Vol: 3, Issue: 1, 2025
Pages:75-88
Received: 17 June 2025
Accepted: 24 June 2025

CURRENT TRENDS IN COMPUTING
Research Article
DOI: 10.71074/CTC.1721800

A NEW METHOD FOR DETERMINING THE TRUST STATUS OF INDIVIDUALS TO EACH OTHER IN SOCIAL NETWORKS

ESRA KARADENİZ KÖSE[1*], ALI KARCI[2]

[1] *Computer Technologies Department, Vocational School of Arapgir, Malatya Turgut Özal University, 44800, Malatya, Türkiye*
[2] *Software Engineering Department, Faculty of Engineering, İnönü University, 44100, Malatya, Türkiye*

ABSTRACT. **Social networks are a concept that has developed in recent years and have become widely used electronic environments in daily life. Social networks are modeled as graphs and problem solutions are handled graph-based. In this study, we seek to answer the questions "Does entity X trust entity Y?" and "Which entity is trustworthy and which is not?" in a social network. In this study, the trust of one entity in the network towards another entity is calculated and expressed numerically. Classical trust inference algorithms eliminate paths with shortest paths, etc., which makes the found trust rate suspicious. In our method, all paths between two individuals are found and taken into account, which makes it unique. Keeping information safe from untrusted users is crucial for social network entities. The method aims to protect the privacy and confidentiality of the entities by detecting the trustworthiness of the entities. The proposed method is applied to standard social networks and the results are presented in this paper.**

1. INTRODUCTION

Due to the value of information today, research has focused on the security, confidentiality and reliability of information sources. Every entity is constantly interacting, communicating and sharing information with other entities. The relationships between entities can be modelled as social networks and analysed using social network analysis techniques. Social networks are used, for example, to extract relationships between proteins in biology, to model academic collaborations, and in e-commerce channels such as "who bought this also bought that". They are also used to detect threat elements and groups in the security sector [1, 2].

Social networks consist of nodes representing entities and links representing relationships between entities, which are modeled and represented as graphs. Examining the network structure such as link prediction, trust calculation, sentiment analysis, community detection etc. and accessing meaningful information is called social network analysis.

E-mail address: esra.karadeniz@ozal.edu.tr (*).
Key words and phrases. Social network analysis, trust, graph navigation, trust in social network.

Trust in social networks directly shapes the effectiveness of relationships established or to be established between entities. Although trust between entities is sometimes explicitly stated, it is often measurable and non-existent, which necessitates the use of implicit trust models [3].

In many real-world social networks, individuals are required to share sensitive or strategic information. However, the lack of a reliable and comprehensive method for quantifying trust between entities poses significant risks. Most existing models rely on shortest-path heuristics or partial observations, which may misrepresent the actual trust dynamics. This study aims to fill this gap by proposing a method that considers all paths between entities to produce a comprehensive trust metric, enabling safer, more informed decision-making in networked systems.

Trust prediction is one of the popular study topics of social network analysis. Hamdi et al. [4] dealt with a specific issue and discussed finding a trustable path in their work. In determining the trust between two entities, Golbeck [5] applies the TidalTrust algorithm, which uses the shortest path, which he argued is the most powerful, instead of traversing the entire network. Similarly, Lin et al.'s study [6] and Massa et al.'s proposed MoleTrust [7] also operates with only the shortest paths. In their study, Oh et al. applied a design combining explicit trust and implicit trust in social networks and found trust connections with trust propagation algorithms [8]. Josang et al. have found trust by considering each of the paths between entities independently of each other, even if they overlap [9]. According to Guha et al. [10] made a study that used trust and distrust together and spread the trust in the network, although they were partially successful in the estimation of trust, which complicates the study since the estimation of distrust is not always possible. Jiang et al. also created a narrow trust graph in a large online social network and called their method SWTrust [11]. In addition, Fatehi et al. [12] stated that using long paths would produce more accurate results when investigating trust in online social networks.

Link prediction methods in social networks are also included in the problem of finding trust, and there are also studies on whether nodes that are not currently connected can establish trustworthy connections in the future [13, 14].

Many of the studies conducted on trust deal with the issue of rating products purchased on marketing sites and the reliability of recommendation systems related to it. For example, in their study [15], the authors present a recommendation method based on implicit trust relationships by using the similarity of interest in the determination of trust between two entities. Likewise, Liu et al. [16] has developed a suggestion mechanism based on subjective and objective trust relationship. Mayadunna et al. [17] have developed a recommendation algorithm based on reinforcement learning in calculating the trust value. Wu et al. [18] used trust relationships and reliability in deciding on the purchase of a product. An inexperienced user can decide whether to buy the product based on the trust information of users who have tried the product before with this study. In [19], a recommendation system based on the trust propagation model was proposed to give right recommendations. Apart from these, there is also a trust-based recommendation technique that uses the Location-Based Social Network (LBSN) users' trust values and recommends a location [20].

There is still a need for a method that measures trust between users by directly detecting trust or using indirect trust [21]. Based on this need, a method that directly estimates trust has been developed in this study. The above studies infer trust by eliminating certain factors. This study differs in that it estimates

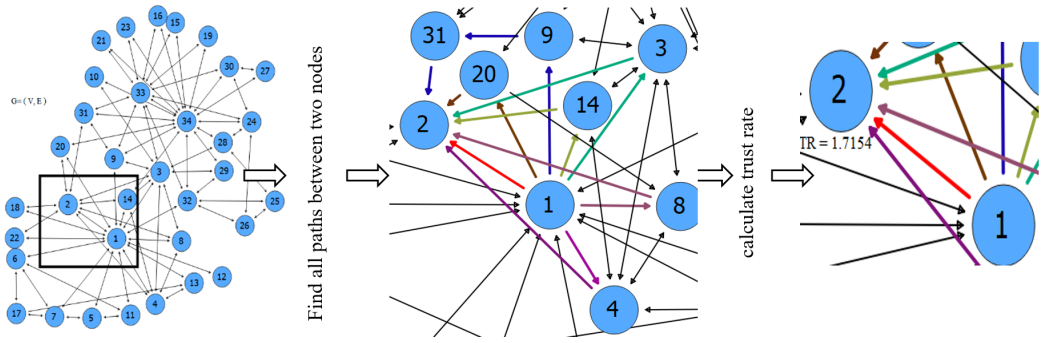

FIGURE 1. **A graphical abstract.**

trust and clearly expresses its value. By taking all connections into account and ensuring no data is lost, the accuracy of the expressed confidence value is strengthened. This makes this study unique. A graphical abstract of the method is given in Figure 1.

To address the challenges of partial or oversimplified trust estimation in social networks, this study proposes a novel trust computation method based on full-path enumeration using graph traversal. Unlike traditional methods that rely on shortest paths, the proposed algorithm evaluates all possible paths between entities, normalizes the results, and expresses the trust score as a percentage. This approach ensures a more holistic, data-complete, and accurate representation of inter-entity trust.

2. CONCEPTS AND DEFINITIONS

2.1. **Trust and Trust in the Social Network.**

We tend to value the opinions of those we trust, which makes trust a very valuable form of social knowledge [22]. Josang says "trust is a subjective concept" and defines trust as the criterion of an entity's subjective opinion about other entities [9]. As can be understood from this, the trust rating of each entity in the network will not be the same for an entity [1, 23]. A trust relationship is established between a trustee and a trusted. The trust characteristics [23] can be listed as bellows:

(1) Directional: To the extent that the trustee trusts the trusted, the trusted may not trust the trustee. The fact that A entity trusts on B entity does not mean that the reverse is also true.
(2) Subjective: Entity A and entity B may not trust on a third entity, entity C, at the same rate.
(3) Depends on the context: Entity A may trust entity B on one issue and not on another.
(4) Measurable: Trust is a phenomenon that can be measured. For measurement, numerical, probabilistic, etc. methods are used.
(5) Depends on the past: What happened in the past affects the current relationship of entity A and entity B
(6) Variable: The fact that two entities trust each other now does not mean that they will trust each other in the future.

(7) Transitivity: If entity A trusts entity B and entity B trusts entity C, it can be said that entity A trusts entity C [24].

2.2. **Graphs and Graphs Navigating.**

The work of Leonhard Euler, who tried to solve the Königsberg bridge problem in 1736, was shown as the beginning of graph theory. The Königsberg bridge problem addresses the question of whether a person can return to the starting point by crossing each of the seven bridges over the Pregel river in the town of Königsberg only once. Euler showed the problem as a graph and was able to analyze the problem easily [25].

Graphs are used to generate algorithmic solutions when analyzing problems in many fields of engineering, basic sciences, social sciences, humanities etc. If the definition of any problem can be in the form of nodes and relations between nodes, graphs and graph-related methods can be used to solve the problem.

Graphs are a mathematical structure consisting of sets V and E and are denoted by G(V, E). The elements of V are called vertices or nodes, and the elements of E are called edges. A graph G with vertices V and edges E is written as G= (V, E) or G (V, E). Let's consider inter-city road maps. It is possible to model these maps as a graph representing cities as nodes and roads as edges. Navigation is made on the graph by means of edges. The navigation on the diagram without node repetition is called path.

There are many forms of expression and modeling of social networks, but graphs are commonly used in the literature. The social network is a graph, the entities in the network are the nodes in this graph, and the relations between entities are expressed as the edges in the graph. Various calculations can be performed with graphs and matrices created from these graphs. For example clustering nodes, similarity calculation, link prediction etc. relationships can be calculated and interpreted [1].

While studying the topic of trust, two directly or indirectly connected nodes are selected from the V set. Let these be s and d. s is source and trustee, d is target and trusted.

The work of navigating between nodes through edges on a graph at the point of solving a problem is called navigating the graph.

Graph navigation algorithms are important and widely used algorithms in various fields such as data intensive, high performance computing problems, protein interactions, land transportation, and social networks [26, 27]. Examples of graph navigation algorithms are depth first search (dfs), breadth first search (bfs), binary search trees, Kruskal's algorithm, Bellman Ford algorithm, B-trees, Dijkstra's algorithm, uniform cost search, Floyd-Warshall algorithm etc. can be given. With these navigating algorithms, in general, the edges to be used in the search process are found without looping.

Traveling all nodes in the graph with the least cost, finding the shortest path between two determined nodes, finding the shortest path between all nodes, calculating the optimum flow between two nodes, finding the path tree, coloring a node or region, exiting a node and making a trip, etc. operations are the subject of graph traversal.

2.2.1. *Depth First Search (DFS).*

The DFS algorithm is a recursive algorithm that uses the idea of feedback [28]. It involves a thorough search of all nodes, going backwards if possible. The DFS transition of a graph produces a spanning tree as the final result. A spanning tree is a graph without loops. To implement the DFS transition, a stack data structure with the size of the total number of nodes in the graph is used.

As the Depth First Search traverses the graph, it goes as far as it can in depth and returns. In the operation of the algorithm, it selects a starting node and assigns all adjacent nodes of that node to a stack. It selects a node from the stack to select the next node to visit, and again assigns all adjacent nodes to a stack. It repeats this process until the stack is empty. However, it should be ensured that the visited nodes are marked. This will prevent the same node from being visited more than once. Otherwise, an infinite loop can be entered.

2.2.2. *Breadth First Search (BFS).*

It is the algorithm that goes to the same level neighbors, starting from the root node (source or starting node) from the tree traversal algorithms. It adds the neighbors of the visited nodes to a queue and runs according to the order in the queue. When the algorithm is running, it is first moved horizontally and all nodes of the current level are visited. It goes to the next level. A graph may contain loops that can bring you back to the same node as the graph is traversed. To prevent the same node from being processed again, an array is used that marks the node after it has been processed.

3. Proposed Method

If the nodes have a lot of common shares, common neighbors, etc., these nodes are similar, closely related, or in other words, they can be said to trust each other. If the neighbors of the nodes are the same, they are similar even if they are not neighbors themselves. For example, this logic is also used in citation analysis approaches. For example, let person X work with person Z and person Y work with person Z again. In this case, it is possible that X and Y will be seen as co-authors in the same study in the near future, since it is assumed that they are working on a similar topic [29]. Apart from this, similarity between nodes can be detected by methods such as measuring the number of independent paths between two nodes, the distance of the shortest path or random walking [30].

If the trust values of the entities in the social network can be clearly established numerically and supported by tangible evidence, behavior can be developed in accordance with the trust value of that entity, if it is safe, a relationship can be advised, if it is unsafe, it can be suggested to stay away. In the study, the trust impressions of the entities in the social network on each other are found numerically and the entities that X entity most trusted and do not trust are found by using all paths. Again, at the last stage, the findings are evaluated collectively and the most trusted and most untrusted entities of the network are determined.

In this study, the process is started with the construction of a model of the social network in the functioning of the method, and the social network is modeled as a graph. Graph;

$$V = \{v_1, v_2, \ldots, v_n\},\ E = \{(v_1, v_2), \ldots, (v_n, v_m)\},\ \forall v_i \in V,\ G = (V, E) \tag{1}$$

is represented as. Then the adjacency matrix of the graph is constructed. In graph theory, the adjacency matrix is a square matrix used to describe a finite graph. The adjacency matrix of a graph, sometimes called a connection matrix, is a matrix consisting of rows and columns labeled with graph nodes that take values of 1 or 0 depending on whether the nodes are adjacent (neighborhood or directly connected). For a simple graph that does not contain a loop in itself, the adjacency matrix must have 0's on the diagonal. The adjacency matrix of an undirected graph is symmetric [31].

A simple social network represented as a graph was shown in Figure 2. In Table 1, the adjacency matrix consisting of nodes and edges of this graph was also shown.

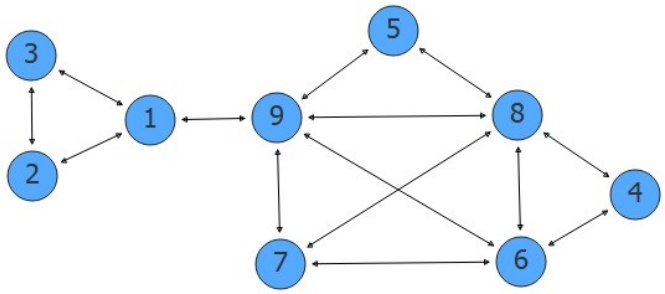

FIGURE 2. **An example of a graph.**

TABLE 1. **Adjacency matrix of the example graph**

	1	2	3	4	5	6	7	8	9
1	0	1	1	0	0	0	0	0	1
2	1	0	1	0	0	0	0	0	0
3	1	1	0	0	0	0	0	0	0
4	0	0	0	0	0	1	0	1	0
5	0	0	0	0	0	0	0	1	1
6	0	0	0	1	0	0	1	1	1
7	0	0	0	0	0	1	0	1	1
8	0	0	0	1	1	1	1	0	1
9	1	0	0	0	1	1	1	1	0

The next step involved developing a method to determine the level of trust between entities in the graph. This method's algorithm is presented as pseudo code in Algorithm 1.

Examining Algorithm reveals that all paths are found using the BFS algorithm with parallel programming (ExecutorService). Unlike the approach in [32], our algorithm spreads trust along network paths rather than working only with the trustee's and the trusted's direct neighbours. All paths to a node are

Algorithm 1 Trust Calculation in Social Network

Start
$G = (V, E)$ ▷ Convert social network to graph
Initialize the ExecutorService according to the optimum number of threads
`printAllPaths(s, d)` ▷ Prints all paths from s to d
Run BFS starting from source s
Store visited nodes in array `path[]`
if destination node is reached **then**
    Print `path[]`
end if
Mark nodes in `path[]` as visited
End ExecutorService
$B \leftarrow \texttt{PathsMatrix}(G)$ ▷ Paths matrix of graph G
for $j \leftarrow 1$ to $s(V)$ **do**
    $C[j] \leftarrow \sum_{i=1}^{s(V)} B[i, j]$ ▷ Sum paths by column
end for
for $t \leftarrow 1$ to $s(V)$ **do**
    for $k \leftarrow 1$ to $s(V)$ **do**
        $D[k,t] \leftarrow \dfrac{B[k,t]}{C[t]} \times 100$ ▷ Calculate percentiles
    end for
end for
$v_x \leftarrow \max(D[i,:])$ ▷ Most trusted entity of v_i
`IncreaseScore(`v_x`)`
$v_y \leftarrow \min(D[i,:])$ ▷ Least trusted entity of v_i
`DecreaseScore(`v_y`)`
End

summed. The paths from each node are proportioned to this sum, with trust expressed as a percentage. The entities that the entity under study trusts and does not trust are determined according to the percentage ratios found.

For example, when calculating the trust between v_1 and v_5, all the paths from v_1 to v_5 were found. There are eight of these paths, which are shown in Figure 3. The paths from v_1 to all nodes and their numbers were found (Table 2), and these numbers were summed. In the sample graph, this sum was determined as 54.

For a graph $G = (V, E)$, where $s, t \in V$, TrustRate$[t]$ represents the trust rate from node s to node t. In Equation (2), C_s denotes the sum of unique paths from node s to all other nodes in the graph, and B_t represents the sum of the unique paths from node s to node t.

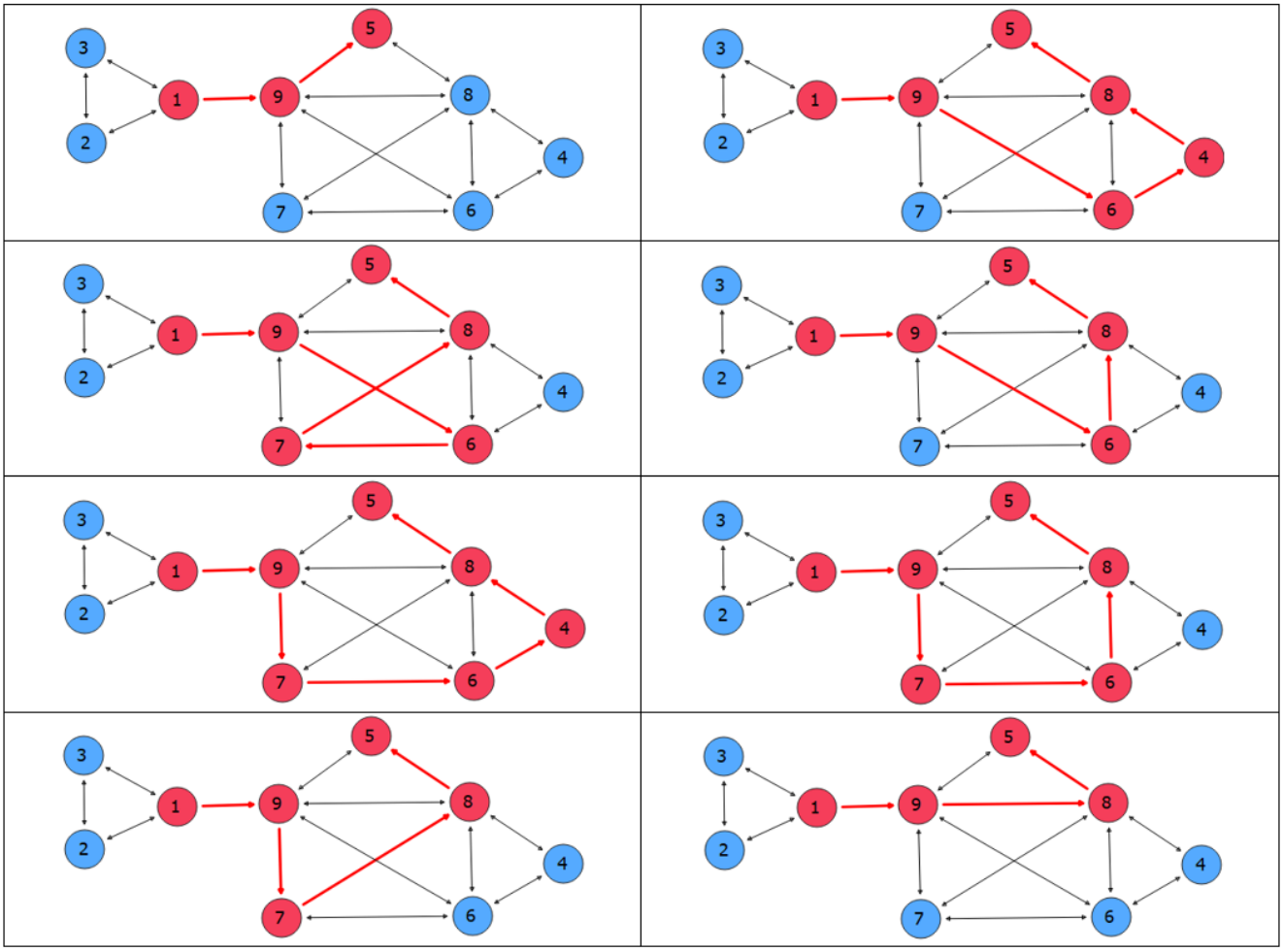

FIGURE 3. **All paths from v_1 to v_5**

$$\text{TrustRate}[t] = \frac{B_t}{C_s \times 100} \tag{2}$$

When calculating the trust of v_1 to v_5, according to this formula, it can be accessed from v_1 to v_5 from 8 different paths. This number of paths was divided by the total number of paths and multiplied by 100 to obtain a percentage expression of 14.81481 and this expression was called the trust rate. In this way, the number of paths evaluated for each node for v_1 was divided into the final total (Figure 4). As a result, v_4 was found to be the most trusted entity by v_1 with 24.07407 percent, and v_9 node was found to be untrusted by 1.851852 percent.

TABLE 2. **In the graph given in Figure2, sets of paths from node 1 to other nodes**

Target Node	**Paths from v_1 to other entities**	$s(K)$
K_2	$\{[1,2],[1,3,2]\}$	2
K_3	$\{[1,2,3],[1,3]\}$	2
K_4	$\{[1,9,5,8,4]$, $[1,9,5,8,6,4]$, $[1,9,5,8,7,6,4]$, $[1,9,6,4]$, $[1,9,6,7,8,4]$, $[1,9,6,8,4]$, $[1,9,7,6,4]$, $[1,9,7,6,8,4]$, $[1,9,7,8,4]$, $[1,9,7,8,6,4]$, $[1,9,8,4]$, $[1,9,8,6,4]$, $[1,9,8,7,6,4]\}$	13
K_5	$\{[1,9,5]$, $[1,9,6,4,8,5]$, $[1,9,6,7,8,5]$, $[1,9,6,8,5]$, $[1,9,7,6,4,8,5]$, $[1,9,7,6,8,5]$, $[1,9,7,8,5]$, $[1,9,8,5]\}$	8
K_6	$\{[1,9,5,8,4,6]$, $[1,9,5,8,6]$, $[1,9,5,8,7,6]$, $[1,9,6]$, $[1,9,7,6]$, $[1,9,7,8,4,6]$, $[1,9,7,8,6]$, $[1,9,8,4,6]$, $[1,9,8,6]$, $[1,9,8,7,6]\}$	10
K_7	$\{[1,9,5,8,4,6,7]$, $[1,9,5,8,6,7]$, $[1,9,5,8,7]$, $[1,9,6,7]$, $[1,9,6,4,8,7]$, $[1,9,6,8,7]$, $[1,9,8,4,6,7]$, $[1,9,8,7]$, $[1,9,7]$, $[1,9,8,6,7]\}$	10
K_8	$\{[1,9,5,8]$, $[1,9,6,4,8]$, $[1,9,6,7,8]$, $[1,9,6,8]$, $[1,9,7,8]$, $[1,9,7,6,4,8]$, $[1,9,7,6,8]$, $[1,9,8]\}$	8
K_9	$\{[1,9]\}$	1

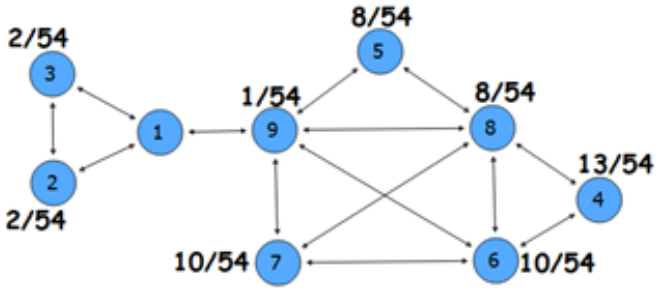

FIGURE 4. **Node v_1 operations.**

The same operations were repeated for all nodes in the graph, paths between nodes were found, and the number of path clusters were calculated (Table 3). Percentages were calculated by taking column-based sums and proportioning each row to its column total (Table 4).

TABLE 3. **The number of paths between the nodes of the sample graph**

	1	2	3	4	5	6	7	8	9
1	0	2	2	13	8	10	10	8	1
2	2	0	2	26	16	20	20	16	2
3	2	2	0	26	16	20	20	16	2
4	13	26	26	0	16	8	14	8	13
5	8	16	16	16	0	13	14	8	8
6	10	20	20	8	13	0	10	8	10
7	10	20	20	14	14	10	0	9	10
8	8	16	16	8	8	8	9	0	8
9	1	2	2	13	8	10	10	8	0

TABLE 4. **Expression of trust rates as a percentage**

	1	2	3	4	5	6	7	8	9
1	0	1.92308	1.92308	10.4839	8.08081	10.101	9.34579	9.87654	1.85185
2	3.7037	0	1.92308	20.9677	16.1616	20.202	18.6916	19.7531	3.7037
3	3.7037	1.92308	0	20.9677	16.1616	20.202	18.6916	19.7531	3.7037
4	24.0741	25	25	0	16.1616	8.08081	13.0841	9.87654	24.0741
5	14.8148	15.3846	15.3846	12.9032	0	13.1313	13.0841	9.87654	14.8148
6	18.5185	19.2308	19.2308	6.45161	13.1313	0	9.34579	9.87654	18.5185
7	18.5185	19.2308	19.2308	11.2903	14.1414	10.101	0	11.1111	18.5185
8	14.8148	15.3846	15.3846	6.45161	8.08081	8.08081	8.41121	0	14.8148
9	1.85185	1.92308	1.92308	10.4839	8.08081	10.101	9.34579	9.87654	0

As can be seen from Table 4, for example, the most trusted node of node v_1 is node v_4. In this case, the score of node v_4 was increased by 1. The untrusted node is node v_9, and the score of node v_9 was decreased by 1. In this way, when an node is marked as trustable by other nodes, its score is increased by 1, and when it is marked as untrustable, its score is decreased by 1. As stated in Algorithm 1, these scores of each node were evaluated and the most trustable nodes and the most untrustable nodes in the whole network were determined. In the example graph in Figure 2, it can be said that while v_2 and v_3 are the most trustable nodes with a score of 4, v_9 and v_1 are untrustable nodes with a score of -5 (Figure 5).

The coloring of the graph (Figure 5) is designed to visually indicate the trustworthiness level of each node. A scale is set between green and red. Nodes with high trust scores are colored in green tones, whereas less trusted nodes are displayed in red tones. This coloring scheme facilitates the rapid interpretation of trust distribution across the network. Although this is not a graph coloring in the traditional algorithmic sense, the visualization benefits from such a representation.

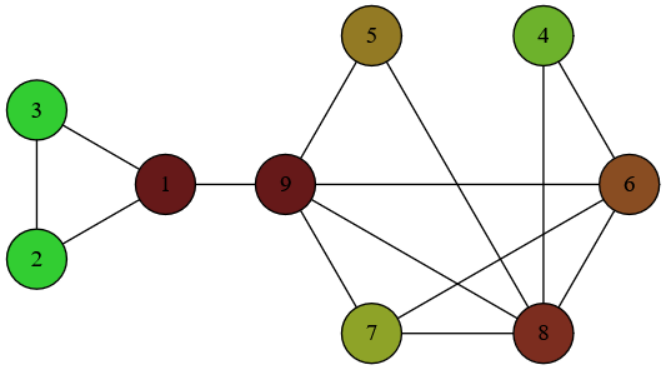

FIGURE 5. **The colored version of the sample graph according to the trust rates.**

4. FINDINGS AND DISCUSSION

This study calculates and expresses the trust between nodes in a network as a percentage. The applied method identifies all the paths between two nodes and then performs the necessary calculations, which is unique in this respect. However, this method has the disadvantage that the algorithm is too complex. However, we have found a solution to the time complexity issue by using parallel programming. When the graph becomes too large, it is necessary to eliminate paths in order to reduce complexity. Studies on trust either make calculations in the shortest path or in certain paths determined by the methods they have developed. Complexity is thus reduced, but this leaves a share of risk in the found trust and does not mean a real result.

The Zachary's Karate Club social network is one of the real-world networks often used in social network analysis. There are 34 students in the karate club and there is an edge between the nodes on the graph for students who are friends with each other. Figure 6 shows the path 1,3,9,33,21,34,14,2, which is one of the paths from node 1 to node 2. Similarly, 80137 different paths from 1 to 2 have been found.

For each of the 34 nodes, the paths to 33 nodes other than itself are calculated, then the sum of these paths is found and the method applied in Algorithm 1 is operated. As a result, the most trusted node of node 1 is found to be 26, and the one it does not trust is found to be 12. The most trusted node of node 2 is 17, and the ones it does not trust are 12, 18 and 22. The most trusted node of node 3 is found to be 17, and the most distrustful is found to be 10, 34. It is found that node 4 trusted 17, and untrusted 1, 12, 13.

Our method is able to express trust values numerically without data loss, but the disadvantage is that the time cost is very high.

5. CONCLUSION

In this study, a novel methodological framework has been proposed to address the challenge of estimating trust levels among individuals within social networks. The developed approach facilitates the quantitative evaluation of mutual trust ratios between entities, offering a more nuanced and measurable

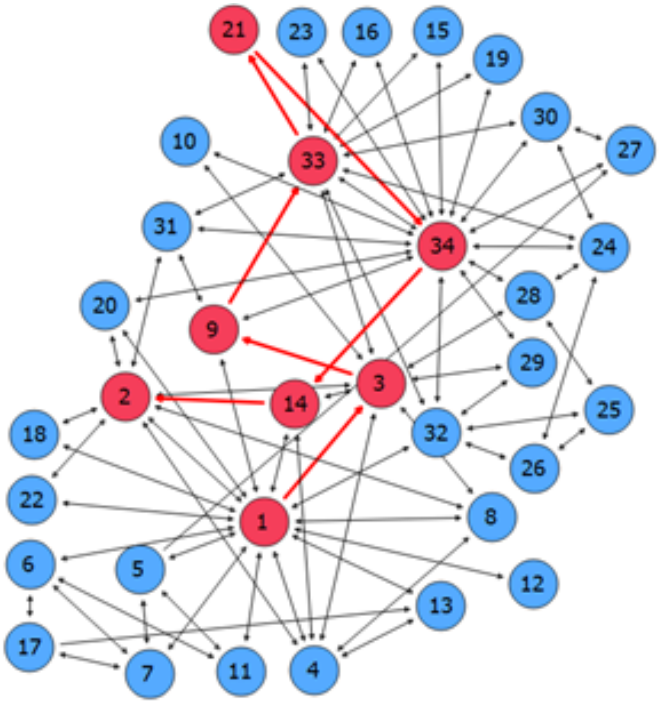

FIGURE 6. **One of the paths from Zachary's Karate Club dataset 1 to 2**

interpretation compared to traditional binary classifications. Unlike many existing methods in the literature, which primarily rely on entity similarity and the transitive nature of trust and are often constrained by incomplete or sparse trust data, the proposed model introduces a more comprehensive and data-driven estimation technique.

In conventional trust inference systems, trust is frequently represented in a dichotomous manner—typically as either "trusted" or "not trusted." However, such simplifications fail to capture the complex and dynamic nature of trust relationships in real-world social structures. In contrast, the method introduced in this research expresses trust on a numerical scale, allowing for the gradation of trustworthiness and enabling a more precise analysis of inter-entity relationships. Again, in the last case, the most trustable and untrustable entities are determined in the network, where the findings are evaluated and analyzed in general.

The application results demonstrate the advantages of our approach over the currently applied methods, both in terms of the discovery of trusted entities the avoidability of untrusted entities. In this way, it is possible to establish new connections with individuals determined as trustable, to take necessary precautions for entities, structures, actors, personnel, etc., identified as untrustable or low trust rate and thus to ensure the security of the network. In the future, we will work to reduce the time cost of the algorithm that we can find confidence rates without data loss.

REFERENCES

[1] E. K. Köse, A. Karci, Sosyal ağlarda güvenilir ve güvenilmez bireylerin tespit edilmesi, Computer Science (Special) (2021) 341–346.

[2] E. Karadeniz, M. M. TEMEL, A. KARCI, Prediction of collaboration between universities of turkey, in: 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), IEEE, 2018, pp. 1–4.

[3] M. U. Demirci, P. Karagoz, Trust modeling in recommendation: Explicit and implicit trust model compatibility and explicit trust prediction, in: Proceedings of the 13th International Conference on Management of Digital EcoSystems, 2021, pp. 8–14.

[4] S. Hamdi, A. L. Gancarski, A. Bouzeghoub, S. B. Yahia, Tison: Trust inference in trust-oriented social networks, ACM Transactions on Information Systems (TOIS) 34 (3) (2016) 1–32.

[5] J. A. Golbeck, Computing and applying trust in web-based social networks, University of Maryland, College Park, 2005.

[6] C.-Y. Lin, N. Cao, S. X. Liu, S. Papadimitriou, J. Sun, X. Yan, Smallblue: Social network analysis for expertise search and collective intelligence, in: 2009 IEEE 25th International Conference on Data Engineering, IEEE, 2009, pp. 1483–1486.

[7] P. Massa, P. Avesani, Trust metrics on controversial users: Balancing between tyranny of the majority, International Journal on Semantic Web and Information Systems (IJSWIS) 3 (1) (2007) 39–64.

[8] H.-K. Oh, J.-W. Kim, S.-W. Kim, K. Lee, A unified framework of trust prediction based on message passing, Cluster Computing 22 (2019) 2049–2061.

[9] A. Josang, R. Hayward, S. Pope, Trust network analysis with subjective logic, in: Conference Proceedings of the Twenty-Ninth Australasian Computer Science Conference (ACSW 2006), Australian Computer Society, 2006, pp. 85–94.

[10] R. Guha, R. Kumar, P. Raghavan, A. Tomkins, Propagation of trust and distrust, in: Proceedings of the 13th international conference on World Wide Web, 2004, pp. 403–412.

[11] W. Jiang, G. Wang, J. Wu, Generating trusted graphs for trust evaluation in online social networks, Future generation computer systems 31 (2014) 48–58.

[12] N. Fatehi, H. S. Shahhoseini, J. Wei, C.-T. Chang, An automata algorithm for generating trusted graphs in online social networks, Applied Soft Computing 118 (2022) 108475.

[13] R. Goyal, A. K. Upadhyay, S. Sharma, P. K. Mishra, Analysis of predicting trust in complex online social networks, Materials Today: Proceedings 29 (2020) 573–580.

[14] S. M. Ghafari, A. Beheshti, A. Joshi, C. Paris, A. Mahmood, S. Yakhchi, M. A. Orgun, A survey on trust prediction in online social networks, IEEE Access 8 (2020) 144292–144309.

[15] Z. Htun, P. P. Tar, A trust-aware recommender system based on implicit trust extraction, Int. J. Innov. Eng. Technol.(IJIET) Technol.(IJIET) 2 (2013) 271–276.

[16] Y. Liu, C. Liang, F. Chiclana, J. Wu, A knowledge coverage-based trust propagation for recommendation mechanism in social network group decision making, Applied Soft Computing 101 (2021) 107005.

[17] H. Mayadunna, L. Rupasinghe, A trust evaluation model for online social networks, in: 2018 National Information Technology Conference (NITC), IEEE, 2018, pp. 1–6.

[18] T. Wu, R. Zhang, X. Liu, F. Liu, Y. Ding, A social commerce purchasing decision model with trust network and item review information, Knowledge-Based Systems 235 (2022) 107628.

[19] M. Ghavipour, M. R. Meybodi, Stochastic trust network enriched by similarity relations to enhance trust-aware recommendations, Applied Intelligence 49 (2019) 435–448.

[20] D. Canturk, P. Karagoz, S.-W. Kim, I. H. Toroslu, Trust-aware location recommendation in location-based social networks: A graph-based approach, Expert Systems with Applications 213 (2023) 119048.

[21] G. Jethava, U. P. Rao, A novel trust prediction approach for online social networks based on multifaceted feature similarity, Cluster Computing 25 (6) (2022) 3829–3843.

[22] B. Yang, Y. Lei, J. Liu, W. Li, Social collaborative filtering by trust, IEEE transactions on pattern analysis and machine intelligence 39 (8) (2016) 1633–1647.
[23] M. Kutay, S. Z. Dicle, M. U. Çağlayan, Çizge tabanlı güven modellenmesi, XIV. Akademik Bilişim Konferansı (2012).
[24] M. Ghavipour, M. R. Meybodi, Trust propagation algorithm based on learning automata for inferring local trust in online social networks, Knowledge-Based Systems 143 (2018) 307–316.
[25] N. Biggs, E. K. Lloyd, R. J. Wilson, Graph Theory, 1736-1936, Oxford University Press, 1986.
[26] I. Herman, G. Melançon, M. S. Marshall, Graph visualization and navigation in information visualization: A survey, IEEE Transactions on visualization and computer graphics 6 (1) (2002) 24–43.
[27] J. Chhugani, N. Satish, C. Kim, J. Sewall, P. Dubey, Fast and efficient graph traversal algorithm for cpus: Maximizing single-node efficiency, in: 2012 IEEE 26th International Parallel and Distributed Processing Symposium, IEEE, 2012, pp. 378–389.
[28] S. Even, Graph algorithms, Cambridge University Press, 2011.
[29] K. İnce, A. Karcı, Modelling and statistical analysis of academic collaborations as a new graph type, Journal of the Faculty of Engineering and Architecture of Gazi University 34 (1) (2019) 439–459.
[30] S. Fortunato, Community detection in graphs, Physics reports 486 (3-5) (2010) 75–174.
[31] G. Chartrand, Introductory graph theory, Courier Corporation, 2012.
[32] K. Akilal, H. Slimani, M. Omar, A very fast and robust trust inference algorithm in weighted signed social networks using controversy, eclecticism, and reciprocity, Computers & Security 83 (2019) 68–78.