance, and is therefore better suited as a generalization method for feature selection in link prediction. The variance in the diagonal group, still reasonably high, would require further break-down for better link prediction feature recommendations.

Instead of following the approach of Ghasemian et al., described by only using the network domain as a way of grouping networks, we grouped networks by how their values are distributed over the global network measures. This method showed to be more effective. By incorporating classes for functional and bipartite networks, we were able to demonstrate the importance of the L3-score for functional networks. We also came up with a recommendation for each group on which measures to use as link prediction features. Another promising find, is that square-node-based measures are found to be good features in a link prediction model. This sheds new light on the idea of link prediction, where link-based measures are generally the first choice as features. Other findings that could not be derived from using network domains as a generalization method, include the high feature importance for degree-based methods on networks with very few triangles and squares.

An accessory our work, but not an unimportant one, is that our model also shows an improvement on predicting missing links when all methods from the latter paper are combined. The main disadvantage of our model, is that its discriminating ability between links and gaps is lower, because we report marginally worse results for predicting false samples. Further research should include a better analysis of networks that have a similar amount of triangles and squares. This network group, that we called the diagonal category, shows no convincing evidence of favoring a certain network measure as a machine-learning feature. Breaking down this category systematically could provide researchers with new handles for approaching link prediction problems on this network group. Another promising contribution that could be made lies in the field on temporal and directed link prediction, where our research now spans converted undirected networks and misses the concept of time.

References

[AA03] Lada A. Adamic and Eytan Adar. Friends and Neighbors on the Web. *Social Networks*, 25(3):211–230, 2003.

[ATS] Clauset Aaron, Ellen Tucker, and Matthias Sainz. The Colorado Index of Complex Networks. `https://icon.colorado.edu/`. Accessed: 8-05-2022.

[BFM+20] Andrea L. Bertozzi, Elisa Franco, George Mohler, Martin B. Short, and Daniel Sledge. The Challenges of Modeling and Forecasting the Spread of COVID-19. *Proceedings of the National Academy of Sciences*, 117(29):16732–16738, 2020.

[BP87] Tim P. Barnett and Rudolph Preisendorfer. Origins and Levels of Monthly and Seasonal Forecast Skill for United States Surface Air Temperatures Determined by Canonical Correlation Analysis. *Monthly Weather Review*, 115(9):1825–1850, 1987.

[Dij59] E. W. Dijkstra. A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik*, 1(1):269–271, 1959.

[ERV05] Ernesto Estrada and Juan A. Rodríguez-Velázquez. Spectral Measures of Bipartivity in Complex Networks. *Physical Review E*, 72(1):46–105, 2005.

[Eul36] Leonhard Euler. Solutio Problematis ad Geometriam Situs Pertinentis. *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, 8(1):128–140, 1736.

[Fac] Facebook Incorporated. Facebook. `https://www.facebook.com/`. Accessed 01-08-2022.

[FBM21] Cynthia Freeman, Ian Beaver, and Abdullah Mueen. Detecting Anomalies in Sequences of Short Text Using Iterative Language Models. *The International FLAIRS Conference Proceedings*, 34(1), 2021.

[GC] Benyamin Ghojogh and Mark Crowley. Linear and Quadratic Discriminant Analysis: Tutorial. `https://arxiv.org/abs/1906.02590`. Accessed: 26-03-2022.

[GHC20] Amir Ghasemian, Homa Hosseinmardi, and Aaron Clauset. CommunityFitNet Corpus Hosted on Github. `https://github.com/Aghasemian/CommunityFitNet`, 2020. Accessed: 17-11-2021.

[GHG+20] Amir Ghasemian, Homa Hosseinmardi, Aram Galstyan, Edoardo Airoldi, and Aaron Clauset. Stacking Models for Nearly Optimal Link Prediction in Complex Networks. *Proceedings of the National Academy of Sciences of the United States of America*, 117(38):23393–23400, 2020.

[GL16] Aditya Grover and Jure Leskovec. Node2vec: Scalable Feature Learning for Networks. In *International Conference on Knowledge Discovery and Data Mining*, volume 2016, pages 855–864, 2016.

[HAMS20] Timothy M. Hospedales, Antreas Antoniou, Paul Micaelli, and Amos J. Storkey. Meta-Learning in Neural Networks: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(1):1–1, 2020.

[HL71] Paul W. Holland and Samuel Leinhardt. Transitivity in Structural Models of Small Groups. *Comparative Group Studies*, 2(2):107–124, 1971.

[HLL83] Paul W. Holland, Kathryn B. Laskey, and Samuel Leinhardt. Stochastic Blockmodels: First Steps. *Social Networks*, 5(2):109–137, 1983.

[Ho95] Tin K. Ho. Random Decision Forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282, 1995.

[HZW+19] Tong Hao, Lingxuan Zhao, Dan Wu, Bin Wang, Xin Feng, Edwin Wang, and Jinsheng Sun. The Protein–Protein Interaction Network of Litopenaeus Vannamei Haemocytes. *Frontiers in Physiology*, 10(1):156, 2019.

[KCKY20] Sanjay Kumar, Sanidhya Chaudhary, Saksham Kumar, and Raj Kumar Yadav. Node Classification in Complex Networks using Network Embedding Techniques. In *2020 5th International Conference on Communication and Electronics Systems*, pages 369–374, 2020.

[KFA+16] Faiza Khan, Madiha Fatima, Usman Alvi, Tahseen Jilani, and Ubaida Fatima. Comparative Study of Similarity Measures in Link Prediction Using Facebook Data. *International Journal of Computer Science and Information Security,*, 14(1):132–143, 2016.

[KGZ15] David B. Kurka, Alan Godoy, and Fernando J. Von Zuben. Online Social Network Analysis: A Survey of Research Applications in Computer Science. *Computing Research Repository*, 2015.

[KLS+19] Istvan Kovacs, Katja Luck, Kerstin Spirohn, Yang Wang, Carl Pollis, Sadie Schlabach, Wenting Bian, Dae-Kyum Kim, Nishka Kishore, Tong Hao, Michael Calderwood, Marc Vidal, and Albert-Laszlo Barabasi. Network-based prediction of protein interactions. *Nature Communications*, 10(1):1240, 2019.

[KTP05] Sotiris Kotsiantis, George Tsekouras, and P. Pintelas. Bagging Model Trees for Classification Problems. In *Advances in Informatics*, pages 328–337. Springer Berlin Heidelberg, 2005.

[KY18] Khushbu Kumari and Suniti Yadav. Linear Regression Analysis Study. *Journal of the Practice of Cardiovascular Sciences*, 4(1):33, 2018.

[Lee75] Douglass B. Lee. Requiem for Large-Scale Models. *SIGSIM Simul. Dig.*, 6(2):16–29, 1975.

[LGH05] Pedro Lind, Marta C. Gonzalez, and Hans Herrmann. Cycles and Clustering in Bipartite Networks. *Physical Review E*, 72(1):56–127, 2005.

[MTH+21] Carolina E. S. Mattsson, Frank W. Takes, Eelke M. Heemskerk, Cees Diks, Gert Buiten, Albert Faber, and Peter M. A. Sloot. Functional Structure in Production Networks. *Frontiers in Big Data*, 2021.

[New03] Mark E. J. Newman. Mixing patterns in networks. *Physical Review E*, 67(1):26–126, 2003.

[NKW18] Stefano Nembrini, Inke König, and Marvin Wright. The Revival of the Gini Importance? *Bioinformatics*, 34(21):3711–3718, 2018.

[Ris01] Irina Rish. An Empirical Study of the Naïve Bayes Classifier. *IJCAI 2001 Work Empirical Methods Artificial Intelligence*, 3, 2001.

[Tho] Martin Thoma. Receiver Operating Characteristic Curve with False Positive Rate and True Positive Rate. `https://commons.wikimedia.org/wiki/File:Roc-draft-xkcd-style.svg`. Accessed: 05-07-2022.

[WS98] Duncan J. Watts and Steven H. Strogatz. Collective Dynamics of 'Small-world' Networks. *Nature*, 393(6684):440–442, 1998.

[YBL19] Hao Yin, Austin R. Benson, and Jure Leskovec. The Local Closure Coefficient: A New Perspective On Network Clustering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, pages 303—-311. Association for Computing Machinery, 2019.

[YvdHL18] Dong Yao, Pim van der Hoorn, and Nelly Litvak. Average Nearest Neighbor Degrees in Scale-free Networks. In *Internet Mathematics*, pages 1–38. Taylor and Francis, 2018.

[ZCR18] Chuan Zhang, Liwei Cao, and Alessandro Romagnoli. On the Feature Engineering of Building Energy Data Mining. *Sustainable Cities and Society*, 39, 2018.