CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Network classification-based structural analysis of real networks and their model-generated counterparts

Marcell Nagy[1] and Roland Molontay[1,2]

[1]Department of Stochastics, Institute of Mathematics, Budapest University of Technology and Economics, Műegyetem rkp. 3., H-1111 Budapest, Hungary, and [2]MTA-BME Stochastics Research Group, Műegyetem rkp. 3., H-1111 Budapest, Hungary
Corresponding author: Marcell Nagy, email: marcessz@math.bme.hu, molontay@math.bme.hu

Action Editor: Christoph Stadtfeld

Abstract

Data-driven analysis of complex networks has been in the focus of research for decades. An important area of research is to study how well real networks can be described with a small selection of metrics, furthermore how well network models can capture the relations between graph metrics observed in real networks. In this paper, we apply machine-learning techniques to investigate the aforementioned problems. We study 500 real-world networks along with 2000 synthetic networks generated by four frequently used network models with previously calibrated parameters to make the generated graphs as similar to the real networks as possible. This paper unifies several branches of data-driven complex network analysis, such as the study of graph metrics and their pair-wise relationships, network similarity estimation, model calibration, and graph classification. We find that the correlation profiles of the structural measures significantly differ across network domains and the domain can be efficiently determined using a small selection of graph metrics. The structural properties of the network models with fixed parameters are robust enough to perform parameter calibration. The goodness-of-fit of the network models highly depends on the network domain. By solving classification problems, we find that the models lack the capability of generating a graph with a high clustering coefficient and relatively large diameter simultaneously. On the other hand, models are able to capture exactly the degree-distribution-related metrics.

Keywords: complex networks; network models; model calibration; model stability; domain prediction; network characteristics; network classification; network similarity

1. Introduction

Data-based analysis of complex networks has attracted a lot of research interest since the millennium when the prompt evolution of information technology made the comprehensive exploration of real networks possible. The study of networks pervades the majority of disciplines, such as Biology (e.g. neuroscience networks), Chemistry (e.g. protein interaction networks), Physics, Information Technology (e.g. WWW, Internet), Economics (e.g. interbank payments), and Social Sciences (e.g. collaboration and acquaintance networks) (Barabási, 2016). Despite the fact that networks can originate from different domains, most of them share a few common characteristics such as scale-free and small-world properties (Barabási & Albert, 1999; Watts & Strogatz, 1998), highly clustered nature (Newman, 2003), and sparseness (Del Genio *et al.*, 2011).

Modeling real networks is of great interest, since it may help to understand the underlying mechanisms and principles governing the evolution of networks. Moreover, such models are mathematically tractable and allow for rigorous analysis. Furthermore, an appropriate model
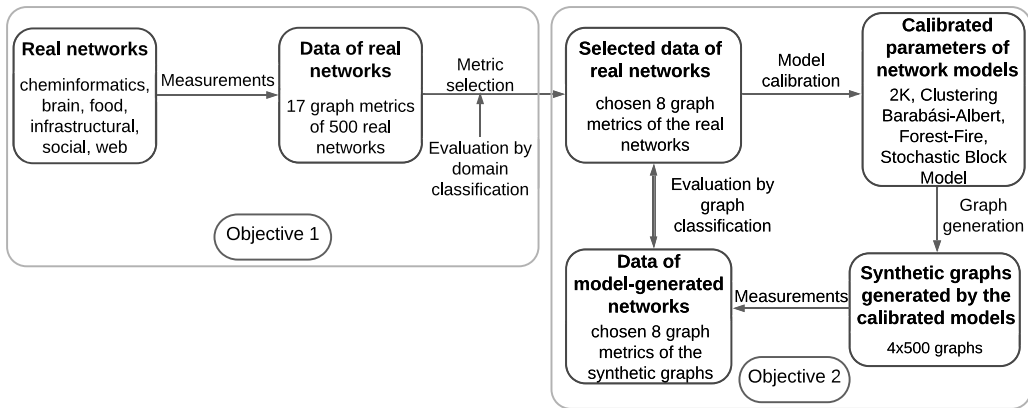
Figure 1. The workflow of the proposed methodology.

preserves the key characteristics, yet ensures the anonymity of the original real network (Fong *et al.*, 2009).

Throughout the years, several network models have been proposed to gain a better understanding of real-world networks, for an extensive overview see the paper of Goldenberg *et al.* (2010). However, without attempting to be comprehensive, the most influential models include the scale-free preferential attachment model introduced by Barabási & Albert (1999), the small-world model proposed by Watts & Strogatz (1998), and the highly clustered community structure model of Newman (2003), each of them was motivated by some of the aforementioned common characteristics of real networks.

In order to further characterize the structure of networks, numerous graph metrics have been introduced in the past two decades such as degree distribution-related measures (Aliakbary *et al.*, 2014) and other node-level centrality measures, local and global clustering coefficients, assortativity coefficient, density (Barabási, 2016), moreover motif-related attributes (Milo *et al.*, 2002; Pržulj, 2007). Naturally, there is significant redundancy among these measures and there is a great deal of effort in studying the correlation between the metrics and identifying a non-redundant selection of measurements that describes every aspect of networks (Bounova & de Weck, 2012; Garcia-Robledo *et al.*, 2013; Jamakovic & Uhlig, 2008; Nagy, 2018).

The present research was driven by the following two main objectives (see Figure 1):

(1) Studying the pair-wise correlations and descriptive power of structural graph metrics of real networks to find a minimal set of metrics that describes networks as accurately as possible.
(2) Understanding how well measurement-calibrated network models can describe real networks from different domains and investigating what structural properties the models can and cannot capture.

The outline of this paper is as follows: in Section 2, we give a thorough overview of related literature categorized by topics. The diverse scope of the literature review is due to the unifying nature of the present paper that joins different branches and building blocks of data-driven analysis of complex networks. In Section 3, we describe our selection of real networks, the investigated graph metrics, and the proposed network model calibration procedures. Furthermore, we show that despite the randomness of these models, their structural properties are stable enough to perform *grid optimization*-based parameter calibration. In Section 4, we investigate the structural properties of real networks by studying the correlation profiles of graph measurements across network domains and compose a minimal set of metrics that has high descriptive power and low

redundancy that we use later in this work (objective 1). In Section 5, we propose a machine-learning-based framework for calibration and evaluation of the network models using a given set of graph features (objective 2). Particularly, we examine how realistic graphs a network model can generate, which domains are easy to model, and what structural characteristics of real networks are difficult to capture with network models. Section 6 concludes the work.

2. Related work

Our paper relates to recent studies that reflect the growing interest in the identification of the most distinguishing structural network features and assessing structural similarity between networks. Our literature review is organized according to the following questions:

(1) What structural properties should be considered in model calibration, i.e. what are the most distinguishing features of networks?
(2) How to measure the similarity between networks, in particular between model-generated graphs and real networks?
(3) What machine learning techniques should be used to distinguish between various classes of networks, in particular between model-generated graphs and real networks?
(4) What time-efficient optimization or statistical approximation method should be used for model parameter calibration?

There are no canonical methods to answer the above questions, several approaches have been proposed in the literature throughout the years, which we summarize in the following subsections. Furthermore, we also highlight how this work contributes to the literature.

2.1 Graph metric selection

There is a great deal of effort related to finding a non-redundant size-independent set of graph measurements. For example, Garcia-Robledo *et al.* (2013) follow a data-driven approach to study the Pearson correlation of different properties of evolving Internet networks and apply clustering techniques to find a non-redundant set of metrics.

Bounova & de Weck (2012) give a great overview of network topology metrics and their normalization. They also carry out correlation and principal component analysis on a relatively large dataset that consists of both model-generated graphs and real networks. Similarly, Filkov *et al.* (2009) investigate both network models and a collection of 113 real networks to find a set of metrics, which enables comprehensive comparison between any two networks. Furthermore, Jamakovic & Uhlig (2008) study the correlation of metrics and visual comparison of distribution node-level features using 13 real networks.

Sun & Wandelt (2014) use a dataset of air navigation route systems and investigate functional dependencies among graph metrics through regression analysis to select a less redundant set of metrics. Their metric selection relies on a graph, where the nodes correspond to the metrics and an edge is drawn between two nodes if the predictive power (measured in R^2) of the corresponding metrics have on each other is above a certain threshold. A metric is selected from each connected component of this auxiliary graph. Harrison (2014) gives an in-depth study of the behavior of network metrics of model-generated graphs. Moreover using statistical learning techniques (Fisher's method and ROC curve analysis), the author also investigates graph similarities through the distinguishing power of different selections of metrics.

Relying on the reviewed literature, we also propose a similar metric selection procedure. However, in contrast to the majority of the aforecited papers, in this paper we use Spearman's rank correlation coefficient instead of the Pearson correlation since rank correlation can measure non-linear relationships as well, moreover, it is less sensitive to outliers (Croux & Dehon, 2008), hence it provides us more useful information about the relationship of the metrics.

2.2 Graph similarity

Another related area of research is concerned with estimating the similarity between graphs with unknown node correspondences. Similarities can be defined in numerous ways, for instance, based on variations of *graph isomorphism*, which involves subgraph and degree iso-morphism (Kelmans, 1976; Wilson & Zhu, 2008) and edit distances (Gao *et al.*, 2010). Another approach utilizes *graph kernels*, which are kernel functions that compute inner products on graphs measuring their similarity (Ralaivola *et al.*, 2005; Vishwanathan *et al.*, 2010; Kashima *et al.*, 2004; Kang *et al.*, 2012; Hammond *et al.*, 2013). *Motif counting* is an alternative frequently used tech-nique (Milo *et al.*, 2002; Pržulj, 2007; Ikehara & Clauset, 2017; Faust, 2006; Bordino *et al.*, 2008; Janssen *et al.*, 2012), which refers to the investigation of the distribution of small "graphlets" (e.g. all graphs of size 3 or 4). NetEmd, a network comparison method introduced by Wegner *et al.* (2018) is also based on the distribution of motifs. *Network Portraits* provide an additional method to compare complex networks as well as to visualize their structural characteristics (Bagrow *et al.*, 2008).

A significantly different technique applied by Leskovec *et al.* (2010) and Sukrit *et al.* (2016), which works by first fitting the Kronecker Graph model to the networks, and then comparing the fitted parameter matrices. Furthermore, a similarity metric can also be defined with the help of *dK*-distributions (Sala *et al.*, 2010; Mahadevan *et al.*, 2006). A drawback of the latter two methods is that they are computationally highly expensive.

In a recent work, Mheich *et al.* (2018) proposes an alternative algorithm called *SimiNet* for measuring similarity between graphs, which takes into account the physical location of the nodes, i.e. *SimiNet* requires prior knowledge of the coordinates of the nodes lying in a 3D coordinate system. Schieber *et al.* (2017) introduce a novel measure for network comparison, which is based on the Jensen–Shannon differences between the sets of node-distance probability distributions defined for each node. Chen *et al.* (2018) propose another method based on the Jensen–Shannon divergence, referred to as *communicability sequence entropy* that considers the communicability between nodes which is related to the path lengths.

In this work, we apply another possible calculation of network similarity, which is based on a feature vector representation of graphs. These feature vectors can contain global information of the graphs (e.g. average degree, diameter, and z-score of a motif) or can be obtained by graph embedding methods (Rossi *et al.*, 2017; Narayanan *et al.*, 2017). The derived vectors from the graphs are usually compared with the traditional distance functions, such as Canberra (Bonner *et al.*, 2016*b*), weighted versions of Manhattan and Euclidean distances (Aliakbary *et al.*, 2015*b*; Sun & Wandelt, 2014), or a suitable distance function can be constructed with "Distance Metric Learning" (Ktena *et al.*, 2017; Aliakbary *et al.*, 2015*a*; Yang & Jin, 2006; Weinberger & Saul, 2009).

By the supervised nature of distance (or similarity) metric learning, it requires pairs of graphs with known distances (or similarities), which are usually unknown in reality. The usual approach to overcome this problem is to consider two graphs similar if they both originate from the same domain or class and dissimilar otherwise. For example, this method is well-suited if one aims to distinguish between certain types of networks, e.g. normal versus cancer cell networks, or con-nectome networks of patients with and without autism spectrum disorder (Ktena *et al.*, 2017). On the other hand, this technique is not applicable in our case, since it is one of our research goals to reveal how similar the network domains are without any prior assumptions about their similarity.

In a recent work, Bai *et al.* (2019) propose an alternative method to calculate graph similarity with efficient time complexity. Their machine-learning-based approach combines two strategies, namely, it performs both graph-level and node-level embedding to obtain fine-grained informa-tion of the graphs. The final similarity score is calculated through a deep neural network whose inputs are the results of the embeddings. In order to train the deep neural network, the authors used a ground-truth similarity score based on the Graph Edit Distance (GED) of the training graphs (Gao *et al.*, 2010). This also can be seen as a disadvantage, since the method actually approximates the GED of two networks. However, as Schieber *et al.* (2017) showed with a simple

example, GED is not a perfect graph distance metric, since graphs that are close to each other according to GED can have totally different structural properties.

Martínez & Chavez (2019) have recently shown that Euclidean distance (defined by the Frobenius norm of the difference of adjacency matrices) of equal-sized graphs can be more effective (or at least provide a better trade-off between accurate and fast performance) than other more complicated methods introduced by Hammond *et al.* (2013) and Schieber *et al.* (2017).

Finally, Soundarajan *et al.* (2014) give a great overview, categorization, and guidance for network similarity methods reflecting the state-of-the-art techniques in 2014.

In this paper, we investigate the similarity of networks stemming from different domains and models based on the Canberra distance, because according to Bonner *et al.* (2016b), it is the most suitable distance metric to compare graphs through their feature vectors.

2.3 Network classification

Using classification techniques to distinguish between real and model-generated graphs has attracted a lot of research interest recently. However, the perspective of these studies differs from ours. The vast majority of related papers do not calibrate the parameters of the models, but rather focus on the *spanning* graph space of the models, i.e. they consider a network model as a special domain of networks. Naturally, with certain parameter settings, every model can generate graphs that are not similar to real graphs at all, which can bias the distinguishing power of graph metrics. Hence, that is why in this work we are focusing on the parameters of the models that induce as realistic graphs as the given model is capable of.

Janssen *et al.* (2012) propose a method to choose a network model that best fits a given real network. Their model selection consists of three steps: first, they generate 6,000 graphs with six well-known models and extract graph metrics and graphlet distributions from these graphs. The parameters of the models are randomly sampled such that the size and the density of the generated graphs are similar to the target networks'. Second, they train an Alternating Decision Tree classifier on the previously obtained dataset, where the target variable is the type of the model. Finally, they compute the same feature vectors for Facebook networks and then feed these vectors to the trained classifier to see how well the models fit the given networks.

Ikehara & Clauset (2017) apply machine learning tools to investigate the clustering coefficient, the distribution of motifs and degrees in 986 real and 575 model-generated graphs in order to find the most distinguishing features of real and synthetic networks. In this work, we do not consider the distribution of motifs because these attributes are rather difficult to interpret, furthermore, their calculation is computationally expensive.

Canning *et al.* (2018) use Random Forest to predict the origin (network domains and type of the models) of 402 real and 383 generated graphs; however, they use metrics such as the total number of triangles and density, which are strongly correlated with the size of the network, which is a very strong distinguishing feature in network classification.

Bonner *et al.* (2016a) propose a Deep Learning based approach, called *Deep Topology Classification*, to classify graphs by both global and local features. They evaluate their methodology on a large dataset consisting of synthetic graphs generated by five well-known network models with heuristically chosen parameters.

Barnett *et al.* (2019) recommend using the random forest algorithms to solve network classification tasks based on manually selected network features. We will follow a similar procedure in this work.

While most of the related works focusing on graph classification investigate a particular network domain such as protein interaction networks (Middendorf *et al.*, 2005), chemical compounds and cell graphs (brain, breast, and bone) (Li *et al.*, 2012; Ralaivola *et al.*, 2005; Ktena *et al.*, 2017) or Facebook networks (Sala *et al.*, 2010; Ugander *et al.*, 2013; Janssen *et al.*, 2012). In this paper, we study six different network domains and their model-generated counterparts.

2.4 Model calibration

A closely related work is by Sala *et al.* (2010) who fit six network models to four large Facebook networks by grid search parameter calibration. For the parameter fitting, they use a *dK*-series based similarity metric (Mahadevan *et al.*, 2006) and evaluate the "fidelity" of the fitted models by calculating the Euclidean distance between graph measurements derived from the target and the model-generated graphs.

Another similar paper is by Bläsius *et al.* (2018) who fit four random graph models to 219 real-world networks (mostly from Facebook) and evaluate their goodness-of-fit with respect to different selections of graphs measurements by investigating the failure rates of a Support Vector Machine classifier that predicts whether the corresponding vector of graph metrics belongs to real or model-generated networks.

Aliakbary *et al.* (2015b) follow a two-step approach to first find the most suitable model for a given real network and secondly to fine-tune its parameters. In their work, first, they generate graphs with different models using randomly chosen parameter settings to obtain the spanning graph space of each model. Then using a distance learning method, a distance metric function is defined that distinguishes well networks that were generated by different models. They use a Nearest Neighbour (kNN) classifier with the previously defined distance function to assign a model to each target real network. The parameters of the previously assigned model are fine-tuned by artificial neural networks (ANN).

In a follow-up paper, Attar & Aliakbary (2017) extend the previous network calibration framework as follows: instead of generating graphs with random parameter settings as an input for the distance learning method, the authors first fit each network model to the target networks (using ANN). To train the distance learning, they generate a few graph instances for each target network with the fitted parameters of each model, and finally, the best performing model is selected with the former kNN classifier. A limitation is that neural networks usually require large datasets to perform well, furthermore as Bläsius *et al.* (2018) also pointed out, this method in its original form is not able to identify the structural characteristics of real networks that models can or cannot capture (cf. objective 2).

While the afore-cited papers apply search optimization techniques, in some cases statistical parameter estimating is also possible. In the following works, a network model is considered as a probability distribution over graphs. Bezáková *et al.* (2006) propose a Maximum-Likelihood-based parameter estimation (and comparison) method for four graph models. Recently, Arnold *et al.* (2021) introduced a likelihood-based approach to find which growing model is more likely responsible for the generation of a network. Their work also involves multiple related tasks such as parameter calibration, graph similarity, and network classification. Fay *et al.* (2014) study Approximate Bayes Computation as a technique for estimating the parameters of network models relatively to an observed graph. An additional brief survey of generating realistic synthetic graphs is given by Lim *et al.* (2015).

In this work, we also apply a sophisticated model calibration technique (see Section 3.4) on 500 networks from six different domains.

3. Data and methodology

This study relies on our large dataset: we calculated 17 metrics of 500 real-world networks and a subset consisting of 8 reasonably chosen (see Section 3.1) metrics for 4×500 synthetic graphs by generating four counterparts for each real network with well-known network models. The dataset of measurements is available online (Nagy & Molontay, 2021). In this section, we detail the composition of our dataset and describe the main challenges and our method of model calibration.

Table 1. Composition of the collected set of real networks

| Domain	Description	Range of network sizes	Number of networks
Brain	Human and animal connectomes	50–2,995 (avg: 946)	100
Cheminformatics	Protein–protein (enzyme) interaction networks	44–125 (avg: 55)	100
Food	What-eats-what, consumer-resource networks	19–1,500 (avg: 117)	100
Infrastructural	Transportation (metro, bus, road, airline) and distribution networks (power and water)	39–40K (avg: 4,562)	68
Social	Facebook, Twitter and collaboration networks	86–34K (avg: 6,183)	118
Web	Samples of the World Wide Web	146–16K (avg: 4,488)	14
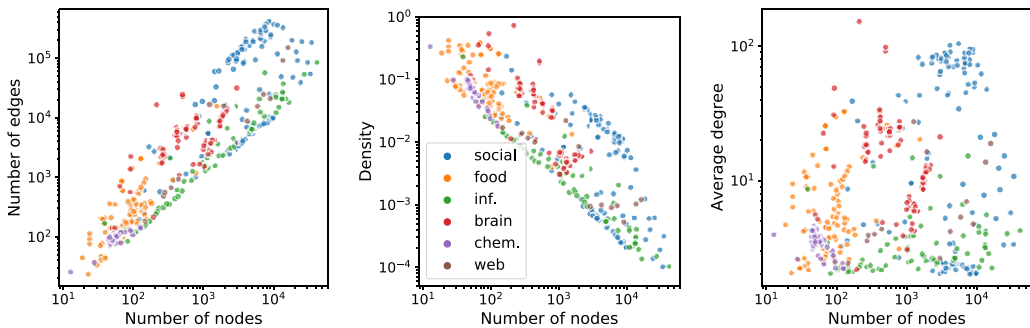
Figure 2. The scatter plot of the real networks. The points are colored according to the network domain. On the left: the log–log plot of the number of edges against the number of nodes. The center figure shows the the log–log plot of the graph density versus the number of nodes. On the right: the log–log plot of the average degree against the number of nodes.

3.1 Data of real networks

Networks from six different domains are studied in this paper, namely food, social, cheminformatics, brain, web, and infrastructural networks. The collected networks are also available online (Nagy & Molontay, 2021). Table 1 briefly describes the composition of network domains.

The graphs were gathered one by one from online databases, namely Network Repository (Rossi & Ahmed, 2015), Index of Complex Networks (Clauset *et al.*, 2016), NeuroData's Graph Database (Kasthuri & Lichtman, 2008), The Koblenz Network Collection (Kunegis, 2013), Interaction Web Database (Diego *et al.*, 2003), Transportation Networks for Research (Stabler *et al.*, 2019), and Centre for Water Systems (Butler & Djordjević, 2014). After importing the graphs, we removed self-loops and treated them as undirected, unweighted graphs.

The left side of Figure 2 shows the range of the number of nodes and edges of the collected networks. In Figure 2, the slope (gradient) of the left scatter plot illustrates the sparseness of real networks, i.e., the number of edges grows rather linearly than quadratically. Furthermore, in the center scatter plot, the slope is roughly minus one which means that the density decays hyperbolically with the number of nodes, i.e., *density* ~ $1/size$. Comparing the center and the right figure of Figure 2, we can clearly observe that the average degree is less size-dependent than the graph density.

3.2 Metrics and notations

First, we calculate the following rich set of metrics for the real networks: assortativity, average clustering coefficient, average degree, average path length, density, global clustering coefficient, four IDPs (Aliakbary *et al.*, 2014), largest eigenvector centrality, maximum degree, maximum edge

and vertex betweenness centralities, number of edges and nodes, and pseudo diameter. If one aims to investigate these measurements for significantly different-sized networks, some normalizations are required, in particular in the case of the average path length, diameter, and maximum degree. Relying on Bounova & de Weck (2012), we normalized the maximum degree with the number of nodes minus one, however, as opposed to Bounova and de Weck, instead of the longest possible path length, we normalized the average path length and the diameter with the logarithm of the size of the graph to obtain less correlated variables with the network size. The reason why the average degree does not require normalization is that its normalized version (with the size minus one) is the graph density (Bounova & de Weck, 2012). The graph density D is defined as the ratio of the number of edges $|E|$ with respect to the maximum possible edges, i.e., $D = |E|/\binom{|V|}{2} = \frac{|E|}{|V|(|V|-1)}$.

In order to quantify the degree distribution of the networks, we apply the *interval degree probability* method introduced by Aliakbary *et al.* (2014), which is the computation of histogram values with special different sized bins that are defined using the mean and the standard deviation of the degrees. In contrast to the work of Aliakbary *et al.* (2014), we defined four bins instead of eight because we found that otherwise many of the bin intervals end up empty. Hence, we quantify the degree distribution of the graphs with four values of IDPs. Formally, the IDPs can be defined as follows:

For a given graph G, let μ and σ be the average and the standard deviation of the degree distribution of G. Then we define the following four intervals $R(i)$, $i = 1, \ldots, 4$:

$$R(i) = \begin{cases} [\,1, \ \mu - \sigma\,), & \text{if } i = 1 \\ [\,\mu - \sigma, \ \mu\,), & \text{if } i = 2 \\ [\,\mu, \ \mu + \sigma\,), & \text{if } i = 3 \\ [\,\mu + \sigma, \ \infty\,), & \text{if } i = 4 \end{cases} \tag{1}$$

The interval degree probabilities $IDP(i)$, $i = 1, \ldots, 4$ are defined as follows:

$$IDP(i) = P(\deg(v) \in R(i)), \quad v \in V(G)$$

3.3 Correlation network

Throughout this paper, we often refer to the *correlation network*, which is frequently used object in the literature (Garcia-Robledo *et al.*, 2013; Jamakovic & Uhlig, 2008; Sun & Wandelt, 2014; Bläsius *et al.*, 2018). A correlation network is an undirected weighted graph with nodes corresponding to variables (in our case graph metrics) and two nodes are connected if their absolute correlation (in our case Spearman's rank correlation) is greater than a given threshold. The weights of the links are the values of the absolute correlations.

In this work, the application of the correlation network is three-folded: we use it to visualize the strong universal correlations of graph metrics, it is also a key component of the metric selection. Furthermore, we propose a novel technique for quantifying a similarity score between groups of networks, based on the adjacency matrices of the correlation networks corresponding to different groups (e.g. domains and types of network models).

3.4 Model calibration

For each real network, we generated additional four graphs by four different models with previously calibrated parameters to mimic real networks as accurately as possible. We study how easy it is to calibrate the network models and what descriptive ability they have with respect to real-world networks. The four chosen models are the clustering version of the Barabási–Albert model (Holme & Kim, 2002), the 2K-Simple model (Gjoka *et al.*, 2015), which captures the joint-degree distribution of a graph, the forest-fire model (Leskovec *et al.*, 2005), which was motivated by citation patterns of scientists, and the (degree-corrected microcanonical) stochastic block model (Peixoto,

2017) that creates graphs with community structure. The 2K and the stochastic block models are more complex models than the other two and can be considered state-of-the-art models for synthesizing real-world networks while preserving their key structural properties.

With a given target network G_T (real network), a network model $M(\theta)$ with parameter vector θ and a similarity s (or distance d) function defined on graphs pairs, the goal of model calibration is to find a parameter vector θ^* of the model, such that the generated graph $G_{M(\theta^*)}$ (realization of $M(\theta^*)$) is as similar (or as close) to the target network as possible. Formally it can be expressed as:

$$\theta^* = \arg\max_{\theta} \; s\big(G_{M(\theta)}, \, G_T\big) = \arg\min_{\theta} \; d\big(G_{M(\theta)}, \, G_T\big) \tag{2}$$

Note that one can maximize the average similarity (or minimize the distance) between the target network and a few realizations of the model with the same parameters. That is:

$$\theta^* = \arg\max_{\theta} \; \frac{1}{n} \sum_{i=1}^{n} s\left(G_{M(\theta)}^{(i)}, \, G_T\right) = \arg\min_{\theta} \; \frac{1}{n} \sum_{i=1}^{n} d\left(G_{M(\theta)}^{(i)}, \, G_T\right) \tag{3}$$

where n is the sample size of the independent identically generated (i.e. generated with the same θ parameter vector) graphs $G_{M(\theta)}^{(1)}, \ldots, G_{M(\theta)}^{(n)}$. However, in this work, we only consider one realization of the models for two reasons, first, we show in the following subsection that the models are stable enough to perform parameter calibration as defined in (2); furthermore, the calculation of average similarity (or distance) would be more computationally expensive.

In this work, the d distance function is defined as the Canberra distance of a reasonably chosen non-redundant selection of graph metrics, formally: $d(G_1, G_2) = d_{\text{Can}}(f(G_1), f(G_2))$, where $f = (f_1, f_2, \ldots, f_k)$, and f_i's are real-valued functions defined on graphs (i.e., graph metrics) and k is the number of selected metrics. The process of metric selection is detailed later in Section 4. The Canberra distance of two k-dimensional vectors is defined as $d_{\text{Can}}(x, y) = \sum_{i=1}^{k} \frac{|x_i - y_i|}{|x_i| + |y_i|}$.

The distance minimization in this paper is carried out with grid search, particularly for a given $M(\theta)$ model, where $\theta = (\theta_1, \theta_2, \ldots, \theta_p)$, we perform a grid search on the p dimensional state space of the parameters. In what follows we briefly outline the parameter calibration of each network model that we investigate in this work.

3.4.1 Clustering Barabási–Albert model
The Clustering Barabási–Albert (CBA) model, introduced by Holme & Kim (2002), has three input parameters: n number of nodes, m new edges of a newcomer vertex, and p probability of triad formation. This p parameter influences the clustering characteristic of the generated network. Holme & Kim (2002) showed that for fixed n and m, there is a linear relationship between p and the average clustering coefficient. The parameter calibration of the CBA model is as follows: Let G_T denote the target network, furthermore $\bar{d}$ denotes the average degree of G_T. Then the parameter n of CBA is simply the size of G_T, and m and p are chosen by:

$$(m, p) = \arg\min_{(k,q) \in M \times P} d\big(G_{\text{CBA}(n,k,q)}, \, G_T\big)$$

where $M = \left\{\left[\bar{d}/2 - 1\right], \left[\bar{d}/2\right], \left[\bar{d}/2 + 1\right]\right\}$, $P = \{0, 0.05, 0.1, \ldots, 1\}$, $M \times P$ is the Cartesian product of M and P and $\left[\,.\,\right]$ denotes the rounding (nearest integer) function. The reason behind the choice of set M is that CBA model with $m = \left[\bar{d}/2\right]$ parameter setting produces graphs with $e = n \cdot \left[\bar{d}/2\right]$ edges, i.e. the average degree in the generated graphs is $2e/n$ which is around $\bar{d}$ as one would desire. We also allow for a small deviation from this value in the minimization that may lead to a better fit overall. For example, it may happen, that if the value of the parameter m is slightly smaller (or larger) than the half of the average degree of the target graph, then it leads to

a better fit regarding the other graph measurements of the generated graphs. For generating clustering CBA graphs, we applied the `powerlaw_cluster_graph` function of NetworkX (Hagberg *et al.*, 2008).

3.4.2 Forest-fire model

The forest-fire model (FF) introduced by Leskovec *et al.* (2005) has two parameters: n number of nodes and p burning probability. Using the previous notations, n is set to be the size of G_T target graph and we minimize in parameter p as follows:

$$p = \arg \min_{q \in Q} d\big(G_{\mathrm{FF}(n,q)}, G_T\big)$$

where $Q = \{0.05, 0.1, 0.15, \ldots, 0.90, 0.95\}$. The reason why q is between 0.05 and 0.95 in the minimization, is that q must be positive and less than 1, moreover, the structure of the generated graph is not sensitive to q if its value is smaller than 0.05 or greater than 0.95. In our analyses, we used the `Forest_Fire` function of python-igraph (Csardi & Nepusz, 2006).

3.4.3 2K model

To fit the 2K model (Gjoka *et al.*, 2015), only the joint degree distribution of the target network is needed. We used the `joint_degree_graph` function of the NetworkX (Hagberg *et al.*, 2008) module to generate simple random graphs with the given joint degree distribution. Let $k_{j,l}$ denote the number of edges in the target graph whose endpoints' degree is j and l. Then the 2K model builds the graph, such that it copies the nodes of the target graph and then in each iteration, the algorithm picks two disconnected nodes v and w of degree j and l correspondingly, for which the $k_{j,l}$ in the random graph has not reached its target yet, and it adds the edge (v, w) to the synthetic graph.

3.4.4 Stochastic block model

The degree-corrected stochastic block model can be fitted by the `minimize_blockmodel_dl`, and random graphs can be generated using the `generate_sbm` functions of the graph-tool package (Peixoto, 2014). We applied the microcanonical version of the model (Peixoto, 2017). The stochastic block model is fitted by minimizing its description length using an agglomerative heuristic.

3.5 Stability of the models

Since the network models generate random graphs, the question naturally comes up: How robust are the graph metrics of the fitted models with fixed parameters? We have analyzed the sensitivity of the models on six different-sized graphs from different domains, the chosen networks are detailed in Table 2. For each of these six real networks, we fitted each network model and then generated 30 graph instances with each model using the previously fitted parameter settings. For the sensitivity analysis, we studied the distribution of the graph measurements of the graph instances.

Figure 3 shows the distribution of the metrics of the measurement-calibrated models for the social network that has 17,903 nodes. Since the 2K model captures all the information about the joint degree distribution, when it generates a connected graph, it is capable of exactly capturing the assortativity, the IDPs, and the degree centrality. Overall, the models could not estimate the average clustering coefficient well, not even the clustering Barabási–Albert model. Besides the clustering coefficient, the models also have difficulties capturing the average path length and the eigenvector centrality. Generally, the SBM and 2K models are more robust and "stable", than the CBA and FF models.

Table 2. Description of chosen graphs for stability analysis. The names of the graphs are also hyperlinks to their sources, moreover, these graphs are also analyzed in the cited papers

Domain	Name	Size	Number of edges
Social	ca-AstroPh (Leskovec *et al.*, 2007)	17,903	196,972
Web	Darkweb (Griffith *et al.*, 2017)	7178	24,879
Brain	Jung2015 (Kiar, 2016)	2989	31,548
Infrastructure	ABN (Chatterjee *et al.*, 2016)	1113	2150
Food	Srep (Dunne *et al.*, 2016)	235	1743
Cheminformatics	Enzymes-g292 (Canning *et al.*, 2018)	60	100
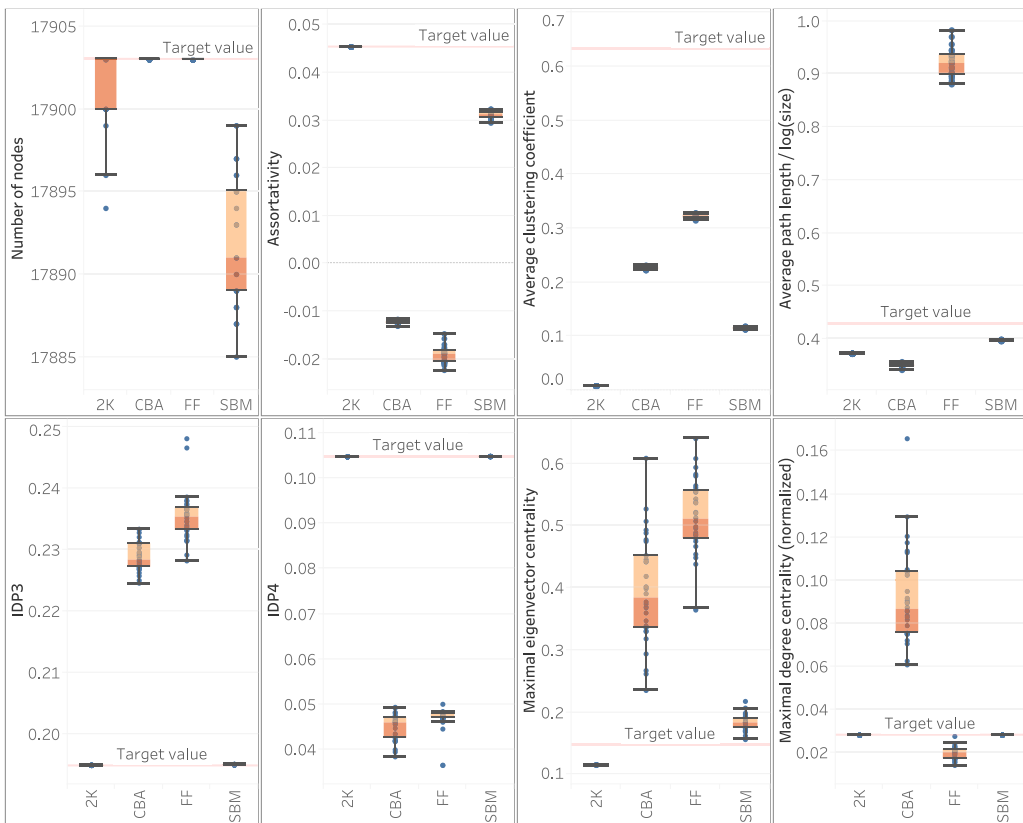
Figure 3. Boxplots of the metrics of the measurement-calibrated models on a social network, namely on the the Astro Physics collaboration network.

The boxplots for the other domains can be found in the supplementary material (Nagy & Molontay, 2021). The results suggest that as the size of the networks increases, the variance of the graph measurements of the fitted models decreases, however, the variance of the metrics is already relatively small on the smaller graphs.

We can conclude, that the characteristics of the calibrated models are stable enough to take a single generated graph as a representative, and it makes it reasonable to analyze the goodness-of-fit of the network models and perform machine learning tasks.

Note that reason why the number of nodes is not fixed for the 2K and the SBM models is that these models do not always generate connected graphs, and in this study we took the largest
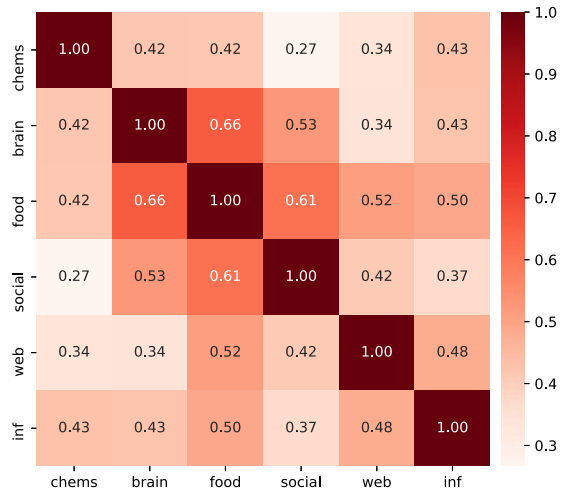
Figure 4. The similarity of the network domains based on their correlation networks. The similarities are calculated as the median adjacency similarities of correlation networks over different threshold settings.

connected components of both the real-world and the model-generated networks. Furthermore, the reason why IDP1 and IDP2 are not shown in Figure 3, is that first, it is enough to study three of these metrics, since the sum of them gives one. Moreover, in this work, we selected IDP1, IDP3, and IDP4 (the selection process is detailed later in Section 4.2). The reason why IDP1 is not shown in Figure 3, is that for this specific network (and its model-generated counterparts), the value of IDP1 is zero in all cases (due to the heavy-tailed degree distribution).

4. Analysis of network domains

In this section, we analyze the correlation profiles of the structural measures across different network domains bearing in mind the following three objectives:

1. Study the structural similarities between domains
2. Identify domain-independent relationships between metrics
3. Choose a less redundant subset of metrics, that can still distinguish the network domains

For analyzing correlation profiles, we investigated the pairwise correlations of the metrics on correlation heatmaps and *correlation networks* based on the Spearman's rank correlation that can also measure nonlinear relationships; moreover, it is less sensitive to outliers (Croux & Dehon, 2008).

4.1 Domain similarity

Figure 4 shows the similarity of the network domains, where the similarity is captured by the pairwise Spearman's correlations of graph metrics across domains. The similarity scores are calculated through the correlation networks of the domains (using all variables detailed in Section 3.2), namely first we created 17 correlation networks over 17 threshold settings (0.1, 0.15, 0.2, ..., 0.85, 0.9) for each domain separately. Then for all thresholds, we calculated the pairwise adjacency similarities of the correlation networks of different domains. The cells of the heatmap correspond to the median of these 17 adjacency similarities.

To calculate the adjacency similarity, we used the `similarity` function of the graph-tool package, which is the ratio of the number of edges that have the same source and target nodes in both graphs and the sum of the number of edges of the graphs. Formally let $A^{(1)}$ and $A^{(2)}$ denote the
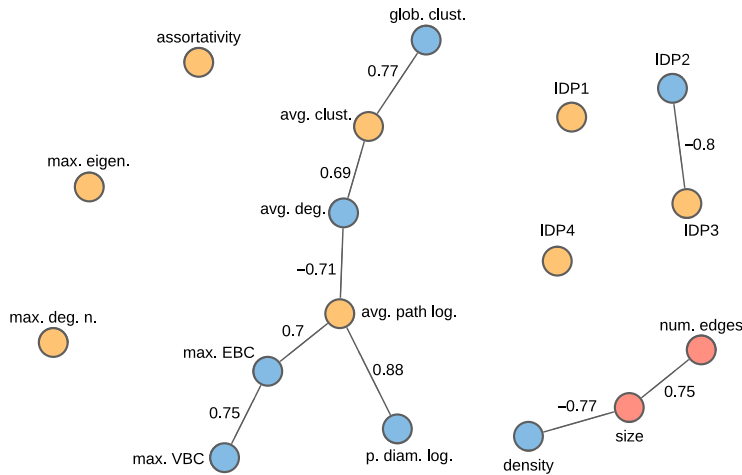
Figure 5. The correlation network of structural metrics. Two metrics are connected if the domain-averaged absolute Spearman's rank correlation is above 0.65. The actual values of the correlations are written on the edges. The selected metrics are colored orange. Note that the max degree, avg. path length, and the pseudo diameter are normalized.

adjacency matrices of the G_1 and G_2 graphs. Then the adjacency similarity S of G_1 and G_2 is defined as follows:

$$S(A^{(1)}, A^{(2)}) = \frac{\sum_{i \leq j} A_{ij}^{(1)} + A_{ij}^{(2)} - \sum_{i \leq j} \left| A_{ij}^{(1)} - A_{ij}^{(2)} \right|}{\sum_{i \leq j} A_{ij}^{(1)} + A_{ij}^{(2)}} = \frac{2 \cdot | E_1 \cap E_2 |}{|E_1| + |E_2|}$$

where E_1 and E_2 are the edge sets of the graphs G_1 and G_2 respectively.

Figure 4 suggests that the correlation profile of social, brain, and food networks are relatively similar, moreover the cheminformatics networks are rather dissimilar to the other domains. The reason behind it is that in the cheminformatics domain the correlations are relatively weak between the metrics, hence larger threshold settings shatter the correlation network into several components. Furthermore, the adjacency similarity only takes into account the mutual edges, i.e., two correlation graphs of isolated nodes are totally dissimilar. It is interesting to note that even though Ikehara & Clauset (2017) applied a totally different approach, the domains that they found to be similar are consistent with our results except for the behavior of the food webs.

4.2 Correlations of graph metrics

Next, we aim to identify the strong domain-agnostic correlations among graph metrics. Here, we consider the relationship of two metrics strong if their absolute Spearman's rank correlation is at least 0.65.

Figure 5 shows the correlation network of the graph metrics, where the links are drawn if the domain-averaged absolute Spearman's rank correlation is above 0.65. Some of the identified strong domain–agnostic relationships are not so surprising, such as the high positive correlation between the average path length and diameter, the average clustering coefficient and global clustering coefficient, and between the maximum vertex betweenness centrality and maximum edge betweenness centrality.

There are also other strong, less trivial relationships among the metrics as well, e.g. the strong negative correlation between density and the number of nodes, which implies that the density

depends on the size of the graphs. This may be because real networks are usually sparse, i.e. the number of edges grows rather linearly than quadratically with the number of nodes, as the slope of the left scatter plot of Figure 2 also illustrates. On the other hand, by the definition of density, the number of edges is normalized by the square of the number of nodes. That is why considering real networks, density decays hyperbolically with the size of the graphs, as it can be seen in the middle of Figure 2 since the slope of the log–log plot of density against size is roughly minus one. This implies that the average degree is a less size-dependent metric than graph density (see Figure 2) which makes it a better measurement for quantifying the extent of relative connectivity of the nodes.

The average degree and the average clustering coefficient are also positively correlated with each other regardless of domains. The reason behind this connection is that both of them measure how dense the network is, however, these are not always correlated with the graph density metric, e.g. in the case of infrastructural networks. On the other hand, given the size of the network, the partial correlation of these metrics with the density is significantly larger than the Spearman's correlations. For a similar reason, there is a strong inverse connection between the average degree (and maximum degree) and the average path length, since the higher the number of edges is and the more hubs are present in the network, the shorter the average distances are. This is consistent with the finding of Bounova & de Weck (2012), which supports that distance-related metrics (e.g. diameter and betweenness centrality) and connectivity-related metrics (e.g. average degree and clustering coefficient) are two highly connected clusters that are strongly negatively correlated, even though Bounova & de Weck (2012) investigate model-generated graphs.

Furthermore, the maximum edge betweenness centrality is always strongly positively correlated with the average path length, which can be easily seen from the definition of the two metrics. Moreover, this is again consistent with the work of Bounova & de Weck (2012) who argued that it can be analytically shown that the average betweenness centrality is linearly proportional to the normalized average path length. Similar observations are made by Jamakovic & Uhlig (2008) and Garcia-Robledo *et al.* (2013) despite the fact that they investigated a different set of networks.

Finally, there are universal connections between the *interval degree probabilities* (IDPs) as well, namely strong negative correlations between IDP1 and IDP2 (-0.61 on average), moreover between IDP2 and IDP3 (-0.8 on average). This implies that IDP2 alone reasonably explains the other two variables. These relations follow from the fact that the sum of these probabilities is one, hence if one of them is relatively large, the others are necessarily low. The reason why IDP4 is usually less correlated with the other probabilities is that only 10–15% of the degrees fall into this interval, and this percentage seems to be stable over all networks, i.e., this IDP has the smallest variance. Thus, the mass of the degree distribution falls into the first three intervals, especially into the second and the third one. Due to the fact that the degree distribution of real-world networks is often heavy-tailed and asymmetric (Barabási, 2016), the first interval often end up empty (Aliakbary *et al.*, 2014). The standard deviation of the degrees is so large, that there are no degrees that are smaller than the difference of the average degree and the standard deviation of the degrees, formally: $\mu_{\deg} - \sigma_{\deg} < \min_v \deg(v)$. Hence if IDP1 is not zero, it can only "steal" degrees from IDP2, that is why these two probabilities are strongly negatively correlated. On the other hand, the degrees corresponding to IDP3 are "closer" to the degrees of IDP2 than the degrees of IDP4, due to the heavy-tailed degree distributions again. Thus in general if IDP3 is high, IDP2 is necessarily low, that is why IDP4 is less influenced by the value of IDP3. Further study of these relationships requires a deeper understanding of the underlying domains and is beyond the scope of this paper.

Overall, the maximum eigenvector centrality, the assortativity, and the IDP4 have the smallest correlation with the other variables. On the other hand, the average degree, average clustering coefficient, and the normalized average path length metrics have the highest correlation with the other variables.

Table 3. The selected graph metrics and the other variables of the narrowed dataset

Name	Description
Assortativity	The assortativity coefficient
Avg. clust.	The average local clustering coefficient
Avg. path log	The average path length divided by the logarithm of the size.
IDP's (1,3,4)	The interval degree probabilities
Max. deg. n.	The normalized maximum degree (max. degree centrality)
Max. eigen.	Maximum eigenvector centrality
Domain	The domain of the real networks: social, food, brain, cheminformatics (chems), infrastructural, and web
Category	Indicates whether the graph is real or generated: real or model
Subcategory	In case of generated graphs, indicates the type of the model as well: real (original), 2K, CBA (clustering Barabási–Albert), FF (forest-fire), SBM (stochastic block model).

4.3 Metric selection

In this section, we aim to select a smaller subset of metrics that describes the networks well but has smaller redundancy. Using a smaller nonredundant set of metrics, the model fitting is less computationally expensive, moreover, we can also avoid collinearity and overfitting. We evaluate our selection via machine learning tools, namely by investigating the predictive power of the chosen metrics with respect to the domains of the networks. In this work, we assume that the reader is familiar with the basic concepts of machine learning, otherwise, for a detailed description of the related concepts, we refer to the book by Friedman *et al.* (2001).

Based on the average absolute correlation network (Figure 5), we select metrics from each connected component of the graph. The metrics from the component containing the size of the network are excluded from the analysis (namely the size, number of edges, and density) since the high correlation with the size implies significant trivial predictive power with respect to network domains due to the size difference of typical networks from different domains (see Figure 2). Moreover, we are more interested in finding distinctive size-independent topological properties. The selected attributes are listed in Table 3. Our selection is slightly different from other work's (Garcia-Robledo *et al.*, 2013; Sun & Wandelt, 2014) since we found that skewness and kurtosis of distributions of node features are highly constrained by the size of the networks. However, assortativity, maximum degree, average path length, average clustering coefficient, and eigenvector centrality are also selected in other works (Garcia-Robledo *et al.*, 2013; Sun & Wandelt, 2014; Harrison, 2014; Sala *et al.*, 2010; Bläsius *et al.*, 2018).

In order to evaluate our metric selection, we predict the domains of the networks, first using all of the metrics, then using only the selected (numerical) attributes from Table 3. Our selection is considered "optimal" if the accuracy of the predictions does not decrease significantly after we drop the redundant attributes. Table 4 contains the average cross-validated accuracy of a few machine learning model. Three important observations can be drawn from Table 4. First, our metric selection achieves very high accuracy. Second, the kNN achieved higher accuracy when it was trained on a smaller set of features. The reason behind this is that kNN is sensitive to irrelevant features (Langley & Iba, 1993), and these features bias the results. Finally, we cannot achieve significantly higher accuracy with random subsampling of the attributes, than the accuracy of our selection. Note that the best-performing random selection is usually quite similar to our selection; however, this set often contains the maximum betweenness centrality, but we would rather not include it in our selection, because the calculation of the betweenness centralities is computationally expensive.

Table 4. Domain prediction performance of the machine learning models using different feature sets. The performance is calculated as the 5-fold cross validated average accuracy . The standard deviation of the accuracy is shown in brackets. We performed 1000 random subsampling of the attributes, with sample size 8 (since the number of selected attributes is also 8). The third and the fourth columns show the maximum and the median accuracy over all of the 1000 random attribute selections correspondingly. We excluded the size, number of edges, and density from each attribute set

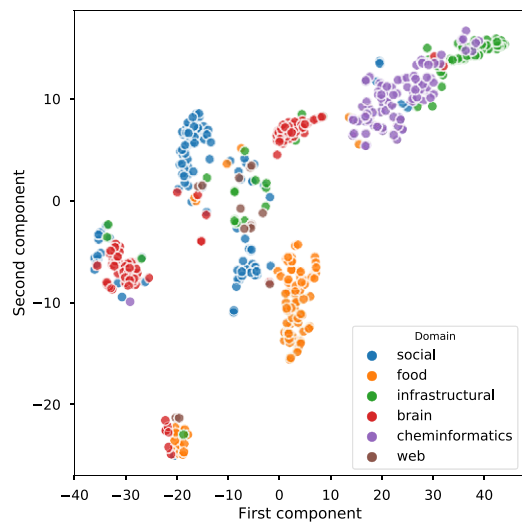| Classifier model	All attributes	Selected attributes	Maximum of randomly selected	Median of randomly selected
kNN	76.8%	84.4%	85.6%	75.2%
	(±3.9%)	(±1.0%)	(±1.6%)	(±3.9%)
Decision tree	82.8%	82.6%	84.8%	79.2%
	(±1.6%)	(±2.6%)	(±2.3%)	(±3.2%)
Random forest	87.6%	89.0%	89.4%	85.6%
	(±2.2%)	(±1.1%)	(±2.4%)	(±4.2%)

Figure 6. The t-distributed stochastic embedding (t-SNE) of the eight selected graph metrics (detailed in Table 3) of the real networks. Apparently, one can distinguish between the domains reasonably well using the selected metrics.

Figure 6 shows the t-distributed stochastic embedding (t-SNE) (Maaten & Hinton, 2008) of real networks represented by the selected eight metrics. It can be seen that the network domains can be relatively clearly distinguished from each other, i.e. this figure also supports the validity of our metric selection.

5. Analysis of network models

In this section, we use machine learning tools to analyze and evaluate the goodness-of-fit of the network models and the applied parameter calibration method. In particular, we solve the following two classification problems:

1. Using the selected graph metrics (Table 3), we aim to solve a binary classification problem, where the target variable indicates whether the given network is *real* or *model-generated*.
2. Based on the selected graph metrics, we aim to identify the *subcategory* of the networks. The values of subcategory attribute can be *real* or the types of the models (see Table 3). Thus, we aim to solve a multiclass classification problem that predicts whether a network is real or model-generated and if it is the latter also the model that generated it.

Table 5. Performance of the machine learning models in the binary and multiclass classification problems

Classification problem	ML Model	Brain	Chem.	Food	Inf.	Social	Web
Binary classification	kNN	0.950	0.968	0.726	0.856	0.879	0.376
(AUC score)	Decision Tree	0.860	0.856	0.647	0.754	0.821	0.354
	Random Forest	0.926	0.970	0.663	0.885	0.902	0.164
Multiclass classification	kNN	89.6%	83.4%	74.8%	68.5%	77.6%	38.6%
(Accurcacy)	Decision Tree	89.0%	79.2%	71.2%	74.7%	83.4%	48.6%
	Random Forest	91.4%	85.8%	74.0%	82.1%	83.7%	55.7%

Solving these classification problems allows us to answer the research objective 2 of this paper. Identifying the reasons why a (well-performing) machine learning model decides that a given graph is rather real than model-generated (or vicaversa) helps us find and understand the structural properties of real networks that the models are unable to capture. These decision rules are easy to retrieve for example by looking at the *feature importances* of the variables. Moreover, the second classification problem helps us understand which models generate the most realistic networks and also which models generate similar graphs to each other. By carrying out the above analyses separately for each domain, we can also study which network domains are easy or difficult to synthesize.

Table 5 shows the performance of the machine learning models in the binary (Problem 1) and multiclass (Problem 2) classification problems. From the table, it is apparent, that the models can relatively easily distinguish whether the graphs are real or model-generated, except for the food and especially the web domain, where both the AUC and accuracy scores are low.

Thus, the performance of the network models depends on the network domain. Figure 8 shows an example of when the network models perform relatively well, and Figure 7 illustrates a counterexample, i.e. when the network models cannot really synthesize the target networks. Hence, as Figure 7 shows, in the cheminformatics domain the network models are not able to capture exactly the structural properties of the real networks since in the confusion matrix the accuracy for the real networks is relatively high and the real networks are rather separated in the t-SNE embedding. On the other hand, as the confusion matrix and the t-SNE embedding in Figure 8 illustrate on the food domain, the stochastic block model can almost perfectly mimic the networks.

According to the Random Forest classifier, when the goal is to predict whether the network is real or model-generated, the most distinguishing graph metrics are the normalized average path length, the average clustering coefficient, the maximum eigenvector centrality, and the assortativity.

Figure 9 illustrates the structural properties of the real networks that the network models cannot capture. For example, in the cheminformatics domain, the real graphs typically have relatively large average path length and a large clustering coefficient at the same time. However, most of the models—except for the forest fire—can either generate a graph with a high clustering coefficient but a small average path length or the other way around.

Furthermore, most of the infrastructural networks have so large average path lengths that none of the models can recreate (Figure 9). The network models usually generate small-world or ultra-small-world networks, while the diameter of a grid-like infrastructure network, e.g. a road network, does not scale logarithmically.

In the case of social networks, average clustering coefficient and assortativity were the most important distinguishing metrics. As the bottom left scatterplot of Figure 9 shows, the stochastic block model and the 2K model can capture the assortativity of the real networks but cannot mimic the high average clustering coefficient. On the other hand, the clustering Barabási–Albert model and the forest-fire model can generate highly clustered networks, but the assortativity of these
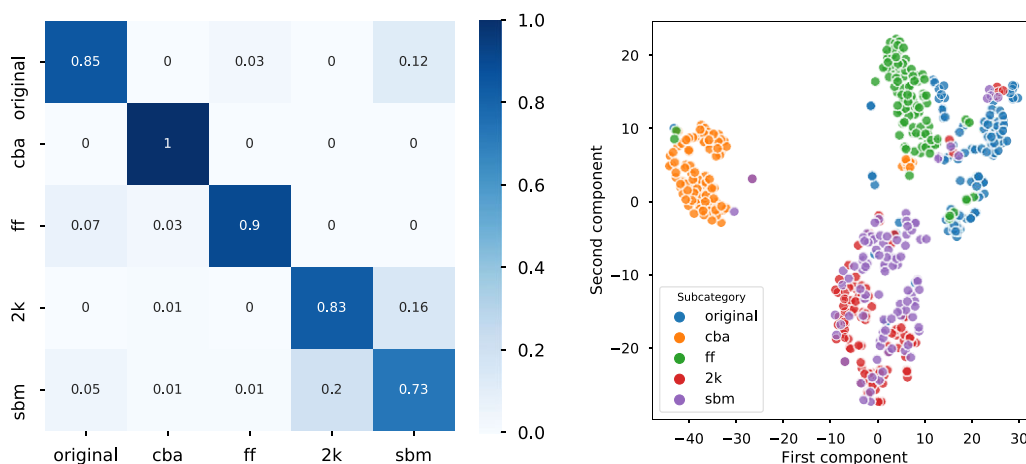
Figure 7. The normalized confusion matrix of the Random Forest on the Cheminformatics network domain (left) and the t-SNE embedding of the corresponding dataset (right).
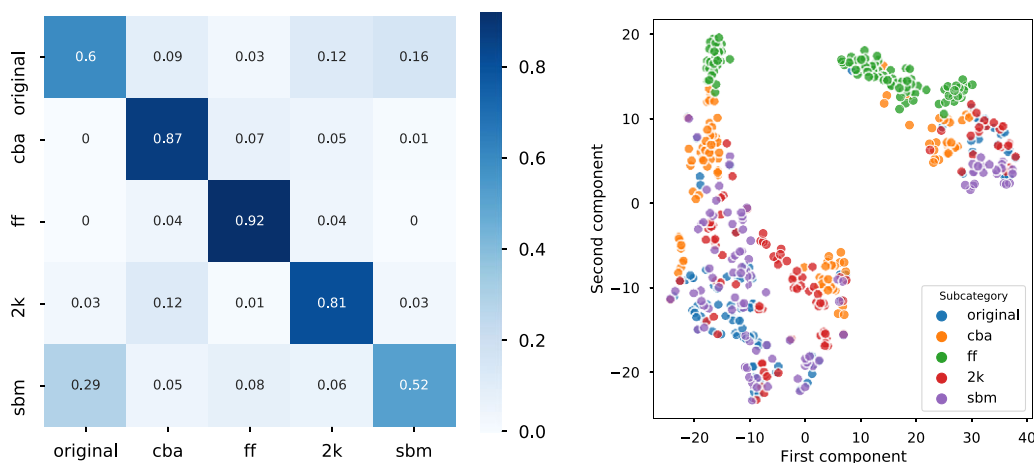

Figure 8. The normalized confusion matrix of the Random Forest on the Food network domain (left) and the t-SNE embedding of the corresponding dataset (right).

model-generated graphs is usually very close to zero. Hence, a subgroup of the social networks with a rather assortative and highly clustered structure cannot be synthesized with these network models.

In the food domain, the two most distinguishing graph metrics are the maximum eigenvector centrality and the normalized average path length. The bottom-right scatter plot of Figure 9 shows that compared to the real food networks, the models either generate graphs with smaller average clustering or with larger eigenvector centrality.

Similarly, in the case of the brain and web domains, the two most important graph metrics are average path length and the average clustering coefficient. The scatter plots for these domains can be seen in the supplementary material (Nagy & Molontay, 2021).

The models, especially the 2K and the stochastic block model, can rather efficiently capture the other metrics, namely the normalized maximum degree and the IDPs. Figure 10 shows that in the
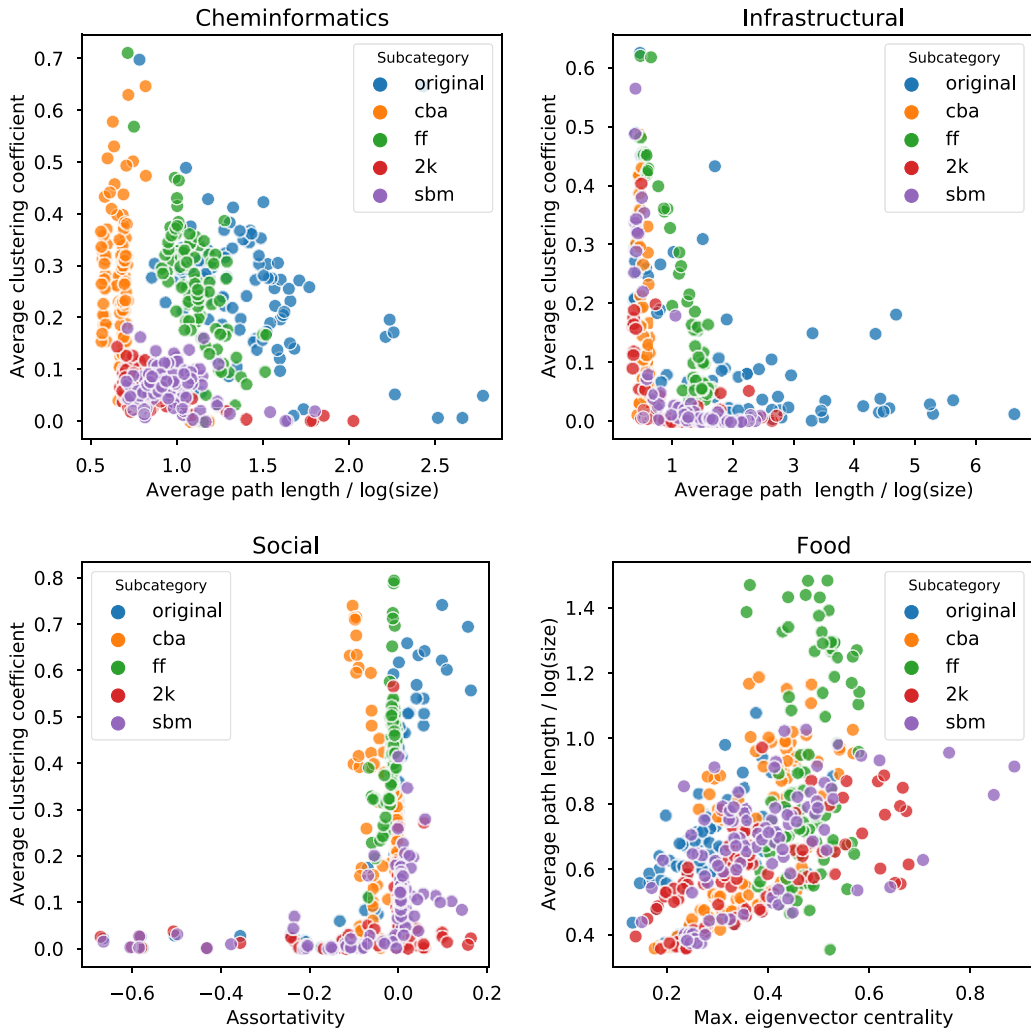
Figure 9. Distribution of the most distinguishing graph metrics of the real networks and the model-generated graphs.

majority of the cases, not so surprisingly the 2K and stochastic block model can exactly capture the aforementioned degree-distribution-related metrics, that is why the corresponding points on the scatter plots completely overlap each other. However, a less sophisticated model, the forest-fire model can also achieve relatively high accuracy regarding these graph metrics, as the bottom-left scatter plot of Figure 10 depicts.

Figure 11 shows the mean Canberra distance between the (selected) metrics of the real target graphs and the fitted model-generated graphs. From the figure, it is apparent that the 2K and the stochastic block models are the most accurate network models in all domains. Moreover, the graphs generated by the forest fire are usually closer to the graphs of the clustering Barabási–Albert model than to the real networks. This may be due to the fact that these two models have a similar growing mechanism. Finally, Figure 10 also indicates that the food and the web domains are the easiest to mimic with these network models, which is also in alignment with Table 5 since the distinguishing ability of the machine learning models was the lowest on these domains.
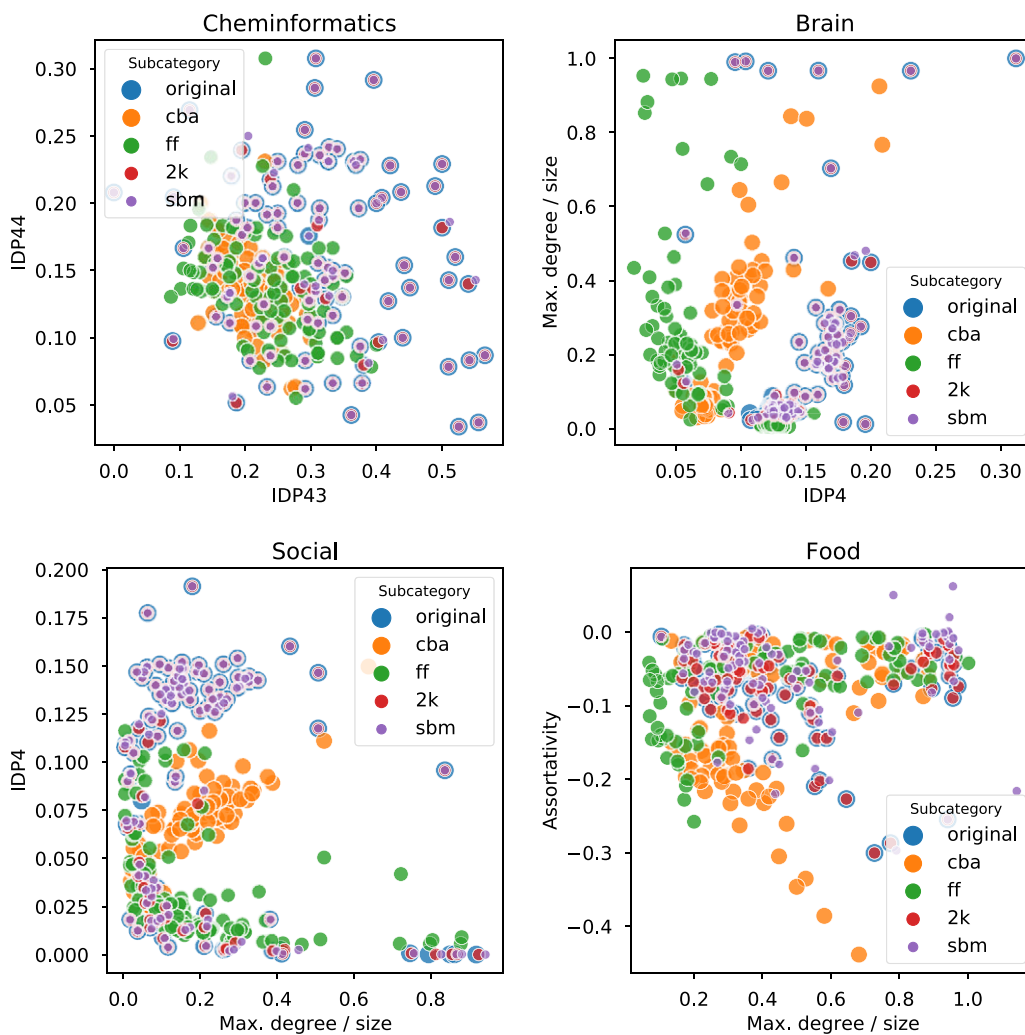
Figure 10. Distribution of the degree distribution-related metrics of the real and the model-generated graphs. The size of the point markers varies only to be able to better see the overlapping points.

6. Conclusion

In this paper, we applied and unified related but so far rather separated branches of data-driven analysis of real and model-generated networks, such as the study of graph metrics, network similarity, network model calibration, and graph classification. First, after a thorough and critical survey of the related state-of-the-art approaches of the literature, we constructed a large dataset to investigate the characteristics and structural properties of real-world networks from six domains and the model-generated counterparts.

By studying the correlation profile of the graph metrics of the real networks, we identified a small, uncorrelated subset of metrics that efficiently describe the real networks. Using the selected nonredundant graph metrics, we fitted each network model to each real network, i.e. we calibrated the models' parameters such that they are as close to the real networks as possible. Note that, we also showed that the distribution of the graph metrics of the random network models is concentrated enough to be able to use them as a base to perform parameter calibration. To be able to
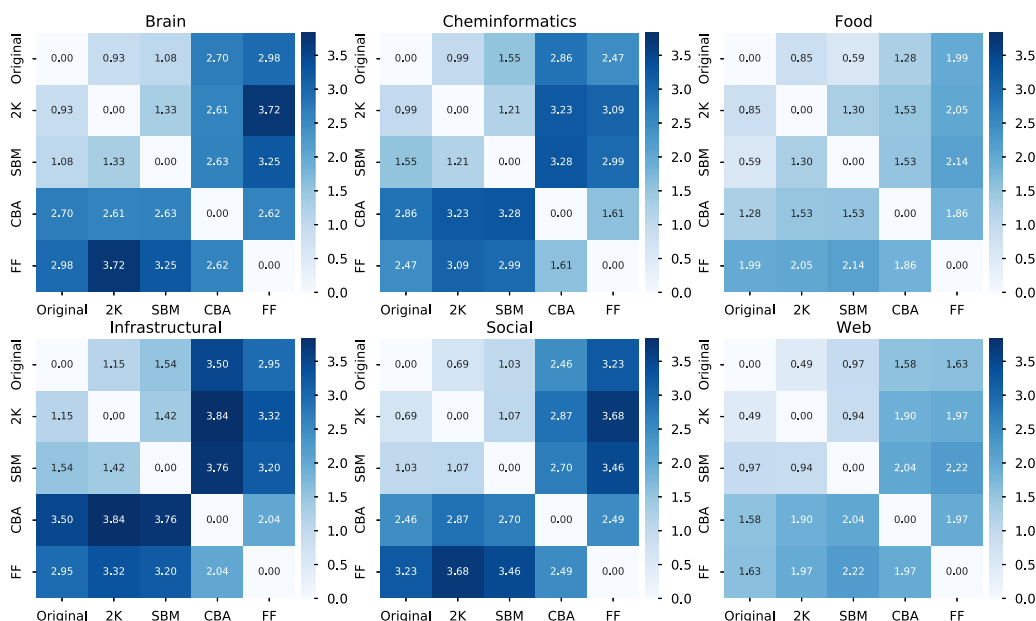
Figure 11. The mean Canberra distance between the real and the corresponding model-generated graphs.

measure the similarity of the graphs, we used the Canberra distance of the vectors of the selected metrics.

With the help of machine learning techniques, we found that the network models are unable to capture the highly clustered structure and relatively large average path length of the real-world networks. On the other hand, the models are able to capture the degree-distribution-related metrics such as the degree centrality, and IDPs. We also found that the networks from different domains are not equally modelable, e.g. the brain and cheminformatics networks are the most difficult and the food and web networks are the easiest to synthesize.

Competing interests. None.

Data availability statement. The datasets analyzed in this study are publicly available at the following GitHub repository: https://github.com/marcessz/Complex-Networks.

References

Aliakbary, S., Habibi, J., & Movaghar, A. (2014). Quantification and comparison of degree distributions in complex networks. In *7th international symposium on telecommunications (IST 2014)* (pp. 464–469). IEEE.

Aliakbary, S., Motallebi, S., Rashidian, S., Habibi, J., & Movaghar, A. (2015a). Distance metric learning for complex networks: Towards size-independent comparison of network structures. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *25*(2), 023111.

Aliakbary, S., Motallebi, S., Rashidian, S., Habibi, J., & Movaghar, A. (2015b). Noise-tolerant model selection and parameter estimation for complex networks. *Physica A: Statistical Mechanics and Its Applications*, *427*, 100–112.

Arnold, N. A., Mondragón, R. J., & Clegg, R. G. (2021). Likelihood-based approach to discriminate mixtures of network models that vary in time. *Scientific Reports*, *11*(1), 1–13.

Attar, N., & Aliakbary, S. (2017). Classification of complex networks based on similarity of topological network features. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *27*(9), 091102.

Bagrow, J. P., Bollt, E. M., Skufca, J. D., & Ben-Avraham, D. (2008). Portraits of complex networks. *EPL (Europhysics Letters)*, *81*(6), 68004.

Bai, Y., Ding, H., Bian, S., Chen, T., Sun, Y., & Wang, W. (2019). Simgnn: A neural network approach to fast graph similarity computation. In *Proceedings of the twelfth ACM international conference on web search and data mining* (pp. 384–392). ACM.

Barabási, A.-L. (2016). *Network science*. Cambridge University Press.

Barabási, A-L., & Albert, R. (1999). Emergence of scaling in random networks. *Science*, *286*(5439), 509–512.

Barnett, I., Malik, N., Kuijjer, M. L., Mucha, P. J., & Onnela, J.-P. (2019). Endnote: Feature-based classification of networks. *Network Science*, *7*(3), 438.

Bezáková, I., Kalai, A., & Santhanam, R. (2006). Graph model selection using maximum likelihood. In *Proceedings of the 23rd International Conference on Machine Learning* (pp. 105–112). ACM.

Bläsius, T., Friedrich, T., Katzmann, M., Krohmer, A., & Striebel, J. (2018). Towards a systematic evaluation of generative network models. In *International workshop on algorithms and models for the web-graph* (pp. 99–114). Springer.

Bonner, S., Brennan, J., Theodoropoulos, G., Kureshi, I., & McGough, A. S. (2016a). Deep topology classification: A new approach for massive graph classification. In *2016 IEEE international conference on big data* (pp. 3290–3297). IEEE.

Bonner, S., Brennan, J., Theodoropoulos, G., Kureshi, I., & McGough, A. S. (2016b). Efficient comparison of massive graphs through the use of 'graph fingerprints'. In *12th international workshop on mining and learning with graphs, KDD 2016*.

Bordino, I., Donato, D., Gionis, A., & Leonardi, S. (2008). Mining large networks with subgraph counting. In *2008 eighth IEEE international conference on data mining* (pp. 737–742). IEEE.

Bounova, G., & de Weck, O. (2012). Overview of metrics and their correlation patterns for multiple-metric topology analysis on heterogeneous graph ensembles. *Physical Review E*, *85*(1), 016117.

Butler, D., & Djordjević, S. (2014). University of Exeter, Centre for Water Systems: Benchmarks.

Canning, J. P., Ingram, E. E., Nowak-Wolff, S., Ortiz, A. M., Ahmed, N. K., Rossi, R. A., Schmitt, K. R. B., & Soundarajan, S. (2018). Predicting graph categories from structural properties. *arxiv preprint arxiv:1805.02682*.

Chatterjee, A., Manohar, M., & Ramadurai, G. (2016). Statistical analysis of bus networks in india. *Plos One*, *11*(12), e0168478.

Chen, D., Shi, D.-D., Qin, M., Xu, S.-M., & Pan, G.-J. (2018). Complex network comparison based on communicability sequence entropy. *Physical Review E*, *98*(1), 012319.

Clauset, A., Tucker, E., & Sainz, M. (2016). The Colorado Index of complex networks.

Croux, C., & Dehon, C. (2008). Robustness versus efficiency for nonparametric correlation measures. *FBE Research Report Kbi_0803*.

Csardi, G., & Nepusz, T. (2006). The igraph software package for complex network research. *International Journal on Complex Systems*, *1695*(5), 1–9.

Del Genio, C. I., Gross, T., & Bassler, K. E. (2011). All scale-free networks are sparse. *Physical Review Letters*, *107*(17), 178701.

Diego, V., Jeremy, G., & Rupesh, N. (2003). Interaction web database.

Dunne, J. A., Maschner, H., Betts, M. W., Huntly, N., Russell, R., Williams, R. J., & Wood, S. A. (2016). The roles and impacts of human hunter-gatherers in north pacific marine food webs. *Scientific Reports*, *6*, 21179.

Faust, K. (2006). Comparing social networks: size, density, and local structure. *Metodoloski zvezki*, *3*(2), 185.

Fay, D., Moore, A. W., Brown, K., Filosi, M., & Jurman, G. (2014). Graph metrics as summary statistics for approximate bayesian computation with application to network model parameter estimation. *Journal of Complex Networks*, *3*(1), 52–83.

Filkov, V., Saul, Z. M., Roy, S., D'Souza, R. M., & Devanbu, P. T. (2009). Modeling and verifying a broad array of network properties. *EPL (Europhysics Letters)*, *86*(2), 28003.

Fong, P. W. L., Anwar, M., & Zhao, Z. (2009). A privacy preservation model for Facebook-style social network systems. In *European symposium on research in computer security* (pp. 303–320). Springer.

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning*. Vol. 1. Springer Series in Statistics. New York, NY, USA: Springer.

Gao, X., Xiao, B., Tao, D., & Li, X. (2010). A survey of graph edit distance. *Pattern Analysis and Applications*, *13*(1), 113–129.

Garcia-Robledo, A., Diaz-Perez, A., & Morales-Luna, G. (2013). Correlation analysis of complex network metrics on the topology of the Internet. In *2013 10th international conference and expo on emerging technologies for a smarter world (cewit)* (pp. 1–6). IEEE.

Gjoka, M., Tillman, B., & Markopoulou, A. (2015). Construction of simple graphs with a target joint degree matrix and beyond. In *2015 IEEE conference on computer communications (INFOCOM)* (pp. 1553–1561). Citeseer.

Goldenberg, A., Zheng, A. X., Fienberg, S. E., *et al.* (2010). A survey of statistical network models. *Foundations and Trends in Machine Learning*, *2*(2), 129–233.

Griffith, V., Xu, Y., & Ratti, C. (2017). Graph theoretic properties of the darkweb. *arxiv preprint arxiv:1704.07525*.

Hagberg, A., Swart, P., & S Chult, D. (2008). *Exploring network structure, dynamics, and function using Networkx*. Tech. rept. Los Alamos National Lab.(LANL), NM, USA.

Hammond, D. K., Gur, Y., & Johnson, C. R. (2013). Graph diffusion distance: A difference measure for weighted graphs based on the graph Laplacian exponential kernel. In *2013 IEEE global conference on signal and information processing* (pp. 419–422). IEEE.

Harrison, K. R. (2014). *Network similarity measures and automatic construction of graph models using genetic programming.* M.Phil. thesis, Brock University.

Holme, P., & Kim, B. J. (2002). Growing scale-free networks with tunable clustering. *Physical Review E*, *65*(2), 026107.

Ikehara, K., & Clauset, A. (2017). Characterizing the structural diversity of complex networks across domains. *arxiv preprint arxiv:1710.11304.*

Jamakovic, A., & Uhlig, S. (2008). On the relationships between topological measures in real-world networks. *Networks and Heterogeneous Media*, *3*(2), 345.

Janssen, J., Hurshman, M., & Kalyaniwalla, N. (2012). Model selection for social networks using graphlets. *Internet Mathematics*, *8*(4), 338–363.

Kang, U., Tong, H., & Sun, J. (2012). Fast random walk graph kernel. In *Proceedings of the 2012 SIAM international conference on data mining* (pp. 828–838). SIAM.

Kashima, H., Tsuda, K., & Inokuchi, A. (2004). Kernels for graphs. *Kernel Methods in Computational Biology*, *39*(1), 101–113.

Kasthuri, N., & Lichtman, J. (2008). Neurodata's graph database.

Kelmans, A. K. (1976). Comparison of graphs by their number of spanning trees. *Discrete Mathematics*, *16*(3), 241–261.

Kiar, G. (2016). *Gremlin: Graph estimation from mr images leading to inference in neuroscience.* Ph.D. thesis, Johns Hopkins University.

Ktena, S. I., Parisot, S., Ferrante, E., Rajchl, M., Lee, M., Glocker, B., & Rueckert, D. (2017). Distance metric learning using graph convolutional networks: Application to functional brain networks. In *International conference on medical image computing and computer-assisted intervention* (pp. 469–477). Springer.

Kunegis, J. (2013). KONECT – The Koblenz network collection. In *Proceedings of international conference on world wide web companion* (pp. 1343–1350).

Langley, P., & Iba, W. (1993). Average-case analysis of a nearest neighbor algorithm. In *International joint conference on artificial intelligence*, vol. 13 (pp. 889–889). Citeseer.

Leskovec, J., Kleinberg, J., & Faloutsos, C. (2005). Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery in data mining* (pp. 177–187). ACM.

Leskovec, J., Kleinberg, J., & Faloutsos, C. (2007). Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *1*(1), 2.

Leskovec, J., Chakrabarti, D., Kleinberg, J., Faloutsos, C., & Ghahramani, Z. (2010). Kronecker graphs: An approach to modeling networks. *Journal of Machine Learning Research*, *11*(Feb), 985–1042.

Li, G., Semerci, M., Yener, B., & Zaki, M. J. (2012). Effective graph classification based on topological and label attributes. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *5*(4), 265–283.

Lim, S.-H., Lee, S. M., Powers, S., Shankar, M., & Imam, N. (2015). Survey of approaches to generate realistic synthetic graphs. *Oak Ridge National Laboratory*.

Mahadevan, P., Krioukov, D., Fall, K., & Vahdat, A. (2006). Systematic topology analysis and generation using degree correlations. In *Acm sigcomm computer communication review*, vol. 36 (pp. 135–146). ACM.

Martínez, Johann H, & Chavez, Mario. (2019). Comparing complex networks: in defence of the simple. *New Journal of Physics*.

Mheich, A., Hassan, M., Khalil, M., Gripon, V., Dufor, O., & Wendling, F. (2018). SimiNet: A novel method for quantifying brain network similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(9), 2238–2249.

Middendorf, M., Ziv, E., & Wiggins, C. H. (2005). Inferring network mechanisms: The drosophila melanogaster protein interaction network. *Proceedings of the National Academy of Sciences*, *102*(9), 3192–3197.

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., & Alon, U. (2002). Network motifs: simple building blocks of complex networks. *Science*, *298*(5594), 824–827.

Nagy, M. (2018). Data-driven analysis of fractality and other characteristics of complex networks. Masters Thesis. *Budapest University of Technology and Economics*.

Nagy, M., & Molontay, R. (2021). *Supplementary Material for Network Classification Based Structural Analysis of Real Networks and their Model-Generated Counterparts. github.com/marcessz/complex-networks*.

Narayanan, A., Chandramohan, M., Venkatesan, R., Chen, L., Liu, Y., & Jaiswal, S. (2017). graph2vec: Learning distributed representations of graphs. *arxiv preprint arxiv:1707.05005.*

Newman, M. E. J. (2003). Properties of highly clustered networks. *Physical Review E*, *68*(2), 026121.

Peixoto, T. P. (2014). The graph-tool python library. *figshare*.

Peixoto, T. P. (2017). Nonparametric bayesian inference of the microcanonical stochastic block model. *Physical Review E*, *95*(1), 012317.

Pržulj, N. (2007). Biological network comparison using graphlet degree distribution. *Bioinformatics*, *23*(2), e177–e183.

Ralaivola, L., Swamidass, S. J, Saigo, H., & Baldi, P. (2005). Graph kernels for chemical informatics. *Neural Networks*, *18*(8), 1093–1110.

Rossi, R. A., & Ahmed, N. K. (2015). The network data repository with interactive graph analytics and visualization. In *Proceedings of the 29th AAAI conference on artificial intelligence.*

Rossi, R. A., Zhou, R., & Ahmed, N. K. (2017). Deep feature learning for graphs. *arxiv preprint arxiv:1704.08829.*

Sala, A., Cao, L., Wilson, C., Zablit, R., Zheng, H., & Zhao, B. Y. (2010). Measurement-calibrated graph models for social network experiments. In *Proceedings of the 19th international conference on world wide web* (pp. 861–870). ACM.

Schieber, T. A., Carpi, L., Díaz-Guilera, A., Pardalos, P. M., Masoller, C., & Ravetti, M. G. (2017). Quantification of network structural dissimilarities. *Nature communications*, *8*, 13928.

Soundarajan, S., Eliassi-Rad, T., & Gallagher, B. (2014). A guide to selecting a network similarity method. In *Proceedings of the 2014 SIAM international conference on data mining* (pp. 1037–1045). SIAM.

Stabler, B., Bar-Gera, H., Sall, E., & Transportation Networks for Research Core Team. (2019). Transportation networks for research.

Sukrit, G., Rami, P., & Konstantin, K. (2016). Comparative network analysis using KronFit. *Complex Networks VII. Studies in Computational Intelligence*, *644*, 363–375.

Sun, X., & Wandelt, S. (2014). Network similarity analysis of air navigation route systems. *Transportation Research Part E: Logistics and Transportation Review*, *70*, 416–434.

Ugander, J., Backstrom, L., & Kleinberg, J. (2013). Subgraph frequencies: Mapping the empirical and extremal geography of large graph collections. In *Proceedings of the 22nd international conference on world wide web* (pp. 1307–1318). ACM.

van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(Nov), 2579–2605.

Vishwanathan, S. V. N., Schraudolph, N. N., Kondor, R., & Borgwardt, K. M. (2010). Graph kernels. *Journal of Machine Learning Research*, *11*(Apr), 1201–1242.

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, *393*(6684), 440.

Wegner, A. E., Ospina-Forero, L., Gaunt, R. E., Deane, C. M., & Reinert, G. (2018). Identifying networks with common organizational principles. *Journal of Complex Networks*, *6*(6), 887–913.

Weinberger, K. Q., & Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, *10*(Feb), 207–244.

Wilson, R. C, & Zhu, P. (2008). A study of graph spectra for comparing graphs and trees. *Pattern Recognition*, *41*(9), 2833–2841.

Yang, L., & Jin, R. (2006). Distance metric learning: A comprehensive survey. *Michigan State Universiy*, *2*(2), 4.