Computer and Information Science; Vol. 18, No. 1; 2025
ISSN 1913-8989 E-ISSN 1913-8997
Published by Canadian Center of Science and Education

Betweenness-based Ranking of Edges using the Principal Components of the Complements of Local Clustering Coefficient and Neighborhood Overlap

Natarajan Meghanathan[1]

[1] Department of Computer Science, Jackson State University, USA

Correspondence: Natarajan Meghanathan, Department of Computer Science, Mailbox 18839, Jackson State University, Jackson, MS 39217, USA. Tel: 1-601-979-3661. E-mail: natarajan.meghanathan@jsums.edu

Received: October 20, 2024 Accepted: December 3, 2024 Online Published: December 20, 2024

doi:10.5539/cis.v18n1p1 URL: https://doi.org/10.5539/cis.v18n1p1

Abstract

Edge betweenness centrality (EBWC) is a computationally-heavy metric used to quantify the contribution of edges for communicating on shortest paths between any two vertices in a network. In this paper, we explore the use of metrics such as the local clustering coefficient (LCC) of a node and the neighborhood overlap (NOVER) scores of the edges as the basis to quantify the contribution of edges for communicating on shortest paths. As vertices with lower LCC and edges with lower NOVER are expected to be unused by their neighbors (and hence unused by any other node in the network as well) and vice-versa for communicating on shortest paths, we propose to develop a principal components analysis (PCA)-based composite betweenness scores for the edges (referred to as PCA_EBW) computed on the basis of a dataset that includes the LCC' (1-LCC) values for the end vertices and the NOVER' (1-NOVER) scores for the edges. When applied over a diverse collection of real-world networks, we notice a moderate-strong Spearman's rank-based correlation between the PCA-EBW scores for the edges and their EBWC values.

Keywords: Edge Betweenness, Local Clustering Coefficient, Neighborhood Overlap, Principal Component Analysis, Spearman's Rank-based Correlation, Kendall's Concordance-based Correlation

1. Introduction

Network Science is a domain of Data Science that deals with analyzing complex relationships in real-world-data sets in the abstract form of a graph comprising of nodes and edges. The edges model the complex relationships between the entities (nodes) in the dataset. Centrality metrics (Newman, 2010) quantify and capture the positional importance of the nodes and edges in a network. While various centrality metrics (such as degree centrality: Newman, 2010; eigenvector centrality: Bonacich, 1987; betweenness centrality: Freeman, 1977; closeness centrality: Freeman, 1979;, Page rank: Page et al., 1998, etc) have been proposed to assess the topological importance of nodes, a relatively fewer number of metrics are available in the Network Science literature to quantify the topological importance of edges. Our focus in this paper is on one such widely used edge centrality metric referred to as Edge betweenness centrality (EBWC: Newman & Girvin, 2002). Throughout the paper, the terms 'network' and 'graph', 'node' and 'vertex', 'edge' and 'link', 'score' and 'value' are used interchangeably. They mean the same.

EBWC captures the contribution of edges to communicating on shortest paths between any two nodes in the network. EBWC is computed as the sum of the fractions of the shortest paths between any two vertices that go through the edge. EBWC is a computationally-heavy metric that cannot be locally computed (i.e., the computation algorithm requires global knowledge of the network). The standard algorithm (Newman & Girvin, 2002; Brandes, 2001) used to compute the EBWC of the edges is a Breadth First Search (BFS)-based algorithm (Cormen et al., 2022) and its time complexity is $\Theta(n*(n+m))$ for a graph of *'n'* vertices and *'m'* edges. The BFS algorithm needs to be run on each node in the network twice: first to determine a BFS tree rooted at each node and then to determine the number of shortest paths from the root node to every other node in the network. All the *'m'* edges in the *'n'* BFS trees are traversed *'n'* more times to compute the flow information (that would add up to the EBWC) in the edges along the shortest paths to the root node. Note that, if the graph is dense (i.e., the # edges, $m \sim n^2$), then the computational-complexity of the BFS-based algorithm to compute EBWC would

become $\Theta(n^3)$.

EBWC is a very informative metric that is used by several algorithms in Network Science. For example, the widely known Girvan-Newman (GN) community detection algorithm (Newman & Girvin, 2002) operates on the basis of the EBWC metric: the edge with the largest EBWC value is conjectured to connect two different communities (that would otherwise be not connected without the edge). In a worst case scenario, the GN algorithm needs to be run *'m'* iterations, with each iteration requiring the computation of the EBWC values of the edges remaining in the graph: leading to an overall-time complexity of $\Theta(m*n*(n+m))$, which is $O(n^5)$ for dense graphs. A closer look at the operating procedure of the GN algorithm indicates that each iteration of the algorithm only requires a ranking of the remaining edges in the graph with respect to the EBWC metric and not the actual EBWC values of the edges. This motivated us to explore the possibility of developing a computationally-light metric that could essentially capture the notion of edge betweenness such that the ranking of the edges based on such a metric would exhibit a moderate-strong correlation to the ranking of the edges based on the EBWC metric. Besides the above use, EBWC is typically used to evaluate the load on the links in communication networks and to identify links that are exposed to a cyber security attack (e.g., Mitchell et al (2019)). Recently, Pournajar et al (2022) reported the use of EBWC to diagnose and predict for component failures in structurally disordered materials (modeled as a network of different components and their connections) under the hypothesis that in the absence of failed edges, the most central edges in a network structure should be load free.

The rest of the paper is organized as follows: Section 2 presents the "principal component analysis (PCA)-based feature engineering" approach taken in this paper to develop a computationally-light metric that could be used to rank the edges (with respect to their betweenness in shortest paths) in lieu of the computationally-heavy EBWC metric. Section 3 vindicates our proposition (from a theoretical perspective by determining the shortest path trees and the minimum/maximum spanning trees for a toy example graph) for using the complements of the local clustering coefficient of the nodes and the neighborhood overlap scores of the edges as the basis to develop a computationally-light metric that could be used to rank edges in lieu of EBWC. Section 4 illustrates the working of the proposed PCA-based feature engineering approach on the toy example graph of Section 3 as well as the Spearman's rank-based correlation analysis and Kendall's concordance-based correlation analysis with the EBWC metric. Section 5 shows the results of executing the proposed approach on a collection of 30 real-world networks of diverse degree distributions and domains. Section 6 presents related work and highlights the contribution of our proposed approach to develop a computationally-light edge betweenness metric. Section 7 presents conclusions and outlines plans for future work.

2. Principal Component Analysis-based Feature Engineering

We propose a "principal component analysis (PCA: Jolliffe, 2002)-based feature-engineering (Kuhn & Johnson, 2019)" approach to develop a computationally-light edge betweenness metric. Our hypothesis is: If a node is not used by its neighbors for communicating on shortest paths, then the node is not going to be used by any other node in the network as well for communicating on shortest paths. Likewise, if the neighbors of the end vertices of an edge do not use the edge for communicating on shortest paths, then the edge is not going to be used by any other node in the network as well for communicating on shortest paths. In this pursuit, we researched on identifying node and edge centrality metrics that could capture the contribution of a node or edge just based on local knowledge. We narrowed down to the local clustering coefficient (LCC: Newman, 2010; Meghanathan, 2017a) metric (a node-level metric) and the neighborhood overlap (NOVER: Newman, 2010; Meghanathan, 2016a) metric (an edge-level metric) that could respectively quantify a node's and edge's contribution for communicating on shortest paths.

The LCC (Newman, 2010) of a node is the probability that any two neighbors of the node are connected; the LCC of a node is computed as the ratio of the actual number of links between the neighbors of the node to the maximum possible number of links between the neighbors of the node. The LCC of a stand-alone node or a node with just one neighbor is considered to be 1.0. If the LCC of a node with at least two neighbors is 1.0, it implies that any two neighbors of such a node are directly connected to each other and would not go through the node for communicating on shortest paths. On the other hand, if the LCC of a node is low, it implies any two neighbors of the node are less likely to be directly connected to each other and would more likely to go through the node for communicating on shortest paths. The NOVER score (Newman, 2010) for an edge is computed as the ratio of the intersection of the neighbors of the end vertices of the edge to that of the union of the neighbors of the end vertices of the edge; the union of the neighbors of the end vertices of an edge excludes the two end vertices and every other neighbor is included exactly once in the union set. If an edge has a larger NOVER score, it implies

the neighbors of the end vertices of the edge are less likely to use the edge for communicating on shortest paths, as there is a lot of overlap in the neighborhood, and the neighbors of the two end vertices are directly reachable to each other without going through the edge. On the other hand, if an edge has a lower NOVER score, then there is less overlap in the neighborhood of the end vertices of the edge and the edge could indeed serve as a bridge edge to connect the two neighborhoods; such edges are expected to be part of several shortest paths, especially those between the vertices on either side of the edge.

In order to stay consistent with the convention in the literature that larger the centrality value for a vertex or an edge, the more important is the vertex or the edge: we seek to use LCC' = 1-LCC and NOVER' = 1-NOVER to respectively capture a node's and edge's contribution for communicating on shortest paths based on just the local neighborhood information. The LCC' value of a node could be thus considered as the probability that the node would be used any two of its neighbors for communicating on shortest paths. If the LCC' of a node with at least two neighbors is 1.0 (i.e., LCC = 0.0), it implies the node's neighbors are not directly reachable to one another and would use the node for communicating on shortest paths. Likewise, the NOVER' value for an edge *u*...*v* could be considered as the probability that any two vertices in the neighborhoods of the end vertices *u* and *v* of the edge would go through the edge *u*...*v* for communicating on shortest paths.

We propose to build a *m* x 3 dataset of *m* records (where *'m'* corresponds to the number of edges in the graph), each record corresponds to an edge and the three features are the LCC' values of the two end vertices of the edge and the NOVER' score for the edge. Principal Component Analysis (PCA; Jolliffe, 2002) is a dimensionality reduction technique used in the Data Science community to maximally capture the information in a dataset with multiple features in terms of a relatively fewer number of principal components. When applied to the *m* x 3 dataset of edges, PCA generates '3' principal components (corresponding to the number of features) that would not be correlated to each other; we retain only those *'d'* principal components (*d* expected to be less than 3) whose variance is greater than or equal to 1.0. We refer to the retained principal components as 'dominating' principal components for the rest of the paper. The *'m'* entries in the dominating principal components (especially, those in the dominating principal component with the largest variance) also reflect the magnitude of the three feature values for these *'m'* records in the original dataset (edges in the graph). Those records/edges for which the three feature values were relatively larger are expected to be represented by larger positive entries in the dominating principal component with the largest variance and vice-versa.

For each edge in the dataset, we propose to compute the PCA_EBW (PCA-based Edge betweenness) value as a weighted average of the entries for the edge in the dominating principal components, with the weights being the variances of the dominating principal components. We propose to use the PCA_EBW values of the edges as a measure of the betweenness of the edges for communicating on shortest paths and claim that a ranking of the edges per this metric would exhibit a moderate-strong correlation with the ranking of the edges based on the computationally-heavy EBWC metric. Highly ranked edges with respect to the PCA_EBW are likely to serve as bridge edges (through which information flow happens in social networks) in the network and the removal of these edges could disintegrate the network to two or more communities, a technique that the community detection algorithms could use. In addition, the maintenance of the high PCA_EBW edges could be critical for ensuring the functionality and connectivity of the network. High PCA_EBW edges could become a bottleneck for transportation networks. In biological networks such as protein networks, high PCA_EBW edges could represent protein-protein interactions that would connect different pathways. In epidemiological networks, edges of high PCA_EBW could be the ones through which the epidemics spread more faster; such edges need to be quickly identified in real-time, especially for larger networks.

Note that PCA involves two computationally-intensive steps (Jolliffe, 2002): the covariance matrix computation (Strang, 2023; Golub et al, 1996) step and the Eigenvalue decomposition (Golub et al, 1996) step. On a dataset of *m* records and *p* features, the time complexity of the covariance matrix computation step is $O(p^2m)$ and the time complexity of the Eigenvalue decomposition step is $O(p^3)$. Put together, the overall time complexity of PCA on a dataset of *m* records and *p* features is $O(p^2m + p^3)$. For the problem in hand, *p* (the number of features) is 3 and *m* corresponds to the number of edges. In such a case, the term p^3 and the factor p^2 in the above time complexity expression could be considered constants and much smaller than the number of edges, *m*. Hence, the proposed PCA_EBW metric could be computed in $O(m)$ time: a linear-time complexity with respect to the number of edges, justifying our claim that the proposed PCA_EBW metric is a computationally-light metric compared to the computationally-heavy $O(n^2+nm)$ EBWC metric for a graph of *n* vertices and *m* edges.

3. Motivating Example: Theoretical Perspective

Figure 1 presents a toy example graph along with the LCC' (1-LCC) values for the vertices and the NOVER' (1-NOVER) scores for the edges. We follow the definitions presented in Section 2 to compute the LCC values for the vertices and the NOVER scores for the edges. For example: consider node 1 in Figure 1. It has five neighbors {2, 3, 5, 6, 9} and a maximum of 5*(4)/2 = 10 edges can be expected between any two of these five neighbors. Of these 10 possible edges, only 4 edges (5-9, 3-5, 5-6, 3-6) exist. Hence, the LCC of node 1 is 4/10 = 0.40 and its LCC' value reported in Figure 1 is thus 1-0.40 = 0.60. Likewise, consider edge 4-8 in Figure 1. The neighbors of the end vertices 4 and 8 are {2, 7, 8} and {4, 7, 10} respectively; the intersection of these neighbor sets is {7} and the union of these neighbors sets, excluding the two end vertices 4 and 8 is {2, 7, 10}. Hence the NOVER score for the edge 4-8 is |{7}| / |{2, 7, 10}| = 1/3 = 0.33 and its NOVER' score reported in Figure 1 is 1-0.33 = 0.67.

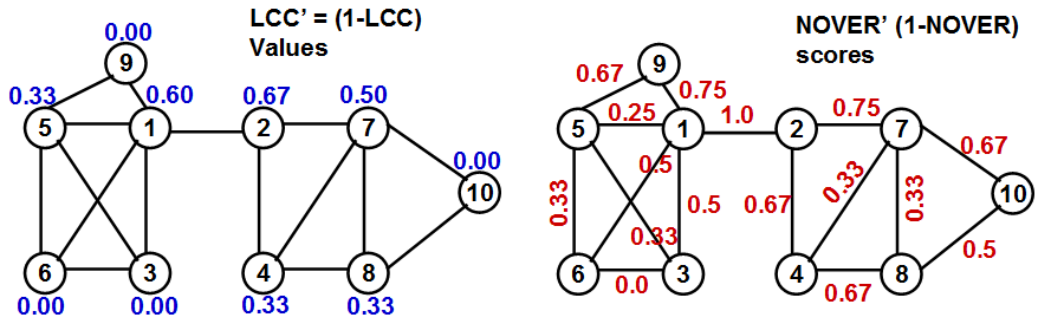
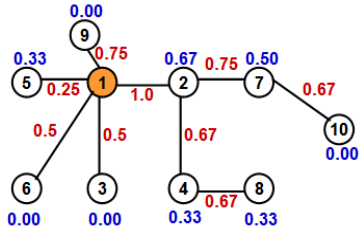
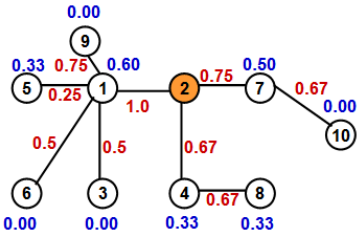


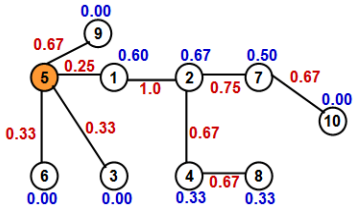
Figure 1. LCC' Values of the Nodes and NOVER' Scores of the Edges for the Toy Example Graph



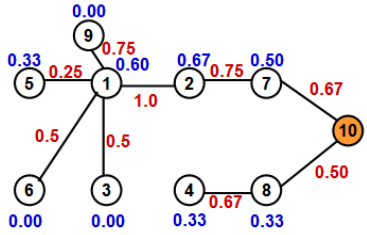
Shortest Path Tree (root: '1')



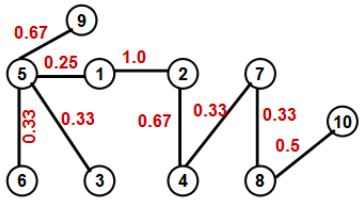
Shortest Path Tree (root: '2')



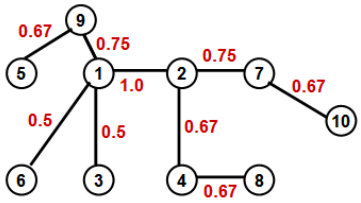
Shortest Path Tree (root: '5')



Shortest Path Tree (root: '10')



Minimum Spanning Tree



Maximum Spanning Tree

Figure 2. Shortest Path Trees (rooted at selected vertices) and the NOVER' Scores-based Minimum and Maximum Spanning Trees for the Toy Example Graph of Figure 1

Figure 2 presents shortest path trees rooted at selected vertices (vertices 1 and 2 at the core/center of the network and vertices 5 and 10 at the edge/periphery of the network) constructed using the classical Breadth First Search (BFS: Cormen et al., 2022) algorithm from Graph Theory, with the ties to pick the predecessor vertices chosen based on the lower id neighbors. Excluding the root nodes, there are a total of 14 intermediate nodes (nodes that have at least one child node further down in the tree) and a total of 22 leaf nodes (nodes that do not have any child node further down in the tree). The LCC' values (colored in blue in Figure 2 and are written next to the vertices) for any intermediate node among the 14 intermediate nodes range from 0.33 to 1.0, with an average of 0.51; whereas, the LCC' values for any leaf node among the 22 leaf nodes range from 0.00 to 0.33, with an

average of 0.11. This corroborates our assertion that the intermediate nodes of the shortest path trees are more likely to comprise of vertices with a larger LCC', and vertices with lower LCC' values are less likely to become intermediate nodes on a shortest path tree.

Figure 2 also shows the minimum and maximum spanning trees (constructed using the classical Kruskal's algorithm: Cormen et. al., 2022) that one could find for the toy example graph of Figure 1 by using the NOVER' scores of the edges as the edge weights. The weights (sum of the edge weights) of the minimum and maximum spanning trees are respectively 4.41 and 6.18. The sum of the edge weights in the four shortest path trees are 5.76 and 5.76 (for the shortest path trees rooted at the center vertices 1 and 2) as well as 5.34 and 5.59 (for the shortest path trees rooted at the peripheral vertices 5 and 10 respectively): all the four weights of the shortest path trees are numerically closer to 6.18 (the weight of a NOVER' scores-based maximum spanning tree, built with the greedy strategy of preferring edges with larger NOVER' scores for inclusion), compared to 4.41 (the weight of a NOVER' scores-based minimum spanning tree, built with the greedy strategy of preferring edges with lower NOVER' scores for inclusion). Also, the average NOVER' score for the edges that connect a leaf node (there are 22 such edges, one edge for each of the 22 leaf nodes, as mentioned above) is 0.55 and the average NOVER' score for the edges that connect two intermediate nodes (there are 14 such edges across the four shortest path trees) is 0.75. All of the above numbers again for the toy example graph again corroborate our assertion that the shortest path trees of a graph are more likely to comprise of edges that have a larger NOVER' score, and moreover, the edges connecting the intermediate nodes are likely to have a larger NOVER' score compared to edges connecting a leaf node with an intermediate node.

4. Execution of the PCA-based Edge Betweenness Computation Approach on the Toy Example Graph

Figure 3 presents the dataset used to conduct PCA. The dataset comprises of the 16 edges as rows (records): though the edges could be considered in any order of the vertices, we follow a convention of writing the lower id end vertex followed by the higher id end vertex, and list the edges in the increasing order of the lower id end vertices. The '3' features for a record/edge *u...v* are the LCC' values of the end vertices *u* and *v* and the NOVER' score for the edge *u...v*. Before conducting PCA, we first standardize the raw dataset using the average and standard deviation of the values corresponding to the '3' features.

Edge #	Edge, u…v	LCC'(u)	LCC'(v)	NOVER'(u…v)
0	1…2	0.60	0.67	1.00
1	1…3	0.60	0.00	0.50
2	1…5	0.60	0.33	0.25
3	1…6	0.60	0.00	0.50
4	1…9	0.60	0.00	0.75
5	2…4	0.67	0.33	0.67
6	2…7	0.67	0.50	0.75
7	3…5	0.00	0.33	0.33
8	3…6	0.00	0.00	0.00
9	4…7	0.33	0.50	0.33
10	4…8	0.33	0.33	0.67
11	5…6	0.33	0.00	0.33
12	5…9	0.33	0.00	0.67
13	7…8	0.50	0.33	0.33
14	7…10	0.50	0.00	0.67
15	8…10	0.33	0.00	0.50

Raw Dataset

Edge #	Edge, u…v	LCC'(u)	LCC'(v)	NOVER'(u…v)
0	1…2	0.7614	1.9971	1.9715
1	1…3	0.7614	-0.8960	-0.0635
2	1…5	0.7614	0.5290	-1.0810
3	1…6	0.7614	-0.8960	-0.0635
4	1…9	0.7614	-0.8960	0.9540
5	2…4	1.0881	0.5290	0.6148
6	2…7	1.0881	1.2630	0.9540
7	3…5	-2.0391	0.5290	-0.7418
8	3…6	-2.0391	-0.8960	-2.0985
9	4…7	-0.4989	1.2630	-0.7418
10	4…8	-0.4989	0.5290	0.6148
11	5…6	-0.4989	-0.8960	-0.7418
12	5…9	-0.4989	-0.8960	0.6148
13	7…8	0.2946	0.5290	-0.7418
14	7…10	0.2946	-0.8960	0.6148
15	8…10	-0.4989	-0.8960	-0.0635

Standardized Dataset

Figure 3. Edge Dataset of the Toy Example Graph of Figure 1 used to Compute the PCA_EBW Values

Figure 4 illustrates the PCA procedure conducted on the standardized dataset (dimensions: 16 x 3) of Figure 3. We first compute the covariance matrix (Strang, 2023) of the 16 x 3 dataset: the covariance matrix (a square matrix: 3 x 3) captures the correlation (Pearson's correlation coefficient: Strang, 2023) between any two of the three features. We then conduct Eigenvalue decomposition (Golub et al., 1996) of the covariance matrix and determine the three Eigenvalues and their corresponding Eigenvectors (EVs). We finally determine the principal components (PCs) corresponding to each of the Eigenvectors by multiplying the standardized dataset with the Eigenvector. We determine the variance of the entries in the three principal components and retain those principal components whose variance is greater than or equal to 1.0. Per this statement, we observe PC-1 is the only dominating principal component and hence there is no need to take a weighted average of the entries in the dominating principal components (with the variances as the weights). We refer to the entries in PC-1 as the PCA-based edge betweenness (PCA_EBW) values of the edges. Note that sum of the PCA_EBW values would

be negative; this is unavoidable as the values of the entries in any principal component have to add to 0. Nevertheless, the edges that report negative PCA_EBW values could be construed as edges that make a relatively weak contribution to shortest path communication compared to edges that report positive PCA_EBW values.

	LCC'(u)	LCC'(v)	NOVER'(u-v)
LCC'(u)	1.0000	0.1911	0.5960
LCC'(v)	0.1911	1.0000	0.2805
NOVER'(u-v)	0.5960	0.2805	1.0000

Covariance Matrix of the '3' Features

Eigenvalues	26.18	12.87	5.94
	Eigenvectors, EV		
	EV-1	EV-2	EV-3
LCC'(u)	-0.6773	-0.3778	0.6313
LCC'(v)	-0.1226	0.9040	0.4095
NOVER'(u-v)	0.7255	-0.2000	0.6586

Eigenvalue Decomposition of the Covariance Matrix

Principal Components and their Variances

Edge #	Edge, u…v	PC-1	PC-2	PC-3
0	1…2	2.5892	1.1059	0.6713
1	1…3	0.0711	-1.0879	-0.4521
2	1…5	-0.0098	0.4158	-1.3664
3	1…6	0.0711	-1.0879	-0.4521
4	1…9	0.7411	-1.2914	0.2860
5	2…4	1.3042	-0.0414	-0.3481
6	2…7	1.8221	0.5410	-0.1903
7	3…5	-1.5630	1.4112	0.7854
8	3…6	-3.0454	0.3821	-0.0223
9	4…7	-0.2815	1.4707	-0.3616
10	4…8	0.3173	0.5493	0.7106
11	5…6	-1.1651	-0.4798	-0.0971
12	5…9	-0.2718	-0.7510	0.8870
13	7…8	-0.0826	0.5252	-0.8028
14	7…10	0.2217	-1.0464	0.3576
15	8…10	-0.7185	-0.6154	0.3950
	Variance	1.7457	0.8581	0.3962

PCA-based Edge Betweenness

Edge #	Edge, u…v	PCA_EBW
0	1…2	2.5892
1	1…3	0.0711
2	1…5	-0.0098
3	1…6	0.0711
4	1…9	0.7411
5	2…4	1.3042
6	2…7	1.8221
7	3…5	-1.5630
8	3…6	-3.0454
9	4…7	-0.2815
10	4…8	0.3173
11	5…6	-1.1651
12	5…9	-0.2718
13	7…8	-0.0826
14	7…10	0.2217
15	8…10	-0.7185

Figure 4. Computation of the PCA-based Betweeness Values for the Edges in the Toy Example Graph

Figure 5 presents the edge betweenness centrality (EBWC) values alongside the PCA_EBW values (shown just in two decimal precision) computed for the edges in the toy example graph. Readers are referred to Meghanathan (2016a) for a sample step-by-step illustration of the BFS-based algorithm (outlined in Section 1 of this paper) to compute the EBWC values for the edges in a graph. Figure 6 presents the computation of the Spearman's rank-based correlation coefficient (that analyzes the similarity in the ranking of the edges; Strang, 2023) between these two metrics. The sum of the squares of the differences in the final rankings of the edges with respect to the two metrics is 29 (see Figure 6 for the calculations). The formula for the Spearman's rank-based correlation coefficient is:

$$1-\frac{6*\sum diff^2}{m(m^2-1)}=1-\frac{6*29}{16(16^2-1)}=0.9574.$$

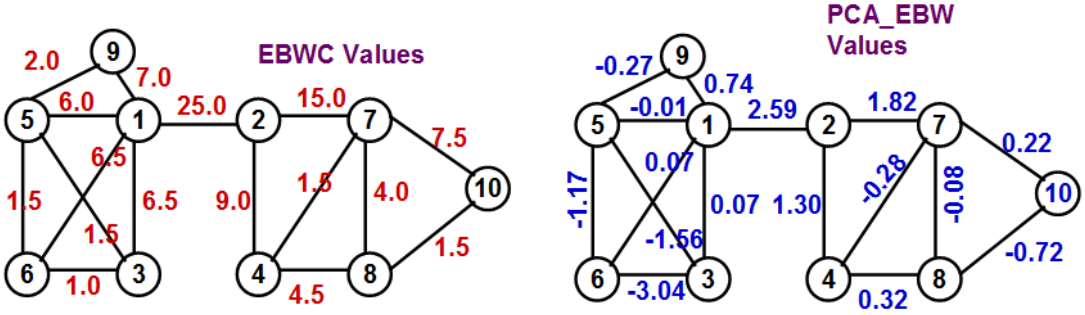


Figure 5. Edge Betweenness Centrality (EBWC) vs. PCA-based Edge Betweenness (PCA_EBW) Values for the Edges in the Toy Example Graph

Edge #	Edge, u…v	EBWC	Tentative EBWC Ranking	Final EBWC Ranking	PCA_EBW	Tentative PCA_EBW Ranking	Final PCA_EBW Ranking	Difference in the Final Ranking, diff	$diff^2$
0	1…2	25	1	1	2.5892	1	1	0	0
1	1…3	6.5	6	6.5	0.0711	7	7.5	-1	1
2	1…5	6	8	8	-0.0098	9	9	-1	1
3	1…6	6.5	7	6.5	0.0711	8	7.5	-1	1
4	1…9	7	5	5	0.7411	4	4	1	1
5	2…4	9	3	3	1.3042	3	3	0	0
6	2…7	15	2	2	1.8221	2	2	0	0
7	3…5	1.5	12	13.5	-1.5630	15	15	-1.5	2.25
8	3…6	1	16	16	-3.0454	16	16	0	0
9	4…7	1.5	13	13.5	-0.2815	12	12	1.5	2.25
10	4…8	4.5	9	9	0.3173	5	5	4	16
11	5…6	1.5	14	13.5	-1.1651	14	14	-0.5	0.25
12	5…9	2	11	11	-0.2718	11	11	0	0
13	7…8	4	10	10	-0.0826	10	10	0	0
14	7…10	7.5	4	4	0.2217	6	6	-2	4
15	8…10	1.5	15	13.5	-0.7185	13	13	0.5	0.25
									Sum = 29

Figure 6. Computation of the Spearman's Rank-based Correlation Coefficient between EBWC and PCA_EBW Values for the Edges in the Toy Example Graph

Figure 7 presents the computation of the Kendall's concordance-based correlation coefficient (Strang, 2023) between the EBWC values and PCA_EBW values for the edges in the toy example graph. We consider every pair of any two edges (say, edge1 and edge2) for the calculation of the correlation coefficient. An edge pair (edge1, edge2) is said to be *concordant* if one of the following criteria are true:

EBWC(edge1) > EBWC(edge2) and PCA_EBW(edge1) > PCA_EBW(edge2) OR

EBWC(edge1) == EBWC(edge2) and PCA_EBWC(edge1) == PCA_EBW(edge2) OR

EBWC(edge1) < EBWC(edge2) and PCA_EBW(edge1) < PCA_EBW(edge2). An edge pair (edge1, edge2) is said to be *discordant* if one of the following criteria are true:

EBWC(edge1) > EBWC(edge2) and PCA_EBW(edge1) ≤ PCA_EBW(edge2) OR

EBWC(edge1) ≥ EBWC(edge2) and PCA_EBW(edge1) < PCA_EBW(edge2) OR

EBWC(edge1) < EBWC(edge2) and PCA_EBW(edge1) ≥ PCA_EBW(edge2) OR

EBWC(edge1) ≤ EBWC(edge2) and PCA_EBW(edge1) > PCA_EBW(edge2).

The Kendall's concordance-based correlation coefficient is calculated as the following fraction: (# concordant pairs - # discordant pairs) / (# concordant pairs + # discordant pairs).

For the toy example graph of Figure 5, we observe a Kendall's concordance-based correlation coefficient of 0.85 on the basis of 111 concordant pairs of edges and 9 pairs of discordant edges. Figure 7 presents the details of the identification of the concordant and discordant pairs. For the same dataset, the Kendall's concordance-based correlation coefficient is typically less than the Spearman's rank-based correlation coefficient (observed by Meghanathan, 2017b). Nevertheless, we observe very high values for both the correlation coefficients, justifying our claim that the proposed PCA_EBW metric could be a computationally-light alternate to rank the edges in a graph in lieu of the computationally-heavy EBWC.

5. Execution of the PCA-based Edge Betweenness Computation Approach on Real-World Networks

We executed the PCA-based Edge Betweenness computation approach on a suite of 30 real-world networks with degree variations (captured by the spectral radius ratio for node degree: λ_{sp}: Meghanathan, 2014) ranging from that of random networks (Erdos & Renyi, 1959) to scale-free networks (Barabasi & Albert, 2002) covering diverse network domains. Figure 8 presents the 30 real-world networks, their number of nodes and edges, the λ_{sp} values, the link density ρ_{link} (calculated by taking the ratio of the actual number of edges in the network to the maximum possible number of edges) and the values for the Spearman's rank-based correlation coefficient between EBWC and PCA_EBW. The maximum possible number of edges used in the link density calculations is $n(n\text{-}1)/2$, where 'n' is the number of nodes in the network.

From Figure 8, the Spearman's rank-based correlation coefficient values observed range from 0.62 to 0.89 and the Kendall's concordance-based correlation coefficient values observed range from 0.54 to 0.79 (both, in a scale

of -1 to 1). Following the standard convention in the literature (e.g., Meghanathan, 2016b), we fix 0.80, 0.60 and 0.40 as cutoffs for very strong, strong and moderate correlations respectively. Accordingly, we observe 12 of the 30 real-world networks to exhibit very strong EBWC vs. PCA_EBW rank-based correlation and the remaining 18 real-world networks to exhibit strong EBWC vs. PCA_EBW rank-based correlation. Overall, all the 30 real-world networks exhibit strong-very strong EBWC vs. PCA_EBW rank-based correlation. On the other hand, as mentioned by Meghanathan (2017b), for any given dataset: the Kendall's concordance-based correlation coefficient values are typically lower than that of the Spearman's rank-based correlation coefficient values. Accordingly, we observe none of the 30 real-world networks to exhibit very strong Kendall's concordance-based correlation. Whereas, we observe 23 of the 30 real-world networks to exhibit strong concordance-based correlation and the remaining 7 of the 30 real-world networks to exhibit moderate concordance-based correlation.

Edge 1	Edge 2	EBWC (Edge 1)	EBWC (Edge 2)	PCA_EBW (Edge 1)	PCA_EBW (Edge 2)	Concordant or Discordant
1…2	1…3	25	6.5	2.5892	0.0711	Concordant
1…2	1…5	25	6	2.5892	-0.0098	Concordant
1…2	1…6	25	6.5	2.5892	0.0711	Concordant
1…2	1…9	25	7	2.5892	0.7411	Concordant
1…2	2…4	25	9	2.5892	1.3042	Concordant
1…2	2…7	25	15	2.5892	1.8221	Concordant
1…2	3…5	25	1.5	2.5892	-1.5630	Concordant
1…2	3…6	25	1	2.5892	-3.0454	Concordant
1…2	4…7	25	1.5	2.5892	-0.2815	Concordant
1…2	4…8	25	4.5	2.5892	0.3173	Concordant
1…2	5…6	25	1.5	2.5892	-1.1651	Concordant
1…2	5…9	25	2	2.5892	-0.2718	Concordant
1…2	7…8	25	4	2.5892	-0.0826	Concordant
1…2	7…10	25	7.5	2.5892	0.2217	Concordant
1…2	8…10	25	1.5	2.5892	-0.7185	Concordant
1…3	1…5	6.5	6	0.0711	-0.0098	Concordant
1…3	1…6	6.5	6.5	0.0711	0.0711	Concordant
1…3	1…9	6.5	7	0.0711	0.7411	Concordant
1…3	2…4	6.5	9	0.0711	1.3042	Concordant
1…3	2…7	6.5	15	0.0711	1.8221	Concordant
1…3	3…5	6.5	1.5	0.0711	-1.5630	Concordant
1…3	3…6	6.5	1	0.0711	-3.0454	Concordant
1…3	4…7	6.5	1.5	0.0711	-0.2815	Concordant
1…3	4…8	6.5	4.5	0.0711	0.3173	Discordant
1…3	5…6	6.5	1.5	0.0711	-1.1651	Concordant
1…3	5…9	6.5	2	0.0711	-0.2718	Concordant
1…3	7…8	6.5	4	0.0711	-0.0826	Concordant
1…3	7…10	6.5	7.5	0.0711	0.2217	Concordant
1…3	8…10	6.5	1.5	0.0711	-0.7185	Concordant
1…5	1…6	6	6.5	-0.0098	0.0711	Concordant

Edge 1	Edge 2	EBWC (Edge 1)	EBWC (Edge 2)	PCA_EBW (Edge 1)	PCA_EBW (Edge 2)	Concordant or Discordant
1…5	1…9	6	7	-0.0098	0.7411	Concordant
1…5	2…4	6	9	-0.0098	1.3042	Concordant
1…5	2…7	6	15	-0.0098	1.8221	Concordant
1…5	3…5	6	1.5	-0.0098	-1.5630	Concordant
1…5	3…6	6	1	-0.0098	-3.0454	Concordant
1…5	4…7	6	1.5	-0.0098	-0.2815	Concordant
1…5	4…8	6	4.5	-0.0098	0.3173	Discordant
1…5	5…6	6	1.5	-0.0098	-1.1651	Concordant
1…5	5…9	6	2	-0.0098	-0.2718	Concordant
1…5	7…8	6	4	-0.0098	-0.0826	Concordant
1…5	7…10	6	7.5	-0.0098	0.2217	Concordant
1…5	8…10	6	1.5	-0.0098	-0.7185	Concordant
1…6	1…9	6.5	7	0.0711	0.7411	Concordant
1…6	2…4	6.5	9	0.0711	1.3042	Concordant
1…6	2…7	6.5	15	0.0711	1.8221	Concordant
1…6	3…5	6.5	1.5	0.0711	-1.5630	Concordant
1…6	3…6	6.5	1	0.0711	-3.0454	Concordant
1…6	4…7	6.5	1.5	0.0711	-0.2815	Concordant
1…6	4…8	6.5	4.5	0.0711	0.3173	Discordant
1…6	5…6	6.5	1.5	0.0711	-1.1651	Concordant
1…6	5…9	6.5	2	0.0711	-0.2718	Concordant
1…6	7…8	6.5	4	0.0711	-0.0826	Concordant
1…6	7…10	6.5	7.5	0.0711	0.2217	Concordant
1…6	8…10	6.5	1.5	0.0711	-0.7185	Concordant
1…9	2…4	7	9	0.7411	1.3042	Concordant
1…9	2…7	7	15	0.7411	1.8221	Concordant
1…9	3…5	7	1.5	0.7411	-1.5630	Concordant
1…9	3…6	7	1	0.7411	-3.0454	Concordant
1…9	4…7	7	1.5	0.7411	-0.2815	Concordant
1…9	4…8	7	4.5	0.7411	0.3173	Concordant

Edge 1	Edge 2	EBWC (Edge 1)	EBWC (Edge 2)	PCA_EBW (Edge 1)	PCA_EBW (Edge 2)	Concordant or Discordant
1…9	5…6	7	1.5	0.7411	-1.1651	Concordant
1…9	5…9	7	2	0.7411	-0.2718	Concordant
1…9	7…8	7	4	0.7411	-0.0826	Concordant
1…9	7…10	7	7.5	0.7411	0.2217	Discordant
1…9	8…10	7	1.5	0.7411	-0.7185	Concordant
2…4	2…7	6	6.5	-0.0098	0.0711	Concordant
2…4	3…5	6	7	-0.0098	0.7411	Concordant
2…4	3…6	6	9	-0.0098	1.3042	Concordant
2…4	4…7	6	15	-0.0098	1.8221	Concordant
2…4	4…8	6	1.5	-0.0098	-1.5630	Concordant
2…4	5…6	6	1	-0.0098	-3.0454	Concordant
2…4	5…9	6	1.5	-0.0098	-0.2815	Concordant
2…4	7…8	6	4.5	-0.0098	0.3173	Discordant
2…4	7…10	6	1.5	-0.0098	-1.1651	Concordant
2…4	8…10	6	2	-0.0098	-0.2718	Concordant
2…7	3…5	6.5	7	0.0711	0.7411	Concordant
2…7	3…6	6.5	9	0.0711	1.3042	Concordant
2…7	4…7	6.5	15	0.0711	1.8221	Concordant
2…7	4…8	6.5	1.5	0.0711	-1.5630	Concordant
2…7	5…6	6.5	1	0.0711	-3.0454	Concordant
2…7	5…9	6.5	1.5	0.0711	-0.2815	Concordant
2…7	7…8	6.5	4.5	0.0711	0.3173	Discordant
2…7	7…10	6.5	1.5	0.0711	-1.1651	Concordant
2…7	8…10	6.5	2	0.0711	-0.2718	Concordant
3…5	3…6	7	9	0.7411	1.3042	Concordant
3…5	4…7	7	15	0.7411	1.8221	Concordant
3…5	4…8	7	1.5	0.7411	-1.5630	Concordant
3…5	5…6	7	1	0.7411	-3.0454	Concordant
3…5	5…9	7	1.5	0.7411	-0.2815	Concordant
3…5	7…8	7	4.5	0.7411	0.3173	Concordant

Edge 1	Edge 2	EBWC (Edge 1)	EBWC (Edge 2)	PCA_EBW (Edge 1)	PCA_EBW (Edge 2)	Concordant or Discordant
3…5	7…10	7	1.5	0.7411	-1.1651	Concordant
3…5	8…10	7	2	0.7411	-0.2718	Concordant
3…6	4…7	9	15	1.3042	1.8221	Concordant
3…6	4…8	9	1.5	1.3042	-1.5630	Concordant
3…6	5…6	9	1	1.3042	-3.0454	Concordant
3…6	5…9	9	1.5	1.3042	-0.2815	Concordant
3…6	7…8	9	4.5	1.3042	0.3173	Concordant
3…6	7…10	9	1.5	1.3042	-1.1651	Concordant
3…6	8…10	9	2	1.3042	-0.2718	Concordant
4…7	4…8	15	1.5	1.8221	-1.5630	Concordant
4…7	5…6	15	1	1.8221	-3.0454	Concordant
4…7	5…9	15	1.5	1.8221	-0.2815	Concordant
4…7	7…8	15	4.5	1.8221	0.3173	Concordant
4…7	7…10	15	1.5	1.8221	-1.1651	Concordant
4…7	8…10	15	2	1.8221	-0.2718	Concordant
4…8	5…6	1.5	1	-1.5630	-3.0454	Concordant
4…8	5…9	1.5	1.5	-1.5630	-0.2815	Discordant
4…8	7…8	1.5	4.5	-1.5630	0.3173	Concordant
4…8	7…10	1.5	1.5	-1.5630	-1.1651	Discordant
4…8	8…10	1.5	2	-1.5630	-0.2718	Concordant
5…6	5…9	1	1.5	-3.0454	-0.2815	Concordant
5…6	7…8	1	4.5	-3.0454	0.3173	Concordant
5…6	7…10	1	1.5	-3.0454	-1.1651	Concordant
5…6	8…10	1	2	-3.0454	-0.2718	Concordant
5…9	7…8	1.5	4.5	-0.2815	0.3173	Concordant
5…9	7…10	1.5	1.5	-0.2815	-1.1651	Discordant
5…9	8…10	1.5	2	-0.2815	-0.2718	Concordant
7…8	7…10	4.5	1.5	0.3173	-1.1651	Concordant
7…8	8…10	4.5	2	0.3173	-0.2718	Concordant
7…10	8…10	1.5	2	-1.1651	-0.2718	Concordant

Figure 7. Identifying the Concordant and Discordant Pairs of Edges to Compute the Kendall's Concordance-based Correlation Coefficient between EBWC and PCA_EBW Values for the Edges in the Toy Example Graph

We opine that there is not much practical utility to assess the "pair-wise" concordance-based correlation between the EBWC and PCA_EBW metrics, as both the metrics are computed synchronously with global knowledge of

the entire network (i.e., we cannot compute the EBWC or PCA_EBW values for any single edge; we have to either compute the values for these two metrics for all the edges or for none of the edges). Hence, the Spearman's rank-based correlation coefficient based assessment of the correlation between these two metrics would be of more practical significance, as we would most of the time need a global ranking of the edges with respect to betweenness (for example, in the case of the Girvan-Newman clustering algorithm: to remove the high betweenness edge), and we observe the PCA_EBW metric to exhibit a strong to very strong correlation to the EBWC metric with respect to global ranking of the edges. Hence, we use the Spearman's rank-based correlation as the measure of correlation being referred to in the rest of this section and paper.

Net #	Network Name	Domain	# Nodes	# Edges	λ_{sp}	ρ_{link}	Spearman's Rank Corr. Coeff.	Kendall's Concordance Corr. Coeff.
1	huckleberryCoappea	Co-appearance	74	301	1.6621	0.1114	0.8913	0.7855
2	WorldTradeNet.txt	Miscellaneous	80	875	1.3774	0.2769	0.8538	0.7391
3	MccartySocialNetJou	Literature	475	625	3.4808	0.0056	0.8496	0.7875
4	FBNet.txt	Social	169	939	2.0738	0.0661	0.8393	0.7230
5	UKFacultyNetwork.tx	Employment	81	577	1.3536	0.1781	0.8339	0.7189
6	lesMisNet.txt	Co-appearance	77	254	1.8198	0.0868	0.8332	0.7260
7	KoreanFamilyPlannin	Social	35	84	1.5252	0.1412	0.8283	0.7592
8	hepatitusCGeneticIn	Biological	105	123	4.1707	0.0225	0.8162	0.6858
9	xenopusGeneticInter	Biological	461	578	7.5333	0.0055	0.8123	0.6802
10	CSDeptAarhusNet.tx	Employment	61	219	2.1200	0.1197	0.8028	0.7182
11	HumanHerpes4Gene	Biological	216	260	6.0706	0.0112	0.8027	0.7566
12	AnnaKarneninaNet.t	Co-appearance	138	493	2.4753	0.0522	0.8024	0.7204
13	copperFieldNetwork.	Co-appearance	87	406	1.8309	0.1085	0.7680	0.7945
14	manufacCompanyEm	Employment	77	2326	1.1205	0.7949	0.7668	0.6489
15	YeastTwoHybridPPIN	Biological	813	843	4.2911	0.0026	0.7662	0.6485
16	TeenageWomenFrien	Friendship	47	122	1.4031	0.1129	0.7573	0.6874
17	facebookNet2.txt	Social	324	2218	1.8109	0.0424	0.7555	0.6033
18	taroExchNet.txt	Social	22	39	1.0579	0.1688	0.7485	0.6201
19	BandJazzNet.txt	Social	198	2742	1.4452	0.1406	0.7341	0.6624
20	SlovenianMagazineN	Literature	124	5972	1.0478	0.7831	0.7273	0.6566
21	geiserCommicNet.txt	Social	165	300	2.5378	0.0222	0.7238	0.6801
22	MacaqueDominancel	Social	62	1167	1.0367	0.6171	0.7130	0.6935
23	MadridTrainBombingI	Friendship	64	486	1.7819	0.2411	0.6993	0.5862
24	WebsterResidenceHa	Friendship	1278	1809	1.2722	0.0022	0.6861	0.6170
25	SanJuanSurNet.txt	Social	75	155	1.2937	0.0559	0.6715	0.5538
26	fraternityCollegeDorn	Friendship	58	967	1.1106	0.5850	0.6585	0.5764
27	footballNet.txt	Miscellaneous	115	613	1.0112	0.0935	0.6581	0.5755
28	FlyingTeamsCadetsN	Friendship	48	170	1.2103	0.1507	0.6416	0.5360
29	MexicanPoliticalNet.	Social	35	117	1.2254	0.1966	0.6412	0.5609
30	USStatesNet.txt	Miscellaneous	49	107	1.2470	0.0910	0.6206	0.5623

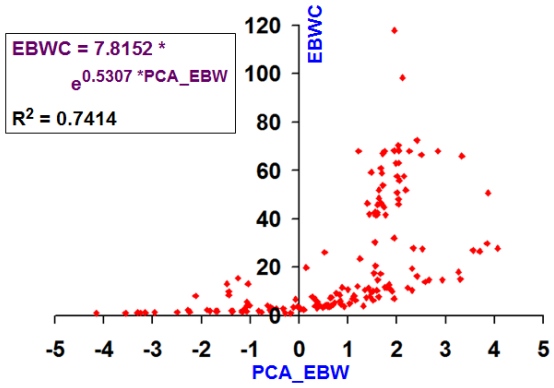
Figure 8. Heat map-based Visualization of the Rank and Concordance-based Correlation Coefficients between EBWC and PCA_EBW for the Real-World Networks and the Impact of the Parameters: Spectral Radius Ratio for Node Degree (λ_{sp}) and Link Density/Fraction of Links (ρ_{link})

The median values for the parameters λ_{sp} and ρ_{link} for the 18 real-world networks that exhibit a strong correlation are 1.2596 and 0.1267 respectively; whereas the median λ_{sp} and ρ_{link} values for the 12 real-world networks that exhibit a very strong correlation are 2.0969 and 0.0765 respectively. This indicates networks with a larger variation of node degree and a lower link density (characteristic of scale-free networks that are peripheral-heavy: comprises of a relatively lower fraction of core nodes of larger degree at the center of the network and a larger fraction of peripheral nodes connected to these core nodes) are more likely to exhibit a very strong correlation between EBWC and PCA_EBW; whereas, networks with a relatively lower variation in node degree and larger link density (characteristic of random networks and scale-free networks that are core-heavy: comprises of a larger fraction of core nodes at the center of the network and a fewer peripheral nodes connected to the core nodes) are more likely to exhibit a strong correlation between EBWC and PCA_EBW.

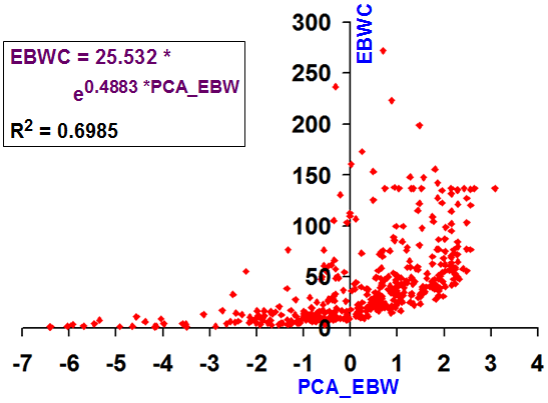
Figure 8 presents a heat map visualization individually for the two parameters λ_{sp} and ρ_{link} as well as for the Spearman's rank-based correlation coefficient and Kendall's concordance-based correlation coefficient. For each of these: the larger the value, the more greener is the cell (indicative of very strong correlation) and the lower the

value, the more reddish is the cell (indicative of moderate correlation); yellow color is for intermediate values (indicative of strong correlation). For the 12 real-world networks exhibiting a very strong rank-based correlation (greener cells), the λ_{sp} cells are predominantly yellow/green (indicative of moderate-larger λ_{sp} values) and the ρ_{link} cells are predominantly red/yellow (indicative of low-moderate ρ_{link} values). On the other hand, for the 18 real-world networks exhibiting a strong rank-based correlation (red-yellow cells), the λ_{sp} cells are predominantly red/yellow (indicative of low-moderate λ_{sp} values) and the ρ_{link} cells are predominantly yellow/green (indicative of moderate-high ρ_{link} values).

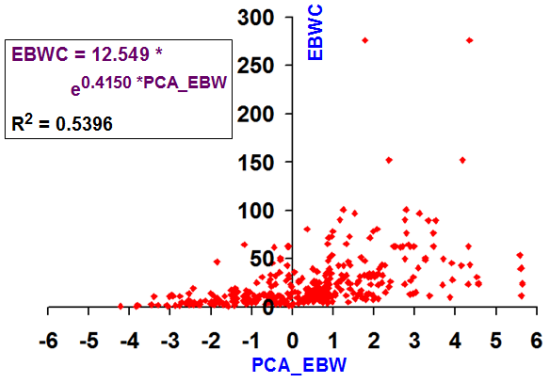
From Figure 8, we could also conclude that the *co-appearance networks* (networks of novels wherein the nodes are the characters and there is an edge between two nodes if the corresponding characters appeared in the same scene or chapter of the novel), *employment networks* (capturing communication relationships between employees within an organization) and *biological networks* (mainly genetic and protein-protein interaction networks) tend to exhibit a very stronger correlation between EBWC and PCA_EBW. On the other hand, social networks and friendship networks are observed to predominantly exhibit a strong correlation between EBWC and PCA_EBW.



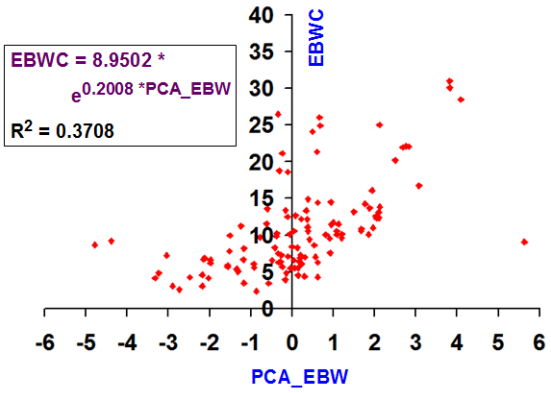
net # 1: huckleberry co-appearance network
(Spearman's rank-based corr. coeff: 0.8913)



net # 12: Anna Karnenina network
(Spearman's rank-based corr. coeff: 0.8024)



net # 23: Madrid train bombing network
(Spearman's rank-based corr. coeff: 0.6993)



net # 29: Mexican political network
(Spearman's rank-based corr. coeff: 0.6412)

Figure 9. Distribution of the PCA_EBW vs. EBWC Values for selected Real-World Networks and the Fitting of an Exponential Function for EBWC Prediction

Figure 9 presents distributions of the PCA_EBW vs. EBWC values for 4 of the 30 real-world networks. The four real-world networks (and their corresponding Spearman's rank-based correlation coefficients) plotted are: net # 1- huckleberry co-appearance network (Knuth, 1993; 0.8913); net # 12 - Anna Karnenina network (Knuth, 1993; 0.8024); net # 23 - Madrid train bombing network (Hayes, 2006; 0.6993) and net # 29 - Mexican political network (Gil-Mendieta & Chmidt, 1996; 0.6412). Net #s 1 and 12 incurred very strong rank correlation and Net #s 23 and 29 incurred strong rank correlation. We also perceive an exponential relation between the two metrics:

i.e., the EBWC metric could be predicted using an exponential value of the PCA_EBW metric as follows: EBWC = $a * e^{b*PCA\_EBW}$, where '*b*' is the coefficient that captures the extent of the exponentiation and '*a*' is simply a proportionality constant; the larger the *'b'* value, the more steeper are the EBWC values as an exponential function of the PCA_EBW values. In Figure 9, we state the exponential functions to which the data points (PCA_EBW vs. EBWC) best fit and the corresponding R^2 values for the four selected real-world networks. We notice the value of the coefficient *'b'* reduces as the rank-based correlation coefficient values decrease (also accompanied by a corresponding decrease in the R^2 values as well). Larger R^2 values for the very strongly rank-correlated networks indicate that the EBWC values for the edges in these networks could be predicted using an exponential function (with the coefficient *'b'* value around 0.50) of the PCA_EBW values obtained using the local knowledge-based LCC' and NOVER' metrics.

Figure 10 presents the computation time (measured in milliseconds) for the EBWC and PCA_EBW metrics for all the 30 real-world networks. The computer on which the code (implemented in Java) for both the metrics are run is a Windows 7 computer with 16 GB memory. We observe the computation time for the PCA_EBW metric to be lower than that of the EBWC metric for all the 30 real-world networks, vindicating our claim that the PCA_EBW metric (even while considering the cost of conducting principal component analysis) would be a computationally-light metric compared to the EBWC metric. We observe the computation time for the PCA_EBW metric to be at least 8% less than that of the EBWC metric and could be as large as 90% less than that of the EBWC metric. As expected, the larger the number of edges, the larger the computation time for both these metrics. With respect to resource usage, the space complexity of PCA (Li et al., 2019) in the context of our m x 3 dataset (where 'm' is the # edges) is O(*m*); whereas, the space complexity to compute the EBWC metric (per the Brandes' algorithm; Brandes, 2001) for a graph of 'n' vertices and 'm' edges is O(*n*+*m*). For larger real-world networks, *n*+*m* >> *m*. Hence, we observe our proposed PCA-based approach to compute a metric that could be used to rank edges in lieu of EBWC is space-efficient as well compared to the approach of directly computing the EBWC metric.

Net #	Computation time (ms) for PC_EBW	Computation time (ms) for EBWC	Net #	Computation time (ms) for PC_EBW	Computation time (ms) for EBWC
1	43.32	72.72	16	1.15	1.33
2	126.35	136.03	17	921.50	991.30
3	135.22	235.87	18	0.80	0.81
4	139.82	148.03	19	522.68	531.24
5	53.69	61.95	20	316.55	365.51
6	18.04	23.79	21	25.64	28.70
7	2.65	4.94	22	31.51	34.80
8	6.05	31.96	23	8.17	8.97
9	120.10	1132.28	24	378.20	383.64
10	13.33	17.42	25	8.66	9.92
11	31.85	122.19	26	24.08	25.08
12	78.06	86.86	27	50.07	55.07
13	37.95	39.93	28	5.82	5.95
14	103.24	107.46	29	4.87	10.63
15	400.03	1303.96	30	3.13	4.08

Figure 10. PCA_EBW vs. EBWC: Computation Time (milliseconds) for the 30 Real-World Networks

6. Related Work

To the best of our knowledge, ours is the first work to focus on developing an edge centrality metric using a feature engineering approach that could serve as an effective alternate for ranking the edges in lieu of the edge betweenness centrality (EBWC). The closest earlier works in the literature (e.g., Meghanathan, 2016a) have largely focused on using some of the fundamental computationally-light edge centrality features such as neighborhood overlap (NOVER) as an alternate for EBWC. The Spearman's rank-based correlation coefficients observed for our PCA_EBW vs. EBWC comparisons are larger than those observed for just the NOVER' vs. EBWC comparisons (Meghanathan, 2016a). Brohl & Lehnertz (2022) propose the notion of nearest neighbor edge centrality (NNEC): a metric designed to primarily capture how center is the location of an edge in the network. The formula to compute the NNEC for an edge *u*...*v* in an undirected network graph is (DEG[*u*] + DEG[*v*] - 2) / (DEG[*u*] - DEG[*v*] + 1), where DEG[*u*] and DEG[*v*] are the degrees of the nodes *u* and *v*. Per

NNEC, an edge is more central if the degrees of both its end vertices are larger and similar. Figure 11 presents the Spearman's rank-based correlation coefficients observed between the EBWC values of the edges vs. the NNEC metric and our proposed PCA_EBW metric for the 30 real-world networks studied in this paper. We observe a weak negative correlation between the NNEC metric and the EBWC metric. Thus, though being computationally-light (as it is just based on the degrees of its end vertices), we show that the NNEC metric cannot be an alternate for EBWC.

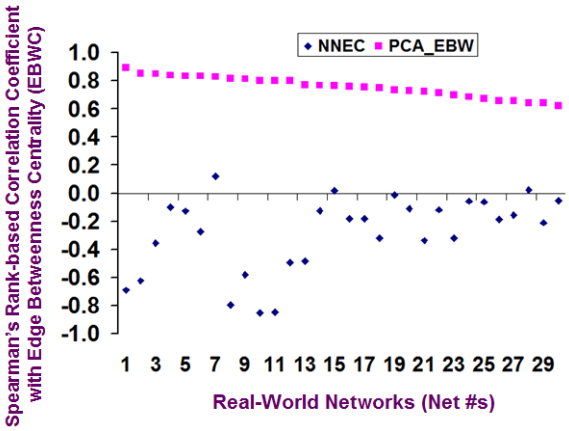


Figure 11. Correlation Analysis of the Nearest Neighbor-based Edge Centrality Metric and the PCA_EBWC Metric vs. EBWC for the 30 Real-World Networks

Mocanu et al (2018) proposed the notion of "Game of Thieves (GoT)" to evaluate the centrality of nodes and edges in a network: Each node is initially assumed to contain a certain number of thieves and diamonds. The computation proceeds for a certain number of epochs. In each epoch, a node picks a random neighbor and sends a thieve (if the former has one) to the chosen neighbor and brings back a diamond (if the latter has one). The centrality of a node is the average number of diamonds the node possessed across all the epochs and the centrality of an edge is the average number of thieves traversing the edge across all the epochs. The GoT approach is parameter-driven and its correlation with EBWC would depend on the distribution function considered for the number of diamonds and the number of thieves vis-a-vis the node degree and the location of the node in the network. On the other hand, our proposed PCA_EBW is not parameter-driven and is not randomized either.

Spanning edge betweenness (SBW: Teixeira et al., 2016) is a notion proposed in the literature to capture the betweenness of edges in graphs; the SBW of an edge is the fraction of the number of spanning trees of the graph that contain the edge to that of the total number of spanning trees of the graph. The number of spanning trees, $\tau(G)$, of a graph G is typically computed by dividing the product of the non-zero Eigenvalues of the Laplacian matrix (Godsil & Royle, 2001) of a connected graph by the number of vertices in the graph. To compute the number of spanning trees of a graph that would contain an edge $u...v$, we adjust the entries in the Laplacian matrix to reflect the removal of the edge $u...v$ and compute the number of spanning trees, $\tau(G\text{-}(u...v))$ of the graph that would not include the edge $u...v$; the difference $\tau(G) - \tau(G\text{-}(u...v))$ would then be the number of spanning trees of the graph that would include the edge $u...v$. If the graph gets disconnected due to the removal of the edge $u...v$, then $\tau(G\text{-}(u...v))$ would be 0, implying the edge $u...v$ would be present in all the spanning trees of the graph and such an edge is called a bridge edge (Cormen et al., 2022; Meghanathan, 2021) in graph theory. However, note that SBW focuses on the presence of an edge in the spanning trees of a graph and not on the shortest paths of a graph. Spanning trees and Shortest paths trees are two different paradigms (Cormen et al., 2022) in Graph theory. Figure 12 presents a plot of the EBWC vs. {SBW, PCA_EBW} values for the toy example graph of Figure 1: Though the edge (1...2) with the largest EBWC also incurs the largest SBW, overall: we do not observe a pattern of increase in the SBW values with increase in the EBWC values. We see that edges with different EBWC values (2.0 to 15.0) incurred SBW values of 0.6 and 0.612. On the other hand, we observe edges with the lowest EBWC values of 1.0 and 1.5 incurred negative PCA_EBW values; whereas, for the rest of the edges we observe positive PCA_EBW values that also increase with increase in the EBWC values. The Spearman's rank-based correlation coefficient between the EBWC vs. SBW values for the edges in the toy example graph is 0.4904, whereas the Spearman's rank-based correlation coefficient between the EBWC vs.

PCA_EBW values is 0.9574. Thus, though the notion of SBW could help to identify the bridge edges of a graph, it could not be used as an alternate for shortest paths-based edge betweenness centrality.

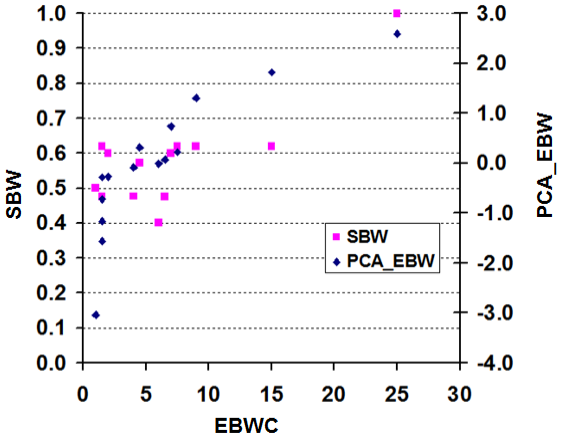


Figure 12. Correlation Analysis of the Spanning Edge Betweenness (SBW) Metric and the PCA_EBWC Metric vs. EBWC for the Toy Example Graph

Yang (2024) proposes two design criteria that could be used for developing effective edge centrality metrics: (1) An edge *u*-*v*, without which, its end vertices *u* and *v* would have a lower degree and would be scarcely connected to the rest of the network needs to be assigned a higher edge centrality score; (2) The centralities of edges incident on the same end vertex should be close to each other. As part of future work, we plan to incorporate these two design criteria as part of the feature set in our underlying dataset for determining PCA_EBW and would evaluate the effectiveness of these criteria in capturing the notion of edge betweenness centrality along with that of the currently used LCC' and NOVER' measures.

Brohl & Lehnertz (2019) proposed an k-core decomposition-style edge centrality-based decomposition technique for complex networks. Conducted iteratively, the edge centrality-based decomposition technique aims to identify webs of edges, with each web comprising of edges whose edge centrality-value is less than or equal to a threshold at the time of the decomposition. The webs of edges are extracted in increasing order of the threshold value for the edge centrality measure considered. The NOVER' measure has not been yet tried with this decomposition technique. With the NOVER' values ranging from 0 to 1, we could try in increments of 0.1 to decompose a complex network whose NOVER' values for the edges are less than or equal to the decomposition threshold. We anticipate the highly central edges to be in the inner web (i.e., extracted for a larger value for the NOVER' threshold), whereas edges connecting the peripheral vertices to the core vertices are expected to be in the outer web of edges.

Kivimaki et. al (2016) evaluated the suitability of the Boltzmann probability distribution-based randomized shortest paths to efficiently evaluate the betweenness centrality of nodes. Matta et. al (2019) compared the speed and accuracy of approaches proposed in the literature to approximate node betweenness centrality, and concluded that most of the algorithms for computing node betweenness centrality are conducive for parallelization and advocated the use of GPU-computing to determine the node betweenness centrality of vertices in larger graphs rather than resorting to approximation algorithms. However, no such works are yet available for edge betweenness centrality, though the existing algorithm seems to be extendible (as mentioned in our plans for Future Work in Section 6).

7. Conclusions and Future Work

The high-level contribution of this paper is the proposal for developing a computationally-light edge betweeness (referred to as PCA_EBW) metric using a principal component analysis (PCA)-based feature engineering approach. Our hypothesis is that vertices and edges that are not part of the shortest paths between the rest of the vertices in the neighborhood are not going to be part of the shortest paths between any two vertices in the network as well. Our hypothesis stems from the proposition that vertices with higher values for the local clustering coefficient (LCC) and edges with higher values for the neighborhood overlap (NOVER) are likely to be not part of shortest paths between any two vertices in the neighborhood. On this basis, we first compile a dataset of the edges whose features are the complements of the local clustering coefficients (LCC' = 1-LCC) of

the end vertices and the complement of the neighborhood overlap (NOVER' = 1-NOVER) scores of the edges and conduct PCA on this dataset. PCA reduced the dimensionality of the dataset as well as quantitatively captured the contribution of the end vertices of the edges as well as the edges themselves in the dominating principal components (those with variance greater than or equal to 1.0). The *'m'* (where *m* is the number of edges) entries in the dominating principal components (especially, those in the dominating principal component with the largest variance) reflect the magnitude of the '3' feature values (based on LCC' and NOVER') for these *'m'* edges/records in the original dataset. The PCA_EBW values for the edges are then computed as the weighted average of the entries for these edges in the resulting dominating principal components.

Through extensive analysis on a suite of 30 real-world networks, we demonstrate that the proposed PCA_EBW metric could serve as an alternate to rank edges in the real-world networks in lieu of the computationally-heavy edge betweenness centrality (EBWC) metric that has been hitherto used as the metric to quantify and rank edges with respect to their participation in shortest paths between any two vertices. We observe the Spearman's rank-based correlation coefficient for PCA_EBW vs. EBWC to be 0.80 or above for 12 of the 30 real-world networks and to be in the range of [0.62, ..., 0.79] for the other 18 real-world networks (i.e., a strong correlation). The Kendall's concordance-based correlation analysis shows 23 of the 30 real-world networks exhibit strong correlation, and the remaining 7 networks exhibit moderate correlation. The 12 real-world networks that exhibited a very strong PCA_EBW vs. EBWC rank-based correlation were observed to be primarily peripheral-heavy scale-free networks (i.e., scale-free networks whose fraction of the peripheral nodes is more than 0.50). On the other hand, the 18 real-world networks that exhibited a strong PCA_EBWC vs. EBWC rank-based correlation were observed to be primarily core-heavy scale-free networks (i.e., scale-free networks whose fraction of the core nodes is more than 0.50) and random networks (networks that exhibit a Poisson-style degree distribution wherein the average degree of the nodes is the mean of the distribution and the degrees of the rest of the nodes are centered around the mean). We also show that for the very strongly rank-correlated networks, the actual EBWC values could be even predicted through an exponential function of the PCA_EBW values. In the Related Work section, we have demonstrated the uniqueness of the proposed PCA_EBW metric (vis-a-vis other edge centrality and edge betweenness-based metrics) through correlation studies on the 30 real-world networks and the toy example graph used in this paper.

We observe that the Breadth First Search (BFS)-based algorithm to determine EBWC could be parallelized by determining the BFS-trees rooted at each vertex in parallel as well as by determining the number of shortest paths from the root as well as the flow information from the rest of the vertices to the root in parallel. Hence, the EBWC computation algorithm could be parallelized as well using high performance computing techniques such as GPU computing and we plan to compare the speed and accuracy of a parallel algorithm for computing EBWC with that of a hybrid algorithm that would compute the NOVER and LCC values for the vertices in parallel and a centralized algorithm that would conduct PCA on the NOVER' and LCC' based edge dataset.

Acknowledgments

Not applicable.

Authors' contributions

Not applicable.

Funding

This work is partly supported through a sub contract received from University of Virginia titled: Global Pervasive Computational Epidemiology, with the National Science Foundation as the primary funding agency. The views and conclusions contained in this paper are those of the authors and do not represent the official policies, either expressed or implied, of the funding agency.

Competing interests

Not applicable/

Informed consent

Obtained.

Ethics approval

The Publication Ethics Committee of the Canadian Center of Science and Education.

The journal's policies adhere to the Core Practices established by the Committee on Publication Ethics (COPE).

Provenance and peer review

Not commissioned; externally double-blind peer reviewed.

Data availability statement

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

Data sharing statement

No additional data are available.

References

Barabasi, A. L., & Albert, R. (2002). Statistical Mechanics of Complex Networks. *Review of Modern Physics*, *74*(47), 47-97. https://doi.org/10.1103/RevModPhys.74.47

Bonacich, P. (1987). Power and Centrality: A Family of Measures. *American Journal of Sociology*, *92*(5), 1170-1182. https://doi.org/10.1086/228631

Brandes, U. (2001). A Faster Algorithm for Betweenness Centrality. *The Journal of Mathematical Sociology*, *25*(2), 163-177. https://doi.org/10.1080/0022250X.2001.9990249

Brohl, T., & Lehnertz, K. (2019). Centrality-based Identification of Important Edges in Complex Networks. *Chaos*, *29*(3), 033115. https://doi.org/10.1063/1.5081098

Brohl, T., & Lehnertz, K. (2022). A Straightforward Edge Centrality Concept Derived from Generalizing Degree and Strength. *Scientific Reports*, *12*(4407), 1-12. https://doi.org/10.1038/s41598-022-08254-5

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to Algorithms*. (4th Ed.) MIT Press.

Erdos, P., & Renyi, A. (1959). On Random Graphs I. *Publicationes Mathematicae*, *6*, 290-297. https://doi.org/10.5486/PMD.1959.6.3-4.12

Freeman, L. (1977). A Set of Measures of Centrality based on Betweenness. *Sociometry*, *40*(1), 35-41. https://doi.org/10.2307/3033543

Freeman, L. (1979). Centrality in Social Networks Conceptual Clarification. *Social Networks*, *1*(3), 215-239. https://doi.org/10.1016/0378-8733(78)90021-7

Gil-Mendieta, J., & Schmidt, S. (1996). The Political Network in Mexico. *Social Networks*, *18*(4), 355-381. https://doi.org/10.1016/0378-8733(95)00281-2

Godsil, C., & Royle, G. F. (2001). *Algebraic Graph Theory*. (1st Ed.) Springer. https://doi.org/10.1007/978-1-4613-0163-9

Golub, G. H., Loan, V., & Charles, F. (1996). *Linear Algebra and its Applications* (6th ed.) Wellesley-Cambridge Press.

Hayes, B. (2006). Connecting the Dots. *American Scientist*, *94*(5), 400-404. https://doi.org/10.1511/2006.61.3495

Jolliffe, I. T. (2002). *Principal Component Analysis* (2nd ed.) Springer.

Kivimaki, I., Lebichot, B., Saramaki, J., & Saerens, M. (2016). Decentralized Dynamic Understanding of Hidden Relations in Complex Networks. *Scientific Reports*, *6*(19968), 1-15. https://doi.org/10.1038/srep19668

Knuth, D. E. (1993). *The Stanford GraphBase: A Platform for Combinatorial Computing*. (1st ed.) Addison-Wesley.

Kuhn, M., & Johnson, K. (2019). *Feature Engineering and Selection: A Practical Approach for Predictive Models*. (1st Ed.) Chapman and Hall/CRC. https://doi.org/10.1201/9781315108230

Li, X., Wang, S., & Cai, Y. (2019). Tutorial: Complexity Analysis of Singular Value Decomposition and its Variants. arXiv:1906.12085v3, [math.NA].

Matta, J., Erical, G., & Sinha, K. (2019). Comparing the Speed and Accuracy of Approaches to Betweenness Centrality Approximation. *Computational Social Networks*, *6*(2), 1-30. https://doi.org/10.1186/s40649-019-0062-5

Meghanathan, N. (2014). *Spectral Radius as a Measure of Variation in Node Degree for Complex Network Graphs*. Paper presented at the 3rd International Conference on Digital Contents and Applications, Hainan, China. https://doi.org/10.1109/UNESST.2014.8

Meghanathan, N. (2016a). A Greedy Algorithm for Neighborhood Overlap-based Community Detection. *Algorithms*, *9*(1), 8: 1-26. https://doi.org/10.3390/a9010008

Meghanathan, N. (2016b). Assortativity Analysis of Real-World Network Graphs based on Centrality Metrics. *Computer and Information Science*, *9*(3), 7-25. https://doi.org/10.5539/cis.v9n3p7

Meghanathan, N. (2017a). A Computationally-Lightweight and Localized Centrality Metric in lieu of Betweenness Centrality for Complex Network Analysis. *Vietnam Journal of Computer Science*, *4*(1), 23-38. https://doi.org/10.1007/s40595-016-0073-1

Meghanathan, N. (2017b). Evaluation of Correlation Measures for Computationally-Light vs. Computationally-Heavy Centrality Metrics on Real-World Graphs. *Journal of Computer and Information Technology*, *25*(2), 103-132. https://doi.org/10.20532/cit.2017.1003492

Meghanathan, N. (2021). Neighborhood-based Bridge Node Centrality Tuple for Complex Network Analysis. *Applied Network Science*, *6*(47), 1-36. https://doi.org/10.1007/s41109-021-00388-1

Mitchell, C., Agrawal, R., & Parker, J. (2019). *The Effectiveness of Edge Centrality Measures for Anomaly Detection*. Paper presented at the 2019 IEEE International Conference on Big Data, Los Angeles, CA, USA. https://doi.org/10.1109/BigData47090.2019.9006468

Newman, M. (2010). *Networks*: *An Introduction*. (1st ed.) Oxford University Press

Newman, M., & Girvan, M. (2002). Community Structure in Social and Biological Networks. *Proceedings of the National Academy of Sciences USA*, *99*(12), 7821-7826. https://doi.org/10.1073/pnas.122653799

Page, L., Brin, S., Motwani, R., & Winograd, T (1998). The PageRank Citation Ranking: Bringing Order to the Web, Technical Report SIDL-WP-1999-0120, Stanford Digital Library Technologies Project. Retrieved from http://ilpubs.stanford.edu:8090/422/

Pournajar, M., Zaiser,M., & Moretti, P. (2022). Edge Betweenness Centrality as a Failure Predictor in Network Models of Structurally Disordered Materials. *Scientific Reports*, *12*(11814), 1-12. https://doi.org/10.1038/s41598-022-15842-y

Strang, G. (2023). *Linear Algebra and its Applications*. (6th Ed.) Wellesley-Cambridge Press.

Teixeira, A. S., Santos, F. C., & Francisco, A. P. (2016). Spanning Edge Betweenness in Practice. *Studies in Computational Intelligence*, *644*, 3-20. https://doi.org/10.1007/978-3-319-30569-1_1

Yang, R. (2024). Effective Edge Centrality via Neighborhood-based Optimization. arXiv:2402.12623 [cs.SI].