Sigma Journal of Engineering and Natural Sciences
Web page info: https://sigma.yildiz.edu.tr
DOI: 10.14744/sigma.2025.00124

Research Article

Kendall rank correlation analysis of Malatya Centrality Algorithm with well-known centrality measures

Selman YAKUT[1,*], Furkan ÖZTEMİZ[1], Ali KARCI[1]

[1]*Department of Software Engineering, Inonu University, Malatya, 44280, Türkiye*

ARTICLE INFO

Article history
Received: 24 July 2024
Revised: 03 October 2024
Accepted: 19 November 2024

Keywords:
Betweenness; Centrality; Closeness; Eigenvector; Kendall Tau; Malatya Centrality; Pagerank

ABSTRACT

The concept of centrality is widely used in graph theory to determine the dominance of nodes within a graph. This concept is crucial for solving many real-life problems that are modeled using graphs. In this study, the effectiveness of a new approach, the Malatya Centrality Algorithm, for determining the centrality of nodes in a graph is examined. This algorithm provides effective solutions to problems in both graph theory and real-life applications. The centrality value in the Malatya Centrality Algorithm is calculated by summing the ratios of the degree of the relevant node to the degrees of its neighboring nodes. To demonstrate the effectiveness of the Malatya Centrality Algorithm, comparisons and analyses were conducted with well-known centrality algorithms in the literature. Various types of graphs, including random graphs, benchmark graphs, social network graphs, and lattice bipartite graphs, were used for these comparisons and analyses. Kendall rank correlation analysis and tests were performed on these different types of graphs for the Malatya Centrality Algorithm and the well-known centrality measures in the literature. The tests conducted on various graphs reveal the ranking of nodes based on their effectiveness. These rankings help identify nodes used in numerous problems. The tests and analyses demonstrate that the Malatya Centrality Algorithm produces results similar to those of established centrality algorithms in the literature and confirms its effectiveness across different types of graphs.

Cite this article as: Yakut S, Öztemiz F, Karcı A. Kendall rank correlation analysis of Malatya Centrality Algorithm with well-known centrality measures. Sigma J Eng Nat Sci 2025;43(4):1339−1354.

INTRODUCTION

Graphs are essential data structures made up of nodes and the connections between them, known as edges. They're commonly used to represent complex relationships in both real-world scenarios and various areas of computer science [1]. Solutions to these problems can be found based on the graph structure. One of the important metrics in graph structures used for solving many problems is centrality. Centrality involves assigning values to the nodes of a graph based on various operations and parameters [2]. In graph analysis, centrality values assigned to nodes are often used to help solve problems related to the structure and dynamics of the graph.

***Corresponding author.**
*E-mail address: selman.yakut@inonu.edu.tr

This paper was recommended for publication in revised form by Editor-in-Chief Ahmet Selim Dalkilic

Published by Yıldız Technical University Press, İstanbul, Turkey

In graph theory, centrality is used to assess how important a node is within the network, which can vary based on the graph's structure and the context in which it's applied [3]. Since graphs consist of nodes and edges, both are utilized to determine the centrality of nodes through various operations performed on them. Centrality plays a crucial role in many fields and problems modeled using graph structures [1]. It is an important metric used to identify critical nodes that provide solutions in various domains and problems, including social networks, biological networks, transportation networks, and communication networks [4-6]. Numerous centrality algorithms have been proposed in the literature to define the centrality of graphs.

Various centrality approaches used in different fields and problems are present in the literature. These centrality algorithms employ different parameters and characteristics to calculate the centrality values of nodes. These approaches are often designed with specific problem types in mind, allowing them to deliver more effective results in those contexts. While some algorithms are suited for directed graphs and tackle challenges unique to them, others work better with weighted graphs and are tailored to the needs of that structure. Certain algorithms offer effective solutions by utilizing parameters such as the distance between nodes or node degrees [7,8]. Consequently, a centrality algorithm that produces effective results for a particular field and problem may not yield similar results in different contexts and problems. Therefore, a centrality algorithm that can be applied to various types of problems and deliver effective solutions in those domains would have significant and widespread impact.

Centrality algorithms in the literature exhibit structural and behavioral differences, but it is possible to compare these measurements to identify influential nodes in a graph [9]. Various metrics are used for comparison, with one of the most commonly employed tests being the Kendall Tau test [10]. This non-parametric correlation test measures the ordinal association between variables and is used to assess the monotonic relationship between them when the variables are of ordinal data type [2]. In some graphs, centrality algorithms in the literature can be compared based on the identified influential nodes and their outcomes [11].

In this study, the effectiveness of the Malatya Centrality Algorithm (MCA), a robust and efficient centrality algorithm, was compared with well-known fundamental centrality measures. The MCA has proven to be effective in tackling a range of complex problems in computer science and graph theory, such as vertex cover [12,13], independent set [13], dominating set [14], text summarization[15], and maximum flow[16]. In this approach, a node's centrality is calculated by summing the ratios between its degree and the degrees of its neighboring nodes. The parameters that determine the centrality of nodes are the node's degree, the number of neighboring nodes, and the degrees of neighboring nodes. The MCA was compared with centrality algorithms used in different fields and problems in the literature. Tests conducted on various types of graphs, including social network graphs, random graphs, benchmark graphs, lattice graphs, and bipartite graphs, show that the MCA produces similar results to these widely accepted algorithms in certain types of graphs. The primary motivation of the study is to determine the extent to which the Malatya Centrality(MC) metric resembles other centrality metrics in different types of graphs. By identifying types of graphs where there is a high degree of similarity, the newly introduced MC is highlighted as an alternative metric to many academic studies and methods developed using other centrality metrics. The results obtained and the effective computational approach demonstrate that the proposed algorithm is an effective centrality algorithm applicable to different problems. To this end, the similarity and outputs of the MCA with the centrality measures in the literature were examined. The most fundamental output of a centrality algorithm is the ranking of the importance of nodes in the graph, which often provides node selection priority in solving graph problems. The MCA was compared with other centrality measures using the Kendall Tau correlation coefficient analysis, and the test results were comprehensively included in the study.

This study is organized into the following sections: Section 2 provides a review of the literature on centrality algorithms. In Section 3, the existing centrality measures in the literature and the Kendall Tau correlation test measure are first discussed. Following that, details of the MCA and the proposed algorithm are presented. Section 4 contains the experimental results and analyses of the proposed algorithm. In the conclusion section, the results of the proposed algorithm are evaluated.

Motivation of the Study

The MCA is a centrality algorithm that provides effective solutions for various graph problems and real-world applications. In this study, MCA was evaluated in comparison with several well-established centrality algorithms from the literature. Experiments conducted on various types of graphs involved ranking the nodes based on their level of influence. The findings demonstrate that MCA yields results comparable to those of other centrality methods, suggesting that it can provide effective solutions in graph-related problems and application domains where these algorithms are typically used.

Moreover, the fact that MCA can provide robust, polynomial-time solutions across various graph types highlights its potential as an effective tool for tackling problems in domains where traditional centrality algorithms are commonly applied. For this purpose, MCA and centrality algorithms in the literature were evaluated in terms of node ranking and outcomes on graphs of different types and classes.

LITERATURE

Graph structures are widely used in modeling and solving various real-life problems. One of the key parameters in solving these problems is the dominance values of the nodes in the graph. Depending on the structure of the problem being modeled and the solution approach, numerous centrality algorithms have been proposed in the literature. These problems and the algorithms used for their solutions occupy a significant place in the literature and vary according to the specific problem. However, this study focuses on well-known fundamental centrality measures used in the literature and the solution approaches for problems where these measures are applied.

Hajarathaiah et al. emphasized the importance of identifying influential nodes to better understand how information spreads within networks. They proposed a novel hybrid centrality measure that integrates both local and global structural features, demonstrating improved performance over existing methods in pinpointing influential nodes in real-world networks [17]. Similarly, Curado et al. explored various centrality measures to assess node importance, with a particular focus on their application to urban networks [9]. Using real-world data from the city of Rome, they introduced a new centrality approach and evaluated its effectiveness against three well-established measures.

In the context of web mining that extracts information from user behavior, W. Xing and A. Ghorbani introduced the Weighted PageRank (WPR) algorithm, which considers the importance of both incoming and outgoing links and distributes ranking scores based on page popularity. Simulation studies have shown that WPR identifies more relevant pages compared to the standard PageRank algorithm, and future research plans to analyze WPR's performance in more detail by using multiple levels of different websites and reference page lists [7].

In this paper, temporal centrality in dynamic complex networks is examined from the perspective of network connectivity, defining two metrics for this purpose: Temporal Degree Centrality and Temporal PageRank Centrality. Evaluations using real-world public transportation network datasets reveal that the results highlight the times of day when the transportation system requires more attention and their relative importance compared to other periods [18].

This paper introduces two new centrality indices, PathRank and Icentr, for ranking graphs connected to metro networks, and explains their application to the metro networks of 34 different cities. PathRank is a generalization of the PageRank algorithm, while Icentr is based on node and edge weights. Both indices determine the importance of nodes by considering the topology of the graphs [6].

Evans and Chen demonstrated a nonlinear relationship between degree and closeness centrality measures, showing that the inverse of closeness is linearly dependent on the logarithm of degree. The results suggest that the measurement of closeness in networks is largely unnecessary and that most networks can be approximately represented by shortest path spanning trees [19].

Jagadishwari and Chakrabarty introduce a new algorithm for link prediction in social networks based on closeness centrality measures. When tested on real-time online social networks, the algorithm demonstrated superior performance compared to baseline methods such as Common Neighbor (CN), Jaccard Coefficient (JC), and Adamic-Adar (AA) [5].

Chen and Dietrich highlight how the positioning of drainage areas within urban networks significantly influences closeness centrality values—a phenomenon they refer to as the 'placement effect' [20]. Their study suggests that employing an idealized network model can help minimize this effect. They also observe that the placement effect is most pronounced within a 100-meter range, beyond which it becomes necessary to define separate drainage areas to ensure meaningful comparisons.

Kotlarz uses percolation theory to examine the modularity and resilience of brain networks by removing nodes with the highest betweenness centrality, thereby identifying the critical points and submodules of the network [21]. The results indicate that brain functions operate at a critical point and that this approach offers a new framework for understanding the computational efficiency of brain networks.

Kintali presents a randomized parallel algorithm and an algebraic method for computing betweenness centrality, which addresses one of the fundamental problems in large-scale network analysis: determining the importance of a specific node. It is also proven that no path comparison-based algorithm can compute betweenness centrality in less than O(nm) time [22]. This paper introduces a variant of betweenness centrality based on the concept of current flow and adapts this measure for applicability in urban networks. The proposed centrality, by considering urban network topology and data associated with nodes, allows for the theoretical prediction of the most frequently used paths and demonstrates how pedestrian flows are influenced by different types of services in the city[4].

Chiranjeevi et al. introduce a novel centrality metric—Isolation-Betweenness Centrality (ISBC)—to assess node influence in complex networks [23]. This measure accounts for both local and global structural characteristics when evaluating a node's impact. Experiments conducted on real-world datasets indicate that ISBC achieves greater propagation efficiency and maintains a reasonable computational complexity compared to both traditional and recently proposed centrality metrics.

Agryzkov et al. [24] tackle the challenge of identifying key activity zones within urban infrastructure by introducing a new centrality measure inspired by the concept of eigenvector centrality. This measure determines the topological importance of nodes in the urban street network by considering the network's topology and the geographic data associated with the nodes. This paper extends existing paired comparison models for predicting tennis matches by using network indicators and proposes a method based on eigenvector centrality [25]. With each new match, this method updates players' ratings, resulting in better prediction accuracy than other models and producing favorable outcomes in betting scenarios.

Anastasiei et al. examine how network density and centrality influence individuals' tendencies to share negative experiences. Their findings suggest that people with high centrality and strong connections are more inclined to spread negative feedback. However, in highly dense networks, prevailing social norms may suppress such behavior [26]. The study underscores the importance for companies to closely monitor and address negative evaluations from well-connected individuals to minimize reputational harm.

Yuliansyah et al. [27] propose a novel approach called Degree Gravity of Link Prediction (DGLP) to mitigate the cold start problem in network analysis. By incorporating degree centrality, common neighbors, and node distance, DGLP significantly enhances link prediction accuracy. Evaluation results indicate a 7.15% improvement in AUC values and a remarkable 99.94% success rate in predicting links between node pairs affected by the cold start issue. Dal Col and Petronetto introduce Graph Layout Centrality, a new centrality measure that considers both local and global node positions [27,28]. When compared with classical centrality metrics and Graph Fourier Transform Centrality, their method proves to be highly scalable for large-scale graphs and robust against disruptions from induced subgraphs. Another study focuses on the ev-degree and ve-degree topological indices of the Sierpiński gasket fractal. It highlights the mathematical significance of these indices and points to the potential of extending this line of research to similar fractal structures [29].

Shang et al. [30] develop a novel technique for detecting influential nodes using the Effective Distance Gravity model. Unlike traditional methods relying on static Euclidean distance, proposed model leverages effective distance to capture dynamic information flow between nodes. Experimental results confirm the model's effectiveness across various scenarios, particularly in dynamic information dissemination contexts. Qi et al. [31] develop a new method to measure the centrality of factors in weighted networks. The method particularly excels in situations requiring the management of limited resources, such as counter-terrorism programs, by evaluating the contribution of nodes to the system. It outperforms other centrality methods and provides a suitable analytical tool for large and sparse social network datasets.

Sarı examines the topological spaces derived from simple, undirected graphs without isolated vertices, addressing the conditions for a point to be an accumulation or interior point [32]. Additionally, by defining relative topology on a subgraph, it is shown that this topology differs from the topology derived from the subgraph itself, and conditions for classifying the space as *T*0, *T1*, or Hausdorff are presented. In their proposed study, Jahanbani and Ediz computed the sigma index of graphs using operations such as Cartesian product, composition, join, and disjunction, and applied their findings to the sigma indices of specific graph classes [33].

Reviews in the literature indicate that each developed centrality method is proposed for solving specific types of problems and has been shown to provide effective solutions for these problem types. Therefore, there is a need for general centrality measures that can offer effective solutions to various problem types. The MCA has been observed to produce effective solutions in many graph problems and real-life scenarios. Compared to other centrality methods, it has achieved similar ranking results. The similarity of this centrality algorithm's results to those of established measures highlights its strong potential for application across a range of problems and domains.

PRELIMINARIES

Graph theory offers a wide range of centrality measures, each serving distinct purposes across various domains [1]. This paper reviews the most fundamental of these measures to provide a clearer understanding of centrality and its practical relevance. Analyzing these measures helps highlight both the similarities and differences with the MCA, while also offering insights into where the MCA can be effectively applied. In particular, examining these well-established approaches is crucial for comparing and interpreting the results produced.

PageRank Centrality Measure

PageRank is a centrality metric introduced by Larry Page and Sergey Brin in 1998 to assess the importance and relevance of web pages based on how they're linked to one another [34]. The algorithm evaluates the influence of each page within a network by analyzing incoming links and distributing scores accordingly. Each page's score is calculated through an iterative process, where importance flows from one page to another, depending on both the quantity and quality of the links. In essence, it's similar to simulating a user randomly clicking through links on the web. A page gains significance when it's linked by other important pages, allowing search engines to rank results with greater accuracy and quality [34].

$$PR(A) = (1 - d) + d\left(\frac{PR(t1)}{C(t1)} + .. + \frac{PR(tn)}{C(tn)}\right) \quad (1)$$

In Equation 1, t1… tn: represent the nodes linking to the node being sought, PR(tn): represents the values of the nodes linking to the relevant node, C(tn): represents the number of links each node gives to other nodes, and d: is the damping factor used to prevent the sum of all ratios from exceeding 1.

Closeness Centrality Measure

The closeness centrality measure is an algorithm that evaluates a node's accessibility in a network by calculating its distances to all other nodes [31]. This measure is determined by taking the inverse of the average distance from a node to all other nodes. It reflects the node's accessibility and centrality within the network. Nodes with high scores are closer to all other nodes and are positioned at the center of the network. This algorithm determines each node's closeness centrality value by computing the shortest paths between all pairs of nodes and inverting this value. Thus, the higher a node's closeness score, the shorter its average distance to other nodes [35].

$$\text{distance}\,[u] = \sum_{\substack{u \epsilon D \\ d_{Ç}(u,v) \neq \infty}} d_{Ç}(u,v) \tag{2}$$

In Equation 2, for the graph G=(D, K), with u ϵ D represents the total distance of the relevant node to other nodes. Equation 3 provides the calculation for cc[u], which determines the Closeness centrality value of the the node.

$$cc[u]\frac{1}{mesafe[u]} \tag{3}$$

Betweenness Centrality Measure

Betweenness centrality measures how often a node acts as a bridge along the shortest path between two other nodes and was introduced by Linton Freeman to identify individuals controlling communication in social networks [36]. Calculating betweenness centrality for all nodes in a graph typically requires computing the shortest paths, which can be done in (O(V^3)) time using the Floyd–Warshall algorithm or in (O(|V||E| + |V|^2 log |V|)) time for sparse graphs using Johnson's algorithm.

$$CB(u) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{4}$$

Eigenvector Centrality Measure

Eigenvector centrality, introduced by Phillip Bonacich in 1987, is a method used to determine the importance of a node based on the influence of the nodes it's connected to [37]. The key idea behind this measure is that not all connections are equal—links to highly influential nodes contribute more to a node's score than links to less important ones. In essence, a node becomes more central if it is connected to other well-connected nodes. This creates a feedback loop where high-scoring nodes tend to be linked with other high-scoring ones, highlighting their overall prominence in the network.

Equations 5 and 6 define the mathematical foundation of this measure, where A stands for the graph's adjacency matrix and λ represents its largest eigenvalue.

$$Ax = \lambda x, \qquad \lambda x_i = \sum_{j=1}^{n} a_{ij} x_{j,} \quad i = 1, .., n \tag{5}$$

$$c(\beta) = \sum_{k=1}^{\infty} \beta^{k-1} A^k 1, \qquad |\beta| < 1/\lambda \tag{6}$$

MATERIALS AND METHODS

The MCA is a newly proposed centrality algorithm designed to calculate node centrality values within a graph. It provides a robust and effective approach to addressing various challenges in graph theory and its practical applications. In this study, MCA was evaluated against several well-established and widely-used centrality algorithms, with a focus on ranking nodes from most to least influential across different graph types and structures. To assess how well these rankings align, the Kendall rank correlation test—a widely recognized method in the literature—was used. For further statistical comparison between MCA and the other algorithms, the Wilcoxon rank-sum test and Welch's two-sample t-test were also employed.

The results and analyses from these tests are presented to demonstrate the performance of MCA in comparison to other centrality measures. As depicted in Figure 1, the overall methodology follows a four-stage process. In Stage 1, sample datasets from various domains were converted into graph structures, consisting of nodes and edges, to be compatible with centrality algorithms. Stage 2 involved applying MCA alongside PageRank, Closeness, Betweenness, and Eigenvector centrality algorithms to these sample graphs. In Stage 3, nodes in each graph were ranked from highest to lowest based on the centrality values calculated by each method. Finally, in Stage 4, Kendall rank correlation coefficients and additional metrics were computed to evaluate the consistency and significance of the node rankings across different algorithms.

The study aims to determine the relationship between the dominance ranking produced by MCA and that generated by other popular centrality algorithms.

Malatya Centrality Algorithm and Centrality Measure

The pseudo code for the MCA is shown in Algorithm 1. This algorithm takes an unweighted, undirected graph as input and computes a centrality value for each node in the graph. The centrality of each node is calculated sequentially. First, a list of all the nodes in the graph is created. Then, for each node, a loop iterates over its neighbors. During this loop, the centrality value is calculated by summing the ratio of the node's degree to the degree of each of its neighbors. This process is repeated for all nodes in the graph, and by the end of the algorithm, the centrality values for every node are determined.

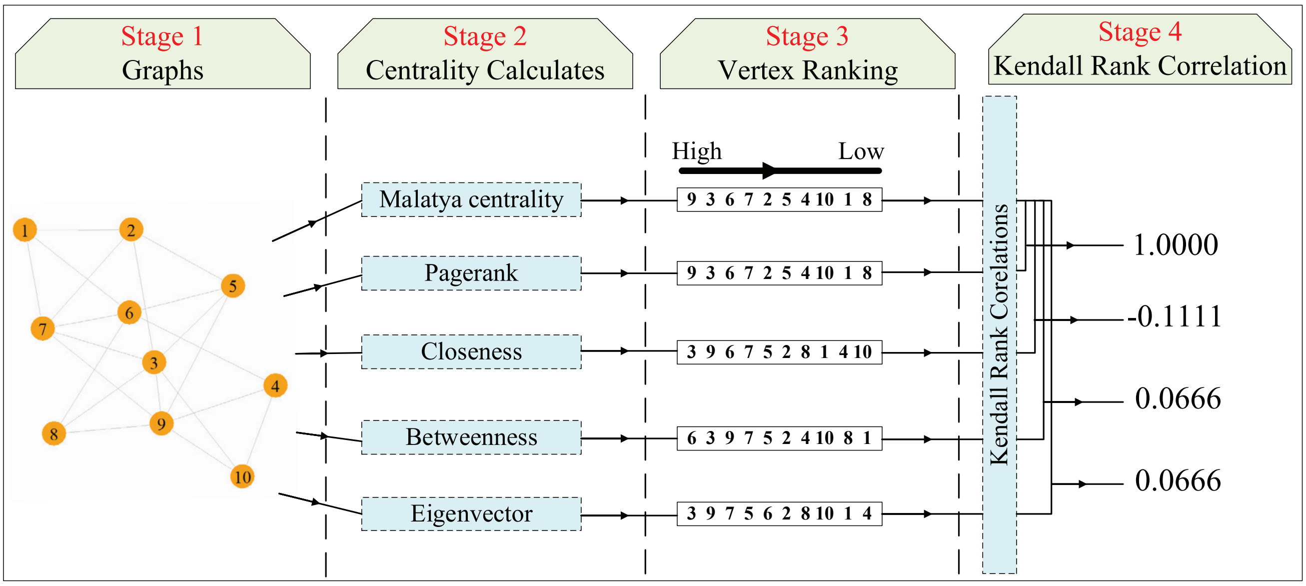

Figure 1. Graphical abstract of the study.

The symbols used in Algorithm 1 and Equation 1 are as follows: G: unweighted and undirected graph, V: the nodes, E: the edges in the graph, |V|: the number of nodes in the graph and $\Psi(v_i)$: the MC value of node v_i, $d(v_i)$: the degree of node v_i, $d(v_j)$: the degree of node v_j.

Algorithm 1. Pseuducode of MCA

Algorithm 1. Malatya Centrality Algorithm(A,**Ψ**)

Input: Adjacency matrix of G is A and G = (V, E)

Output:**Ψ**

1. i←1, …, |V|
2. $\boldsymbol{\Psi}(v_i) = \sum_{\forall v_j \in N(v_i)} \frac{d(v_i)}{d(v_j)}$
3. Output=**Ψ** (…)

The structure of the MCA formula is presented in Equation 7. As observed in this equation, the centrality value for each node in the graph can be expressed as the sum of the ratios of the degree of the node itself to the degrees of its neighboring nodes. Thus, the degrees of all neighboring nodes contribute to determining a node's centrality value. Consequently, the centrality of a node within the graph depends on both the number of neighboring nodes and their respective degrees. Using this equation, which forms the foundation of the MCA, the resulting value for each node can be considered its MC measure.

$$\Psi(v_i) = \sum_{\forall v_j \in N(v_i)} \frac{d(v_i)}{d(v_j)} \quad (7)$$

In Figure 2, the MCA is explained in a practical manner on the example graph from Figure 1. The MC values of the graph depicted in Figure 2 are also illustrated on the graph itself. Specifically, the calculation of the MC value for node (Vertex) 1 is visualized. The blue arrows represent the edge relations forming the degree of node 1, while the green arrows indicate the edge relations forming the degrees of its neighboring vertices. When calculating the MC value for node 1, its degree is proportionally compared to the degrees of its neighboring vertices, and these ratios are summed. The neighbors of node 1 are vertices 2, 6, and 7, with respective degrees of 4, 5, and 5. Thus, the MC value for node 1 is calculated as 3/4 + 3/5 + 3/5 = 1.95, The MC values of the other vertices are computed in the same manner. The MC values for the sample graph are shown in Figure 2 as indicated.

Given that MCA is applied to various graph problems and application areas, yielding effective solutions, its analyses can be performed on these problems. Accordingly, the time complexity and analysis of MCA are provided for the Minimum Vertex-Cover Problem (MVCP) [12].

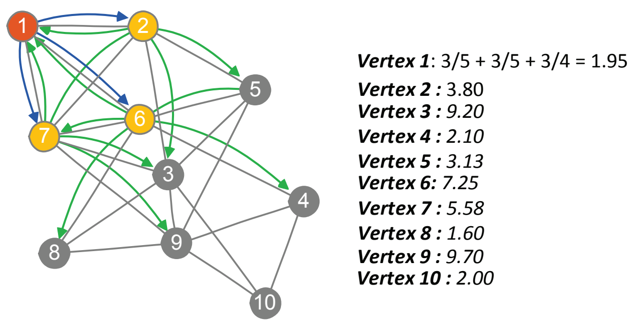

Figure 2. MCA implementation phases.

Theorem 1: MCA has effective and polynomial time complexity for MVCP[12].

Proof: Assume that $G_0 = (V, E)$ is a simple graph. $\delta(G_0)$ is minimum node degree and $\Delta(G_0)$ is maximum node degree.

$V_c = \phi$ // V_c is a vertex-cover set

$v_j \in V$ *and* v_j = *arg max*$\{ v_i \,|\Psi(v_i) = \sum_{\forall v_j \in N(v_i)} \frac{d(v_i)}{d(v_j)}$, $1 \le i \le |V|$ *and* $1 \le i \le |V|, i \ne j \}$

$$V_c = V_c \cup \{v_j\}$$

$G_1 = \{V_1, E_1\}$ *and* $V_1 = V - v_j, E_1 = E - \{\forall(v_r, v_j) \in E, v_r \in N(v_j)\}$

This step requires maximum $|V|.\Delta(G)$ aritmetrik operations.

If $E_1 \ne \phi$, the same process will be applied to G_1

Assume that $v_j \in V_1$ and v_j = *arg max*$\{\Psi(v_i) = \sum_{\forall v_j \in N(v_i)} \frac{d(v_i)}{d(v_j)}, 1 \le i, j \le |V_1|,\ i \ne j \}$

$$V_c = V_c \cup \{v_j\} \text{ and}$$

$G_2 = \{V_2, E_2\}$ *and* $V_2 = V_1 - v_j$ *and* $E_2 = E_1 - \{\forall(v_r, v_j) \in E, v_r \in N(v_j)\}$

The number of arithmetic operations is less then $|V|.\Delta(G)$.

If $E_1 \ne \phi$, the same process will be applied to G_2, otherwise, algorithm is terminated.

Assume that the obtained graph series is $G_0, G_1, G_2, \ldots G_m$.

The total number of arithmetic operations is

$$\sum_{i=0}^{m-1} |V_i|.\Delta(G) < \sum_{i=0}^{m-1} |V|.\Delta(G) = m|V|.\Delta(\mathrm{G}). \quad (5)$$

Finally, $T(n) = O(\sum_{i=0}^{m-1} |V|.\Delta(G))$ and T(n) is the time complexity of algorithm.

Corollary: Malatya centrality algorithm has polynomial space complexity.

Proof: Malatya centrality algorithm uses at most two adjacency matrices. So space complexity is $O(|V|^2)$.

Kendall Rank Correlation Coefficient

The Kendall rank correlation coefficient is a statistical measure that assesses the relationship between two ranked variables. This coefficient evaluates the differences in the ranking order of two variables. For two variables X and Y, Kendall (τ), considers ranking pairs (x_i,y_i) and (x_i,y_i). These pairs are classified based on whether they are in order as follows:

Concordant Pairs: $(x_i - x_j)(y_i - y_j) > 0$

Discordant Pairs: $(x_i - x_j)(y_i - y_j) < 0$

Kendall (τ) is calculated using the difference between the number of Concordant and Discordant pairs.

$$\tau = \frac{(C - D)}{\sqrt{(C + D + T)(C + D + U)}} \quad (8)$$

Here: C = number of Concordant pairs, D = number of Discordant pairs, T = number of Tied pairs $(x_i=x_j)$, U = number of Tied pairs $(y_i=y_j)$ [38]. Alternatively, the simplified form, ignoring the effect of ties, can be written as follows.

$$\tau = \frac{2(C - D)}{n(n - 1)} \quad (9)$$

Here, n represents the number of observations. The obtained Kendall coefficient value ranges between -1 and 1.

If τ = 1, it means that the rankings of the two variables are exactly the same. All pairs are concordant and there are no discordant pairs. If τ = -1, the rankings of the two variables are completely opposite. All pairs are discordant and there are no concordant pairs. When. τ = 0, it indicates that there is no ordinal relationship between the two variables. The number of concordant and discordant pairs is equal, neutralizing each other [10].

In summary:

- τ > 0: Positive ordinal relationship
- τ = 0: No or very weak ordinal relationship
- τ < 0: Negative ordinal relationship

Statistical Comparison Methods for Two Independent Samples: Welch T-Test and Wilcoxon Rank Sum Test

Comparative analyses of the MCA with other centrality measures included the results of the Wilcoxon Rank Sum and Welch Two Sample t-tests. This addition enhanced the analysis by providing insights both in terms of rank-based and group-level results. The Welch Two Sample t-test is typically considered a parametric test or a test of mean differences, while the Wilcoxon Rank Sum Test falls under non-parametric tests or tests for ordinal data. Additionally, these tests are often categorized as rank-based tests.

Welch two sample t-test

Welch's t-test is a parametric test used to evaluate whether the means of two independent samples are equal. Unlike the standard t-test, it does not assume that the variances of the two sample groups are equal, making it suitable for cases where the variances between the two samples differ [39]. Mathematically, it is expressed as shown in Equation 10 [39].

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \quad (10)$$

Here, $\bar{X}_1$ and $\bar{X}_2$ represent the group means, s_1^2 and s_2^2 represent the group variances, and n_1 and n_2 represent the sample sizes. The p-value obtained from the test indicates whether the difference between the two groups is statistically significant.

Wilcoxon rank sum test (mann-whitney u test)

The Wilcoxon Rank Sum test is a non-parametric test used to compare the central tendency between two independent samples. It is employed in place of Welch's t-test when the data do not follow a normal distribution. Instead of comparing means, this test compares the median ranks between groups [40].

The Wilcoxon Rank Sum test is based on ranking the data and calculating the sum of ranks. The two groups are ranked jointly, after which the sum of ranks for each group is compared. The formula is expressed as shown in Equation 11 [40].

$$U = n_1 n_2 + \frac{n_1(n_1+1)}{2} - R_1 \quad (11)$$

Here, n_1, n_2 denote the sizes of the two groups, and R_1 represents the rank sum for the first group.

EXPERIMENTAL RESULTS

The MCA offers solutions to a variety of graph problems independent of graph type. The behavior and outcomes of the MCA across different classes and densities of graphs can potentially be effective in solving other problems within various graph classes. Therefore, during testing, a wide range of graphs of varying classes and densities were evaluated, without limiting the process to specific graph types or sizes. These various types of graphs are widely used in defining and solving many problems in the literature. Additionally, test results for benchmark graphs, such as those from DIMACS, are included.

The analysis was conducted using a broad range of graphs, including random, social, DIMACS benchmark, bipartite, and lattice graphs. This diverse selection of graphs helped produce more consistent results and allowed for meaningful comparisons of the methods' similarities. Accordingly, various test outcomes are provided to facilitate comparisons and evaluations between MCA and centrality algorithms widely recognized in the literature. These results offer the opportunity to conduct extensive analyses and evaluations of MCA in relation to established centrality algorithms.

For instance, in Table 1, graphs were generated with different vertices and densities using the Erdos-Renyi model to reveal the relationships between the methods. A graph with the parameter 50*0.40 has 50 vertices and 502 edges.

Table 1. Random graphs Kendall ranking scores

	Random Graphs	V	E	Pagerank	Closeness	Betweenness	Eigenvector
Malatya Centrality	50*0.20	50	250	0.6228571	0.1787755	0.1869388	0.06938776
	50*0.40	50	502	0.6522449	0.2457143	0.317551	-0.03183673
	50*0.50	50	598	0.7322449	0.03673469	0.2604082	0.2179592
	50*0.60	50	733	0.7028571	0.2636735	0.1657143	0.1771429
	50*0.75	50	946	0.9559184	0.4514286	0.2620408	0.1428571
	100*0.20	100	1000	0.7608081	0.1151515	0.1123232	0.08444444
	100*0.40	100	1915	0.6036364	0.2181818	0.109899	0.04848485
	100*0.50	100	2493	0.909899	0.1757576	0.05818182	0.03838384
	100*0.60	100	2953	0.8836364	0.1232323	0.1050505	0.04646465
	100*0.75	100	3711	0.9482828	0.08686869	0.2044444	0.05818182
	500*0.20	500	24895	0.3073828	0.1246012	0.08301403	0.07852505
	500*0.40	500	50051	0.6803848	0.0816513	0.05749098	0.03743487
	500*0.50	500	62589	0.7738036	0.09125451	0.06888978	0.061499
	500*0.60	500	75030	0.8132104	0.08211623	0.1271343	0.06401603
	500*0.75	500	93431	0.888994	0.1099319	0.05590381	0.08327054
	2000*0.20	2000	400763	0.2335968	0.01392796	0.02581391	0.02157079
	2000*0.40	2000	799134	0.569958	0.01857629	0.02732466	0.009121561
	2000*0.50	2000	1000139	0.7715238	0.04033817	0.03755378	0.02983792
	2000*0.60	2000	1197775	0.843964	0.0218089	0.02928764	0.01924762
	3000*0.20	3000	899687	0.2080182	0.01102412	0.01743603	0.0282383
	4000*0.20	4000	1600045	0.1889085	0.00561665	0.01640535	0.009877969
	5000*0.20	5000	2497811	0.1780263	0.00377707	0.01896523	-0.01223461
	6000*0.20	6000	3603003	0.1699664	0.01321642	0.007047841	0.01228538
	8000*0.20	8000	6402266	0.1684382	0.01555576	-0.00764739	-0.00515101
	10000*0.20	10000	9997928	0.1267672	0.00249076	0.01327261	0.01341162

In the test conducted on this graph, the τ value between MCA and PageRank was found to be 0.652. The values with other methods were 0.245 with Closeness, 0.317 with Betweenness, and -0.031 with Eigenvector. These values indicate a strong positive relationship between MCA and PageRank for this graph. In other words, there is a significant similarity in the effectiveness rankings of the vertices. Although there is also a positive relationship with Closeness and Betweenness methods, it is weaker compared to PageRank. The result of the Eigenvector algorithm is negative and very close to 0. This indicates a weak negative relationship between MCA and the Eigenvector algorithm. In other words, a node that is effective for MCA is seen as ineffective according to the Eigenvector algorithm.

For the larger and denser random graph with the parameter 500*0.75, which consists of 500 vertices and 93,431 edges, the Kendall τ values for MC were found to be 0.888 with PageRank, 0.109 with Closeness, 0.055 with Betweenness, and 0.08327054 with Eigenvector. When examining these values, it is evident that MC has a strong positive relationship with PageRank and a weak positive relationship with the other methods. Overall, the table shows that there are strong correlations between MC and PageRank algorithm results for all graphs. As the density of the graphs increases, the similarity between the results of MC and PageRank algorithms generally increases. Based on this inference, applying the MC algorithm for problems solved with the PageRank algorithm in high-density graphs will be a significant alternative. Although Closeness has a weaker relationship compared to PageRank, it is observed that the MC Kendall value increases as density increases. This trend is not as evident for the other methods. According to the table results, Eigenvector and Betweenness methods have a weak positive or negative relationship with MC. In light of these results, the MC method is also applicable to many real-world problems modeled with high-density graphs and solved using PageRank.

To determine the similarities of MC with different graph types, Kendall τ scores were calculated for 25 different types of graphs under the categories of social, benchmark, bipartite, and lattice. When examining the Kendall τ scores specified in Table 2, it is observed that MC generally has a strong or moderate positive relationship with social, bipartite, and lattice graphs. For DIMACS benchmark graphs, however, it

Table 2. Kendall ranking scores in specific graphs

		Graphs	V	E	Pagerank	Closeness	Betweenness	Eigenvector
Malatya Centrality	Social	Zachary Kareta	34	78	0.422459	0.3939394	-0.1550802	0.00178253
		Dolphin	62	159	0.230037	-0.098889	0.001586462	-0.0163934
		Zebra	27	111	0.475783	-0.0370370	0.3048433	0.4301994
		Complex	70	133	0.180124	-0.007039	-0.00207039	-0.0633540
		Tribes	16	58	0.466666	0.3	0.3	0.25
	DIMACS	johnson8.2.4	28	210	-0.1428571	-0.1269841	0.4814815	0.7407407
		MANN.a9	45	918	0.5171717	0.0686868	-0.1838384	-0.1919192
		hamming6.2	64	1824	0.0843254	0.0317460	0.1309524	0.8769841
		hamming6.4	64	704	-0.3392857	0.6021825	0.2748016	0.203373
		johnson8.4.4	70	1855	-0.0973084	-0.0095238	0.2778468	0.8890269
		johnson16.2.4	120	5460	0.2252101	0.0117647	0.1784314	-0.2703081
		C125.9	125	6963	-0.0601290	-0.0056774	-0.09083871	-0.0345806
		keller4	171	9435	0.0355693	0.0307533	-0.0897832	0.03446852
		c.fat200.1	200	1534	0.0893467	0.141407	0.139598	-0.0036180
		brock200.1	200	14834	0.0990954	-0.0353768	-0.08361809	-0.0730653
	Bipartite & Lattice	Hypercube(Q6)	64	192	0.0605158	1	0.09920635	0.2599206
		Grid(9*9)	81	144	0.0617284	0.308642	0.06666667	0.01049383
		Knight(5*5)	25	48	0.3733333	1	-0.1133333	0.3133333
		King(4*5)	20	55	0.0736842	0.3894737	0.3578947	0.4315789
		Triangular(10)	55	135	0.3185185	0.0585858	0.1973064	-0.0505050
		Bethe Lattice	190	189	0.924812	0.7540518	0.7540518	0.467892
		Banana Tree(4*4)	17	16	1	0.7058824	0.7058824	0.25
		Folkman	20	40	0.1368421	0.4210526	1	0.1368421
		Hoffman	16	32	0.8833333	1	0.1	-0.1
		Horton	96	144	0.3289474	-0.1179825	-0.07105263	0.1574561

has weak positive or negative relationships. With Closeness and Betweenness methods, there is a largely strong positive relationship in bipartite and lattice graphs, whereas it is not possible to generalize for social and DIMACS graphs. As for the Eigenvector centrality method, it acts independently across all graph types and does not form a strong relationship with MC except for a few graphs.

When the table is examined overall, it is observed that the MC and PageRank algorithms generally produce similar results in high-density graphs. The Closeness and Betweenness methods produce similar results in certain graph types such as bipartite and lattice graphs, while they produce different results independent of MC in random and benchmark graphs. The Eigenvector method, on the other hand, has produced results that are distant from and dissimilar to MC in almost all types of graphs tested. In other words, regardless of the graph type, size, and density, the Eigenvector centrality method does not produce similar results to the Malatya centrality. In fact, in some graphs, it has produced solutions that are opposite to those of MC.

In Figure 2, to enhance the comprehensibility of the presented study, centrality methods have been applied to three different graphs, and the results are visually depicted. The size of the vertices in the visuals indicates the high centrality value of the respective node. For example, in the Lattice-Grid graph consisting of 24 vertices and 46 edges, vertices 6, 7, 18, and 19 are the most influential according to the centrality values, while they are less influential according to the Closeness centrality measure. This distinction is clearly highlighted in the visuals.

In the Bipartite-Knight graph, consisting of 16 vertices and 24 edges, the Kendall correlation ranking score between the MC and Betweenness methods was found to be 1, meaning that the ranking results of these two methods are exactly the same. For this graph, MC establishes a strong positive relationship with the Closeness method, with a value of 0.66, while it forms a weak negative relationship with the results of the Eigenvector and PageRank algorithms.

In this section, methods are further analyzed graphically according to the Kendall score values based on graph densities. This approach provides more detailed analysis results regarding the variation of Kendall scores of the MCA in relation to density. In Table 3, the analysis process is conducted on graphs generated by the Erdos-Renyi model consisting of 200 vertices and edges of varying densities. An examination of the results in Table 3 reveals a direct correlation between the MCA and Pagerank algorithms. As the graph densities increase, the Kendall score values also rise, indicating a high similarity in the ranking outcomes of MCA and Pagerank in densely connected graphs. In contrast, an analysis of the results for other centrality methods, such as Closeness, Betweenness, and Eigenvector, shows that the Kendall score values are not dependent on graph density.

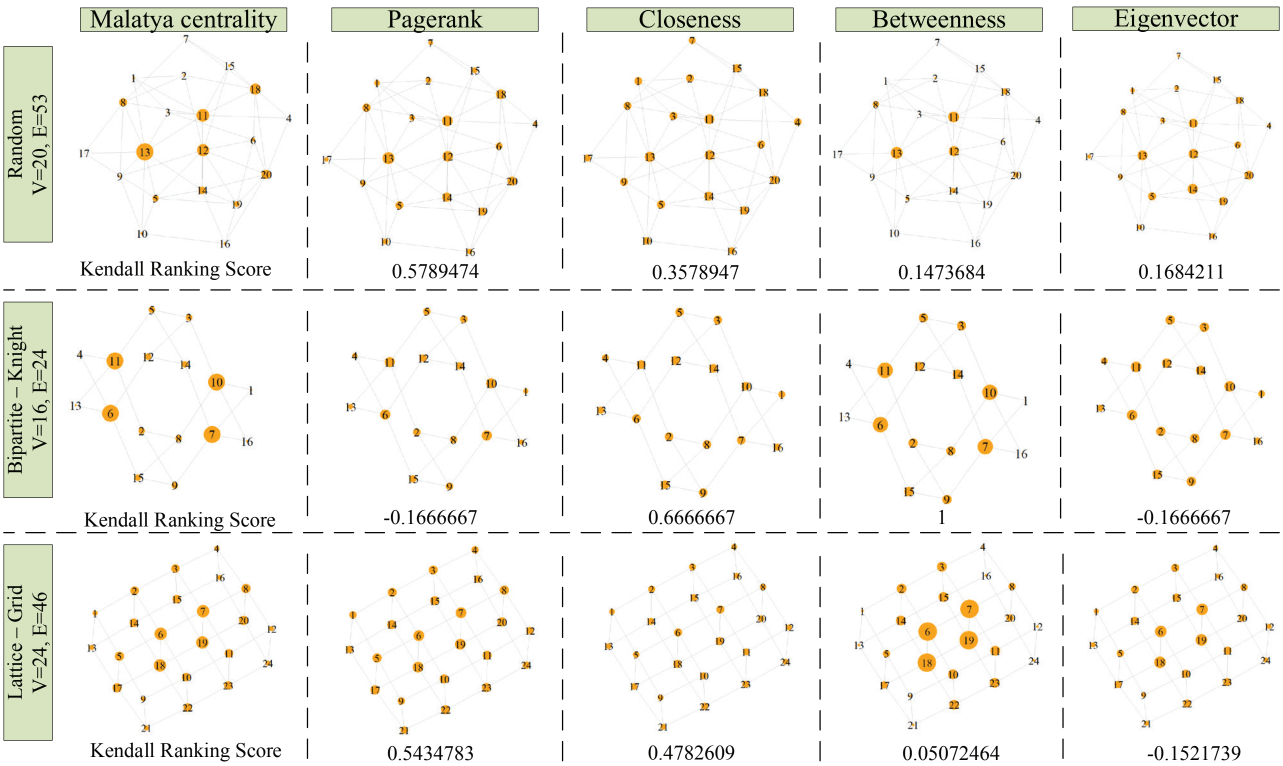

Figure 3. Kendall (τ) score of centrality methods and visually dominant values of vertices.

Table 3. Kendall score based on graph density

	Approximate Density	V	E	Pagerank	Closeness	Betweenness	Eigenvector
Malatya Centrality	5%	200	987	0.0747738	-0.001708543	-0.001306533	0.04
	10%	200	1945	0.1438191	0.05105528	0.0198995	0.004321608
	15%	200	3023	0.3251256	0.07507538	0.1239196	0.0840201
	20%	200	4078	0.3926633	0.09145729	0.1110553	0.09417085
	25%	200	4917	0.5588945	0.06472362	0.06361809	0.001708543
	30%	200	5891	0.4821106	0.06060302	0.09055276	0.009547739
	35%	200	6989	0.7557789	0.1779899	0.1476382	0.1542714
	40%	200	8034	0.7464322	0.2282412	0.1678392	-0.02301508
	45%	200	8879	0.7460302	0.1115578	0.05829146	0.06452261
	50%	200	9894	0.8540704	0.2091457	0.04502513	0.02090452
	55%	200	10842	0.7654271	0.1705528	0.06321608	0.03125628
	60%	200	11814	0.798191	0.08361809	0.06	0.003517588
	65%	200	12995	0.8311558	0.2661307	0.1302513	0.06432161
	70%	200	14011	0.8029146	0.08753769	0.05788945	0.1732663
	75%	200	15055	0.8685427	0.1071357	0.1055276	0.08361809
	80%	200	15888	0.9177889	0.1461307	0.007437186	0.01427136
	85%	200	16956	0.8454271	0.01296482	0.004522613	0.04944724
	90%	200	17905	0.8770854	0.1251256	0.07708543	0.201206
	95%	200	18896	0.9446231	0.05125628	0.07698492	0.1031156
	100%	200	19900	1	1	1	0.5739698

In Figure 3, the graph of Kendall scores based on the graph densities provided in Table 3 is presented. Upon examining the linear graph, it is observed that as density increases, the Kendall rank correlation coefficient between the MC and PageRank algorithms generally increases. Conversely, the results of the Closeness, Betweenness, and Eigenvector algorithms appear to be independent of the graph densities. The analysis results highlight that in high-density graphs, the MC results are significantly similar to those of the PageRank algorithm.

The Figure 4 included in the analysis in Table 4 are derived from the network repository [41], a source indexed in the Data Citation Index. Examples were selected under various categories to enrich the types of graphs used in the analysis.

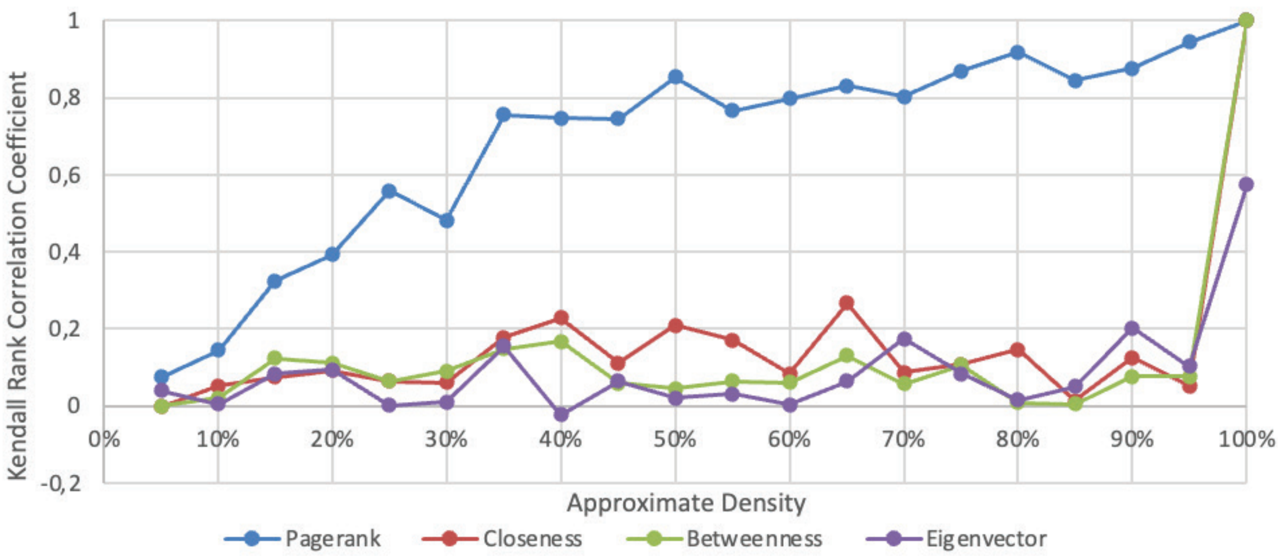

Figure 4. Kendall score based on graph density.

Table 4. Wilcoxon rank sum test results

		Graphs	V	E	Pagerank	Closeness	Betweenness	Eigenvector
Malatya Centrality	Special Networks	cat-mixed-species-brain-1	65	1139	0.2846154	-0.00384615	-0.01923077	-0.05192308
		ENZYMES-g118	95	242	0.2451754	0.06929825	0.1377193	0.1153509
		cit-DBLP	12591	49743	0.1403004	0.09917071	0.2332719	0.1017
		email-univ	1133	5451	0.2022181	0.2647463	0.2083309	0.2543733
		130bit	584	6120	0.4284405	0.2269908	0.4136023	0.2369417
		infect-dublin	410	2765	0.1066372	-0.03440873	0.004949609	0.008145984
		soc-wiki-Vote	889	2914	-0.0210885	-0.01905167	0.02636326	-0.02600857
		socfb-Reed98	962	18812	0.0138434	-0.00137806	-0.04876461	0.006055283

Wilcoxon Rank Sum and Welch Two Sample T-Test

In this section, Wilcoxon Rank Sum and Welch Two Sample t-test analyses were conducted to compare the MCA with the PageRank, Closeness, Betweenness, and Eigenvector centrality methods. Through these tests, differences between the MCA and other methods were identified. For the testing process, the results of all centrality methods were normalized between 0 and 1 to eliminate the large differences in dominance values produced by the various centrality methods. The tests were conducted on the Zachary Karate Club, Dolphin, Zebra, Triangular(10), and MANN-a9 graphs previously chosen in earlier sections.

Examining the test results on the Zachary Karate Club graph in Table 5, the PageRank method yielded a W value of 192, a P value of 2.248e-06, an effect size of 0.574, and a magnitude classified as large. These results indicate a statistically significant difference between the MCA and PageRank. The P value is very low, and the effect size is classified as large, meaning the MCA demonstrates significantly different performance compared to PageRank.

Table 5. Wilcoxon rank sum test results

Wilcoxon rank sum test						
	Graphs	**Centrality Metric**	**W**	**P**	**Effsize**	**Magnitude**
Malatya Centrality	Zachary Kareta	Pagerank	192	2.248e-06	0.574	large
		Closeness	113	1.186e-08	0.692	large
		Betweenness	680.5	0.2092	0.153	small
		Eigenvector	143	9.78e-08	0.647	large
	Dolphin	Pagerank	1001	4.22e-06	0.413	moderate
		Closeness	476	5.035e-13	0.649	large
		Betweenness	2153	0.2491	0.104	small
		Eigenvector	1575	0.08333	0.156	small
	Zebra	Pagerank	184	0.001825	0.425	moderate
		Closeness	581.5	0.0001722	0.512	large
		Betweenness	590	8.916e-05	0.534	large
		Eigenvector	307	0.3236	0.136	small
	Triangular(10)	Pagerank	1305	0.214	0.119	small
		Closeness	1758	0.1412	0.141	small
		Betweenness	1974	0.00565	0.264	small
		Eigenvector	1983	0.004788	0.269	small
	MANN-a9	Pagerank	1548	5.236e-06	0.481	moderate
		Closeness	1012.5	1	0	small
		Betweenness	1552.5	4.077e-06	0.486	moderate
		Eigenvector	1606.5	6.411e-07	0.525	large

For the Closeness algorithm, a W value of 113, a P value of 1.186e-08, an effect size of 0.692, and a magnitude classified as large were obtained. The difference between Closeness and the MCA is highly significant. The very low P value indicates a strong difference, and the high effect size suggests that the MCA may have a distinct advantage over Closeness.

When analyzing the Betweenness algorithm, the P value is above 0.05, and the effect size is small. This suggests that the two algorithms yield similar results or that the difference is not statistically significant. In the case of the Eigenvector algorithm, the P value is very low, and the effect size is large, indicating a significant difference between the results of the MCA and Eigenvector algorithms.

Table 6. Welch two sample t-test results

Welch Two Sample t-test								
	Graphs and MC mean value	**Centrality Alg. and mean value**	**t**	**df**	**p**	**confidence interval**	**Effsize**	**Magnitude**
Malatya Centrality	Zachary Kareta (0.09745462)	Pagerank (0.21725293)	-2.0981	65.774	0.03974	-0.23380802 -0.00578860	-0.509	moderate
		Closeness (0.49914053)	-6.8653	65.301	2.94e-09	-0.5185266 -0.2848452	-1.67	large
		Betweenness (0.10055460)	-0.05766	65.746	0.9542	-0.1104324 0.1042324	-0.014	negligible
		Eigenvector (mean)	-4.2803	64.973	6.273e-05	-0.3719367 -0.1352753	-1.04	large
	Dolphin (0.1880573)	Pagerank (0.4135502)	-4.9302	113.8	2.821e-06	-0.3160999 -0.1348861	-0.885	large
		Closeness (0.5382324)	-8.9625	122	4.489e-15	-0.4275207 -0.2728295	-1.61	large
		Betweenness (0.1582461)	0.78424	121.58	0.4344	-0.04544128 0.10506361	0.141	negligible
		Eigenvector (0.2862439)	-2.1599	114.3	0.03287	-0.18823722 -0.00813604	-0.388	small
	Zebra (0.3534010)	Pagerank (0.5585025)	-2.8178	51.881	0.006827	-0.35117124 -0.05903182	-0.767	moderate
		Closeness (0.1692843)	2.1405	48.95	0.03732	0.0112546 0.3569788	0.583	moderate
		Betweenness (0.1166862)	3.3768	51.138	0.001408	0.09599118 0.37743832	0.919	large
		Eigenvector (0.5191267)	-1.6116	42.428	0.1145	-0.37318925 0.04173781	-0.439	small
	Triangular (10) (0.5844156)	Pagerank (0.6251156)	-0.81444	107.41	0.4172	-0.13976211 0.05836199	-0.155	negligible
		Closeness (0.5182810)	1.2967	107.89	0.1975	-0.03496449 0.16723370	0.247	small
		Betweenness (0.4763705)	2.0145	107.51	0.04646	0.00172667 0.21436348	0.384	small
		Eigenvector (0.4413064)	2.714	107.87	0.00774	0.03858852 0.24762982	0.518	moderate
	MANN-a9 (0.8)	Pagerank (0.8)	1.5622e-13	88	1	-0.1694766 0.1694766	3.29e-14	negligible
		Closeness (0.8)	0	88	1	-0.1694766 0.1694766	0	negligible
		Betweenness (mean)	3.9056e-14	88	1	-0.1694766 0.1694766	8.23e-15	negligible
		Eigenvector (mean)	1.8096e-13	88	1	-0.1694766 0.1694766	3.81e-14	negligible

In Table 6, Welch Two Sample t-tests were conducted to compare the MCA with other centrality metrics. The tests were conducted on the same graphs listed in Table 5, enabling a more thorough analysis of the comparison results. The results under different categories are shown in Table 6. The mean value represents the average result for each centrality algorithm, and the mean value for the MCA is included in the column corresponding to each graph name. For example, for the Zachary Karate Club graph, the mean value of MC was determined to be 0.09745462.

The t-value is the statistical outcome of the Welch Two Sample t-test, indicating the magnitude and direction of the difference between the MCA and the specified centrality metric. A negative t-value shows that the MCA has a lower value than the centrality metric in question. The degrees of freedom (df) parameter affects the reliability of the test; a higher degree of freedom generally indicates more observations and a stronger test. The p-value represents the significance level of the test, and values below 0.05 are generally considered statistically significant, indicating a meaningful difference between the MCA and the compared centrality metric. Low p-values suggest a very low probability that the observed differences are due to chance.

The confidence interval represents the test result's confidence range, showing where the mean difference between the two algorithms will lie. For instance, a confidence interval between -0.23380802 and -0.00578860 indicates that the mean difference between these two algorithms lies within this range with 95% confidence. The effect size shows the magnitude of the difference between the two centrality metrics, reflecting how much the MCA differs from the other metric. A high effect size indicates a meaningful performance difference between the two algorithms. The negative sign in this column indicates that the compared centrality metric has a higher value than the MCA. Magnitude classifies the effect size as follows: Negligible (no significant effect), Moderate (moderate effect), and Large (strong effect, indicating a notable difference).

Upon examining the test results for the Zachary Karate Club graph in Table 6, PageRank has a mean value of 0.21725293, a t-value of -2.0981, a df of 65.774, a p-value of 0.03974, a confidence interval between -0.23380802 and -0.00578860, an effect size of -0.509, and a magnitude of moderate. These findings indicate a statistically significant difference between PageRank and the MCA. The effect size is moderate, showing a statistically significant performance difference, though not extremely high.

The results for the Closeness algorithm, with a mean value of 0.49914053, show a highly significant and large effect difference with the MCA, suggesting that the MCA exhibits a distinctly different performance compared to Closeness. For the other centrality metrics, there is no significant difference between Betweenness and the MCA, indicating no notable performance difference between the two algorithms. However, a significant difference was found between Eigenvector and the MCA. The high effect size here demonstrates a distinct performance difference between these two algorithms.

CONCLUSION

The MCA is an effective centrality algorithm that can be used in various fields such as graph theory and real-life problems. This algorithm calculates the centrality values of any node in the graph using the degree of the node and the degrees of its neighboring nodes. The Kendall tau test has shown for which types and densities of graphs the MCA produces similar results to other centrality metrics. Using this test, experiments were conducted on different types of graphs, including random graphs, benchmark graphs, social network graphs, bipartite graphs, and lattice graphs.

The tests and resulting solution sets suggest that MCA's performance on benchmark graphs is largely independent of other centrality measures, showing little to no significant similarity. However, when it comes to bipartite and lattice graphs, the results are mixed—some graphs exhibit strong similarities, while others show weaker correlations. In the case of social networks, a generally positive relationship is observed. Notably, in high-density random graphs, MCA demonstrates a strong positive correlation with PageRank. For other metrics, the relationships are generally moderate or weak.

The similarity of results produced by these well-known centrality measures, developed for different purposes, with the MCA in various graph types demonstrates the effectiveness of the MCA. Similar and effective solution sets suggest that the proposed method can provide solutions to graph theory and real-life problems addressed by centrality measures in the literature. For example, the MCA can be applied as an alternative to PageRank applications on graphs like Hoffman and Bethe lattices. The MCA has low notation in both spatial and temporal complexity. Therefore, it can provide computational and temporal advantages over other metrics in many large and complex graph types. The tests and analyses not only demonstrate the success of the proposed algorithm but also indicate the potential applicability of the MCA in various fields and problems.

MCA's ability to produce effective and robust results indicates its potential to yield impactful solutions for fundamental problems in graph theory and their application domains. Specifically, the algorithm provides efficient and reliable solutions for NP-hard problems, including the independent set problem, vertex cover problem, dominating set problem, and graph coloring problem. Additionally, its applicability extends to various domains, such as text summarization, social network analysis, and link prediction, where these graph problems are commonly employed, demonstrating its versatility and effectiveness across diverse application areas.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

STATEMENT ON THE USE OF ARTIFICIAL INTELLIGENCE

Artificial intelligence was not used in the preparation of the article.

REFERENCES

[1] Ullah A, Wang B, Sheng J, Long J, Khan N. Identification of ınfluential nodes via effective distance-based centrality mechanism in complex networks. Complexity 2021;2021:1–15. [CrossRef]

[2] Soltani-Sarvestani M, Azimifar Z, Wong A. Variable Centrality: A scenario based centrality calculation method. J Comput Sci 2022;62:101681. [CrossRef]

[3] Keng YY, Kwa KH, McClain C. Convex combinations of centrality measures. J Math Sociol 2021;45:195–222. [CrossRef]

[4] Agryzkov T, Tortosa L, Vicent JF. A variant of the current flow betweenness centrality and its application in urban networks. Appl Math Comput 2019;347:600–615. [CrossRef]

[5] Jagadishwari V, Chakrabarty S. Link Prediction using Influencer nodes of a Social Network. Proc 2nd Int Conf Inven Res Comput Appl ICIRCA 2020 2020:927–930. [CrossRef]

[6] Mussone L, Viseh H, Notari R. Novel centrality measures and applications to underground networks. Phys A Stat Mech Its Appl 2022;589:126595. [CrossRef]

[7] Xing W, Ghorbani A. Weighted PageRank algorithm. Proc - Second Annu Conf Commun Networks Serv Res 2004:305–314. [CrossRef]

[8] Saxena A, Iyengar S. Centrality Measures in Complex Networks: A Survey. arXiv:2011.07190 2020. doi: 10.48550/arXiv.2011.07190

[9] Curado M, Tortosa L, Vicent JF, Yeghikyan G. Analysis and comparison of centrality measures applied to urban networks with data. J Comput Sci 2020;43:101127. [CrossRef]

[10] Franceschini F, Maisano D. Aggregating multiple ordinal rankings in engineering design: the best model according to the Kendall's coefficient of concordance. Res Eng Des 2021;32:91–103. [CrossRef]

[11] Schoch D, Valente TW, Brandes U. Correlations among centrality indices and a class of uniquely ranked graphs. Soc Netw 2017;50:46–54. [CrossRef]

[12] Yakut S, Öztemiz F, Karci A. A new robust approach to solve minimum vertex cover problem: Malatya vertex-cover algorithm. J Supercomput 2023;79:19746–19769. [CrossRef]

[13] Karcı A, Yakut S, Öztemiz F. A new approach based on centrality value in solving the minimum vertex cover problem: Malatya centrality algorithm. Comput Sci 2022;55:1–100. [CrossRef]

[14] Okumuş F, Karcı Ş. MDSA: A dynamic and greedy approach to solve the minimum dominating set problem. Appl Sci 2024;14:9251. [CrossRef]

[15] Bakan CT, Yakut S. development of text summarization method based on graph theory and Malatya centrality algorithm. J Comput Sci 2023:90–99. [CrossRef]

[16] Öztemiz F. Maximum flow with bipartite matching. J Comput Sci 2023;8:102–109. [Turkish]

[17] Hajarathaiah K, Enduri MK, Anamalamudi S, Sangi AR. Algorithms for finding influential people with mixed centrality in social networks. Arab J Sci Eng 2023;48:10417–10428. [CrossRef]

[18] De Souza ACM, Vieira AB, Wehmuth K, Ziviani A. Connectivity-based time centrality in time-varying graphs. J Complex Netw 2021;9:1–25. [CrossRef]

[19] Evans TS, Chen B. Linking the network centrality measures closeness and degree. Commun Phys 2022;5:1–11. [CrossRef]

[20] Chen HH, Dietrich U. Normalized closeness centrality of urban networks: impact of the location of the catchment area and evaluation based on an idealized network. Appl Netw Sci 2023;8:1–14. [CrossRef]

[21] Kotlarz P, Febo M, Nino JC. Brain network modularity and resilience signaled by betweenness centrality percolation spiking. Appl Sci 2024;14:104197. [CrossRef]

[22] Kintali S. Betweenness centrality: Algorithms and lower bounds. arXiv:0809.1906 2008. doi: 10.48550/arXiv.0809.1906

[23] Chiranjeevi M, Dhuli VS, Enduri MK, Hajarathaiah K, Cenkeramaddi LR, Member S. Quantifying node ınfluence in networks: Isolating-betweenness centrality for ımproved ranking. IEEE 2024;12:93711–93722. [CrossRef]

[24] Agryzkov T, Tortosa L, Vicent JF, Wilson R. A centrality measure for urban networks based on the eigenvector centrality concept. Environ Plan B Urban Anal City Sci 2019;46:668–689. [CrossRef]

[25] Arcagni A, Candila V, Grassi R. A new model for predicting the winner in tennis based on the eigenvector centrality. Ann Oper Res 2023;325:615–632. [CrossRef]
[26] Anastasiei B, Dospinescu N, Dospinescu O. Word-of-mouth engagement in online social networks: Influence of network centrality and density. Electron 2023;12:12857. [CrossRef]
[27] Yuliansyah H, Othman ZA, Bakar AA. A new link prediction method to alleviate the cold-start problem based on extending common neighbor and degree centrality. Phys A Stat Mech Its Appl 2023;616:128546. [CrossRef]
[28] Dal Col A, Petronetto F. Graph regularization centrality. Phys A Stat Mech Its Appl 2023;628:129188. [CrossRef]
[29] Yamaç K, Cancan M. On EV-degree and VE-degree based topological properties of the Sierpiński Gasket Fractal. Sigma J Eng Nat Sci 2019;37:1275–1280. [CrossRef]
[30] Shang Q, Deng Y, Cheong KH. Identifying influential nodes in complex networks: Effective distance gravity model. Inf Sci (Ny) 2021;577:162–179. [CrossRef]
[31] Qi X, Fuller E, Luo R, Zhang CQ. A novel centrality method for weighted networks based on the Kirchhoff polynomial. Pattern Recognit Lett 2015;58:51–60. [CrossRef]
[32] Sari HK. Some properties of the topological spaces generated from the simple undirected graphs. Sigma J Eng Nat Sci 2023;41:266–270. [CrossRef]
[33] Jahanbani A, Ediz S. The Sigma index of graph operations. Sigma J Eng Nat Sci 2022;27:58–66.
[34] Brin S, Page L. The Anatomy of a Large-Scale Hypertextual Web Search Engine. Comput Networks ISDN Syst 1998;96:179–185. [CrossRef]
[35] Sariyuce AE, Kaya K, Saule E, Catalyiirek UV. Incremental algorithms for closeness centrality. Proc - 2013 IEEE Int Conf Big Data Big Data 2013 2013:487–492. [CrossRef]
[36] Freeman L. A set of measures of centrality based on betweenness. Sociometry 1977;40:35–41. [CrossRef]
[37] Bonacich P. Some unique properties of eigenvector centrality. Soc Netw 2007;29:555–564. [CrossRef]
[38] De Raadt A, Warrens MJ, Bosker RJ, Kiers HAL. A comparison of reliability coefficients for ordinal rating scales. J Classif 2021;38:519–543. [CrossRef]
[39] Keselman HJ, Othman AR, Wilcox RR, Fradette K. The new and improved two-sample t test. Psychol Sci 2004;15:47–51. [CrossRef]
[40] Harris T, Hardin JW. Exact Wilcoxon signed-rank and Wilcoxon Mann-Whitney ranksum tests. Stata J 2013;13:337–343. [CrossRef]
[41] Network repository. Available at: https://networkrepository.com/. Accessed on Oct 26, 2024.