Classification of Complex Networks Using Structural Analysis of Random Graph Models

Conference Paper · February 2021

DOI: 10.1109/SIU49456.2020.9302344

3 authors, including:

Ali Baran Taşdemir
Hacettepe University
7 PUBLICATIONS **0** CITATIONS

SEE PROFILE

Lale Ozkahya
Hacettepe University
28 PUBLICATIONS **212** CITATIONS

SEE PROFILE

Read, See Below for Comments.

H1: How, given an input graph (G) can we categorize it into what type of graph it is?

H2:

So what
↳ Link to Josh Olfi's work on deriving graph parameters.

Category:
① Data-generated
② Model-generated.

Classification of Complex Networks Using Structural Analysis of Random Graph Models

H3: - Use synthetic graphs from each popular random graph distribution
= Train log Reg classifier using Feature set of:
- Max Degree
- Avg Cluster Coefficent
- Avg Eigenvector Centrality.

Ali Baran Taşdemir, Barkın Atasay, Lale Özkahya

Hacettepe University,
Department of Computer Engineering,
06810 Beytepe, Ankara
alibaran@tasdemir.us, barkinatasay96@gmail.com, ozkahya@cs.hacettepe.edu.tr

Abstract

· Presence of several 5 vertex pattern Motifs.

Pre-processing this must be Gross.

Complex networks representing social interactions, brain activities, molecular structures have been studied widely to be able to understand and predict their characteristics as graphs. In this study, various real-world networks have been classified according to random graph models that represent them the best. Synthetic graphs generated by the random graph models are used in order to increase the success rate of the classification. It is observed that using higher order graph features, such as 4-motifs and 5-motifs, yields more accurate results. The most distinctive graph features in the classification process are determined by making use of various machine learning algorithms and statistical tools. With the use of different classifier algorithms, our framework is shown to provide higher accuracy and more robust performance.

1 Introduction

Complex networks representing social interactions, brain activities, molecular structures have been studied widely to be able to understand and predict their characteristics as graphs. While having many similarities, the networks from different domains have varying characteristics, such as the degree-distribution, average clustering coefficient. In this work, we use learning algorithms to obtain a multiclass classification model confirming this claim. Moreover, we gain a deeper understanding of the graph theoretical parameters on this classification by designing a rigorous feature selection framework. Our multiclass classification model is learned using 400 networks from three categories.

Complex networks arise in many domains of real-life applications, such as behavioral networks (Bernard, Killworth, and Sailer 1979), financial networks (Boginski, Butenko, and Pardalos 2005), and citation and dynamic networks (Stix 2004). Thus, the structural analysis of real-life networks such as these has been studied widely. Moreover, it is found useful in computational biology (Abu-Khzam et al. 2005; Eblen et al. 2012; Yeger-Lotem et al. 2004), including the detection of protein-protein interaction complex, clustering protein sequences, searching for common cis-regulatory elements (Baldwin et al. 2004), and others (Bomze et al. 1999). In particular, the local parameters such as vertex degree, the number of triangles containing a vertex, are helpful in predicting the future connections in a network.

Studies on the problem of predicting the domain (category) of arbitrary networks using a small set of graph features give a better understanding of complex networks. In (Bonato et al. 2016; Rossi and Ahmed 2019), real-world networks from various domains are observed to have distinct structural characteristics that are useful in predicting the category of an arbitrary network with high accuracy. These results motivate a more rigorous investigation of the network properties that are more essential in distinguishing the categories of networks by using both real complex networks and synthetic graphs generated by network models.

It is still an open question what an optimal set of graph features is that will give the most accurate classification results with a reasonable computational cost. The study in (Rossi and Ahmed 2019) investigates a classification model using only four features for predicting the category of unknown networks: density, average degree, assortativity, and maximum k-core.

The counting problem of higher order graph structures such as motifs and graphlets have been studied widely (Milo et al. 2002; Ahmed et al. 2015; Pinar, Seshadhri, and Vishal 2017). In our study, we use the frequencies of 4-motifs and 5-motifs in addition to other graph features to obtain more accurate results and a refined analysis for the feature selection problem. We use various classifiers trained on synthetic networks to predict the domain of new networks. We provide a feature selection framework that lets us achieve a classification accuracy with an overall minimum around 95 % for predicting the domain (or network model) of real complex networks.

1.1 Related Work

Previous research has mainly focused on classification of synthetic graphs (Bonner et al. 2016b) or graphs within a particular category/domain such as molecular graphs (Vishwanathan et al. 2010; Ralaivola et al. 2005; Lee, Rossi, and Kong 2017). Other examples on the classification of graphs include the distinction problem between brain or breast cancer cells (Li et al. 2012) or between different social structures (Ugander, Backstrom, and Kleinberg 2013). In general, these studies involve synthetically generated graphs (Bonner et al. 2016a), as they are created and customized easily. The study of graphs from the same domain are also extended to chemical compounds or protein interactions (Guo and Zhu 2013; Li et al. 2012).

In (Bonato et al. 2016), social networks of characters in books and movies are classified according to the type of the random graph model that represents those networks the best. In (Rossi and Ahmed 2019), the problem of predicting the domain (category) of arbitrary networks is investigated by using a small set of graph features. This leads to studying questions related to the classification of complex networks and the network features that play a more significant role in distinguishing the categories of these networks.

In that sense, our work extends earlier studies using the frequencies of higher order graph features, called motifs, providing a thorough performance analysis of different classification algorithms.

1.2 Our Contribution

We make use of various machine learning algorithms to be able to determine the most distinctive graph features in the classification process. By using higher order graph features, such as 4-motifs and 5-motifs, we obtain more accurate results and a refined analysis for the feature selection problem. We investigate different classifier algorithms for our classification framework to provide higher accuracy and a more robust performance.

2 Methodology

2.1 Graph features

Besides the graph size of a graph $G = (V, E)$, denoted by $n := |V|$ and $m := |E|$, we use the following graph features in our study. The graph features listed in Table 1 are density, maximum vertex degree, average degree, maximum k-core, average clustering coefficient, average distance, number of triangles, average eigenvector centrality and the frequencies of the connected motifs on 4 and 5 vertices.

Features

The *degree* of a vertex is the number of edges incident to a vertex. Graph features such as maximum and average degrees, denoted by d_{max} and d_{avg}, respectively, are crucial parameters in graph optimization problems, such as graph coloring and graph matching. The *density*, ρ, of a graph is given by $m/\binom{n}{2}$ and it plays an important role in the design of algorithms for graphs. We use two features related to distances, i.e., the length of the shortest paths between vertex pairs: the diameter and the average distance. The maximum of all distances between vertex pairs is called the *diameter* of a graph G, denoted by $diam(G)$. The *average distance* over all vertex pairs is denoted by $dist_{avg}$ and provides information about the level of connectedness between the vertices.

The clustering coefficient is a good measure of the clustering tendency among vertices (Watts and Strogatz 1998). The *local clustering coefficient* $C(v)$ is defined for any vertex v as $C(v) = T(v)/W(v)$, where $T(v)$ and $W(v)$ denote the number of triangles and the number of wedges (open triangle) containing v at its center, respectively. We also let $C(G) = \sum_{v \in V(G)} C(v)$ and $T(G)$ denote the average clustering coefficient and the number of triangles in G, respectively. The *eigenvector centrality* $\mathbf{v}$ is the eigenvector of the adjacency matrix A of G with the largest eigenvalue λ, i.e., it is the solution of $A\mathbf{v} = \lambda\mathbf{v}$. The ith entry of $\mathbf{v}$ is the *eigencentrality* of vertex v. The higher values of eigencentrality mostly occur at a vertex in a dense subgraph of G. Thus, having a large eigencentrality for a vertex indicates the high connectedness of that vertex to other vertices, possibly with high degrees. As a graph feature, we also consider the average of the eigenvector centrality over the vertex set, denoted by EC_{avg}.

Motifs are small, connected, non-isomorphic subgraphs which appear in a larger graph. We use the frequencies of the motifs listed in Figure 1 and 2 as graph features, that are calculated by the algorithm in (Pinar, Seshadhri, and Vishal 2017). In addition, we use the maximum k-core of a graph as a feature (Sarıyüce, Seshadhri, and Pinar 2017; Liu and Sarıyüce 2019), where a *k-core* of a graph is a maximal subgraph with minimum degree at least k.

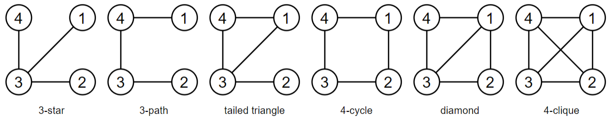

Figure 1: Connected 4-vertex patterns.

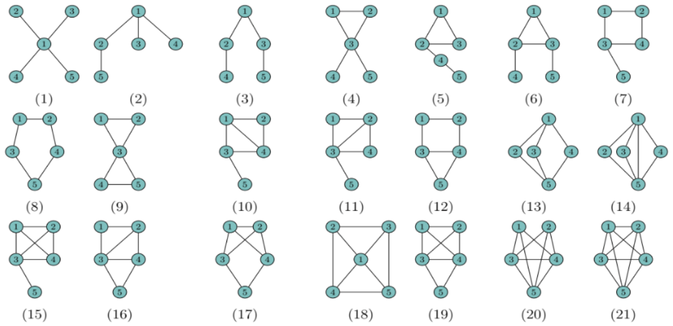

Figure 2: Connnected 5-vertex patterns (Pinar, Seshadhri, and Vishal 2017).

2.2 Synthetic graph models and settings

Preferential Attachment) Model (PA) (Albert and Barabási 2002): This model matches expected scale-free degree distributions. It starts with a connected network of one or more nodes, then adding vertices one by one such that new edges are chosen with probability proportional to the degree of each vertex before the new vertex arrived. Thus, the new vertex has a preference to connect to the vertices with many neighbors. If m is chosen satisfying the following relation

$$\frac{2}{n} + 2m = \frac{2|E|}{n},$$

then the number of edges will match that of the original graph ($|E|$) in expectation.

Table 1: The characteristics of the brain networks (Macaque-rhesus-brain-2, mouse-brain-1), cheminformatic networks (ENZYMES-g300, ENZYMES-g540), and biological networks (celegans-dir, soc-dolphins).

Datasets.

Network (G)	$\lvert V\rvert$	$\lvert E\rvert$	ρ	d_{max}	d_{avg}	Max. k-core	$C(G)$	$Diam(G)$	$dist_{avg}$	$T(G)$	EC_{avg}
macaque-rhesus_brain_2	91	582	0.1421	87	12.79	11	0.86005	3	1.8681	1902	0.0848
mouse_brain_1	213	16089	0.7126	205	151.07	111	0.75826	2	1.2874	622414	0.0677
ENZYMES_g300	49	93	0.0791	7	3.80	3	0.13780	16	5.8061	17	0.0854
ENZYMES_g540	49	92	0.0782	6	3.76	3	0.27551	17	6.1607	23	0.0880
celegans-dir	453	2025	0.0198	237	8.94	10	0.64646	7	2.6638	3284	0.0311
soc-dolphins	62	159	0.0841	12	5.13	4	0.25896	8	3.3570	95	0.0908

Erdős-Rényi Model (ER) (Erdös and Rényi 1959) Each of the $\binom{n}{2}$ edges are added with probability p To match the average degree of the original network, we use $p = |E|/\binom{n}{2}$.

Chung-Lu Model (CL) (Chung and Lu 2002): This model generates graphs with a given expected degree sequence. Each pair of vertices $\{i, j\}$ is connected with probability $p_{ij} = d_i d_j/(2m)$, where d_i denotes the degree of any vertex i.

Configuration Model (CFG): This model makes a uniform random selection from the graphs that matches the given degree distribution exactly. The outcome degree distribution may slightly vary considering that the loops and multi-edges are omitted in the generated graph.

2.3 Classification Framework

We generate samples and train a machine learning algorithm to identify each model to determine the best random graph model fitting the data. We then ask the algorithm to classify the real graph. First, 100 random graphs from each model are used to train a machine learning classifier. In the test step, the classifier predicts a class label for the original real-world network. This provides a measure of which random graph model best fits the character network. We make use of various machine learning algorithms to be able to determine the most distinctive graph features in the classification process. The algorithms we use comprise Support Vector Machines (SVM), logistic regression, decision tree, random forest, AdaBoost, and k-Nearest Neighbor (kNN).

In the training process, the features are normalized. We train a different classifier given by each of the various algorithms listed above. The training is made by using synthetic data by generating 100 graphs using each random graph model, with a total of 400 graphs. The hyperparameters of the training are tuned using 5-fold cross validation. The testing is done on real-world networks. The comparison of the performances of the different learning algorithms are also presented.

We determine the significance of the graph features by making use of the χ^2-test and RFE: recursive feature elimination (Guyon et al. 2002), so that some of the insignificant features are eliminated from the learning process. The performance of these selected set of features are studied by using the logistic regression and random forest classification algorithms. As a result of this procedure, we obtain several reductions from the original feature vector $\mathbf{x_0}$ containing all of the listed features.

3 Experimental Results

The data are obtained from the Network Repository (NR) (Rossi and Ahmed 2015). The classification model is experimented on three main categories: brain, cheminformatic, and biological networks. All networks are undirected, unweighted and connected. Table 1 shows the main characteristics of each network.

Table 2: Model selection F1-scores for random graph models using various vectors.

	$\mathbf{x_1}$	$\mathbf{x_2}$	$\mathbf{x_3}$	$\mathbf{x_0}$	$\mathbf{y}$
CL	0.9686	0.9372	0.9217	0.9930	0.9802
CNFG	0.9849	0.9678	0.9634	0.9833	0.9695
GNP	0.9720	0.9358	0.9239	0.9783	0.9563
PA	0.9986	0.9988	0.9939	0.9976	0.9986
Average	0.9810	0.9599	0.9507	0.9881	0.9762

We study various feature vectors and eliminate the insignificant features through a careful process. We obtain several reductions from the original feature vector $\mathbf{x_0}$ containing all of the listed features. The first reduction yields $\mathbf{x_1}$, which contains all features in $\mathbf{x_0}$ except the motifs numbered 4, 9, 13, 14 in Figure 2. Similarly, $\mathbf{x_2}$ is reduced from $\mathbf{x_1}$ by eliminating the graph density and the motifs numbered 5, 8, 10, 15 in Figure 2. Finally, $\mathbf{x_3}$ is the reduced from $\mathbf{x_2}$ by eliminating the average degree, maximum k-core, average distance, number of triangles, and the motifs numbered 1, 3, 7, 11, 12, 16, 18, 19 in Figure 2. This chain of reductions yields a feature vector, $\mathbf{x_3}$, with only 8 features: the maximum degree, average clustering coefficient, average eigenvector centrality, and the motifs numbered 2, 6, 17, 20 and 21 in Figure 2.

Keep Features to classify Graphs.

We also make a comparison using a feature vector, $\mathbf{y}$, that has all features in $\mathbf{x}$ except the frequencies of the 5-motifs in Figure 2. We observe in the results that the classification accuracy given by the learning algorithms using two different feature vectors, $\mathbf{x_3}$ and $\mathbf{x_0}$, are almost the same. It is observed in Table 2 that the F1-scores given by the learning algorithms have an overall minimum around % 95. The recall and precision values are also observed to have a minimum as % 95 and %96, respectively. Our results show that the training can be done using smaller feature vectors $\mathbf{x_2}$ and $\mathbf{x_3}$, without sacrificing from accuracy.

Table 3: Model selection scores using $\mathbf{x_3}$ as feature vector for the cheminformatic and biological networks.

Network	Classifier	CL	CFG	ER	PA
ENZYMES G-300	SVM L1	0.50	-48.38	-34.35	**8.95**
	SVM L2	1.98	-32.19	-34.69	**13.68**
	Logistic Regression	4.10	-21.32	-107.61	**24.18**
	Adaboost	17.94	-19.46	-21.18	**22.71**
	Random Forest	0.15	0.15	0.18	**0.52**
	kNN	0.0	0.0	0.0	**1.0**
ENZYMES G-540	SVM L1	-13.26	-13.97	-33.80	**7.16**
	SVM L2	-15.73	-17.43	-34.29	**17.53**
	Logistic Regression	-1.66	-22.58	-107.30	**30.52**
	Adaboost	3.31	-7.89	-5.37	**9.95**
	Random Forest	0.0	0.0	0.0	**1.0**
	kNN	0.0	0.0	0.0	**1.0**
soc-dolphins	SVM L1	**21.45**	-9.97	-7.91	-6.64
	SVM L2	**16.75**	-5.10	-5.48	-2.10
	Logistic Regression	**41.59**	-14.27	-16.17	-5.95
	Adaboost	**8.50**	3.24	-7.35	-4.39
	Random Forest	**0.56**	0.26	0.04	0.14
	kNN	**1.0**	0.0	0.0	0.0

The results in Table 3 show model selection scores for the setup described in Section 2, where we train on the entire random graph data, and test on a real-world complex network. The results are presented by choosing different score scales for each chosen classifier. For every classifier, the higher scores indicate more confidently that the original graph does belong to the model. The scores for the first two networks are as expected, since the preferential attachment phenomenon is widely observed on cheminformatic networks (Light, Kraulis, and Elofsson 2005). We similarly observe scores as expected for the third network. The soc-dolphin network represents social interactions of dolphins and the Chung-Lu model is shown to be the best fitting model. This is expected mainly due to the fact that this network has the characteristics of a social network such as the power-law degree distribution, which is captured accurately by the Chung-Lu model.

We analyze the results further by studying possible correlations between the graph features in Figure 3. We measure the pairwise Pearson correlation between each pair of features. The (i, j)th entry in this matrix is a similarity score represented by the Pearson correlation, where 1 is a positive linear correlation, 0 is no correlation, and -1 is negative correlation. We observe the correlations very accurately, such as the direct relation between the average degree and the density of the graph, as well as the maximum k-core and the density. Similarly, negative correlations are observed between expected pairs, such as the average distance and the density.

4 Conclusions

We investigated whether the domain of a complex network can be accurately predicted using mainly the frequencies of higher order graph structures as features. Our results indicate that networks from different domains can be distinguished with high accuracy by using only a few graph features. In particular, the frequencies of some 5-motifs in graphs are shown to be significant in graph classification problems. The results of our work can also be used in many other ways, e.g., to recommend networks that are structurally similar to an unknown network given as input by the user (graph search engine).

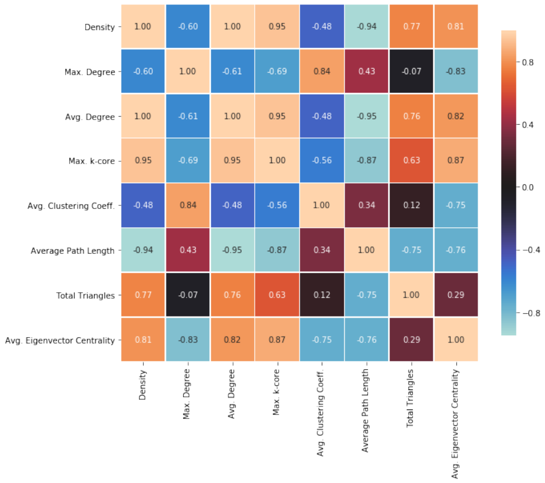

Figure 3: Structural feature correlations for the ENZYMES g300 network. We measure pairwise Pearson correlation between each pair of structural features.

Acknowledgments

The research of the third author was supported by the TUBITAK Grant 11E283.

References

Abu-Khzam, F. N.; Baldwin, N. E.; Langston, M. A.; and Samatova, N. F. 2005. On the relative efficiency of maximal clique enumeration algorithms, with application to high-throughput computational biology .

Ahmed, N. K.; Neville, J.; Rossi, R. A.; and Duffield, N. 2015. Efficient graphlet counting for large networks. In *2015 IEEE International Conference on Data Mining*, 1–10. IEEE.

Albert, R.; and Barabási, A.-L. 2002. Statistical mechanics of complex networks. *Reviews of modern physics* 74(1): 47.

Baldwin, N. E.; Collins, R. L.; Langston, M. A.; Symons, C. T.; Leuze, M. R.; and Voy, B. H. 2004. High performance computational tools for motif discovery. In *18th International Parallel and Distributed Processing Symposium, 2004. Proceedings.*, 192. IEEE.

Bernard, H. R.; Killworth, P. D.; and Sailer, L. 1979. Informant accuracy in social network data IV: A comparison of clique-level structure in behavioral and cognitive network data. *Social Networks* 2(3): 191–218.

Boginski, V.; Butenko, S.; and Pardalos, P. M. 2005. Statistical analysis of financial networks. *Computational statistics & data analysis* 48(2): 431–443.

Bomze, I. M.; Budinich, M.; Pardalos, P. M.; and Pelillo, M. 1999. The maximum clique problem. In *Handbook of combinatorial optimization*, 1–74. Springer.

Bonato, A.; D'Angelo, D. R.; Elenberg, E. R.; Gleich, D. F.; and Hou, Y. 2016. Mining and modeling character networks. In *International workshop on algorithms and models for the web-graph*, 100–114. Springer.

Bonner, S.; Brennan, J.; Theodoropoulos, G.; Kureshi, I.; and McGough, A. S. 2016a. Deep topology classification: A new approach for massive graph classification. In *2016 IEEE International Conference on Big Data (Big Data)*, 3290–3297. IEEE.

Bonner, S.; Brennan, J.; Theodoropoulos, G.; Kureshi, I.; and McGough, A. S. 2016b. GFP-X: A parallel approach to massive graph comparison using spark. In *2016 IEEE International Conference on Big Data (Big Data)*, 3298–3307. IEEE.

Chung, F.; and Lu, L. 2002. Connected components in random graphs with given expected degree sequences. *Annals of combinatorics* 6(2): 125–145.

Eblen, J. D.; Phillips, C. A.; Rogers, G. L.; and Langston, M. A. 2012. The maximum clique enumeration problem: algorithms, applications, and implementations. In *BMC bioinformatics*, volume 13, S5. Springer.

Erdös, P.; and Rényi, A. 1959. On random graphs I. *Publicationes Mathematicae (Debrecen)* 6: 290–297.

Guo, T.; and Zhu, X. 2013. Understanding the roles of subgraph features for graph classification: an empirical study perspective. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 817–822.

Guyon, I.; Weston, J.; Barnhill, S.; and Vapnik, V. 2002. Gene selection for cancer classification using support vector machines. *Machine learning* 46(1-3): 389–422.

Lee, J. B.; Rossi, R.; and Kong, X. 2017. Deep graph attention model. *arXiv preprint arXiv:1709.06075* .

Li, G.; Semerci, M.; Yener, B.; and 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.

Light, S.; Kraulis, P.; and Elofsson, A. 2005. Preferential attachment in the evolution of metabolic networks. *Bmc Genomics* 6(1): 159.

Liu, P.; and Sarıyüce, A. E. 2019. Analysis of Core and Truss Decompositions on Real-World Networks. *MLG workshop at Knowledge Discovery and Data Mining Conference, Anchorage, AK, USA* .

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

Pinar, A.; Seshadhri, C.; and Vishal, V. 2017. Escape: Efficiently counting all 5-vertex subgraphs. In *Proceedings of the 26th International Conference on World Wide Web*, 1431–1440. International World Wide Web Conferences Steering Committee.

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

Rossi, R. A.; and Ahmed, N. K. 2015. The Network Data Repository with Interactive Graph Analytics and Visualization. In *AAAI*. URL http://networkrepository.com.

Rossi, R. A.; and Ahmed, N. K. 2019. Complex networks are structurally distinguishable by domain. *Social Network Analysis and Mining* 9(1): 51.

Sarıyüce, A. E.; Seshadhri, C.; and Pinar, A. 2017. Parallel local algorithms for core, truss, and nucleus decompositions. *arXiv. org e-Print archive, https://arxiv. org/abs/1704.00386* .

Stix, V. 2004. Finding all maximal cliques in dynamic graphs. *Computational Optimization and applications* 27(2): 173–186.

Ugander, J.; Backstrom, L.; and 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*, 1307–1318.

Vishwanathan, S.; Schraudolph, N.; Kondor, R.; and Borgwardt, K. 2010. Graph kernels. The Journal ofMachine Learning Research .

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

Yeger-Lotem, E.; Sattath, S.; Kashtan, N.; Itzkovitz, S.; Milo, R.; Pinter, R. Y.; Alon, U.; and Margalit, H. 2004. Network motifs in integrated cellular networks of transcription–regulation and protein–protein interaction. *Proceedings of the National Academy of Sciences* 101(16): 5934–5939.