Identifying the Top-\textit{k} Influential Spreaders in Social Networks: A Survey and Experimental Evaluation

KAMAL TAHA, (Senior Member, IEEE)
Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi, United Arab Emirates
e-mail: kamal.taha@ku.ac.ae

\textbf{ABSTRACT} Identifying the influential and spreader nodes in complex networks solves many types of complex scientific problems. In social networks, identifying the influential individuals can be useful for structuring techniques that accelerate or hinder information propagation. Each node in the network has unique characteristics that reflect its importance. These characteristics are used by researchers to design many different centrality algorithms. Unfortunately, current survey papers categorize these algorithms into broad classes and do not draw distinguishable boundaries among the specific techniques adopted by them. This can result in misclassifying unrelated algorithms into the same analysis category. To overcome this, we introduce a methodology-based taxonomy for classifying the algorithms that identify top-\textit{k} influential spreaders into hierarchically nested, specific, and fine-grained categories. We survey 184 papers and discuss their algorithms, which fall under 26 specific techniques. Our methodological taxonomy classifies the algorithms hierarchically into the following manner: Analysis type \textarrow analysis scope \textarrow analysis approach \textarrow analysis category \textarrow analysis sub-category \textarrow analysis specific technique. We introduce in this paper a comprehensive survey, review, and experimental evaluation of the recent and state-of-the-art algorithms that identify the top-\textit{k} and influential spreader nodes in social networks.

\textbf{INDEX TERMS} Social networks, top-\textit{k} nodes, influential nodes, centrality measures, influence maximization.

\section{I. INTRODUCTION}

Online social networks (OSNs) are the platforms for billions of worldwide users’ dynamic social interaction. It is considered the most widespread phenomenon characterizing modern society. These social interactions have generated a gigantic amount of data that can be used for gaining insight into human behavioral patterns [139]. They also provide a means for propagating information and maximizing influence in modern society. This exponential explosion of social interaction data has made it almost impossible and impractical to analyze the data to deduce useful information due to resource constraints. As an alternative, an idea has been materialized for analyzing the data by exploring the networks representing it to identify its top-\textit{k} influential nodes.

It has been demonstrated and proven that some people are more influential than others [67]. Similarly, it has been demonstrated that only a few influential individuals are capable of disseminating information in such a way that the opinion of a large population is shaped and changed based on it [67]. Identifying the influential users in a social network or the influential nodes in a network, in general, solve many different types of complex scientific problems [3], [4]. That is why the research area of identifying influential nodes has attracted considerable attention.

Identifying the most influential spreaders can be used for reaching a maximum spreading ability. Influential nodes in complex networks have the most spreading ability. Thus, there is a high correlation between the centrality of a node and its Influence Maximization (IM). The correlation between nodal centrality and influence spread is investigated in many works [90], [14], [132], [165]. Identifying influential nodes
can be useful for planning and structuring techniques that accelerate the information propagation in marketing applications [121, 171, 163] or hinder the propagation of unwanted information [64, 21, 146, 135]. Targeting the influential individuals of a social community can help companies in viral marketing to expand their business potential and to promote their products by triggering a cascade of influence in the community [133]. It can also help in studying behavior patterns that contribute to the spread and outbreak of a disease. Planning for an anticipated epidemiological outbreak requires studying the behavior patterns of the key players that may lead to accelerating the disease spread [11, 133, 151].

We introduce in this paper a comprehensive survey, review, and experimental evaluation of the recent and state-of-the-art algorithms for identifying the top-\(k\) influential spreaders in social networks. We introduce a methodology-based taxonomy for classifying the algorithms into hierarchically nested, specific, and fine-grained categories. We surveyed 184 papers and discuss their algorithms, which fall under 26 specific techniques. For each different technique, we performed the following: (1) searched reputable publishers, such as IEEE and ACM, for papers reporting algorithms that employ the technique, (2) ranked the papers based on their degree of state of the art and recency, and (3) selected the top ones, whose overall content have sufficient variety of information about the technique.

**A. MOTIVATION AND KEY CONTRIBUTIONS**

Current survey papers on the topic categorize algorithms into broad classes. Unfortunately, they do not draw distinguishable boundaries among the specific techniques adopted by these algorithms. This can result in misclassifying unrelated algorithms into the same analysis category. It can also lead to incorrectly assessing the quality and qualitative behaviors of algorithms adopting different techniques using the same metrics. This problem can be overcome if these algorithms are categorized into fine-grained classes. Also, some of these survey papers categorize algorithms without regards of their specific adopted techniques.

To overcome the above limitations of current survey papers on the topic, we introduce in this paper a methodological taxonomy that classifies algorithms hierarchically into fine-grained categories and 26 specific techniques. The methodological taxonomy classifies the algorithms hierarchically into the following manner: Analysis type \(\rightarrow\) analysis scope \(\rightarrow\) analysis approach \(\rightarrow\) analysis category \(\rightarrow\) analysis sub-category \(\rightarrow\) analysis specific technique. To the best of our knowledge, this is the first survey paper that classifies the algorithms for the identification of top-\(k\) influential spreaders using the above categorization fashion.

We provide a comprehensive survey on the different algorithms that employ the same analysis specific technique, the different techniques that fall under the same analysis sub-category/category, the different analysis categories that fall under the same analysis approach, the different analysis approaches that fall under the same analysis scope, and the different analysis scopes that fall under the same analysis type. We experimentally compared and ranked the following:

1) The different algorithms that employ the same specific analysis technique.
2) The different techniques that fall under the same analysis sub-category/category.
3) The different analysis categories that fall under the same analysis approach.
4) The different analysis approaches that fall under the same analysis scope.
5) The different scopes that fall under the same analysis type.
6) The different analysis types.

Our methodology-based taxonomy can help a researcher to gain insight into the following: (1) the specific analysis technique, analysis category/sub-category, analysis approach, analysis scope, and analysis type under which the algorithm proposed by the researcher falls, (2) the advantages and limitations of the specific technique, analysis category, analysis approach, analysis scope, and analysis type under which the algorithm proposed by the researcher falls, and (3) the algorithms and techniques in the literature most comparable to the ones proposed by the researcher.

**B. CURRENT SURVEY PAPERS ON THE TOPIC**

Razis et al. [115] presented a review on online social influence and the role of social semantics in online social networks (OSNs) based on social authority, diffusion, and topology. The authors investigated current approaches for predicting online social influence and the methodologies used for measuring influence. They classified the current approaches into four topics: information flow and influence, influence metrics, network properties, and applications. They classified social semantics roles in the qualitative assessment of viral user-generated content into three topics: community detection, social matching, and social modeling.

Bian et al. [12] reviewed existing literature on top-\(k\) nodes identification. The authors classified current methods for identifying top-\(k\) influential nodes into two broad categories: top-\(k\) influential nodes and top-\(k\) significant nodes. They classified top-\(k\) influential nodes methods as follows: (1) greedy (improved greedy, simple greedy, new greedy), (2) centrality measure (four selection strategies, semi-local centrality measure, evidential centrality measure), (3) network content, (4) topic, and (5) dynamic network.

Dey et al. [26] compared seven centrality measures: eigenvector, node degree, closeness, PageRank, clustering coefficients, betweenness, and k-core. They employed four information propagation methods for the comparison: random walk, forest fire, breadth-first search, and susceptible-infected-removed. They measured the similarity between the results detected by the centrality measures with the corresponding ones derived by the LargeStar-SmallStar algorithm using Twitter Stream Data.
TABLE 1. Comparing related surveys in terms of ten criteria.

| Information Flow and Influence | Social Matching | Considered Metrics | Targeted Applications | Covered Topics | Range of Social Networks | Application Domain | Considered State of Social Networks | Approaches Enable Information Retrieval | Adopted Hierarchical Methodological Taxonomy |
|--------------------------------|----------------|--------------------|-----------------------|----------------|-------------------------|-------------------|-----------------------------|-----------------------------------|----------------------------------|
| Propagation-oriented approaches | Diffusion-oriented approaches | User-oriented approaches | Topics and event-oriented approaches | Network-based metrics | Influencer ranking applications | Recommendations | Top-k nodes identification | Significant nodes identification | Community detection |
| [13] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| [26] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| [14] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| [113] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| [8] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| [69] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| [121] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| [116] | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| This article | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

Bhavnani et al. [13] provided a systematic analysis of the methods that employ Gaussian kernel, grey wolf optimization, Gaussian kernel, and bipartite networks for identifying the influential nodes in a network. They reviewed the grey wolf optimization methods, which employ wolves-like parameters for achieving an optimal solution. The authors provided a systematic review on the methods that employ Gaussian kernel, which use negative and positive signs to indicate whether or not the user is influential. They surveyed the methods that use SVDA-rank algorithms for identifying the influential nodes in bipartite networks.

Al-Garadi et al. [7] analyzed the methods that identify the influential nodes in OSNs. They classified these methods based on their working principles, as follows: (1) short path–based measures (betweenness centrality, closeness centrality, Katz algorithm), (2) local measures, (3) machine-learning algorithms, (4) coreness-based measure, and (5) iterative calculation–based measures (LeaderRank, PageRank-like algorithms, eigenvector centrality, TunkRank, TwitterRank, TURank). They discussed the deviations and commonalities of these methods based on adopted metrics, types of networks, and evaluation models.

Kumar et al. [70] provided a systematic review on the metrics and the factors that impact user influence in OSNs. They classified these metrics according to the following criteria: (1) neighborhood attributes (number of influencers, exposure to indirect and direct influence), (2) decay and locality influence, (3) structural diversity that estimates a community’s activity, (4) cascade-based criteria (messages’ size and path length), (5) temporal measures, and (6) metadata existence.

Singer [121] provided a systematic review on adaptive seeding-based methods for the influence maximization problem. The author reviewed propagation-oriented methods based on information flow and influence for identifying influential nodes and influence maximization.

Riquelme and Gonzalez-Cantergian [118] provided a systematic review on user influence measurements on Twitter. The authors review includes the measurements of users’ influences, activities, and popularities on Twitter.

In Table 1, we compare the related surveys described above in terms of ten criteria that convey their extent of covering the topic. One of these criteria conveys whether distinguishable boundaries are drawn among the techniques reviewed in the survey, which is a key objective in this work.

C. STRUCTURE OF THE PAPER

The rest of this paper is organized as follows. Section II provides preliminaries and basic concepts of social networks and influential nodes identification. Section III surveys the algorithms that employ adaptive analysis using the local topology. Section IV surveys the algorithms that employ adaptive analysis using the global topology. Section V surveys the algorithms that employ static analysis using the local topology. Section VI surveys the algorithms that employ static analysis using the global topology. Section VII surveys the algorithms that employ adaptive content-based analysis. Section VIII surveys the algorithms that employ static content-based analysis. Section IX presents the empirical experiments that evaluate and compare the different analysis types, scopes, approaches, categories, and techniques. Section X presents our conclusions.
D. OUR PROPOSED METHODOLOGY-BASED TAXONOMY

Fig. 1 shows our proposed methodology-based taxonomy, which classifies the algorithms hierarchically as follows:

II. BASIC CONCEPTS

Weighted and un-weighted social network: In this survey, we denote a social network to be $N = (V, E)$, where $V$ represents a (finite) set of nodes (depicting individuals) $\{v_1, v_2, \ldots\}$, and $E$ represents a (finite) set of edges depicting the social interactions between the individuals $\{e_1, e_2, \ldots\}$. In a weighted social network, each edge $(u, v)$ has a numeric weight $w(u, v)$ with $w(u, v) > 0$ iff each edge $(u, v) \in E$. In an unweighted social network, edges are not associated.
with weights. In other words, an unweighted social network can be considered as a weighted social network where every edge has weight 1.

**Undirected social network:** It is a social network that consists of a finite set \( V \) of nodes and a finite set \( E \) of edge, where each edge is associated with an unordered pair of nodes. That is, if an edge \((u, v) \in E\) exists between two nodes \(u \) and \(v \in V\), the edge \((v, u) \in E\) also exists.

**Static social network and dynamic social network:** A dynamic network depicts an evolving network with transient interactions between its nodes. Transient nodes and/or links between the nodes are kept being added to the network. A static network has fixed information, structure, and topology. It can represent a snapshot of a dynamic network at a certain time.

**Degree of a node:** The degree of a node \( v \) in a network is the total number of neighboring nodes of \( v \).

**In-degree of a node:** The in-degree of a node \( v \) is the total number of nodes that have directed edges towards \( v \).

**Out-degree of a node:** The out-degree of a node \( v \) is the total number of nodes that have directed edges from \( v \).

**Path between two nodes:** We call \( n_1, n_2, \ldots, n_k \) a path between nodes \( n_1 \) and \( n_k \), if \( \forall j, n_j \) is linked to \( n_{j+1} \), where \( j \in (1, k - 1) \). A node in the path cannot be revisited.

**Walk between two nodes:** We call \( n_1, n_2, \ldots, n_k \) a walk between nodes \( n_1 \) and \( n_k \), if \( \forall j, n_j \) is linked to \( n_{j+1} \), where \( j \in (1, k - 1) \). The distance of this walk is \( k - 1 \).

**Distance between two nodes:** The distance between two nodes \( u \) and \( v \) is the minimum number of edges in any path between \( u \) and \( v \). That is, it is the number of edges in the shortest path between \( u \) and \( v \).

**Average distance in a network:** It is the average number of edges for traversing between two nodes in a network. It is the average of the distance of the shortest path between every two nodes in a network, and it is defined as shown in Equation 1:

\[
\frac{1}{n(n-1)} \sum_{d(u, v) \forall u \neq v} \sum_{i j} \in E \in \sigma_{i j}(v) \sum_{(u, v) \in E} \sigma_{i j} \frac{1}{(n-1)(n-2)}/2 \tag{2}
\]

- \( \sigma_{i j} : \) Number of shortest paths between nodes \( i \) and \( j \) that pass through node \( v \).
- \( \sigma_{i j} : \) Overall number of shortest paths between nodes \( i \) and \( j \).

**Edge betweenness centrality:** The betweenness centrality of an edge \( e \) is defined as shown in Equation 3:

\[
EB(e) = \sum_{i \in V} \sum_{j \in V} \sigma_{i j}(e) \sigma_{i j} \tag{3}
\]

- \( V \): The set of nodes.
- \( \sigma_{i j} : \) Number of shortest paths between any two nodes \( i \) and \( j \).
- \( \sigma_{i j}(e) : \) Number of shortest paths between \( i \) and \( j \) that pass through \( e \).

**Node degree centrality:** The degree of a node \( v \) is the number of nodes that are connected to \( v \). That is, it is the number of neighbours \( v \) has.

**Node closeness centrality:** The closeness centrality of a node \( v \) is the average distance of the shortest paths between \( v \) and all reachable nodes of \( v \) in the network.

**Node strength:** The strength of a node \( v \) in a weighted network is the sum of the weights of all the edges that \( v \) has. The in-strength of node \( v \) is the sum of weights of all the edges that have directed edges towards \( v \). The out-strength of node \( v \) is the sum of weights of all the edges, which \( v \) has directed edges towards them.

### III. TOPOLOGY-BASED ADAPTIVE METHODOLOGY USING LOCAL SCOPE

These algorithms use available local information to adaptively explore the locality property of nodes and perform local updates accordingly. They employ adaptive seeding strategies based on the topology of the local network and quantify the influence degrees of the seed nodes.

#### A. CORE DECOMPOSITION-BASED APPROACH

These algorithms are based on the observation that nodes with core positions tend to be influential in the network. They solve the problem of core-decomposition maintenance update adaptively by exploiting the locality property of core (i.e., decomposition with local updates). They tend to update the number of core nodes according to the insertion or/deletion of edges or/and nodes based on the available local information. Considering that the network is constantly changing as nodes and edges are deleted or inserted, these
algorithms tend to renew nodes’ core number while avoiding coreness re-computation.

1) CORE DISTRIBUTION ANALYSIS TECHNIQUE
Taha [148] proposed a method that detects communities and their influential nodes by optimizing, among others, the partitions separability of communities. The method does so by selecting well-distributed core nodes. First, it ranks nodes according to their influences. Then, it selects well-distributed top-ranked ones to be the core seeds. Then, it quantifies the degrees of relationship between influential edges and the seed nodes. It considers a node \( u \) to belong to a community \( c \), if each edge in the shortest path from \( u \) to the most influential seed node \( v \) of \( c \) has a high degree of relationship with \( v \). The time complexity of the method is \( O(nm) \), where \( n \) and \( m \) are the number of nodes and edges, respectively.

Chen et al. [17] developed an IM algorithm that optimizes the distribution and spacing between core seeds through seed colonization. The authors defined IM as such: given a number the distribution and spacing between core seeds through seed respectively.

Chang et al. [19] proposed a method that identifies the influential nodes in a multi-layer mapping service network. The set of influential nodes contains nodes that are well-distributed, community leaders, and inner-relational ones. These nodes are selected according to quantitative Cost-Failure and Service. To identify leader nodes, core degree is employed to determine influential communities. Then, leader nodes are identified in these communities. The degree of influence of a node on its neighborhood reflects its strength.

Li et al. [75] introduced a community model based on the concept of \( k \)-core. The model can identify the top-\( r \) \( k \)-influential nodes and their communities. It defines a community as a connected subgraph, whose nodes have degrees at least \( k \) (\( k \) determines a community’s cohesiveness). A \( k \)-influential community should not be included in a super-\( k \)-influential one. This is achieved by making sure the selected seed nodes are well distributed. The time complexity of the model is \( O(Nkm) \), where \( N_k \) is the number of \( k \) influential communities and \( m \) is the number of edges.

2) CORE CONNECTIVITY ANALYSIS CATEGORY
2.1) SHELL/DEGREE-BASED ANALYSIS SUB-CATEGORY
2.1.1) MIXED DEGREE ANALYSIS TECHNIQUE
Zhang et al. [181] proposed a method based on multiple local attributes centrality and information entropy for identifying influential nodes. Local attributes centrality is achieved by combining clustering coefficient and degree measure. Neighbors that are one-hop neighbors are added to the search. Both direct and indirect influences of a node are considered. Clustering coefficient and degree constitute the direct influence of a node. Two-hop clustering coefficient and two-hop degree constitute the indirect influence of a node. The final centrality of a node is the sum of these measures. The two-hop degree of a node \( x \) (\( \text{th}_{k} \)) is shown in Equation 4. The two-hop clustering coefficient of node \( x \) (\( \text{th}_{k} \)) is defined as shown in Equation 5. The time complexity of the method is \( O(n) \), where \( n \) is the number of nodes.

\[
\text{th} = \text{NN}_x - \text{NS}_x - 1 \quad (4)
\]

- \( \text{NN}_x \): The number of two-hop neighbors of node \( x \).
- \( \text{NS}_x \): The number of same neighbors.

\[
\text{th}_{k} = \sum_{j \in N(x)} \text{cl}_{y} \quad (5)
\]

- \( \text{cl}_y \): The clustering coefficient of node \( y \).
- \( N(x) \): The neighbors of node \( x \).

Namithra et al. [106] proposed a method that combines the degrees of neighbors and \( k \)-shell decomposition for identifying influential nodes. A node’s centrality is influenced by its degree and \( k \)-shell and the degree and \( k \)-shell of its neighbors. The influence of a node is quantified as follows. Nodes with degrees \( k = i \ (i = 1, 2, \ldots) \) are removed from the network and are assigned the \( k \)-shell index of \( ks = i \). The network is updated accordingly. This procedure is repeated iteratively until all nodes are removed from the network.

Zeng et al. [180] proposed a modified version of the \( k \)-shell method that employs a mixed degree decomposition mechanism. It considers both exhausted degree and residual degree. Nodes are iteratively removed based on the mixed degree procedure. The removal of nodes continues until \( k_m \) of the remaining nodes become larger than \( M \), where \( m \) is the number of residual and exhausted nodes, and \( M \) are the nodes with the smallest \( k_m \). Hou et al. [44] introduced a method for identifying influential nodes using the concept of all-around score. A node’s importance is determined based on mixed degree decomposition, \( k \)-shell index, and betweenness centrality. Then, an all-around distance is presented for quantifying the influences of nodes.

2.1.2) EDGE-WEIGHTED DEGREE ANALYSIS TECHNIQUE
Zhou et al. [190] proposed a method that employs weighted core decomposition in a weighted graph for identifying influential nodes. The core maintenance approach proposed by the authors is based on the observation that only a few nodes keep updating their core numbers in response to the insertion or deletion of a single edge. The algorithm splits edges to k-favorable edges and multiple k-edges. Taha et al. [133] proposed a method for detecting influential nodes and the smallest cross-communities after analyzing the hierarchical relationships among the social
profiles of these communities. The method is based on core decomposition and edge-weighted degrees. It first depicts an edge-weighted social network using the k-core model (in terms of the k-clique model). The cross-users of communities $S$ and $\bar{S}$ are identified by the influence weight of the Association Edge connecting $S$ and $\bar{S}$. This weight is computed based, in part, on the weighted degrees of all the edges connected to the end points of the Association Edge.

Antonios et al. [6] proposed a generalized weighted k-shell decomposition method for detecting influential nodes and computing k-shell in weighted networks. It is based on the degree and weight of a network (the degrees of nodes and the weights of their links). The method ranks nodes by putting the nodes with high spreading in shells close to the core.

Wei et al. [164] introduced an edge-weighting k-shell decomposition method that considers both edge-weight and node-degree. An edge’s weight is the sum of the degrees at its end points. The method employs a tuning parameter in the range of 0-1. The 0 parameter implies that edge-weights are given complete importance. The 1 parameter implies that nodes’ degrees are given complete importance.

Liu et al. [71] proposed a method that improves the accuracy of k-shell using edge-weighted degree and removing redundant links. It filters out redundant links with low diffusion power before applying k-shell decomposition.

2.1.3) SEMI-LOCAL CENTRALITY ANALYSIS TECHNIQUE

Zhang et al. [176] proposed a centrality measure that detects influential nodes according to the influences of their neighboring nodes as well as the community to which they belong. It is based on the notion that the more important a node is, the more it influences its neighboring nodes and the entire network. A node’s influence in the network is the sum of the community influence of all its neighbors. Specifically, a node’s influence is the average k-shell value of the community containing the node’s neighbors.

Tatti [142] proposed a method for density-friendly graph decomposition, a modification of k-core decomposition. It defines a locally-dense subgraph as one whose density does not change when nodes are added and deleted. Two algorithms are introduced to determine such a decomposition. The first one employs a minimum cut procedure. The second one orders nodes by iteratively deleting a node that has the smallest degree before detecting the densest subgraph. The time complexity of the method is $O(n^2 m)$, where $n$ and $m$ are the number of nodes and edges, respectively.

Bae et al. [8] introduced a k-shell decomposition method that considers both degree and weight of a network. It computes the shortest distance between each node and the nodes whose maximum k-core values need to be updated when edges are removed and inserted during the decomposition. It locally finds the root node’s subcore before applying the procedure. The core degree of a node is the number of its neighbors whose k value is greater than the root’s core value.

2.1.4) DISTANCE-BASED ANALYSIS TECHNIQUE

Nan et al. [103] proposed a k-shell multi-attribute ranking method based on nodes’ positions and their neighborhoods’ local information. It utilizes the iterative information during the decomposition procedure. It employs a Sigmod function to process the iteration information. Then, it combines the position index’s distance and shell value to identify the position attribute. Information entropy weighting is employed to weight the neighbor and attributes. The time complexity of the method is $O(n^2)$, where $n$ is the number of nodes.

Li and Sun [79] proposed a ranking method based on k-shell decomposition that considers the local structure and distances. Nodes are labelled with indices, $ks$, representing their location distance in the network. Nodes that have low $ks$ are located at the periphery while the ones that have high $ks$ are located in the center of the network. First, the $ki$ of each node is calculated. Then, nodes with the lowest $ki$ are removed and given the k-shell. This process is repeated until each node is assigned to a k-shell.

Jian-Guo et al. [61] presented an improved k-shell method that ranks nodes to estimate their spreading influence. It computes the shortest distance between each node and the nodes that have the largest k-core values. The idea is based on the intuition that nodes that have the same k-core values and are located close to the core of the network have greater spreading influences.

2.2) S-SHELL ANALYSIS TECHNIQUE

Li and Sun [79] proposed a method to rank node influence based on the s-shell dynamic decomposition scheme and the local structure’s change during decomposition. It ranks nodes based on the $s$ and assigns them to the corresponding s-shell. During the process, nodes with the lowest $s$ are removed and assigned to the s-shell. This procedure is repeated recursively until all nodes are assigned to the s-shell. At each step of the decomposition, the network’s local characteristics change as nodes are removed. Nodes are ranked based on $s$.

Liu et al. [89] introduced a method that ranks spreaders and identifies the influential ones by employing the s-shell decomposition process. The weighted coreness or the s-shell index of a node is computed as follows. First, the strength weight $si$ is computed for each node. Then, the nodes with a strength less than $sm$ are removed along with their links repeatedly until the network is empty. A node’s s-shell index
and order of removal represents its hierarchy in the influence spreading. It can be considered as its weighted coreness.

Wang et al. [161] presented a refined shell index and a reject neighbors-based algorithm for identifying key nodes. It employs the concept of rejecting neighbors to filter out seed nodes which decentralize node selection. First, the number of shell layers $s$ is set to 1. Then, nodes with a degree value of $s$ are iteratively removed from the network along with their links. These nodes are the s-shell layer. The time complexity of the algorithm is $O(n)$, where $n$ is the number of nodes.

B. ADAPTIVE SEEDING ANALYSIS APPROACH

These algorithms quantify the influence degree of seed nodes adaptively based on the influential structural changes in the network observed from previous iterations. They employ adaptive seeding strategies based on the topology of the local network and the observed influence feedback information collected from prior rounds. That is, they employ an adaptive sequential decision making procedure to select the seed nodes based on the evolution patterns of influential seed nodes.

1) SEED SELECTION ORDER ANALYSIS TECHNIQUE

Yalavarthi and Khan [171] proposed a dynamic framework for influence maximization. As the communication patterns and structure of the network change, local updates are performed to adjust the top-k influencers. Only seed nodes impacted by the dynamic updates are iteratively selected. The seed nodes are updated by adding sub-routines to the static influence maximization algorithm. The infection propagation concludes if old seed nodes are determined and discarded from the seed set and the new seed nodes are selected. The time complexity of the method is $O(n \left( |E| + |V| \log |V|^2 \right))$, where $V$ and $E$ are the number of nodes and edges, respectively.

Enyu et al. [28] proposed a re-ranking algorithm to detect the influential spreaders in a network based on an iterative seed selection strategy. Using the information spreading probability function, a seed node with the largest score is repeatedly selected. Using the local paths of a selected node, the scores of other nodes are reduced. The probability of information spreading from node $v_i$ to node $v_j$ is defined as shown in Equation 6. The time complexity of the algorithm is shown below, where $n$, $m$, and $r$ are the number of nodes, edges, and initial infected nodes, respectively.

$$O \left( r \cdot \left( \frac{m}{n} \right)^2 \right)$$

$$INF(v_i, v_j) = 1 - \prod_{l=1}^{L} (1 - \alpha^l)^{P_{ij}^{(l)}}$$  \hspace{1cm} (6)

- $\alpha$: Probability of spreading information to a neighboring node.
- $P_{ij}^{(l)}$: Number of paths from node $v_i$ to node $v_j$ with length $l$.

Goeppeppert et al. [38] proposed an update mechanism for iterative influence maximization. Each node is iteratively assigned a score that reflects its influence on the nodes located within a certain hop radius. The score is the sum of the node’s neighbors’ weighted scores. When selecting the next seed node, its score is recalculated for the third hop. The method performs updates within two hops of a selected seed node. It also performs a forward update when a new seed node is added. The portion of the scores of the neighbors of a selected neighbor whose scores are equal to the edge probability is iteratively removed. The time complexity of the method is $O(k \cdot h \left| E \right|)$, where $E$, $h$, and $k$ are the number of edges, hop neighbors, and seed nodes, respectively.

Jiachen et al. [60] proposed temporal-network surveillance strategies based on the friendship paradox theory using the local information of individuals and their surroundings. The first strategy selects the neighbor that has the largest number of contracts of a randomly selected node as a sentinel. The other strategy chooses the most recent contact of a randomly selected node’s neighbor as a sentinel.

2) INFORMATION DIFFUSION MODELING ANALYSIS CATEGORY

2.1) VOTER MODEL ANALYSIS TECHNIQUE

He et al. [49] proposed an opinion maximization method that maximizes the follow rate of products. It employs the weighted voter model to dynamically produce opinion series. First, influence power is computed by employing maximum likelihood estimation to estimate the unknown parameter. Then, influence overlapping is eliminated by deleting the influence overlapping between nodes and the selected seed nodes. The weighted voter model updates the opinion series.

Rawai and Khan [117] formulated a method that employs the voter model for simulating the opinion diffusion of two groups with contradicting views. It also identifies the influential seed nodes that maximize the diffusion. The voter model is employed to determine the change of a user’s opinion based on the influences of his neighbors. Each node is given a value between -1 and 1 that reflects its opinion. The time complexity of the method is $O(|E| \cdot \tau)$, where $\tau$ is time stamp and $E$ is the number of edges.

Yu and Zhang [172] proposed a method, which is a generalization of the Majority Vote Model for identifying the influential nodes in social networks under the Coordination Game model. The authors defined the total payoff of a player as the sum of the payoffs he gets from all coordination games with his neighbors. Each node is associated with a threshold and a “local” spreading function. Initially, there is a seed set less than the number of nodes. The spreading process stops when each node becomes a choice node. The time complexity of the method is $O(n^2)$, where $n$ is the number of nodes.

2.2) INDEPENDENT CASCADE ANALYSIS TECHNIQUE

Menta and Singh [92] proposed an approximate solution to the $t$-influence maximization problem that causes the seed nodes to change a few of their neighbors to active state using the independent cascade model. The freshly active nodes then
spread the influence to some of their neighbors. This process continues. A greedy algorithm is employed for the t-influence maximization problem. The method can allocate seeds with maximum total influence and also ensure that no node can decide the overall influence of the network.

Jing et al. [59] have proposed a lightweight hop-based method to address the influence maximization problem using the Independent Cascade model. It focuses on the influence propagation up to a given number of $h$ hops starting from the initial seed set. With one hop, a node can only be activated directly by its inverse neighbors. With two hops, a non-seed node may be activated directly by a seed node or indirectly via a neighbor of a seed node. The time complexity of the method is $O(k (|V| + |E| + \sum_{w \in V} (|I_w| + |N_w|)))$, where $N_w$ is the set of nodes of $w$’s neighbors, $I_w$ is the $w$’s inverse neighbors, $V$ is the set of nodes, $E$ is the set of edges, and $k$ is the number of seed nodes.

Wilder et al. [165] presented an end-to-end agent for influence maximization using a variant of the classical independent cascade model. Activated nodes can influence others. An activated node can adaptively activate its neighbors with a certain probability. This process takes place at time steps in the range $1-T$, where $T$ is a time horizon. The activation process of a node stops when either the time horizon is reached, or its neighbor is influenced.

IV. TOPOLOGY-BASED ADAPTIVE METHODOLOGY USING GLOBAL SCOPE

These algorithms assume that the influences of nodes change according to the structural changes of the network. Under this assumption, the structural changes lead to changes in the functions of nodes. To account for the functional changes, these algorithms employ an adaptive seeding strategy to assess the influences of nodes based on the topological properties of the network. They use this information to learn the seeding pattern. They construct global representations in either a bipartite or a unipartite model.

A. FEEDBACK-BASED MODEL ANALYSIS APPROACH

These algorithms consider the influences of nodes change due to the deletion or addition of edges or nodes. Towards this, these algorithms employ feedback schemes that learn from prior rounds. To account for the functional changes of nodes or edges, most of these algorithms employ an adaptive seeding strategy to assess the influences of nodes based on the outcome feedbacks from past rounds to learn the seeding pattern. Some of them employ a greedy/aggregation policy that uses activated nodes in prior rounds as feedback for identifying the seed set.

1) NODE FEEDBACK-BASED MODEL ANALYSIS CATEGORY

1.1) AGGREGATE-BASED ANALYSIS TECHNIQUE

Maurya et al. [91] employed a graph neural network (GNN) for solving the betweenness and closeness approximation problem and outputting ranking scores for nodes. Nodes aggregate the nodes’ features in their multi-hop neighborhood. The feature aggregation mechanism serves as a feedback scheme that learns the number of nodes that are reachable to a certain node. To approximate the betweenness and closeness measures, the flow of feature information is restricted to edges located on shortest paths. The authors use a ranking loss function to compute predicted scores’ loss of the actual closeness and betweenness scores. For the model score $S_i^{(m)}$ and the actual betweenness score $S_i^{(b)}$ the loss function is defined as shown in Equation 7:

$$\text{Loss}(x, y) = \max(0, -y * (S_i^{(m)} - S_i^{(b)}) + \text{Margin}),$$

$$y = \begin{cases} 1 & \text{if } S_i^{(m)} \text{ should be ranked higher than } S_i^{(b)} \\ -1 & \text{if } S_i^{(b)} \text{ should be ranked higher than } S_i^{(m)} \end{cases}$$

Lellis and Porfiri [77] focused on network synchronization for determining the influential nodes from the time series of experimental nodes’ feedbacks, where each node is plagued by noise. A node’s influence is defined as the extent to which adding noise to the node affects the overall synchronization of the entire network. Using these time series, several feedback inference tasks are undertaken from reconstructing the network topology to discovering hidden nodes. The feedback inference of causal influence does not require tailored experimental manipulations.

Guillaume et al. [34] presented an adaptive greedy strategy for influence maximization, where a realistic feedback model called myopic is considered. The strategy is achieved by maximizing an alternative utility function that considers the aggregate number of active nodes through the time. A modified version of the IC model is used, where a node has several opportunities to influence its neighbors.

Gui et al. [35] proposed an overlapping community detection algorithm. It employs label propagation and a complete subgraph. The algorithm first searches for a complete subgraph and allocates unique labels for each subgraph. Then it updates each label by observing its adjacent nodes’ labels. Specifically, the method aggregates all nodes whose degree is not zero and marks their neighbors. It joins all the new nodes that form the complete subgraph until the nodes that do not meet the criteria remain. The procedure is repeated until all nodes are marked and the search is ended.

Wang et al. [158] proposed a method that identifies influential nodes based on nodes’ feedback in the time window in propagation tree (e.g., users’ retweeting and comment behaviors). It is based on the incremental influential degree. It aggregates nodes’ feedback in the prior increment states (i.e., time window). The method identifies local influential nodes in the propagation tree. It also identifies global influential nodes for information diffusion. The time complexity of the method is $O(\max w \times n (k \times w^2))$, where $k$ is the number of influential nodes, $n$ is the number of nodes, and $w$ is the number of windows in the graph.
Fushimi et al. [31] proposed a measure for a dynamic network that detects influential structural changes by quantifying the influence degree of each node. Under this assumption, structural changes create influence in node function. The method calculates the influence degree of the whole network by aggregating the functional change of each node. The correlation distance between convergence curves is calculated before and after the change occurs.

1.2) GREEDY-BASED ANALYSIS TECHNIQUE

Guo and Wu [33] proposed an adaptive influence maximization method that attempts to repeatedly activate selected nodes that do not want to be influencers. Expected influence spread is estimated by combining sampling and adaptive greedy techniques. In each iteration, a selected node is determined whether or not it is willing to be a seed. If it is, its influence diffusion feedback is observed. The feedback helps in determining the status of edges going out it and can be reached by the nodes in the current graph structure.

Han et al. [45] proposed a method that identifies the influential nodes in a social network by integrating the greedy mechanism with the nearest neighborhood degree centrality. Let \( S \) be the set of neighborhood nodes for a current optimal temporary seed node. The method selects the node in \( S \) that has the largest degree value. Based on the feedback received from this node, a greedy selection mechanism is employed to replace the current optimal seed node with the selected node. The time complexity of the method is \( O(k^2 \cdot \log k \cdot N \cdot D \cdot g_{\text{max}}) \), where \( k \) is the size of the seed set, \( N \) is the size of particle swarm, \( D \) is the dimension of the problem space, and \( g_{\text{max}} \) is the number of iterations. A node’s neighborhood centrality is defined as in Equation 8:

\[
C_i^u(\theta) = \sum_{j \in \Gamma_i} \theta_j + \sum_{l \in \Gamma_i \cap j} \theta_l + \ldots + \sum_{s \in \Gamma_{\Gamma_i \cup s}} \theta_s \quad (8)
\]

- \( \theta \): Benchmark centrality’s metric.
- \( \Gamma_i \): Node-set of nearest neighbors of node \( i \).

Tong et al. [137] proposed a greedy adaptive seeding strategy by introducing the concept of a seeding pattern to maximize the influence spread in dynamic social networks. Both the seeding pattern and seeding strategy can be adaptively constructed based on the outcomes of the past rounds. The authors studied the adaptive seeding strategies which make seeding decisions step by step according to the observed influence diffusion feedback. In each seeding step of the algorithm, a node is selected to maximize the marginal profit conditioned on the observed events feedback.

Song et al. [124] introduced a method that dynamically identifies the influential nodes maximizing the spread of influence based on the interchange greedy procedure. Influential nodes are detected by analyzing the similarity feedback from prior influential seed set. These nodes replace current influential nodes to enhance the influence coverage. Until the budget is reached, the algorithm keeps repeatedly replacing the nodes by choosing the node that has the largest marginal gain and includes it to the seed set. The time complexity of the method is \( O(kn) \), where \( n \) is the number of nodes and \( k \) is the size of the seed set.

He et al. [47] proposed a greedy approximation method for influence maximization. It identifies the smallest number of influential nodes that can positively influence each node in the network. It considers positive and negative influences.

Kuhlman et al. [62] proposed a method that inhibits undesirable information spread in social networks using a greedy set cover computation. A greedy approach is used for finding a sub-collection containing sets that satisfy the convergence requirements. The time complexity of the method is \( O(|V| + |E|) \), where \( V \) is the number of nodes and \( E \) is the number of edges.

2) EDGE FEEDBACK-BASED MODEL ANALYSIS CATEGORY

2.1) AGGREGATE-BASED ANALYSIS TECHNIQUE

Wang et al. [159] studied the problem of ranking graph elements (e.g., edges, nodes, subgraphs). The authors tackled the problem by finding a set of graph elements such that the ranking result will have the greatest change upon the perturbation/removal of the set of graph elements. A node’s influence is the aggregation of all inbound and outbound edges that connect to the node. And the influence of a subgraph is defined as the aggregation of all edges in the subgraph. The influence for both nodes and subgraphs can be computed based on the edges’ influences. The time complexity of the method is \( O(mn^2) \), where \( n \) is the number of nodes and \( m \) is the number of edges.

Taha [144] proposed an agglomerative-like method that detects cohesive communities in a social network. An association score of each two vertices is computed based on the feedbacks of aggregating the betweenness centralities of the edges connected to these two vertices. The influence score \( S(u) \) of a vertex \( u \) is computed based on the feedback of aggregating the betweenness centralities of the incoming and outgoing edges to and from \( u \). Vertices are ranked based on their betweenness centralities. The time complexity of the method is \( O(n^3) \), where \( n \) is the number of nodes.

Tixier et al. [145] proposed a method inspired by bootstrap aggregating (bagging) that first creates many perturbed versions of a given graph, then applies a node scoring function separately to each graph, and finally aggregates the results back into more robust scores. It identifies spreaders by aggregating node scores computed from multiple perturbed versions of the original network based on the feedbacks of edges’ deletion/addition. A node scoring function is applied to each graph based on the feedbacks of edges’ deletion/addition. Finally, the results are aggregated. The time complexity of the method is \( O(|E| \cdot \log (|V|)) \), where \( E \) is the number of edges and \( V \) is the number of nodes.

Wei et al. [160] introduced a centrality method for identifying influential nodes. Network representation learning is employed for identifying overlapping communities and influential nodes. Let \( w \) be the weight of a node \( u \) with regard to a community \( c \). The larger \( w \), the higher the probability that \( u \) has an edge with the nodes of the community \( c \).
2.2) GREEDY-BASED ANALYSIS TECHNIQUE

Sun et al. [128] proposed an adaptive Multi-Round Influence Maximization method. An advertiser selects seed nodes adaptively based on the previous rounds’ propagation feedbacks. Feedback about changes in edges and nodes are obtained following the propagation. The method employs a greedy policy that uses activated nodes in prior rounds as feedback for identifying the seed set that maximizes the expected marginal gain. For the first round, seed nodes are selected greedily. Then, seed nodes are kept being selected greedily for the subsequent rounds. The time complexity of the method is $O(k^3 T n^2 m \log(nT)/\epsilon^2)$, where $k$ is the number of seed nodes in each round, $T$ is the number of rounds, and $\epsilon$ is the approximation accuracy.

Loukides and Gwadera [78] introduced a method that estimates the impact of edge deletion using PageRank for identifying influential nodes. It uses an approximation algorithm. It employs the aggregate path probability’s gain after choosing an edge in its edge selection criterion to estimate the gained benefit. The benefit is assessed for the decrease in the activation probability of all vulnerable nodes (the ones that have an activation probability higher than a threshold). The time complexity of the method is $O(|E| \cdot |E'| \cdot T_{D})$, where $E'$ is the set of deleted edges and $T_{D}$ is the maximum time needed to compute the deleted edges.

Kimura et al. [64] presented a method that employs a greedy strategy for the problem of average contamination minimization. It minimizes the spread of undesirable information by blocking some links in the network. It identifies the links, whose blocking will minimize the degree of contamination (e.g., undesirable information).

Wu et al. [152] proposed a method that identifies the influential seed users in evolving social networks from multiple rounds. It adopts a greedy algorithm to iteratively and adaptively selects users who will cover a large seed set. It does so by learning from the influence probabilities feedbacks in prior diffusion rounds. It employs the Upper Confidence Bound approach for learning from prior rounds’ diffusions. Feedback gives information about the influenced users who successfully influenced their neighbors through social links.

B. NETWORK-BASED MODEL ANALYSIS APPROACH

These algorithms employ either eigenvector or semi-local centrality measures for identifying the influential nodes based on the topological properties of the network. These algorithms construct global representations in either a bipartite or a unipartite model. Most of the ones that employ Eigenvector Centrality rank the eigenvalues of a matrix. Some of these algorithms employ an adaptive iterative semi-local-based procedure that measures the influences of nodes in certain networked dynamics, by constructing a local network within $k$-hop distance for a node under consideration. They estimate the degree to which influential nodes influence their neighboring nodes (i.e., they quantify the centrality of a node based on how it influences its neighbor nodes). The algorithms that employ eigenvector measure construct a graph’s adjacency matrix eigenvalues.

1) UNIPARTITE-BASED ANALYSIS CATEGORY

1.1) EIGENVECTOR-BASED ANALYSIS TECHNIQUE

Adebayo and Sun [2] proposed an eigenvector centrality measure for detecting the influential nodes in a power network. It utilizes the network topological properties of an electric power system. First, a matrix that captures the information between an electrical interconnection is established. The eigenvector centrality measure is formulated based on the established matrix. Then, the eigenvector corresponding to the maximum eigenvalue is identified.

Kamath and Mahadevi [65] introduced a method for ranking the nodes of a network by analyzing the impact of the deletion of nodes on network connectivity using the energy of graphs. A graph’s energy is the absolute sum of the graph’s adjacency matrix eigenvalues. This absolute sum gives the graph’s energy. The centrality value of a deleted node is given by subtracting the energy of sub-graph from the total energy. Nodes resulted in lower sub-graph energy are highly influential. The time complexity of the algorithms is $O(n^3)$, where $n$ is the number of nodes in the network.

Harigovindan et al. [46] introduced a method that detects the persuasive nodes in a network after employing the Eigen Centrality to measure nodes’ importance. Kosaraju’s algorithm is used to identify strongly connected components. Eigenvector Centrality is employed to identify each node’s degree. The merge sort is used based on nodes’ degrees. Finally, the highest persuasive node is identified.

Moradi et al. [99] introduced a measure, which is denoted as the Eigenratio Sensitivity Index (ESI) for the centrality of nodes for pinning controllability (nodes that have large ESI are drivers’ candidates). ESI is computed based on the Laplacian matrix’s eigenvectors. The eigenvectors correspond to the smallest and highest Laplacian matrix’s eigenvalues.

Li et al. [81] proposed a centrality measure, which expands the Principal Component Centrality (PCC), for weighted social networks. It is based on the leading $P$ eigenvectors of an adjacent matrix. Let $N$ be eigenvalues of a matrix ranked in descending order. Then, the corresponding eigenvectors are $x_1, x_2, \ldots, x_N$. A node’s PCC is its Euclidean distance from the origin in the $P$-dimensional eigenspace.

1.2) SEMI-LOCAL-BASED ANALYSIS TECHNIQUE

Li et al. [82] proposed an adaptive semi-local-information-based method that measures the influences of nodes in certain networked dynamics, in which the links in the subgraph are made up of the second-order neighbors of nodes. These links are given different weights adaptively. A selected node may infect its susceptible neighbors with a certain probability. And in the next step, this node will be recovered and will never
participate in the dynamics with a different probability. The spreading process continues until the network does not have any infected nodes. The time complexity of the method is $O(\sqrt{V} \ k^2)$, where $k$ is the average degree and $V$ is the number of nodes. The adaptive weighted link of node $i$ is defined as shown in Equation 9:

$$awl(i) = \alpha k_i + (1 - \alpha) \left[ \left( \sum_{j \neq k, j < k} R_i + \frac{1}{k_i} \right) \right]$$

where $R_i = \left( \sum_{j \in \Gamma_i, j \neq k} a_{jk} \right) / 2$, $\Gamma_i$ is the set of the nearest neighbor of node $i$, and $\alpha \in [0, 1]$.

Liu et al. [83] developed a method that identifies the influential spreaders in complex networks. It employs a semi-local iterative algorithm. The propagation ability of nodes is assumed to be proportional to its neighbors’ number after a ground node is added to the network. First, the out-degree of nodes is computed. Then, a ground node is included in the network, which bidirectionally connects with all ordinary nodes. Finally, the importance of nodes is successively updated until they reach their steady states.

Niu et al. [102] proposed a semi-local method that identifies influential nodes in dynamic networks. It considers only nodes within $k$-hop distance from a node under consideration. A local network is constructed for each node by getting all nodes within less than $k$-hop from it. The centrality index of a node is the sum of mutual influence values between it and the remaining nodes in the local network. The time complexity of the method is $O(n^3)$, where $n$ is the number of nodes.

Ibrahim et al. [51] presented a centrality measure to quantify the importance of a given node using the concept of lattice formulation of a network, which facilitates the extraction of its cliques and bridges through symmetrical concepts. The measure is based on quantifying the centrality of nodes based on how it influences its neighbor nodes through its cross-cliques while connecting the densely connected regions. It extracts from the lattice the set of concepts that represent cross $k$-cliques and face bridges in the social graph. It calculates the Cross-face centrality measure to identify the key nodes. The time complexity of the method is $O(n \times n_2)$, where $n$ is the number of nodes, and $n_2$ is the number of symmetrical concepts.

Weiyan et al. [162] proposed a semi-local centrality index for identifying the influential nodes in the urban road network. It is used for a weighted road network that integrates the characteristics of the urban roads’ networks. It uses the information of a node’s fourth-order neighbors.

Ahmad et al. [5] proposed a measure for the specification of a node’s spreading influence in a network. It is based on the notion that the spreading influence of a node is great if its neighboring nodes have high degrees. Towards this, the measure considers the neighbors’ degrees. Nodes’ centralities are calculated based on the degrees of their first-order and second-order neighbors.

2) BIPARTITE-BASED ANALYSIS CATEGORY

2.1) EIGENVECTOR-BASED ANALYSIS TECHNIQUE

Song et al. [125] proposed a method for the bipartite synchronization and the convergent behavior of a signed network composed of harmonic oscillators. It determines the converged trajectory using the right and left eigenvectors of the zero eigenvalue of the Laplacian matrix of a signed graph. When the signed graph is structurally balanced, the bipartite synchronization problem for the signed network can be solved only when the time delay is chosen from an interval.

Guo et al. [36] presented a spectral analysis technique for detecting spammers using a user relation graph model based on a bipartite graph built directly from review data. The method focuses on each eigenvector and its neighborhood for testing whether an eigenvector is an abnormal dimension and whether its entries follow a normal distribution. A certain number of eigenvectors with the largest Shapiro-Wilk statistic is selected as the abnormal dimensions.

Jiangxia et al. [56] developed a local-global infomax objective that maximizes the mutual information (MI) among global and local bipartite graph embedding representations. This maximization enables bipartite graph’s nodes to be relevant globally. It preserves the community structures of homogeneous nodes. The authors introduced a bipartite graph encoder to learn initial node representations. The method first generates a global representation. Then, sampled edges are encoded as local representations using a subgraph-level attention technique. The time complexity of the method is $O(k(|E| + |\tilde{E}|)d^2)$, where $k$ is the number of layers in the encoder, $d$ is the embedding size, $E$ is the set of edges, and $\tilde{E}$ is the set of corrupted edges.

Pesantez-Cabrera et al. [111] presented a multi iterative and phase algorithm for the problem of detecting communities and influential nodes in bipartite networks, whose interactions are between a pair of different nodes’ type. Each phase of the algorithm is a series of iterations. The transition from one phase to another is done through a graph compaction procedure. Using a modularity gain function, a node decides locally on its community. The time complexity of the algorithms is $O((n_1 + n_2)n_1 n_2)$, where $n_1$ and $n_2$ are the number of nodes in node sets 1 and 2, respectively.

2.2) SEMI-LOCAL-BASED ANALYSIS TECHNIQUE

Ibrahim et al. [51] presented a bipartite centrality method for identifying the influential nodes in a network. It combines global network flow and local cohesiveness using mathematical formalization. It estimates the degree into which influential nodes influence their neighboring nodes through refined bicliques. It also links the dense substructures in the network via influential bridges. It is based on the observation that influential bicliques and bridges that contain many influential neighbors are likely to have central nodes. The time complexity of the method is $O(m^2 (n_1 + n_2))$, where $n_1$ is the number of nodes of types 1, $n_2$ is the number of nodes of types 2, and $m$ is the number of attributes.
Huang et al. [42] introduced a ranking algorithm for node-weighted online bipartite matching. The algorithm focuses on the scenario where online nodes come in random order. The authors designed a two-dimensional gain sharing function based on nodes’ online arrival time and offline ranking. The algorithm lowers the bound pair of neighbors combined to gain over nodes’ random ranking.

Rödter et al. [113] introduced a rankings method for identifying influential nodes in a directed bipartite social affiliation network. The method uses an entropy-based up-and-coming procedure for modeling a bipartite network. It replaces directed edges with if-then condition. Probabilistic modeling is used to analyze directed social affiliation.

Paudel et al. [112] presented a linear-time algorithm that computes the most influential node of a bipartite graph after removing a subset of vertices of a given size such that the residual graph has minimum pairwise strong connectivity. If the graph is not strongly connected, the algorithm processes a subset of vertices of a given size that causes the largest overall drop in the connectivity value. The time complexity of the algorithms is \( O(k(m + n)) \), where \( n \) is the number of nodes, \( m \) is the number of edges, and \( k \) is the number of removed nodes.

V. TOPOLOGY-BASED STATIC METHODOLOGY USING LOCAL SCOPE

These algorithms use the local and static information of the network. They employ static centrality measures for identifying the influential nodes in networks using the networks’ semi-local information and/or nodes’ degrees. Most of these algorithms that use the semi-local information considers the contribution of the local neighbors by combining nodes’ influences and the contributions from the nearest and next-nearest neighboring nodes. Most of them adopt decision-making methods that can modify the structure of the network.

A. LINKED CONNECTIONS-BASED MODELLING ANALYSIS APPROACH

These algorithms consider the degree and/or the neighbors’ information (within \( k \)-hop distance). They consider the contributions from the nearest and next-nearest neighboring nodes. Most of the ones that use degree-based mechanisms, consider nodes’ degrees and the degrees of their neighbors.

1) SEMI-LOCAL CENTRALITY ANALYSIS TECHNIQUE

Yang et al. [175] proposed a centrality measure that considers the semi-local information for identifying the influential nodes. It considers the degree and clustering coefficient of the target node as well as neighbor information. Specifically, it considers the second-level neighbors’ clustering coefficient and the second-level neighbors’ sum of the clustering coefficient. The entropy technology is adopted to assign weights. The clustering coefficient indicates the density of connections between the target node and its neighbors. The time complexity of the method is \( O(nk^2) \), where \( k \) is the average degree of the graph. The authors defined the centrality measure as in Equation 10:

\[
DCC(i) = \alpha I_D(i) + \beta I_C(i), \quad \alpha + \beta = 1 \tag{10}
\]

- \( I_D(i) = C_D(i) + \sum C_D(j) \): The effect of degree and neighbors’ degree of node \( i \).
- \( I_C(i) = e^{-C_i} \sum_{j \in r_i^2} C_j \): The effect of clustering coefficient and second-level neighbors’ clustering coefficient of node \( i \).
- \( \alpha \) and \( \beta \): The weights of \( I_D(i) \) and \( I_C(i) \), respectively.

Dai et al. [23] proposed a method for identifying the influential nodes in a network based on the contribution of local neighbors. The method combines nodes’ influences and the contributions from the nearest and next-nearest neighboring nodes. A degree is used as the fundamental influence indicator of nodes. The contribution probability is set to be reciprocal to the degree of the node.

Dong et al. [25] introduced a semi-local centrality ranking method to identify influential nodes. Using random walk, the method identifies influential nodes by locating the influential surround nodes. A random walk is performed for each node \( u \) to gather semi-local information. The selected nodes in the walk path are considered influential surrounding nodes of \( u \).

Oridi et al. [108] proposed an algorithm that generates an optimal seed set able to maximize influence by quantifying and assigning specific weights to social actions carried out among nodes. The influence power of each node is computed. Then, a set of nodes is generated, where each node has an influence power value that is higher than the mean of its neighborhood. This leads to identifying the influential nodes.

Basaras et al. [9] proposed a method for identifying the influential spreaders in multilayer networks. The authors introduced centrality-like measures for local computation only. They devise locally computed measures that consider the inter and intra connections for the vicinity of nodes.

Wang et al. [155] proposed an index to identify influential nodes combined with the structural hole theory in sociology. The algorithm evaluates node importance by considering its neighbors connected to its communities and other communities. The influence of a node is identified based on its internal degree and external degree. The time complexity of the method is \( O(m(m + n)) \), where \( n \) is the number of nodes and \( m \) is the number of edges.

Muhuri and Chakraborty [96] introduced a method that uses Edge Contribution Factor (ECF) for identifying the influential node in an online dynamic network. The ECF of a node \( v \) is computed as the ratio of the sum of the neighboring nodes’ edge contribution and \( v \)’s edge contribution to the graph’s number of edges. The time complexity of the method is \( O(n^2) \), where \( n \) is the number of nodes.

2) DEGREE-BASED ANALYSIS TECHNIQUE

Wang et al. [157] proposed an algorithm that identifies the influential spreaders based on nodes’ degrees and the degrees of their neighbors. First, a node’s weighted degree is computed based on its degree. Then, the node’s location is
estimated using the weighted clustering coefficient. A node’s clustering degree is determined based on its weighted degree and weighted clustering coefficient. Finally, the node’s propagation capability is assessed based on its clustering degree and the contributions of its neighbors.

Hafiene and Karoui [48] presented a new approach that identifies influential nodes. First, the method identifies leader nodes, whose degree centralities are higher than their neighbors. The degree of influence equals a community’s number of active nodes divided by the overall number of nodes. The time complexity of the method is \( O(n^2 \log (M) + M(m + n) + M^2 T) \), where \( n \) is the number of nodes, \( m \) is the number of edges, \( T \) is the shortest path time between a leader node and an active node, and \( l \) is the number of the leader nodes in each community \( M \).

Tiwari et al. [135] introduced a method that blocks the nodes that spread rumors by considering the network’s shield nodes. First, a minimal subset of influential shield nodes that have an influence tendency is identified. Then, rumors’ propagation is reduced by anti-rumors spread using the shield nodes. The method calculates the average degree of the graph. This will be considered as the threshold for eliminating nodes that do not have a central tendency.

Mihara et al. [95] proposed a method for influence maximization in a network by relying only on node degrees local information. It greedily identifies the node with the highest expected degree. It also greedily selects the inactive node that has the largest expected degree using the results of prior influence spread and probing.

Tulu et al. [143] introduced a method that selects the important nodes in a large network according to the degree of neighbor nodes. Selected nodes connect communities to each other. Influential nodes are determined based on how many other influential nodes it is essential to connect to.

B. INFORMATION DIFFUSION-BASED MODELING ANALYSIS APPROACH

These algorithms employ information diffusion-based analysis procedure under an enhanced version of the linear threshold or independent cascade models uses in the local and static information. They inherit the key features of the linear threshold or independent cascade models. Most of the independent cascade-based algorithms adopt decision-making methods that can modify the structure of the network. Most of the linear threshold-based methods employ mechanisms that govern the influence spread over a network and detect the nodes, whose influences exceed a certain threshold.

1) INDEPENDENT CASCADE ANALYSIS TECHNIQUE

Li [80] propose an extended version of the Independent Cascade (IC) to measure an individual’s influential capabilities in a local and static social network, which provides evidence for pheromone distribution in a decentralized manner. It inherits the key features from the Independent Cascade model, where influence diffusion is demonstrated as a hopping and infecting process. It incorporates two types of agents: user agents and ant agents. The former provides information actively for ant agents when requested. While the latter traverses the network based on heuristics, seeking the influencers in a social network. The time complexity of the method is \( O(l \cdot s) \), where \( l \) is the path length, and \( s \) is the cardinality of the neighborhood.

Wu et al. [156] proposed a decision-making method based on the Independent Cascade that can modify the structure of a network to select sources. It can modify the structure of a network after deleting edges, adding edges, and increasing the infection influences of nodes on their neighbors. The procedure begins with an initial set of infected nodes. It concludes when all nodes become infected.

Sela et al. [123] revisited the problem of network seeding using an expanded version of the Independent Cascade model. It considers the timing aspect through the creation of social hypes which harness herd behavior tendencies. Messages arrive from different sources at adjacent time periods form an illusion of hype. The method assumes that a spread is possible only within a limited number of time steps.

2) LINEAR THRESHOLD ANALYSIS TECHNIQUE

Talukder et al. [134] proposed linear threshold estimation models based on influence-weight and degree distribution. The threshold values provided by the models are small and specific. In one of the models, a node is activated by comparing its neighbors’ aggregated influence values with the threshold value. In another model, a systematic sampling technique is employed after determining a representative sample size. A threshold \( \theta \) of a node \( v \) is an expected influence value \( w_{uv} \) of randomly selected nodes from in-neighbor \( n^{-1}(v) \) of \( v \). The threshold is defined as in Equation 11:

\[
\theta = \frac{1}{N} \sum_{v \in V} \sum_{u \in n^{-1}(v)} w_{uv} x_u
\]  

- \( N \): Number of nodes.

Yang et al. [170] proposed a method based on the linear threshold model. It detects a network’s influential nodes and locates the nodes, whose influences exceed a user-specified threshold. It can update randomly selected sets based on the changes in the network’s structure. It adopts a polling-based strategy that maintains randomly selected sets in such a way that nodes’ influences are approximated. The time complexity of the method is \( O(|E|^2 \cdot \Delta) \), where \( E \) is the number of edges.

Khurana et al. [66] proposed a content-aware variation of the linear threshold model that governs influence spread over a network based on the features associated with both network nodes and propagated items. An item will be set to propagate the network nodes and propagated items. An item will be set to propagate.
Blesa et al. [10] proposed centrality measures based on the Linear Threshold model that distinguishes between outgoing and incoming neighborhoods when selecting the initial set of nodes that activate the spreading process. The aim is to experimentally check how different these measures are, and their relationships with the other centrality measures.

VI. TOPOLOGY-BASED STATIC METHODOLOGY USING GLOBAL SCOPE

These algorithms identify influential nodes by analyzing the global structure of a static network. They employ global-based influence centrality methods from the perspective of data reconstruction, greedy thought, or random walks on networks.

A. INFORMATION PROPAGATION-BASED MODELLING ANALYSIS APPROACH

Most of the algorithms in this category that adopt a randomized strategy employ a sampling technique. Some of them employ a seeding randomized walk sampling strategy. Most of the algorithms that identify influential nodes from the perspective of data reconstruction reconstruct the data iteratively through subset sampling. Other algorithms identify influential nodes based on greedy thoughts or new greedy.

1) RANDOM FILTERING ANALYSIS TECHNIQUE

Wang et al. [153] proposed an influence centrality method based on random walks without a back-trace. Centrality estimation is carried out using the Monte Carlo technique. Consider that the walker is currently at node x. The next stop is selected at random from the set of nodes connected to the hyperedge that includes node x. First, a hyperedge h associated with node x is selected. Then, the random walk’s next stop is carried out by selecting a node associated with h. The method uses a greedy-based algorithm to identify the influential nodes in a network. The time complexity of the method is \( O(\gamma \log(\gamma) + \lambda n (n_0 + \gamma)) \), where \( \gamma \) is the adjacent pairs of nodes, \( n_0 \) is the maximum degree of nodes in the network, \( n \) is the number of nodes, and the coarsest ratio \( \lambda \in [0, 1] \).

Marshall et al. [94] proposed a method that, given a small scale random seeding of activity in a network, it determines the influential nodes in the network to inhibit the spread of the activity amongst the general population of agents (e.g., obstruct the percolation process). The method involves agent-based modelling of bootstrap percolation on hyperbolic random geometric graphs, to observe how the underlying spatial configuration of a network will impact the process.

Ji et al. [57] designed a diffusion-aware random walk strategy that generates the sequences of nodes more in line with the diffusion semantics. The strategy considers the sampling process of nodes as a diffusion process among nodes and the sequences of nodes as a description of the series of possible diffusion paths. In the sampling phase, a diffusion-aware random walk sampling strategy is designed considering the diffusion preference of nodes and diffusion path among nodes. In the training phase, the generated sequences of nodes are used as the corpus, and the embedding vector of nodes is learned by the Skip-gram model. The time complexity of the method is \( O(\gamma \log(\gamma) + \lambda n (n_0 + \gamma)) \), where \( \gamma \) is the adjacent pairs of nodes, \( n_0 \) is the maximum degree of nodes in the network, \( n \) is the number of nodes, and the coarsest ratio \( \lambda \in [0, 1] \).

2) GREEDY-BASED ANALYSIS TECHNIQUE

Nguyen et al. [107] proposed an IM framework that unifies the approaches introduced in [15, 149, 150] to better characterize the number of Reverse Influence Sampling (RIS) that achieves \( (1 - 1/e - \varepsilon) \)-approximation guarantee. In the original RIS, the standard greedy algorithm is employed for deriving the approximate solution \( (1 - 1/e) \) after random RR sets were generated and \( k \) nodes covering the maximum number of these sets were selected. The authors proposed the Stop-and-Stare Algorithm (SSA) and its dynamic version, D-SSA, which both guarantee a \( (1 - 1/e - \varepsilon) \) approximate solution. Both SSA and D-SSA proved to achieve the minimum number of RIS samples. The improvement of D-SSA over SSA is attributed to its automatic and dynamic selection of the best parameters for the RIS framework.

Lin et al. [84] proposed a greedy algorithm to identify influential nodes based on the greedy thought. The algorithm seeks to rank nodes based on their importance by minimizing the value of unique robustness. From an empty network, the algorithm constructs the network’s original structure in a backward manner by keep adding nodes gradually. This is the opposite of the strategy that deletes nodes from the network in a forward manner. It keeps adding nodes in such a way that connected nodes grow slowly. The time complexity of the method is \( O(n^2 (n + m)) \), where \( n \) is the number of nodes.

Shahsavari and Golpayegani [122] proposed a linear threshold diffusion model that adopts an improved greedy algorithm to find the k-most influential users. The improved algorithm decreases the total amount of computations. For each node, the algorithm computes how many of its neighbors can be activated by this node. First, it computes the number of nodes activated by each node \( v \) for each context by
considering the node’s behavioral pattern. After computing 3% for all nodes, the greedy algorithm is reapplied.

Roy and Pan [119] proposed a greedy approach for the influence maximization problem and finding the top-α influential nodes. The approach continuously updates the fitness value with a predefined number of iterations, which will result in the evolution of a seed set.

Sun et al. [120] proposed a new greedy framework for influence maximization. It selects the most k influential nodes from weighted graphs to maximize the overall influence spread. It selects the seed nodes with the highest marginal gain based on the seed set of the graph to that it belongs to. Each seed set is responsible for promoting one single product category. It may also be a seed node in another graph.

3) DATA RECONSTRUCTION ANALYSIS TECHNIQUE

Wang et al. [163] introduced a framework for influence maximization from the perspective of data reconstruction. It finds the influential nodes by minimizing the reconstruction error. It first constructs an influence matrix, each row of which is a node’s influence on other nodes. It selects the k most informative rows to reconstruct the influence matrix and find the influential seed nodes with the data reconstruction procedure. Consider that \( x_i \in \mathbb{R}^1 \times N \) is the influence of node \( i \). Selecting the \( k \) most influential rows for reconstructing the influence matrix \( X \in \mathbb{R}^k \times N \) is defined as in Equation 12:

\[
\min_{J(A, \beta)} \sum_{i=1}^{N} \left\| x_i - X a_i \right\|^2 + \sum_{i=1}^{N} \frac{a_{i,j}^2}{\beta_j} + \gamma |\beta| \tag{12}
\]

s.t. \( a_{i,j} \geq 0, \beta_j \geq 0, a_i \in \mathbb{R}^n, i = 1 \ldots n, j = 1 \ldots n \); where \( A^T = [a_1, \ldots, a_n] \). \( \beta = [\beta_1, \ldots, \beta_n]^T \) is an auxiliary variable to control nodes selection.

Guo et al. [37] proposed a method that generates node set using random reverse reachable (RR) procedure under the influence maximization model. The method reduces the computational cost for the random RR set by reconstructing the data through effective subset sampling. A random RR set \( R \) is reconstructed as follows: (a) selecting a node \( u \) at random, and (b) sampling the set \( R \) of nodes that may be activated by \( u \) reversibly in such a way that the probability for each node \( v \) to appear in \( R \) is the same as the probability that \( v \) can activate \( u \). The time complexity of the method is \( O(k \cdot n \log (n/\varepsilon^2)) \), where \( k \) is the number of seed nodes, \( n \) is the number of nodes, and \( \varepsilon \) is the number of edges.

Gao et al. [39] investigated the dynamics of information diffusion in community-based networks. The authors argue that the topological structure of the link between two nodes can determine their centralities. Such structure can reveal, for instance, that nodes with relatively least centrality can have greater propagation impact than nodes with high diffusion speed at the initial stage due to their relatively greater centrality. Therefore, nodes that have relatively least centrality should be considered when computing information propagation due to their potential great global diffusion influence. The authors reached this conclusion after investigating the nonlinear crossover of the diffusion processes of boundary nodes and central nodes.

Lohia et al. [85] introduced a method that identifies and ranks nodes to maximize the influence of reaching targeted nodes and minimize the number of undesirable recipients of the information. It iteratively identifies nodes by reconstructing the data by varying the graph structure. Once an influential node is selected, all reached matching nodes are removed. Then, it iteratively finds the next node that can allow reachability to another set of targets. The iteration continues until the budget is reached. The time complexity of the method is \( O(m + n) \), where \( n \) is the number of nodes and \( m \) is the number of edges.

Lei and Wei [86] proposed a method to identify influential communities. Every community is converted to a node by the renormalization procedure and its data is reconstructed. In the renormalized network, a node’s influence is decided by removing this node from the network and determining the remaining nodes’ state using the State of Critical Functionality (SCF) procedure. The renormalization network is removed one node \( i \) at each time, and the value of SCF is re-calculated \( i = 1, 2, \ldots, n \), where \( n \) is the number of nodes of the renormalization network. Influential communities are determined according to the value of SCF of nodes.

B. SHORT PATH-BASED ANALYSIS APPROACH

These algorithms identify influential nodes by analyzing the global structure of a static network using a short path-inspired approach. Most of these algorithms employ either the closeness or betweenness centrality measures. Some of these algorithms analyze the closeness centrality behavior and its correlation with the network’s structural properties. They consider the connectivity of nodes and the neighbors’ closeness. Most of the other algorithms select seed nodes based on their betweenness centrality. They may employ an approximation technique.

1) CLOSENESS-BASED ANALYSIS TECHNIQUE

Hu et al. [43] introduced a centrality measure based on information-theory. It adopts multiple-route distance diminishment. It is based on the observation that the distance between a pair of nodes decreases if there are many paths between the pair. A resistor network is associated with each network. This is carried out by replacing edges with resistors. A pair of nodes’ resistance distance in the network is the resistance between the corresponding pair in the resistor network. The resistance distance between a pair of node decreases if there are several routes between them. The time complexity of the method is \( O(\log r) \), where \( r \) is the rank of the symmetric positive-definite matrix.

Luan et al. [76] introduced a centrality index based on closeness centrality. It aims at studying how identifying the influential spreaders in a network is influenced by the number of shortest paths (NSP). That is, the proposed improved
closeness centrality index incorporates the effect of the NSPs. A node’s importance is impacted by other nodes’ influences determined by the lengths of their shortest paths.

Dai et al. [24] proposed a method that considers the influence of nodes strongly correlated with their neighbors’ closeness. The method identifies the influential nodes by computing the closeness of neighboring nodes. It computes nodes’ influences by considering the mutual impact and the distance between the nodes and their neighbors. The time complexity of the method is $O(nk^2)$, where $k$ is the average number of the first-order neighbor nodes and $N$ is the total number of nodes in the network.

Ni et al. [105] proposed an evolving closeness centrality measure for social networks modelled as time-evolving graphs. It traverses the nodes of a graph from shorter to longer distances. It associates distance labels with time intervals. Its time complexity is $O(K^2 \cdot (|V| + |E|))$, where $K$ is the maximum number of distance labels of a node, $V$ is the set of nodes, and $E$ is the set of edges.

Putnam et al. [110] introduced a method that incrementally updates the closeness centrality values in evolving weighted networks. It includes work-based filtering for outgoing and incoming distances to decrease the computational workload. It considers distances resulting from multiple connected components. Its time complexity is $O(m(n + D) \cdot \log(n + D))$, where $n$, $m$, and $D$ are the number of nodes, edges, and shortest paths, respectively.

Li et al. [87] introduced a centrality measure that considers the closeness and connectivity of nodes for essential proteins identification.

Mittal and Bhatia [97] proposed a closeness centrality metric called cross-layer closeness centrality (CCC) for multiplex social networks. The CCC measure computes a node’s closeness degree to each other node of the multiplex network. The method starts with measuring the shortest path from each node to every other node of the multiplex network. The time complexity of the metric is $O(|V| \cdot |E|)$, where $V$ is the set of nodes and $E$ is the set of edges.

Saxena et al. [130] analyzed the closeness centrality behavior and its correlation with the network’s structural properties. The authors proposed a method that selects $k$ nodes randomly and calculates their closeness centralities. The average closeness of these $k$ nodes is used for estimating the network’s middle-ranked node closeness centrality. The time complexity of the method is $O(nm + n)$, where $n$ is the number of nodes and $m$ is the number of edges.

Crescenzi et al. [16] studied how adding new edges incident to a node may increase its centrality. Specifically, they investigated how adding new edges to a network can maximize a predefined node’s closeness. The time complexity of the method is $O(\Gamma_{uv}(S))$, where $\Gamma_{uv}(S)$ is the number of edges incident to nodes.

Alzabii et al. [4] introduced an enhanced version of the Closeness Centrality that computes the relative centrality of a node, by quantifying the weights of edges connected to it. It is based on the observation that the larger the path between a pair of nodes, the weaker the relation between the pair. Towards this, a penalty parameter is used in the measure that penalizes long paths. The measure rewards shorter paths.

Cohen et al. [20] proposed a solution that approximates the classic closeness centrality of a network’s nodes. It picks a small uniform sample of nodes. From each sampled node, it computes single-source shortest paths. The time complexity of the method is $O(nk)$, where $n$ is the number of nodes and $k$ is the number of seed nodes.

2) BETWEENNESS-BASED ANALYSIS TECHNIQUE

Taha and Yoo [136] presented a forensic analysis method that can detect the influential criminals in a criminal network by analyzing the network’s important communication paths. First, the method calculates nodes’ betweenness centralities to estimate their influences on the network’s flow of information. The method determines whether a path $p$ between two nodes is important based on the betweenness centralities of the set $s$ of nodes in the path $p$ as and the betweenness centralities of the nodes linked to $s$ by edges.

Taha and Yoo [141] presented a forensic analysis method that can detect the influential criminals in a criminal network based on edge betweenness centrality. For each edge, the method computes its shortest-path edge betweenness. An edge’s weight is its “shortest-path edge betweenness”. Then, the network’s Minimum Spanning Tree (MST) is constructed accordingly. Each node $u$ is given a score, which is the number of nodes, whose existence in the MST is dependent on $u$. Influential nodes are identified by ranking nodes according to their scores.

Dey and Roy [21] presented an approach that selects the influential nodes for information blocking in a social network. Seed nodes are selected based on their betweenness centrality. The method blocks the spread of information by blocking the information flow through influential nodes and influencing edges (edges with high betweenness).

Mumtaz and Wang [98] introduced a method that identifies the influential nodes in networks with respect to Betweenness Centrality (BC) using an approximation technique. The method finds the set of $k$ nodes with the largest BC. It uses a progressive random sampling which reduces the sample size by early stopping conditions. The time complexity of the method is $O(mn + \log n)$, where $n$ is the number of nodes and $m$ is the number of edges.

Zhang et al. [177] introduced a method that identifies the influential nodes in a network with the highest ego-betweennesses. The authors developed a top-k search framework with a static upper bound that uses local-update to maintain the ego-betweennesses for all nodes. The time complexity of the method is $O(\alpha \cdot n \cdot d_{\text{max}})$, where $\alpha$ is the arboricity of the graph, $n$ is the number of nodes, and $d_{\text{max}}$ is the maximum degree of the nodes.

VII. CONTENT-BASED ADAPTIVE METHODOLOGY

These algorithms identify the influential nodes in a network by analyzing the content of nodes or links and the
structural information of the network. They use adaptive analysis under an enhanced version of PageRank, LeaderRank, random walk, or grey optimization-based method. Some of them consider neighbors’ information.

A. NON-AUTHORITATIVE-BASED ANALYSIS APPROACH

These algorithms use adaptive analysis under an enhanced version of a random-based or a grey optimization-based method. Some of them determines a random variable by the events’ information content and probability distribution. They may employ Shannon entropy or random walk to estimate nodes’ centralities and chaotic degree of the ranking. Others employ the Grey Wolf Optimizer algorithm, which mimics the hunting mechanism of grey wolves and includes searching, encircling, and attacking a prey.

1) RANDOM-BASED ANALYSIS TECHNIQUE

Zhao et al. [178] introduced a method that identifies influential nodes in a certain event using evidence theory. A random variable is determined by the events’ information content and probability distribution. The random variable’s expectation is the entropy (the generated information content by the distribution). Shannon entropy is adopted to measure the chaotic degree of the ranking where ranking weights are assigned. Participating methods are selected based on needs. The time complexity of the method is \( O(n^2) \), where \( n \) is the number of nodes. The Shannon entropy and the weight of each method are computed as shown in Equation 13:

\[
S = \sum_{i=1}^{N} S_{Mi} \\
W_{Mi} = \frac{S - S_{Mi}}{N_i S} \tag{13}
\]

- \( Mi \): A selected method.
- \( S_{Mi} \): Shannon entropy of \( Mi \).
- \( S \): Sum of Shannon entropy.
- \( W_{Mi} \): Weight of \( Mi \).
- \( N_i \): Number of methods involved with fusion.
- \( N \): Number of nodes in the graph.

Xin et al. [168] proposed a network immunization method that considers the topology of the network as well as the heterogeneity of nodes. Specifically, the method estimates the influence of a node based on its structural centrality, extent of activity, and spreading ability. The structure centrality reflects the node’s topology factor. Both activity and spreading capability of a node reflect its topology factor. First, a subset of nodes is immunized based on these different factors. Then, two nodes are chosen at random to serve as the infected nodes that start the propagation. That is, two nodes are selected randomly as the infection sources. The authors demonstrated that the centrality of a node is influenced by both the topology factor and the heterogeneity factor.

Lv et al. [72] proposed a Random Walk with Restart link prediction algorithm for measuring the influence of a current node and its neighbors on the transition probability. First, the transfer probability between nodes is calculated using transfer probability. Then, the calculated transfer probability is adjusted to improve the new transfer probability matrix.

Katsimpras et al. [68] proposed a method that determines topic-sensitive influential users in social networks. Using supervised random walks, topic-sensitive influence is determined based on prior information. Users are ranked accordingly. The method uses each node’s textual content and structural information to identify its topic-sensitive influence.

Zhu et al. [179] introduced a pricing technique for selecting the nodes that maximize marketing campaigns. Random reverse reachable sets are employed to calculate nodes’ prices. The method computes the difference between a randomly selected seed set’s price and its expected marketing value. Samples are generated adaptively based on the outcomes, until their sum becomes higher than a threshold. The average of the samples is computed. The time complexity of the method is \( O(\gamma / \min s_i \in C \mu_i) \), where \( \gamma \) is a threshold, \( C \) is the set of seed nodes, and \( \mu_i \) is the target node \( i \).

Wakisaka et al. [154] proposed a method that derives the expected size of influence cascade. It is based on random node sampling and degree-based seed node selection. Let \( M \) be randomly sampled nodes from a ground-truth network with degree distribution. The top-k nodes are selected from \( M \) nodes based on their degrees. Finally, the expected size of influence cascades triggered by the top-k nodes is obtained.

Zou et al. [182] proposed a heuristic algorithm called Random Node Recommend to select seed set quickly for influence maximization. It considers the impact of the breadth of information spreading and the depth of information dissemination. First, it divides the network into several communities. Then, it selects randomly several target nodes in every community. Finally, it searches for the most influential neighbor of each target nodes as seeds. The time complexity of the algorithm is \( O(nc) \), where \( n \) is the number of nodes and \( c \) is the number of clusters.

Han et al. [41] proposed a method for identifying the influential users in mobile social networks. Using basic properties of random walks, the method adopts distributed solutions mechanisms that have low control-message overhead. An individual’s centrality is estimated based on how many times the random-walk that probe messages visited his mobile device. It adopts fixed-distance random walks.

2) GREY OPTIMIZATION-BASED ANALYSIS TECHNIQUE

Aghadam et al. [1] proposed a method for the opinion analysis under an enhanced version of PageRank, LeaderRank, random walk, or grey optimization-based method. Some of them consider neighbors’ information.

Katsimpras et al. [68] proposed a method that determines topic-sensitive influential users in social networks. Using supervised random walks, topic-sensitive influence is determined based on prior information. Users are ranked accordingly. The method uses each node’s textual content and structural information to identify its topic-sensitive influence.

Zhu et al. [179] introduced a pricing technique for selecting the nodes that maximize marketing campaigns. Random reverse reachable sets are employed to calculate nodes’ prices. The method computes the difference between a randomly selected seed set’s price and its expected marketing value. Samples are generated adaptively based on the outcomes, until their sum becomes higher than a threshold. The average of the samples is computed. The time complexity of the method is \( O(\gamma / \min s_i \in C \mu_i) \), where \( \gamma \) is a threshold, \( C \) is the set of seed nodes, and \( \mu_i \) is the target node \( i \).

Wakisaka et al. [154] proposed a method that derives the expected size of influence cascade. It is based on random node sampling and degree-based seed node selection. Let \( M \) be randomly sampled nodes from a ground-truth network with degree distribution. The top-k nodes are selected from \( M \) nodes based on their degrees. Finally, the expected size of influence cascades triggered by the top-k nodes is obtained.

Zou et al. [182] proposed a heuristic algorithm called Random Node Recommend to select seed set quickly for influence maximization. It considers the impact of the breadth of information spreading and the depth of information dissemination. First, it divides the network into several communities. Then, it selects randomly several target nodes in every community. Finally, it searches for the most influential neighbor of each target nodes as seeds. The time complexity of the algorithm is \( O(nc) \), where \( n \) is the number of nodes and \( c \) is the number of clusters.

Han et al. [41] proposed a method for identifying the influential users in mobile social networks. Using basic properties of random walks, the method adopts distributed solutions mechanisms that have low control-message overhead. An individual’s centrality is estimated based on how many times the random-walk that probe messages visited his mobile device. It adopts fixed-distance random walks.

2) GREY OPTIMIZATION-BASED ANALYSIS TECHNIQUE

Aghadam et al. [1] proposed a method for the opinion analysis under an enhanced version of PageRank, LeaderRank, random walk, or grey optimization-based method. Some of them consider neighbors’ information.
sensor network. First, several unknown nodes are randomly initialized. Then, the location of the localized node is measured. Then, the localization error is determined by computing the interval between the evaluated and the original areas of the node. These steps are repeated until all unknown/target nodes are localized.

Zhang et al. [184] proposed a coverage optimization method that adopts the grey wolf technique for wireless sensor networks. After establishing the coverage optimization and ending the siege behavior, the simulated annealing procedure is included in the grey wolf. Eventually, the method updates the grey wolf for enhancing the ability of global optimization.

Zivkovic et al. [185] applied enhanced grey wolf optimizer swarm intelligence metaheuristics for social networks’ clustering. For each solution in the population, an additional parameter trial was added. If the current solution cannot be improved, the value of the parameter trial is incremented. Once the value of the parameter trial reaches a predetermined threshold, the solution is replaced with a randomly generated solution inside the search space.

B. AUTHORITATIVE-BASED ANALYSIS APPROACH

These algorithms identify influential nodes using adaptive analysis under an enhanced version of PageRank or LeaderRank. Some of them use the LeaderRank or PageRank methods to rank entities according to the information of the links. Most of them consider neighbors’ information (e.g., PageRank values, degree, or clustering coefficient).

1) PAGERANK-BASED ANALYSIS TECHNIQUE

Fang et al. [29] introduced a sorting and recognition algorithm for identifying the important nodes on the large-scale Internet. It is based on the graph theories of Assortativity and PageRank. It integrates node attributes information into the Markov chain model for detecting the influential nodes on the Internet. The similarity of the two nodes is calculated by the Radial Basis Function Kernel. The time complexity of the algorithm is $O(Nk^2)$, where $N$ is the number of nodes and $k$ is the number of seed nodes.

Jin and Wang [53] devised a single-dimensional method NodeRank, which is an improved algorithm for ranking node influence based on PageRank. The method identifies influential nodes concerning the network load changes. The most influential node in a network is the one whose corresponding vector in the weight matrix has the maximum corresponding cosine similarity with respect to the principal component directions of the matrix.

Zhong and Lv [187] introduced a method that identifies the influential nodes in a network using an improved PageRank mechanism based on the resource allocation. Initially, the method assigns each node a unit resource. According to the neighbors’ PageRank values, a node can allot its resource to its neighbors. Finally, a node’s influence is identified based on the resources it obtained.

2) LEADERRANK-BASED ANALYSIS TECHNIQUE

Zhang et al. [188] introduced an improved weighted LeaderRank method that considers indegree, clustering coefficient, and neighbors’ influences. The method employs a function that depicts the links’ weights based on the influences of nodes. It considers the differences in users’ influences across the whole network by giving more scores to those influential users.

Jia et al. [54] proposed a method to disambiguate mentions in documents by combining LeaderRank with entity popularity to rank entities. The method considers three features: text similarity, entity popularity, and entity-relationship. It adopts the LeaderRank algorithm on the graph model to rank entities according to the link information among entities. At last, the global text similarity is used to improve the ranking result by disambiguating mentions. The entity popularity formula is as shown in Equation 14:

$$ep(e_{i,j}) = \frac{\text{count}(e_{i,j})}{\sum_{e_{i,k} \in E_i} \text{count}(e_{i,k})}$$  \hspace{1cm} (14)$$

- $E_i$: The entity set of a mention $m_i$.
- $\text{count}(e_{i,j})$: Number of links whose anchor text is $m_i$ for entity $e_{i,j}$.

Fan et al. [30] proposed an algorithm of Weighted LeaderRank with Neighbors to select seed users in Device-to-Device (D2D) mobile social networks. It promotes offline multimedia content propagation based on realistic large-scale D2D sharing data. It takes users’ L-hop neighbors’ importance into account.

Yang et al. [173] introduced a modified LeaderRank method to select the most influential “core teams” from a modularized organizational network. The method captures the technical communication dependencies among teams via the extended multi-domain matrix. It employs a similarity index as a measure of the local and global importance of a team in the organizational network. It employs the Jaccard similarity coefficient to measure the overlap between two teams.

Li et al. [88] introduced an improved LeaderRank method by including a weighting mechanism. The method gives degree-based weights to the edges of the ground node. After the weight of each edge is identified, a biased random walk strategy is applied. The influences of nodes are identified using the steady-state final scores.

VIII. CONTENT-BASED STATIC METHODOLOGY

These algorithms employ static-based analysis procedure and consider both the network structure and content-related factors for identifying influential nodes. By analyzing the network’s content semantic, these algorithms infer node behavior using content analysis.

A. TEXTUAL CONTENT-BASED ANALYSIS APPROACH

These algorithms assess the influence of nodes based on the extent of similarity of their content, feature content, or
associated textual semantic information with other nodes. Most of them use techniques (e.g., Shannon entropy) to drive content information and to analyze the network structure.

1) **TOPICAL AFFINITY PROPAGATION ANALYSIS TECHNIQUE**
Zhang et al. [186] introduced a centrality measure based on Local Fuzzy Information Centrality. The measure employs an improved Shannon entropy and fuzzy sets for analyzing the network structure and computing the information contained in boxes. Shannon entropy is employed to drive the information contained in a message. Based on the shortest paths to the center, fuzzy sets distribute weights to neighboring nodes. The more a node’s box contains information, the greater the influence of the node.

Wagih et al. [165] introduced a semantic-influence measurement method for identifying the influential nodes in a social network. It is based on a user’s associated textual semantic information and geospatial information. It considers friends’ spatial and social information.

Munger and Zhao [101] developed a method that identifies the influential users in support forums. It employs social network analysis and topical expertise. It determines which users in an on-line forum are influential in a particular topic based on their topical expertise and the reach of their social network. It considers the amount of their post content.

2) **TOPIC AWARE CASCADING ANALYSIS TECHNIQUE**
De Salve et al. [22] proposed a framework that defines a workflow to deal with the peculiar aspects of the influencers’ prediction process. It models the collection of information about users’ activities from online social networks. It also models the collected information based on the similarity between users’ topics. It predicts future influential members. Given the list of members and the scores resulting from the prediction task, it identifies the k most influential members.

Mao et al. [93] proposed a method for predicting the influential nodes in social networks (SN) for brand communication. For a specific brand, the method can extract all posts content related to the brand from the SN and construct a weighted network model. The method quantitatively measures the individual values of nodes by considering both the network structure and content-related factors. It uses the topological potential theory to evaluate the importance of the nodes by their values.

Saha and Bandyopadhyay [127] presented an algorithm that calculates the degree centrality of nodes. It ranks the nodes after integrating their data and content features from multiple networks. The ranking method imports a weight for every node. The intra-layer connections are imported from each network by computing its topic-aware correlations. Two nodes are linked if there is a correlation between their textual contents. The strength of a link represents a high similarity between the textual contents of the nodes at its end points.

Saleem et al. [129] introduced a unified model composed of three layers to capture diverse topic-aware information propagation scenarios and influence maximization. First, the influential and influenced nodes are identified using the node dimension. Next, the characteristics dimension filters the influential and influenced nodes on the required attributes (i.e., topics) such as age and gender of users. Finally, the nodes capable of influencing many nodes are selected as the top-k influential nodes.

**B. NETWORK CONTENT-BASED ANALYSIS APPROACH**
To evaluate and rank the importance of nodes, most of these algorithms aggregate the nodes’ feature and attribute information. Some of them use PageRank or h-index to evaluate a node’s global property and the network’s indirect influences. Some of them consider a node’s multi-hop influence and aggregate the graph to obtain its information.

1) **PAGERANK-BASED ANALYSIS TECHNIQUE**
Huang et al. [50] proposed a framework that identifies influential nodes in social networks by employing the weighted linear of PageRank and CO-influence. It combines content information, spammer detection, node behavior, and control theory concepts. It builds node behavior using content analysis. It employs the control theory to identify the influential nodes. PageRank is used to evaluate a node’s global property.

Yin et al. [174] proposed a method based on the PageRank algorithm for influence maximization in signed social networks. The algorithm identifies influential nodes based on negative and positive links. It characterizes information propagation for the recommendation of ads. After ranking nodes based on their influences, it selects the top-ranked ones as the initial seeds for influence maximization.

Jaouadi and Romdhane [55] proposed an information diffusion method that controls the information propagation through seed nodes. The method models the semantics of a network based on users’ interests. Candidate nodes are determined using the PageRank algorithm. The PageRank algorithm is applied to take a community as input and outputs a measure of the influence of each node.

Chen and He [18] introduced a method based on PageRank to study the influence maximization on signed social networks. It considers hostile relations and friends, which are characterized as negative and positive edges. Each node’s first iterative ranking is computed using PageRank values. Thereafter, each subsequent iteration ranking is performed based on the predecessor iteration, until the estimator of PageRank converges to its steady value.

2) **H-INDEX-BASED ANALYSIS TECHNIQUE**
Zhang et al. [189] introduced a method that identifies power systems’ vulnerable lines based on H-index. It takes into consideration the contribution of the weights of edges and adjacent nodes’ strengths. It considers the target node’s influence on its adjacent nodes. The nodes’ outdegree indicates the influence on other nodes. The larger the weight of the node’s outdegree, the greater the influence on the nodes adjacent to it. Weighted H-index conveys the original network’s vulnerability in cascading failure process. The weighted H-index is
defined as shown in Equation 15:

\[
W_i \cdot H = \left\{ (w_{ij} \cdot S_{ij})_{j \in \mathcal{E}} \right\}
\]

\[
S_{ij} = \left( w_{ij} \cdot S_{ij}^2 \right)^{\frac{1}{p}}
\]

(15)

- \( W_i \): The weighted H-index of node \( i \).
- \( H \): The computation operator of weighted H-index.
- \( w_{ij} \): The weight of the edge connecting nodes \( i \) and \( j \).
- \( S_{ij} \): The strength of node \( j \) relative to node \( i \).
- \( \alpha \) and \( \beta \): Adjustment factors.

Teng et al. [140] introduced a centrality metric based on HI-index to estimate a network’s indirect influence. It considers friends of friends’ multi-hop influence. Users that have direct neighbors, whose gender ratio is close to the target node, are given more weight. The metric differentiates between comments and likes interaction types. A user is penalized if he interacts with direct neighbors, whose female ratio is different than the target ratio.

Jia et al. [58] proposed a method based on H-index for measuring the influence of a paper and enhancing three classical link prediction methods. The method computes a node’s influence using link prediction mechanism. It focuses on those link prediction mechanisms that mainly employ degree to compute the importance of nodes.

3) AGGREGATE-BASED ANALYSIS TECHNIQUE

Sun et al. [131] introduced a method that detects influential nodes by employing weighted formal concept analysis. It converts a network’s binary relationships among nodes into a hierarchy. It aggregates the nodes in terms of their attribute information. To rank node importance, nodes are aggregated according to their attribute information. This hierarchy characterizes the generalization instantiation relationships between the concepts of nodes. This allows the clustering based on their identical attributes’ information.

The time complexity of the method is \( O(n^2 + n^2L + nL) \), where \( n \) is the number of nodes and \( L \) is the length of the concept lattice.

Maurya et al. [100] proposed a method that uses feature aggregation to identify the number of nodes reachable to a certain node. It approximates the betweenness centrality using node features’ constrained message passing. Nodes aggregate other nodes’ features in a multi-hop neighborhood. Each node accumulates the feature vectors of its multi-hop neighbors as the number of layers increases. This flow of feature information is employed for modeling the graph’s paths. The time complexity of the method is \( O(|V| \cdot |E|) \), where \( V \) is the set of nodes and \( E \) is the set of edges.

Liu et al. [73] proposed a generalized residual vector quantization method to learn effective encodings with an aggregative model. It performs vector quantization for the analysis of content-based information retrieval. It uses a locally aggregating encoding criterion that measures the quantization quality. To render beam search efficient, the neighboring vectors of each node aggregates in fewer subtrees. It iteratively selects a codebook and optimizes it with the current residual vectors, then re-quantizes the dataset to obtain the new residual vectors for the next iteration.

Tan et al. [147] introduced a method that applies the historical connection information of an opportunistic network data to evaluate the influence of its nodes. An opportunistic network is sliced into opportunistic network units which are modeled by aggregating the graph to present the network’s information. The network embedding model is employed to extract the temporal and structural information among the opportunistic network units. The method takes the network attribute tensor as input and produces an attribute embedding matrix composed of node attribute embedding vectors.

Zhang and Gan [183] proposed a heterogeneous graph embedding model that aims at fusing heterogeneous information and aggregating different kinds of neighbors. It takes into consideration node content information. It selects the important nodes from meta-path based random walk sequences and leverages the attention mechanism to aggregate the nodes’ features. First, the model utilizes multiple meta-paths to generate the candidate meta-neighbor set and chooses the top \( k \) most frequently appeared neighbors for each target node. Second, the attention mechanism is used to assign weights for meta-neighbors.

Xuguang et al. [167] introduced a factor analysis approach to mine the relationships of multiple metric measurements, and to determines node importance in complex networks. It aggregates the multi-attributes for the evaluation of the importance of nodes. The specific steps of the method are: (1) input a complex network, (2) calculate the different centrality values, (3) implement the method of factor analysis to different centrality values, and (4) compute the separation from the positive ideal alternative.

IX. EMPIRICAL EXPERIMENTS AND EVALUATIONS

We ran the algorithms adopting the different techniques described in the article using Dell Precision 7730 with Intel(R) Core(TM) i7-6820HQ processor and 2.60GHz, 2592 Mhz, 6 Core(s) and 16 GB RAM under Windows 10 Pro.

A. DATASETS

We conducted the experiments using five real network datasets. The topological features of these networks are shown in Table 2. We selected these real networks because, mainly, they are drawn from disparate fields. Moreover, they are popular and used extensively in evaluating the performance of a wide-range of centrality measures. Below are descriptions of the five networks:

- **Zachary’s karate club network [191]:** It is an undirected and unweighted network constructed by Wayne Zachary [191] based on observing 34 members of a karate club and their split into two groups due to a disagreement.
- **NetScience [104]:** It is a scientists’ co-authorship network compiled from two review articles’ bibliographies. The network is weighted and undirected.
• **Epinions [116]:** It is a network of product review, where users choose to trust or distrust the reviews written by others. It is a directed and unweighted network.

• **Slashdot [74]:** It is a network of users’ friend/foe links in Slashdot, which is a technology news website. It is a directed and unweighted network.

• **USAir [69]:** It is a network of flights between airports in the USA in 2010. The network’s edges depict the flights’ traffic. The nodes represent the airports. It is an undirected and unweighted network.

### B. EVALUATION SETUP
We performed the following methodology for the experimental evaluations:

- **For each analysis-specific technique:** we selected one of the algorithms adopting the technique. That is, for each specific technique, we selected a paper, whose proposed algorithm adopts this technique. We considered the selected algorithm as a representative of the technique. Among all papers proposing algorithms adopting the same technique, we selected the most influential paper. We based the influence of a paper on different factors, especially its degree of state of the art and recency.

- **For each analysis category:** we ranked the different techniques that fall under the same category. We computed the ranking by averaging the scores achieved by the selected algorithm adopting the technique.

- **For each analysis approach:** we ranked the different analysis categories that fall under the same approach. We computed the ranking by averaging the scores achieved by the selected algorithms that fall under the category.

- **For each analysis scope:** we ranked the different analysis approaches that fall under the same scope. We computed the ranking by averaging the experimental scores achieved by the selected algorithms that fall under the approach.

- **For each analysis type:** we ranked the different analysis scopes that fall under the same analysis type.

- We ranked the analysis types.

### C. MODEL FOR SIMULATING THE SPREADING ABILITY AND METRICS FOR EVALUATING THE PERFORMANCE OF THE ALGORITHMS
To evaluate the accuracy of the list of nodes ranked by one of the selected algorithms, we compared the list against a real spreading process of the nodes using the following widely adopted procedure [27]:

1. Recording the list of nodes ranked by each algorithm.
2. Employing the SIR model [27] for simulating nodes' spreading ability. In this model, each node is assigned to a susceptible state: infected state and recovered state. In each state, we considered only a single node to be infected while the remaining are susceptible ones. Each infected node can infect its susceptible neighbors with a certain spreading probability. An infected node can infect neighboring susceptible nodes. In the experiments, we set the spreading probability $\beta = 0.1-0.2$, and the recovery probability $\mu = 1$. Initially, we set the top-$k$ ranked nodes to be infected, where $k = 1\% \times n$ ($n$ is the number of nodes). Thereafter, the number of infected nodes increases based on the SIR model.
3. Using the list of nodes ranked by one of the algorithms and the corresponding one ranked by the SIR model, we recorded the pair scores in a list $S = \{(x_i, y_i)\}_{i=1}^n$, where $x_i$ and $y_i$ belong to the algorithm and SIR model for a node $i$, respectively.

In this survey, the following three metrics are adopted for evaluating the performance of the algorithms:

- **Kendall’s tau correlation coefficient** $\tau$ [63]: It measures the correlation between two variables. It measures the similarity of data orderings ranked in two quantities. The value of $\tau$ is in the range $\{-1, 1\}$. The higher the value of $\tau$, the more accurate is the centrality measure generated the ranked list. It is defined as in Equation 16:

  \[
  \tau = \frac{N_1 - N_2}{0.5 N (N - 1)}
  \]  

  where $N_1$ and $N_2$ are the number of concordant and discordant pairs, respectively.

- **Monotonicity index** [192]: It is a metric used for quantifying the resolution of different indices. The monotonicity index $M(L)$ for a ranking list $L$ of nodes lies in the range $[0, 1]$. The higher the value of $M$, the better is the ranking. It is defined as shown in Equation 17:

  \[
  M(L) = \left[ 1 - \frac{\sum_{l \in L} |V_l| \times (|V_l| - 1)}{|V| \times (|V| - 1)} \right]^2
  \]  

  where $|V_l|$ is the number of nodes that have the same index rank $l$ in the ranked list $L$; and $V$ is the number of nodes.

### TABLE 2. Topological features of the five real network datasets used in the experiments.

| Dataset                     | # Nodes | # Edges | Avg degree | Max degree |
|-----------------------------|---------|---------|------------|------------|
| Zachary Karate Club         | 34      | 78      | 4.558      | 17         |
| NetScience                  | 379     | 914     | 4.8232     | 34         |
| USAir                       | 332     | 2126    | 12.807     | 139        |
| Epinions                    | 75879   | 508837  | 6.71       | 3079       |
| Slashdot                    | 82168   | 504230  | 13.03      | 2543       |
- **Percentage average absolute error [130]**: It is a numerical amount of the discrepancy between an exact value and the corresponding estimated one. The absolute error ($AE(v)$) for a node $v$ is defined as in Equation 18:

$$AE(v) = |EST(v) - ACT(v)|$$  

(18)

The percentage average absolute error $PAAE(v)$ for the node $v$ is defined as shown in Equation 19:

$$PAAE(v) = \frac{\text{Average absolute error}}{\text{network size}} \times 100\%$$  

(19)

### D. THE EXPERIMENTAL RESULTS

Tables 3-6 and Figs. 2-7 present the results of the evaluations, as follows:

- Table 3 shows the Kendall’s coefficient, monotonicity, and percentage average absolute error scores achieved by each algorithm representing a **topology-based** technique. The table also shows the ranking of the different analysis techniques, categories, approaches, scopes, and types.
- Figs. 2, 3, and 4 show the Kendall’s tau correlation coefficient, monotonicity, and % average absolute error scores, respectively, of the selected algorithms representing the **topology-based** techniques, grouped based on the analysis approaches they fall under.
- Table 4 shows the Kendall’s coefficient, monotonicity, and percentage average absolute error achieved by each algorithm representing a **content-based** technique. The table also shows the ranking of the different analysis techniques, analysis approaches, and analysis types.
- Figs. 5, 6, and 7 show the Kendall’s tau correlation coefficient, monotonicity, and percentage average absolute error scores, respectively, of the selected algorithms representing the **content-based** techniques, grouped based on the analysis approaches they fall under.

### E. THE EXECUTION TIMES OF THE ALGORITHMS

Table 5 shows the average execution time of each algorithm representing a topology-based technique. The table also shows the execution time ranking of the different analysis techniques, analysis categories, analysis approaches, analysis scopes, and analysis types. Table 6 shows the average execution time of each algorithm representing a content-based technique. The table also shows the ranking of the different analysis techniques, analysis approaches, and analysis types.

### F. PARAMETRIC TEST OF STATISTICAL SIGNIFICANCE

We used One-way ANOVA parametric test of statistical significance [169] to shed the light on whether the differences between each algorithm’s individual accuracy scores in the different tests reported in the previous subsections are large enough to be statistically significant. Specifically, we want to know the following: (1) whether the accuracy scores for an algorithm using the same dataset but different $k$ (i.e., top-$k$) are large enough to be statistically significant, and (2) whether the accuracy scores for an algorithm using different datasets are large enough to be statistically significant. We ran each algorithm four times against each of the five datasets described in Section IX.A and computed the accuracy scores in terms of the three metrics described in Section IX.C. In the first run, the value of $k$ was 5. In the second, third, and fourth run, it was 10, 15, and 20 respectively. For each algorithm, we considered the number of ANOVA groups as the number of datasets (i.e., 5). Therefore, ANOVA Between Groups Degrees of Freedom was 4. Since the number of accuracy scores was 20 (i.e., 4 accuracy scores for each of the 5 datasets) and the number of groups was 5, ANOVA Within Groups Degrees of Freedom was 15 (i.e., 20-5 = 15).

Tables 7, 8, and 9 show the results for the Kendall’s tau correlation coefficient $\tau$, monotonicity index, and percentage average absolute error, respectively. Ideally, the smaller is the within group mean square and the larger is the $F$-statistic, the better is the algorithm. As the tables show, the algorithms that adopt adaptive analysis-based methods achieved better within group mean square and $F$-statistic than the algorithms that adopt static analysis-based methods. In general, the algorithms that employ the feedback-based model achieved the best results. We considered a p-value that is greater than the significance level 0.01 as an indicative there is not enough evidence to reject the null hypothesis.

### G. DISCUSSION OF THE RESULTS

We outline in this section our analysis of the surveyed algorithms and our observations of the experimental results. Since we evaluated and ranked all the algorithms based on the SIR model, we would like to point out that some of these algorithms were evaluated in their respective original papers using different models (namely, the Independent Cascade model or/and the Linear Threshold model). Therefore, we need to acknowledge that some of these algorithms might have achieved better ranking, if we adopted the same models used for their evaluations. We list in Table 10 the model(s) used for evaluating each algorithm as revealed in its respective original paper. We used the SIR model because it is the most popular one in simulating the dynamics and spread ability. It has been studied on networks of various kinds. It has been effectively and accurately used for predicting different scenarios related to various factors (e.g., epidemic) and the possible outcomes that assess their spread. It yields several fundamental insights into some spread.

For each of the 12 analysis approaches described in this survey and for each set of the selected algorithms that fall under the approach, we outline our conclusions inferred from the papers reported these algorithms and our findings of the experimental results. Moreover, we outline the consensus among the characteristics of the algorithms that fall under a same approach to serve as a verification of our proposed taxonomy. We argue that if the set of algorithms falling under one of our proposed analysis approaches exhibit
significant common characteristics, this can signify the correctness of grouping these algorithms under a same approach, which in turn confirms the correctness of our proposed taxonomy.

1) Adaptive Core Decomposition using Local Topology

Survey conclusions: Local core distribution can significantly boost diffusion throughout the network. A modular community structure is likely to have
TABLE 4. The kendall’S coefficient, monotonicity, and % average absolute error scores of each algorithm representing a content-based technique. The table also shows the ranking of the different analysis techniques, analysis approaches, and analysis types.

| Analysis Type       | Analysis Approach | Specific Technique          | Kendall Coefficient | Monotonicity | % Avg. Absolute Error | Analysis Approach Rank | Analysis Type Rank |
|---------------------|-------------------|-----------------------------|---------------------|--------------|-----------------------|------------------------|--------------------|
| Static Analysis-Based Methods | Agent-Based | Random-Based [177] | 0.7993              | 0.9958       | 5.088                 | 1                      | 1                  |
|                     | Grey Optimization-Based [11] | Kendall coefficient | 0.7689            | 0.9946       | 6.648                 | 2                      | 2                  |
| Adaptive Analysis-Based Methods | PageRank-Based [29] | Kendall coefficient | 0.6970           | 0.9932       | 7.1092                | 2                      | 2                  |
|                     | LeaderRank-Based [187] | Kendall coefficient | 0.7874           | 0.9940       | 5.983                 | 1                      | 1                  |
| Textual Content-Based | Topical Affinity Propagation [185] | Kendall coefficient | 0.7695           | 0.9936       | 4.769                 | 2                      | 2                  |
|                     | Topic Aware Cascading [22] | Kendall coefficient | 0.8116           | 0.9952       | 3.976                 | 1                      | 1                  |
| Network Content-Based | PageRank-Based [50] | Kendall coefficient | 0.8179           | 0.9949       | 2.1094                | 2                      | 1                  |
|                     | h-index-Based [188] | Kendall coefficient | 0.8264           | 0.9957       | 1.784                 | 1                      | 1                  |
|                     | Aggregate-Based [131] | Kendall coefficient | 0.7738           | 0.9937       | 4.1086                | 3                      | 3                  |

well-distributed core nodes. The highest diffusion performance can be maintained by assigning more or fewer core nodes per community, which can lead to modifying the break-down of the structures of communities.

- **Experimental conclusions:** The correlation between the ranking of nodes of the SIR model and the algorithms that fall under this approach is modest for different $\beta$. That is, the accuracy of their node rank is modest. The distance-based core decompositions worked better...
than the degree-based core decomposition in identifying influential nodes. S-shell decomposition showed better performance than shell/degree-based decomposition in identifying influential nodes. As the number of nodes in different ranks becomes larger, the further increase of ranking done by the shell/degree-based algorithms stayed rather limited. Shell/degree-based algorithms have a slight edge when $\beta$ is between 0.13 and 0.15. However, they performed poorly in the 3% top-ranked nodes.

- Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy: (1) All these algorithms identified correctly many influential nodes that have low degrees (weak nodes), (2) the neighbors of most of the influential nodes identified by these algorithms are located near the networks’ cores, (3) the good performance of the search performed by the semi-local-based and mixed-degree-based algorithms took place in the range of 3-hopm (their worst performance took place in 4-hop, where it is not that the larger the hop range is, the better the performance), and (4) as $k$ increases, these algorithms get a closer performance to the SIR model.

2) Adaptive Seeding using Local Topology

- Survey conclusions: The algorithms that fall under this approach work well for localized batch updates. Compared to other algorithms, these algorithms improve the updating time of the top-$k$ influencers by 1~2 orders of magnitude. They can effectively locate the set of seed
nodes with the highest expected number of influenced nodes.

- **Experimental conclusions:** The adaptive voter model’s expected influence spread outperformed the other algorithms that fall under the same adaptive seeding approach. This is because these algorithms estimate unknown parameters by employing the maximum likelihood estimation model to consider influence power.

- **Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy:**
  1. The expected influence spread of the algorithms that fall under this approach increases with budget $k$,
  2. these algorithms’ largest growth took place after selecting the first few seed nodes,
  3. the variation rate of the seed set selected by these algorithms decreases very slowly with the evolutionary effect, and
  4. the number of nodes activated by the selected influential nodes increases significantly within a few walking rounds, and changes slightly.

3) **Adaptive Feedback Model using Global Topology**

- **Survey conclusions:** These algorithms construct the initial community structure quickly which results in
reducing the follow-up processing. However, the overhead of this follow-up processing can have high computational time. The density of a graph has a great impact on the performance of these algorithms. Most of these algorithms detect concentrated changes in networks rather than distributed ones. The information can propagate more quickly and broadly from the seed nodes selected by these algorithms. The performance of the link-removal heuristics performed by these algorithms is greatly impacted by the network structure.

To make it less sensitive to the structure of the network, these algorithms consider the dynamics of information diffusion process. These algorithms assume that information diffusing influences all nodes to limit the impact of edge deletion. That is, they assume that deleting an edge has the same impact on all the graph’s information propagation properties.

- Experimental conclusions: These algorithms demonstrated a fast convergence, which is an indicative of good adaptability. In a few iterations, they...
Table 9. One way ANOVA table for the overall average values of the percentage average absolute error.

| Analysis Type          | Analysis Scope           | Analysis Approach                | Analysis Technique          | Technique Req. Algorithm | Sum of Square (SS) | Mean Square (MS) | F-Statistic | p-Value |
|------------------------|--------------------------|----------------------------------|----------------------------|--------------------------|--------------------|-----------------|-------------|---------|
|                        |                          | Core Decomposition-Based Methods | Local-Based                |                           |                    |                 |             |         |
|                        |                          |                                  |                           | Violin Degree            | 1[90]              | 0.0224          | 0.0078      | 0.0296   | 0.0227  | 2.9102 | 0.0575 |
|                        |                          |                                  |                           | Edge Weighted Degree     | 1[92]              | 0.0102          | 0.0019      | 0.0310   | 0.0081  | 4.2631 | 0.0167 |
|                        |                          |                                  |                           | Semi-Local Centrality-Based | 1[75]             | 0.0120          | 0.0197      | 0.0784   | 0.0502  | 2.5482 | 0.0825 |
|                        |                          |                                  |                           | Distance-Based           | 1[103]             | 0.0061          | 0.0017      | 0.0239   | 0.0051  | 3.6900 | 0.0527 |
|                        |                          |                                  |                           | S-Shell-Based            | 1[79]              | 0.0078          | 0.0126      | 0.0490   | 0.0744  | 5.9947 | 0.0046 |
|                        |                          |                                  |                           | Core Distribution        | 1[148]             | 0.0113          | 0.0287      | 0.0384   | 0.0297  | 1.5348 | 0.4215 |
|                        |                          |                                  |                           | Short Selection Order    | 1[97]              | 0.0092          | 0.0024      | 0.0234   | 0.0071  | 2.8983 | 0.0549 |
|                        |                          | Adaptive Seeding                 |                            | Voter Model              | 1[49]              | 0.0122          | 0.0048      | 0.0287   | 0.0217  | 4.5260 | 0.0315 |
|                        |                          |                                  |                           | Independent Cascades     | 1[92]              | 0.0201          | 0.0053      | 0.0273   | 0.0557  | 10.5094| 0.0003 |
|                        |                          | Feedback-Based Model             | Local-Based                | Aggregating-Based-Node Feedback | 1[91]             | 0.0109          | 0.0037      | 0.0301   | 0.0282  | 7.6216 | 0.0044 |
|                        |                          |                                  |                           | Greedy-Based-Node Feedback | 1[33]              | 0.0068          | 0.0031      | 0.0319   | 0.0378  | 12.1915| 0.0001 |
|                        |                          |                                  |                           | Aggregating-Based-Edge Feedback | 1[159]            | 0.0067          | 0.0014      | 0.0284   | 0.0103  | 7.5751 | 0.0017 |
|                        |                          |                                  |                           | Greedy-Based-Edge Feedback | 1[128]             | 0.0078          | 0.0018      | 0.0291   | 0.0307  | 17.5555| 0.0001 |
|                        |                          | Network-Based Models             | Local-Based                | Laplacian-Based          | 1[2]               | 0.0128          | 0.0072      | 0.0367   | 0.0336  | 4.6666 | 0.0326 |
|                        |                          |                                  |                           | Semi-Local-Based         | 1[2]               | 0.0089          | 0.0012      | 0.0291   | 0.0073  | 6.0833 | 0.0040 |
|                        |                          |                                  |                           | Degree-Based             | 1[174]             | 0.0127          | 0.0031      | 0.0246   | 0.0068  | 2.1935 | 0.1191 |
|                        |                          |                                  |                           | Independent Cascade-Based | 1[57]              | 0.0132          | 0.0055      | 0.0499   | 0.0304  | 0.6019 | 0.4371 |
|                        |                          |                                  |                           | Information Diffusion Modelling | 1[60]             | 0.0136          | 0.0244      | 0.0277   | 0.0293  | 1.2088 | 0.3509 |
|                        |                          |                                  |                           | Linear Threshold Model    | 1[34]              | 0.0123          | 0.0136      | 0.0487   | 0.0121  | 1.0431 | 0.3472 |
|                        |                          |                                  |                           | Random Filtering-Based   | 1[53]              | 0.0185          | 0.0374      | 0.0568   | 0.0186  | 0.4973 | 0.7381 |
|                        |                          |                                  |                           | Greedy-Based             | 1[107]             | 0.0228          | 0.0435      | 0.0520   | 0.0424  | 1.0426 | 0.4175 |
|                        |                          |                                  |                           | Data Reconstruction-Based | 1[163]             | 0.0212          | 0.0286      | 0.0476   | 0.0295  | 1.0314 | 0.4231 |
|                        |                          | Static Analysis-Based Methods    | Global-Based               | Closeness-Based          | 1[43]              | 0.0206          | 0.0052      | 0.0667   | 0.0238  | 0.4524 | 0.7692 |
|                        |                          |                                  |                           | Betweenness-Based        | 1[36]              | 0.0129          | 0.0043      | 0.0581   | 0.0421  | 1.0193 | 0.4288 |

Converged to an optimal solution. To enhance the efficiency and expedite the speed of convergence in the large-scale networks, these algorithms simply add more nodes/edges. As the number of inserted or deleted edges by these algorithms gets larger, the average processing time for a single edge decreases. This is because the more edges are inserted or deleted, the more edges/nodes are processed simultaneously in each iteration. Hence, these algorithms have a good scalability. The adaptivity of these algorithms is advantageous in obtaining more knowledge about networks gradually.

- Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy: (1) The impact of the size of data on the performance of these algorithms is insignificant, (2) in higher sparsity networks with many separated components, the number of nodes influenced by the seed nodes selected by these algorithms gets smaller as time steps gets larger, and (3) the identified seed set by these algorithms is continuously being enhanced until reaches a stable status.

4) Adaptive Network Model using Global Topology

- Survey conclusions: These algorithms outperform other centrality measures in identifying the influential nodes in weighted social networks. They demonstrated that sub-graph density and node centrality are inversely proportional. They are not affected by the impact of random perturbations on a network’s edges and nodes. They outperform other centrality measures in assigning distinct ranks to nodes that have different spreading capabilities. They achieve a good balance between sorting accuracy and computational complexity. The Bi-face score employed by most of these algorithms can determine the influential nodes more accurately than the other centrality measures, including vote-rank, eigenvector, betweenness, closeness, and degree.

- Experimental conclusions: These algorithms exhibited good performance under different propagation probabilities. They could quantify the structural dependencies of dense regions in a network by: (1) capturing the local information to gain knowledge about the way nodes influence their immediate influential neighbors through bridges-like structures, and (2) assessing the global roles of nodes according to the passing of information along key bridges-like structures. The extent of infection triggered by these algorithms is influenced by a node’s number of neighbors and the capability of these neighbors to propagate.

- Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy: (1) These algorithms are effective in identifying highly central nodes located in cluster centers and nodes that function as bridges, and (2) the semi-local-based methodology adopted by some of these algorithms played a significant role in their performance.

5) Static Linked Connections-Based Modelling using Local Topology

- Survey conclusions: The performance indicators of these algorithms in networks with unclear community structure is low. However, their propagation range and transmission rates on other networks are somewhat acceptable. These algorithms can summarize the connectivity around a node efficiently without the need for examining the topology of the entire network.
The propagation ability of the top-$k$ nodes identified by these algorithms decreases steadily as $k$ increases. Influential nodes determined by these algorithms are likely to connect different communities.

- **Experimental conclusions:** These algorithms considered only local structural information and did not consider the network’s global structure. They selected some nodes with large degrees as influential. However, the real influences of these nodes were low.
- **Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy:** (1) The larger the degree of spreading, the lower the accuracy of the local search of these algorithms, where 4-hop is the worst, (2) the performance of these algorithms for distinguishing the spreading ability of nodes is not good, (3) as $k$ increase, the degree of propagation of the top-$k$ nodes identified by these algorithms decreases steadily, (4) the performance of these algorithms decreases as the number of nodes in the dataset gets larger, and (5) these algorithms performed badly in networks, whose nodes’ degrees are relatively small.

6) **Static Information Diffusion-Based Modelling using Local Topology**

- **Survey conclusions:** These algorithms demonstrated that the threshold value for IM under the Linear Threshold model depends on the influence weights and degrees of the nodes as well as the application, for which the IM is used. They demonstrated that the running time of the Independent Cascade increases more than linear. They also demonstrated that the spread of nodes changes more significantly in the Independent Cascade model than in the Linear Threshold model. Moreover, they demonstrated that messages’ arrival time and sources play an important role in the process of spreading information and that the number of these messages sent in a certain period of time can be limited according to the type of application domain.
- **Experimental conclusions:** The range of the threshold values generated by these algorithms was small and distributed around relatively small values, which is good for real influence maximization. However, this is disadvantageous for decreasing the effect of insufficient influence because of the narrower range of smaller threshold values.
- **Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy:** (1) Most of the seed nodes selected by these algorithms are widely spread and have a great impact on information dissemination, (2) the estimated seeding cost using threshold values generated by these algorithms is about 35% lower than the ones generated by most existing IM models, and (3) these algorithms perform well in networks with dense connectivity but not in networks with sparse connectivity.

7) **Static Information Propagation-Based Model using Global Topology**

- **Survey conclusions:** Influential nodes selected by these algorithms have been demonstrated to have a greater impact on spreading information than nodes with high degree, betweenness, eigenvector, or/and PageRank. The algorithms in this approach that employ greedy strategy have been demonstrated to be more efficient than the classic greedy algorithm. They achieve better accuracy than the IM-based algorithms that have similar running time. The algorithms in this approach that employ random filtering have been demonstrated to be more efficient than most popular IM algorithms; yet these algorithms have the same accuracy as these IM algorithms. The algorithms that employ data reconstruction schemes have been demonstrated to outperform other heuristic algorithms and to be as effective as the traditional greedy algorithms.
- **Experimental conclusions:** At the different steps of the propagation process, these algorithms generated a considerable number of infected nodes. The propagation range of these algorithms was good relative to the single-point contact SIR model.
- **Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy:** (1) These algorithms identify the top-10 nodes with high accuracy and provide reasonable node rank list, (2) the algorithms in this approach that adopt random-based filtering procedure significantly outperform the algorithms in other approaches that adopt also random-based filtering, (3) these algorithms are effective in selecting high-degree nodes as seed nodes; however, they showed instability in several networks because of the random selection strategy, and (4) these algorithms are effective in large-scale networks.

8) **Static Short Path-Based Model using Global Topology**

- **Survey conclusions:** The algorithms in this approach that employ closeness centrality demonstrated that the closeness rank of a node can be computed efficiently. They also demonstrated that the closeness centrality versus reverse ranking follows a sigmoid pattern. They proved that adding a few edges to a node can significantly increase its closeness centrality and ranking. The algorithms in this approach that employ betweenness centrality demonstrated that edge-betweenness has better accuracy and efficiency in blocking information than other centrality measures. They proved that removing a certain node can increase the influence of another node. They demonstrated the efficiency of identifying the top-$k$ nodes with the highest ego-betweennesses. They showed that the top-$k$ ego-betweenness results are very similar to the top-$k$ betweenness results; yet they are much more efficient.
- **Experimental conclusions:** The node importance indicators based on these algorithms performed well on the SIR model. The performance of the betweenness-based
mechanism of these algorithms was bad in many networks because information does not flow along the shortest paths of these networks. Besides, the betweenness centrality may not predict well low influence nodes. These algorithms considered weak ties in evaluating centrality, which lowered their results’ accuracy.

- **Consensus among the characteristics of these algorithms:** (1) The efficiency of these algorithms in the first half of the dissemination is low, but it keeps improving in the latter half, (2) the performance of these algorithms is excellent in networks, whose connectivity is good and in networks, whose global connectivity is complete, and (3) these algorithms considered weak ties in evaluating centrality, which lowered their results’ accuracy.

9) **Adaptive Non-Authoritative Content-based**

- **Survey conclusions:** These algorithms achieve good ranking in identifying influential users based on their topic-sensitive interests. In general, the algorithms in this approach demonstrated that they could distinguish the spreading ability of nodes very good. The algorithms in this approach that employ the grey wolf method demonstrated that they could avoid the problem of falling into local optimum. They achieved good performance with regards to the maximum number of localized nodes.

- **Experimental conclusions:** In general, the performance of these algorithms was reasonably acceptable. The relatively good performance of these algorithms was due, in part, to their using of percolation properties-like procedure, which avoid the edges used in clustering.

- **Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy:** (1) Initially, the seed nodes selected by these algorithms influence many nodes in sparse networks; thereafter, the influence cannot be sustained due to the sparsity of the networks, and (2) in sparse networks, the long random walks used by these algorithms cannot return to the same nodes, which cause the influence to subside over time.

10) **Adaptive Authoritative Content-based**

- **Survey conclusions:** The algorithms in this approach that employ improved PageRank mechanisms have been demonstrated to outperform the traditional PageRank algorithm in identifying influential nodes. Moreover, they outperform the Degree, Betweenness, Closeness, and Eigenvalue centralities. However, they have higher computational time than the decomposition-based algorithms. The algorithms in this approach that employ improved LeaderRank mechanisms have been demonstrated to outperform traditional LeaderRank algorithm in identifying influential spreaders. They have higher tolerance to noisy data.

- **Experimental conclusions:** The initial nodes selected by these algorithms were scattered. This caused information to spread faster. Some of the seed nodes selected by these algorithms had a significant impact on information dissemination.

- **Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy:** (1) The nodes selected by these algorithms have a significant impact on information dissemination, and (2) these algorithms outperform other content-based algorithms in networks that have high unbalanced degree distribution.

11) **Static Textual Content-based**

- **Survey conclusions:** These algorithms demonstrated superiority over existing heterogeneity-oriented methods and single topology-based immunization strategies. They are more effective in networks with insignificant variation in the structure centrality. They demonstrated that their identified top-10 influential nodes infect a wide range of nodes. The algorithms in this approach that employ topical-aware cascade have been demonstrated to have the highest accuracy results among the widely used centrality measures.

- **Experimental conclusions:** In the initial steps, these algorithms did not perform well in disseminating information. However, after some steps, they tended to perform better in spreading information throughout the network. These algorithms did not perform well in networks that have many nodes with similar centrality values, where they could not differentiate between the ranking of most of these nodes.

- **Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy:** (1) These algorithms can improve the updating time of the top-\(k\) influencers by 1~2 magnitude orders compared to the content-based algorithms, (2) the quantitative performance of these algorithms is not good, especially in networks with high degrees.

12) **Static Network Content-based**

- **Survey conclusions:** The algorithms in this approach that employ PageRank-based mechanisms demonstrated effectiveness and efficiency due to successful selection of influential initial seed nodes. The algorithms in this approach that employ aggregation-based techniques demonstrated efficiency in quickly selecting influential seed nodes and evaluating the importance of nodes.

- **Experimental conclusions:** By measuring the hub scores of nodes, these algorithms achieved acceptable performance in identifying influential nodes. The index values generated by these algorithms achieved modest best-fitted distribution.

- **Consensus among the characteristics of these algorithms that justify grouping them in the taxonomy:**
taxonomy: (1) The index values generated by these algorithms achieve modest best-fitted distribution, and (2) the ranking correlation of these algorithms and the SIR model is relatively higher when the propagation probability is small.

H. DISCUSSION OF THE EXECUTION TIMES OF THE ALGORITHMS

The algorithms that adopt global adaptive feedback schemes to learn from prior rounds achieved the lowest computational times. Most of these algorithms employ an adaptive seeding strategy to learn the seeding pattern. Specifically, the algorithms in this category that employ edge feedback-based schemes achieved the lowest computational times. This is because the flow of feature information is restricted to edges located on the shortest paths. The algorithms in this category generate and store compact summaries of input data and prior running rounds. This leads to efficiently capturing the clustering information and reducing the computational cost. This is because learning from prior rounds of diffusions helps in avoiding the computational cost of searching for new diffusion information.

The category of algorithms that adopt global static shortest-bath-based search had the highest execution costs. These algorithms employ global-based influence centrality schemes from the perspective of data reconstruction, greedy thought, or random walks. Most of them reconstruct the data by varying the graph structure, which is a time consuming task. Some of them constructs the network’s original structure in a backward manner by keep adding nodes gradually, which further increases the computational cost. Some of them samples the set of nodes that may be activated reversibly, which is also a time-consuming task. Some of these algorithms computes the shortest-path edge/node betweenness for each edge/node, which is also time consuming. These algorithms require huge amount of storage, which leads to high computational time in retrieving, analyzing, and processing the data. The algorithms that employ filtering procedure achieved high computational time, which is attributed largely to the overheads of: (1) their internal evaluation objective function to perform the filtering procedure, and (2) their repetitive feedback cycles.

I. THE METHODS THAT USE NODE BEHAVIORAL OR CONTENT FEATURES ALONG WITH THE NETWORK STRUCTURAL FEATURE

Considering the behavioral and content features of nodes is important for designing analytical models that characterize the dynamic relationship between nodes, which can be used for ranking these nodes based on their influences. The table categorizes node behavioral features into four classes: node interest-based behavioral feature, node type-based behavioral feature, and node dynamicity-based behavioral feature.

X. CONCLUSION

We introduced in this survey paper a methodology-based taxonomy for classifying the algorithms that identify top-\(k\) influential spreaders into hierarchically nested, specific, and fine-grained categories. We surveyed 184 papers and discussed their algorithms, which fall under 26 specific techniques. We surveyed and experimentally compared and ranked the following: (1) the different algorithms that employ the same analysis specific technique, (2) the different analysis techniques that fall under the same analysis sub-category/category, (3) the different analysis categories that fall under the same analysis approach, (4) the different analysis approaches that fall under the same analysis scope, (5) the different analysis scopes that fall under the same analysis type, and (7) the different analysis types.

Here below are highlights of some of our major findings: (1) the distance-based core decomposition worked better than the degree-based core decomposition for identifying influential nodes, (2) s-shell decomposition showed better performance than shell/degree-based decomposition for identifying influential nodes, (3) the adaptive voter model’s expected influence spread outperformed the other algorithms that adopt the same adaptive seeding approach, (4) the algorithms that employ the Adaptive Feedback model demonstrated a fast convergence, which is indicative of a good adaptability; also, the impact of the size of data on these algorithms

| TABLE 10. The diffusion model used for evaluating each algorithm in its respective original PAER. Some of these algorithms were evaluated using multiple diffusion models. |
|-----------------------------------------------|
| SIR model | Independent Cascade model | Linear Threshold model |
|------------------------------------------------|
| [180], [79], [82], [103], [175], [174], [189], [157], [80], [33] | [92], [91], [128], [163], [170], [153], [107], [174] | [134], [170], [153], [107] |

| TABLE 11. The methods that employ node behavioral or content features along with the network structural features to rank nodes. |
|-----------------------------------------------|
| Node Behavior-based Feature | Node Type-based Behavioral Feature | Node Dynamicity-based Behavioral Feature | Node Monotone-based Behavioral Feature |
|-----------------------------------------------|
| [22], [48], [55], [66], [68], [85], [108], [113], [120], [133], [134], [158], [168], [174] | [24], [29], [36], [47], [51], [52], [56], [97], [111], [131], [140], [153], [165], [167], [173] | [31], [34], [35], [62], [77], [79], [80], [91], [103], [137], [147] | [32], [33], [57], [64], [123], [125] |
| [18], [30], [50], [54], [58], [73], [93], [100], [101], [129], [183], [189] | | | |
is insignificant, (5) the performance of the algorithms that employ the Static Linked Connections-based model in distinguishing the spreading ability of nodes is not good, (6) the algorithms that employ the Static Information Propagation-based model identified the top-10 nodes with high accuracy and provided a reasonable node rank list, and (7) the Adaptive Authoritative Content-based algorithms outperformed the other content-based algorithms when networks have high unbalanced degree distribution.

REFERENCES

[1] S. Aghdam, F. Gharehchopogh, and M. Masardi, “Opinion leaders selection with grey wolf optimizer algorithm on social networks,” Int. J. Ind. Math., vol. 13, no. 2, pp. 163–174, 2021.

[2] I. G. Adedayo and Y. Sun, “A novel approach for the identification of influential node in an electric power grid,” in Proc. 14th IEEE Conf. Electron. Appl. (ICEIA), Xi’an, China, Jun. 2019, pp. 718–723.

[3] S. Ahajjam and H. Badir, “Identification of influential spreaders in complex networks using HybridRank algorithm,” Sci. Rep., vol. 8, no. 1, p. 11932, Dec. 2018.

[4] M. Alzaabi, K. Taha, and T. A. Martin, “CISRI: A crime investigation system using the relative importance of information spreaders in networks depicting criminals communications,” IEEE Trans. Inf. Forensics Secur. vol. 10, no. 10, pp. 2196–2211, Oct. 2015.

[5] A. Zareie, A. Sheikhahmadi, and A. Fatemi, “Influential nodes ranking in complex networks: An entropy-based approach,” Chaos, Solitons Fractals, vol. 104, pp. 485–494, Nov. 2017.

[6] A. Garas, F. Schweitzer, and S. Havlin, “A k-shell decomposition method for weighted networks,” New J. Phys., vol. 14, no. 8, Aug. 2012, Art. no. 083030.

[7] M. A. Al-Garadi, K. D. Varathan, S. D. Ravana, E. Ahmed, G. Mujtaba, M. U. S. Khan, and S. U. Khan, “Analysis of online social network connections for identification of influential users: Survey and open research issues,” ACM Comput. Surv., vol. 51, no. 1, pp. 1–37, 2018.

[8] J. Bae and S. Kim, “Identifying and ranking influential spreaders in complex networks by neighborhood coreness,” Phys. A. Statist. Mech. Appl., vol. 395, pp. 548–559, Feb. 2014.

[9] P. Basaras, G. Iosifidis, D. Katsaros, and L. Tassiulas, “Identifying influential users in social networks by neighborhood coreness,” IEEE Trans. Netw. Sci. Eng., vol. 6, no. 1, pp. 31–45, Jan. 2019.

[10] M. J. Blesa, P. García-Rodríguez, and M. Serna, “Forward and backward linear threshold ranks,” in Proc. 17th Int. Conf. Inf. Integ. Web-Based Appl. Serv., Dec. 2015, p. 16.

[11] M. R. Bouadjenek, H. Hacid, and M. Bouzeghoub, “Social networks and complex networks: An entropy-based approach,” IEEE/ACM Trans. Netw., vol. 23, no. 5, pp. 121–129, Dec. 2015, p. 16.

[12] J. Dong, F. Ye, W. Chen, and J. Wu, “Identifying influential nodes in complex networks via semi-local centrality,” in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), Florence, Italy, May 2018, pp. 1–5.

[13] P. Dey, S. Bhattacharya, and S. Roy, “A survey on the role of centrality as seed nodes for information propagation in large scale network,” ACM/IMS Trans. Data Sci., vol. 2, no. 3, pp. 1–25, Aug. 2021.

[14] J. Berahmand, A. Bouyer, and N. Samadi, “A new centrality measure depicting criminals communications,” in Proc. CIAT, 2020, pp. 121–129.

[15] H. Fan, X. Tong, Q. Zhang, T. Chang, C. Wang, and X. Wang, “Identification of influential users in mobile device-to-device social networks to promote offline multimedia content propagation,” in Proc. IEEE Int. Conf. Multi-media Expo (ICME), Shanghai, China, Jul. 2019, pp. 1720–1725.

[16] T. Fushimi, T. Satoh, K. Saito, and K. Kazama, “Comparison of influence measures on structural changes focused on node functions,” in Proc. 17th Int. Conf. Inf. Integ. Web-Based Appl. Serv., Dec. 2015, p. 16.

[17] J. Guo and W. Wu, “Influence maximization: Seeding based on community structure,” ACM Trans. Knowl. Discovery Data, vol. 14, no. 6, pp. 1–22, Dec. 2020.

[18] J. Guo and W. Wu, “Adaptive influence maximization: If influential node unwilling to be the seed,” ACM Trans. Knowl. Discovery Data, vol. 15, no. 5, p. 84, 2021.

[19] G. Salhu, N. Tziortziotis, and M. Vazirgiannis, “Adaptive submodular influence maximization with myopic feedback,” in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM), Barcelona, Spain, Aug. 2018, pp. 455–462.

[20] Q. Gui, R. Deng, X. Cheng, and Y. Lv, “A new method for overlapping community detection based on complete subgraph and label propagation,” in Proc. 3rd Int. Conf. Intell. Inf. Process., Guilin, China, May 2018, pp. 127–134.

[21] B. Guo, H. Wang, Z. Yu, and Y. Sun, “Detecting spammers in E-commerce website via spectrum features of user relation graph,” in Proc. 5th Int. Conf. Adv. Cloud Big Data (CBD), Shanghai, China, Aug. 2017, pp. 324–330.

[22] Q. Guo, S. Wang, Z. Wei, and M. Chen, “Influence maximization revisited: Efficient reverse reachable set generation with bound tightened,” in Proc. ACM SIGMOD Int. Conf. Manag. Data, Jun. 2020, pp. 2167–2181.

[23] H. Geppert, S. Bhowmik, and K. Rothermel, “Large-scale influence maximization with the influence maximization benchmark suite,” in Proc. 4th ACM SIGMOD Joint Int. Workshop Graph Data Manage. Expert. Syst. (GRADES) Netw. Data Anal. (NDA), Jun. 2021, pp. 1–11.
B. Hou, Y. Yao, and D. Liao, “Identifying all-around nodes for spreading
Y. Hu, J. Li, and Y. Ruan, “Finding influencers in complex networks: A novel method based on information theory,” IEEE Internet Things J., vol. 16, no. 2, pp. 3372–3380, Jun. 2022.

B. Hou, Y. Yao, and D. Liao, “Identifying all-around nodes for spreading
dynamics in complex networks,” Phys. A, Stat. Mech. Appl., vol. 391, no. 15, pp. 4012–4017, Aug. 2012.

B. Hou, Y. Zhao, J. Tang, X. Yang, and H. Huang, “Identifying top-k influential nodes based on discrete particle swarm optimization with local neighborhood degree centrality,” IEEE Access, vol. 9, pp. 21345–21356, 2021.

M. G. Harigovindan, M. S. Naveen, and R. C. Jisha, “A novel method to find the most influential node in a complex network,” in Proc. 4th Int. Conf. Comput. Methodologies Commun. (ICCMC), Erode, India, Mar. 2020, pp. 1–4.

J. S. He, M. Han, S. Ji, T. Du, and Z. Li, “Spreading social influence with both positive and negative opinions in online networks,” Big Data Mining Anal., vol. 2, no. 2, pp. 100–117, Jun. 2019.

N. Hafiene and W. Karoui, “A new structural and semantic approach for identifying influential nodes in social networks,” in Proc. IEEE/ACSSC 14th Int. Conf. Comput. Syst. Appl. (AICSCS), Hammamet, Tunisia, Oct. 2017, pp. 1338–1345.

Q. He, X. Wang, B. Yi, F. Mao, Y. Cai, and M. Huang, “Optimization maximizing through unknown influence power in social networks under weighted voter model,” IEEE Internet Things J., vol. 14, no. 2, pp. 1874–1885, Jun. 2020.

F. Huang, Y. Yang, Z. Zheng, G. Wu, and S. Mumtaz, “Recognizing influential nodes in social networks with controllability and observability,” IEEE Internet Things J., vol. 8, no. 8, pp. 6197–6204, Apr. 2021.

M. H. Ibrahim, R. Missaoui, and J. Vaillancourt, “Identifying influential nodes in two-mode data networks using formal concept analysis,” IEEE Access, vol. 9, pp. 159549–159565, 2021.

M. H. Ibrahim, R. Missaoui, and J. Vaillancourt, “Cross-face centrality: A new measure for identifying key nodes in networks based on formal concept analysis,” IEEE Access, vol. 8, pp. 206901–206913, 2020.

T. Jin and C. Wang, “Ranking node influence according to dynamic network load,” in Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS), Apr. 2019, pp. 1–6.

B. Jia, B. Wu, J. Lv, P. Zhou, Y. Bu, and X. Ying, “An entity disambiguation method based on LeaderRank,” in Proc. IEEE Conf. Int. Big Data, Dec. 2017, pp. 4337–4342.

M. Jaouadi and L. Ben Romdhane, “DIN: An efficient algorithm for detecting influential nodes in social graphs using network structure and attributes,” in Proc. IEEE/ACS 13th Int. Conf. Comput. Syst. Appl. (ACS/CA S), Agadir, Morocco, Nov. 2016, pp. 1–8.

J. Cao, X. Lin, S. Guo, L. Liu, T. Liu, and B. Wang, “Bipartite graph embedding via mutual information maximization,” in Proc. 14th ACM Int. Conf. Web Search Data Mining, Mar. 2021, pp. 635–643.

Y. Ji, L. Pan, and P. Wu, “Influence maximization on large-scale networks with a group-based method via network embedding,” in Proc. IEEE 4th Int. Conf. Data Sci. Cyberspace (DSC), Hangzhou, China, Jun. 2019, pp. 177–182.

Y. Jia and L. Qu, “Improve the performance of link prediction methods in citation network by using H-index,” in Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discovery (CyberC), Chengdu, China, Oct. 2016, pp. 220–223.

J. Tang, X. Tang, and J. Yuan, “Influence maximization meets efficiency and effectiveness: A hop-based approach,” in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining, Sydney, NSW, Australia, Jul. 2017, pp. 64–71.

J. Geng, Y. Li, Z. Zhang, and L. Tao, “Sentinel nodes identification for infectious disease surveillance on temporal social networks,” in Proc. IEEE/WIC/ACM Int. Conf. Web Intell., Thessaloniki, Greece, Oct. 2019, pp. 493–499.

J.-G. Liu, Z.-M. Ren, and Q. Guo, “Ranking the spreading influence in complex networks,” Phys. A, Statist. Mech. Appl., vol. 392, no. 18, pp. 4154–4159, 2013.

C. J. Kuhlman, V. S. A. Kumar, M. V. Marathe, S. S. Ravi, and D. J. Rosenkrantz, “Inhibiting diffusion of complex contagions in social networks: Theoretical and experimental results,” Data Mining Knowl. Discovery, vol. 29, no. 2, pp. 423–465, Mar. 2015.

M. G. Kendall, “The treatment of ties in ranking problems,” Biomietrika, vol. 33, no. 3, pp. 239–251, 1945.

M. Kimura, K. Saito, and H. Motoda, “Blocking links to minimize contamination spread in a social network,” ACM Trans. Knowl. Discovery Data, vol. 3, no. 2, pp. 1–23, 2009.

S. S. Kamath, “Graph energy based centrality measure to identify influential nodes in social networks,” in Proc. IEEE 5th Int. Conf. Converg. Technol. (ICTC), Bombay, India, Mar. 2019, pp. 1–6.

A. Khurana, A. Logins, and P. Karra, “Selecting influential features by a learnable content-aware linear threshold model,” in Proc. 29th ACM Int. Conf. Inf. Knowl. Manag., Oct. 2020, pp. 635–644.

E. Katz, Personal Influence: The Part Played by People in the Flow of Mass Communications. Piscataway, NJ, USA: Transaction Publishers, 1955.

G. Katsimpras, D. Vogiatzis, and G. Paliouras, “Determine influential users with supervised random walks,” in Proc. 24th Int. Conf. World Wide Web, May 2015, pp. 787–792.

J. Kunegis, “KONECT: The Koblenz network collection,” in Proc. 22nd Int. Conf. World Wide Web, May 2013, pp. 1343–1350.

N. Kumar, R. Guo, A. Alaei, and P. Shakarian, “An empirical evaluation of social influence metrics,” in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM), San Francisco, CA, USA, Aug. 2016, pp. 1329–1336.

Y. Liu, M. Tang, T. Zhou, and Y. Do, “Improving the accuracy of the k-shell method by removing redundant links: From a perspective of spreading dynamics,” Sci. Rep., vol. 5, no. 1, p. 13172, Oct. 2015.

L. Lv, C. Yi, B. Wu, and M. Hu, “An improved link prediction algorithm based on comprehensive consideration of joint influence of adjacent nodes for random walk with restart,” in Proc. 6th Int. Conf. Comput. Artif. Intell., New York, NY, USA, Apr. 2020, pp. 380–387.

S. Liu, J. Shao, and H. Lu, “Generalized residual vector quantization and aggregating tree for large scale search,” IEEE Trans. Multimedia, vol. 19, no. 8, pp. 1785–1797, Aug. 2017.

J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, “Community scale-up in large networks: Natural cluster sizes and the absence of one-size-fits-all well-defined clusters,” Internet Math., vol. 6, no. 1, pp. 29–123, 2009.

R.-H. Li, L. Qin, J. X. Yu, and R. Mao, “Influential community search in large networks,” Proc. VLDB Endowment, vol. 8, no. 5, pp. 509–520, 2015.

Y. Luan, Z. Bao, and H. Zhang, “Identifying influential spreaders in complex networks by considering the impact of the number of shortest paths,” J. Syst. Sci. Complex., vol. 34, no. 6, pp. 2168–2181, Dec. 2021.

P. De Leillis and M. Porfiri, “Detection of influential nodes in network dynamical systems from time series,” IEEE Trans. Control Netw. Syst., vol. 8, no. 3, pp. 1249–1260, Sep. 2021.

G. Loukides and R. Gwadera, “Preventing the diffusion of information to vulnerable users while preserving PageRank,” Int. J. Data Sci. Anal., vol. 5, no. 1, pp. 19–29, Feb. 2018.

X. Li and Q. Sun, “Identifying and ranking influential nodes in complex networks based on dynamic node strength,” Algorithms, vol. 14, no. 3, pp. 82, Mar. 2021.

W. Li, Q. Bai, and M. Zhang, “SIminer: A stigmergy-based model for mining influential nodes in dynamic social networks,” IEEE Trans. Big Data, vol. 5, no. 2, pp. 223–237, Jun. 2019.

J. Li, L. Yu, J. Zhao, and C. Wen, “A new node centrality evaluation model for multi-community weighted social networks,” in Proc. IEEE 86th Veh. Technol. Conf. (VTC-Fall), Tokyo, ON, Canada, Sep. 2018, pp. 1–6.

Z. Li, T. Ren, Y. Xu, B. Chang, D. Chen, and S. Sun, “Identifying influential spreaders based on adaptive weighted link model,” IEEE Access, vol. 8, pp. 66608–666073, 2020.

C. Liu, P. Wang, and A. Chen, “Identifying influential spreaders in complex networks by an improved spectralrank algorithm,” in Proc. 40th Chin. Control Conf. (CCC), Jul. 2021, pp. 736–741.
A. Namtirtha, A. Dutta, B. Dutta, A. Sundararajan, and Y. Simmhan, “Efficient closeness centrality computation in complex networks,” in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining, Aug. 2019, pp. 1018–1025.

P. Pesantez-Cabrera and A. Kalyanaraman, “Detecting communities in biological bipartite networks,” in Proc. 7th Int. Conf. Inf. Bioinf., Comput. Biol., Health Inform., Seattle, WA, USA, Oct. 2016, pp. 98–107.

N. Paudel, L. Georgiadis, and G. F. Italiano, “Computing critical nodes in directed graphs,” ACM J. Exp. Algorithms, vol. 23, pp. 1–24, Nov. 2018.

W. Rödder, A. Dellinitz, F. Kullmann, S. Litzinger, and E. Reucher, “Bipartite structures in social networks: Traditional versus entropy-driven analyses,” Entropy, vol. 21, no. 3, p. 277, Mar. 2019.

M. Raja, D. Chaudhary, P. Darawachelin, and T. Vengattaraman, “GW0-LPWSN: Grey wolf optimization algorithm for node localization problem in wireless sensor networks,” J. Comput. Netw., vol. 2017, pp. 1–10, Aug. 2017.

G. Razis, I. Anagnostopoulos, and S. Zeadally, “Modeling influence with semantics in social networks: A survey,” ACM Comput. Surv., vol. 53, no. 1, pp. 1–38, Jan. 2021.

M. Richardson, R. Agrawal, and P. Deshpenges, “Trust management for the semantic web,” in Proc. Semantic Web, May 2003, pp. 351–368.

K. Rawal and A. Khan, “Maximizing contrasting opinions in signed social networks,” in Proc. 10th Int. Conf. Ubiquitous Inf. Manag. Commun., Jan. 2016, pp. 1–8.

Y. Singer, “Influence maximization through adaptive seeding,” ACM SIGecom Exchang. vol. 15, no. 1, pp. 32–39, Sep. 2016.

M. Shalavvi and A. H. Golpayegani, “Finding K-most influential users in social networks for information diffusion based on network structure and different user behavioral patterns,” in Proc. IEEE 14th Int. Conf. E-Bus. Eng. (ICEBE), Shanghai, China, Nov. 2017, pp. 220–225.

A. Sela, I. Ben-Gal, A. S. Pentland, and E. Shmueli, “Improving information spread through a scheduled seeding approach,” in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining, Aug. 2015, pp. 629–632.

G. Song, Y. Li, X. Chen, H. Je, and T. Jiang, “Influential node tracking on dynamic social network: An interchange greedy approach,” IEEE Trans. Knowl. Data Eng., vol. 29, no. 2, pp. 359–372, Feb. 2017.

Q. Song, G. Lu, W. Wen, J. Cao, and F. Liu, “Bipartite synchronization and convergence analysis for network of harmonic oscillator systems with signed graph and time delay,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 66, no. 7, pp. 2732–2734, Jul. 2019.

A. E. Sarýüce, B. Gedik, G. Jacques-Silva, K.-L. Wu, and U. V. Çatalyurek, “Streaming algorithms for k-core decomposition,” Proc. VLDB Endowment, vol. 6, no. 6, pp. 433–444, Apr. 2013.

S. Saha and S. Bandyopadhyay, “Versatility-preserving multi-omics data analysis by ranking the nodes in multilayer networks,” in Proc. IEEE 5th Int. Conf. Comput. Commun. Autom. (ICCCA), Oct. 2020, pp. 617–622.

L. Sun, W. Huang, P. S. Yu, and W. Chen, “Multi-round influence maximization,” in Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Jul. 2018, pp. 2249–2258.

M. A. Saleem, R. Kumar, T. Calder, X. Xie, and T. B. Pedersen, “IMixer: A unified system for evaluating influence maximization in location-based social networks,” in Proc. ACM Int. Conf. Inf. Knowl. Manag., Singapore, Nov. 2017, pp. 2523–2526.
A. Saxena, R. Gera, and S. R. S. Iyengar, “Fast estimation of closeness centrality ranking,” in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining, Jul. 2017, pp. 80–85.

Z. Sun, B. Wang, J. Sheng, Y. Hu, Y. Wang, and J. Shao, “Identifying influential nodes in complex networks based on weighted formal concept analysis,” IEEE Access, vol. 5, pp. 37–389, 2017.

C. Salavati, A. Abdollahpour, and Z. Manzari, “BridgeRank: A novel fast centrality measure based on local structure of the network,” Phys. A, Stat. Mech. Appl., vol. 496, pp. 635–653, Apr. 2018.

K. Taha, P. D. Yoo, F. Z. Eddinnari, and S. Nedunkuthalil, “Inferring the densest multi-profiled cross-community for a user,” Knowl.-Based Syst., vol. 237, Feb. 2022, Art. no. 107681.

A. Talukder, M. G. R. Alam, N. H. Tran, D. Niyato, G. H. Park, and C. S. Hong, “Threshold estimation models for linear threshold-based influential user mining in social networks,” IEEE Access, vol. 7, pp. 105541–105461, 2019.

K. Tiwari, M. Singh, and A. Bharti, “A time based greedy approach to reduce the influence of malicious nodes in a social networks,” Int. J. Eng. Sci. Comput., vol. 10, no. 6, pp. 1–4, Jun. 2020.

K. Taha and P. D. Yoo, “Shortlisting the influential members of criminal organizations and identifying their important communication channels,” IEEE Trans. Inf. Forensics Security, vol. 14, no. 8, pp. 1988–1999, Aug. 2019.

G. Tong, W. Wu, S. Tang, and D. Du, “Adaptive influence maximization in dynamic social networks,” IEEE/ACM Trans. Netw., vol. 25, no. 1, pp. 112–125, Feb. 2017.

M. M. Tulu, R. Hou, and T. Younas, “Identifying influential nodes based on community structure to speed up the dissemination of information in complex network,” IEEE Access, vol. 6, pp. 7390–7401, 2018.

K. Taha, “Static and dynamic community detection methods that optimize a specific objective function: A survey and experimental evaluation,” IEEE Access, vol. 8, pp. 98330–98358, 2020.

Y.-W. Teng, H.-W. Chen, D.-N. Yang, Y.-A. Pignolet, T.-W. Li, and L. Chen, “On influencing the influential: Disparity seeding,” in Proc. 30th ACM Int. Conf. Inf. Knowl. Manag., Oct. 2021, pp. 1804–1813.

K. Taha and P. D. Yoo, “Using the spanning tree of a criminal network for identifying its leaders,” IEEE Trans. Inf. Forensics Security, vol. 12, no. 2, pp. 445–453, Feb. 2016.

N. Tatti, “Density-friendly graph decomposition,” ACM Trans. Knowl. Discovery Data, vol. 13, no. 5, pp. 1–29, Oct. 2019.

M. M. Tulu, R. Hou, and T. Younas, “Finding important nodes based on community structure and degree of neighbor nodes to disseminate information in complex networks,” in Proc. 3rd IEEE Int. Conf. Comput. Commun. (ICCC), Chengdu, China, Dec. 2017, pp. 269–273.

K. Taha, “Disjoint community detection in networks based on the relative association of members,” IEEE Trans. Computat. Social Syst., vol. 5, no. 2, pp. 493–507, Jun. 2018.

A. J.-P. Tixier, M. E. G. Rossi, F. D. Malliaros, J. Read, and K. Taha, “Influence maximization in networks by accounting the spreading heterogeneity and evolution of the network,” in Proc. 17th Int. Conf. Auto. Agents Multiagent Syst. (AAMAS), Richland, SC, USA, 2018, pp. 1414–1422.

H. Wei, Z. Pan, G. Hu, L. Zhang, H. Yang, X. Li, and X. Zhou, “Identifying influential nodes based on network representation learning in complex networks,” PLoS ONE, vol. 13, no. 7, Jul. 2018, Art. no. e0200091.

D. Wang, J. Yan, D. Chen, B. Fang, and X. Huang, “RNA: A reject neighbors algorithm for influence maximization in complex networks,” Math. Oper. Res., vol. 5, no. 8, pp. 1313, Aug. 2020.

L. Weiyan, L. Bin, and L. Tao, “Mining influential nodes in urban road networks based on virtual semi-local centrality,” in Proc. 10th Int. Conf. Commun., Circuits Syst. (ICCCAS), Dec. 2018, pp. 222–226.

Z. Wang, H. Wang, Q. Liu, and E. Chen, “Influential nodes selection: A data reconstruction perspective,” in Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Jul. 2014, pp. 879–882.

B. Wei, J. Liu, D. Wei, C. Gao, and Y. Deng, “Weighted k-shell decomposition for complex networks based on potential edge weights,” Phys. A, Stat. Mech. Appl., vol. 521, pp. 277–283, Feb. 2019.

H. M. Waghi, H. M. O. Molokar, and S. Ghoniemy, “SIMBA: A semantic influence measurement based algorithm for detecting influential diffusion in social networks,” in Proc. IEEE 13th Int. Conf. Semantic Comput. (ICSC), Jan. 2019, pp. 178–182.

B. Wilder, L. Onasch-Vera, J. Hudson, J. Luna, N. Wilson, R. Petering, D. Woo, M. Tambe, and E. Rice, “End-to-end influence maximization in the field,” in Proc. 17th Int. Conf. Auto. Agents Multiagent Syst. (AAMAS), Richland, SC, USA, 2018, pp. 1414–1422.

W. Xuguang, “Identify influential nodes in complex networks based on modified TOPSIS,” in Proc. 36th Chin. Control Conf. (CCC), Jul. 2017, pp. 1474–1479.

X. Yin, C. Gao, Z. Wang, X. Zhen, and X. Li, “Discerning influential spreaders in complex networks by accounting the spreading heterogeneity of the nodes,” IEEE Access, vol. 7, pp. 92070–92078, 2019.

J. H. Xue and D. M. Titterington, “t-tests, F-tests and Otsu’s methods for image thresholding,” IEEE Trans. Image Process., vol. 20, no. 8, pp. 2392–2396, Aug. 2011.

Y. Yang, Z. Wang, J. Pei, and E. Chen, “Tracking influential individuals in dynamic networks,” IEEE Trans. Knowl. Data Eng., vol. 29, no. 11, pp. 2615–2628, Nov. 2017.

Y. K. Yalavarthi and A. Khan, “Steering top-k influencers in dynamic graphs via local updates,” in Proc. IEEE Int. Conf. Big Data (Big Data), Dec. 2018, pp. 576–583.

Y. Zhang and Y. Zhang, “Top-k influential nodes in social networks: A game perspective,” in Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., Tokyo, Japan, Aug. 2017, pp. 1029–1032.

N. Yang, Q. Yang, M. Razika, and T. Yao, “Identifying core teams in product development organizational networks from the perspectives of dependency and similarity,” in Proc. 3rd Int. Conf. Manag. Eng., Softw. Eng. Service Sci. (ICMSS), 2019, pp. 97–101.
[174] X. Yin, X. Hu, Y. Chen, X. Yuan, and B. Li, “Signed-PageRank: An efficient influence maximization framework for signed social networks,” IEEE Trans. Knowl. Data Eng., vol. 33, no. 5, pp. 2208–2222, May 2021.

[175] Y. Yang, X. Wang, Y. Chen, M. Hu, and C. Ruan, “A novel centrality of influential nodes identification in complex networks,” IEEE Access, vol. 8, pp. 58742–58751, 2020.

[176] L. Zhang, X. Sun, P. Wang, J. Hou, and G. Sun, “Identifying influential nodes with a community structure measure,” in Proc. IEEE 4th Int. Conf. Data Sci. Cyberspace (DSC), Hangzhou, China, Jun. 2019, pp. 264–269.

[177] Q. Zhang, R.-H. Li, M. Pan, Y. Dai, G. Wang, and Y. Yuan, “Efficient top-k ego-betweenness search,” 2021, arXiv:2107.10052.

[178] J. Zhao, Y. Song, and Y. Deng, “A novel model to identify the influential nodes: Evidence theory centrality,” IEEE Access, vol. 8, pp. 46773–46780, 2020.

[179] Y. Zhu, J. Tang, and X. Tang, “Pricing influential nodes in online social networks,” Proc. VLDB Endowment, vol. 13, no. 10, pp. 1614–1627, Jun. 2020.

[180] A. Zeng and C.-J. Zhang, “Ranking spreaders by decomposing complex networks,” Phys. Lett. A, vol. 377, pp. 1031–1035, Jun. 2013.

[181] J. Zhang, Q. Zhang, L. Wu, and J. Zhang, “Identifying influential nodes in complex networks based on multiple local attributes and information entropy,” Entropy, vol. 24, no. 2, p. 293, Feb. 2022.

[182] H. Zou and M. Zheng, “Random node recommend algorithm for influence maximization in social network,” in Proc. 9th Int. Conf. Inf. Technol. Med. Educ. (ITME), Hangzhou, China, Oct. 2018, pp. 921–925.

[183] J. Zhang and X. Gan, “Meta-neighbor aggregated graph attention network for heterogeneous graph representation,” in Proc. IEEE/CIC Int. Conf. Commun. China (ICCC), Jul. 2021, pp. 248–253.

[184] Y. Zhang, L. Cao, Y. Yue, Y. Cai, and B. Hang, “A novel coverage optimization strategy based on grey wolf algorithm optimized by simulated annealing for wireless sensor networks,” Comput. Intell. Neurosci., vol. 2021, pp. 1–14, Mar. 2021.

[185] M. Zivkovic, N. Bacanin, T. Zivkovic, I. Strumberger, E. Tuba, and M. Tuba, “Enhanced grey wolf algorithm for energy efficient wireless sensor networks,” in Proc. Zooming Innov. Consum. Technol. Conf. (ZINC), Novi Sad, Serbia, May 2020, pp. 87–92.

[186] H. Zhang, S. Zhong, Y. Deng, and K. H. Cheong, “LFIC: Identifying influential nodes in complex networks by local fuzzy information centrality,” IEEE Trans. Fuzzy Syst., vol. 30, no. 8, pp. 3284–3296, Aug. 2022.

[187] L. Zhong and F. Lv, “An improved pagerank for identifying the influential nodes based on resource allocation in directed networks,” in Proc. 14th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process. (ICCWAMTIP), Dec. 2017, pp. 42–45.

[188] Z.-H. Zhang, G.-P. Jiang, Y.-R. Song, L.-L. Xia, and Q. Chen, “An improved weighted LeaderRank algorithm for identifying influential spreaders in complex networks,” in Proc. 7th IEEE Int. Conf. Comput. Sci. Eng. (CSE) IEEE Int. Conf. Embedded Ubiquitous Comput. (EUC), Jul. 2017, pp. 748–751.

[189] Q. Zhang, W. Fan, Z. Qiu, Z. Liu, and J. Zhang, “A new identification approach of power system vulnerable lines based on weighed H-index,” IEEE Access, vol. 7, pp. 121421–121431, 2019.

[190] W. Zhou, H. Huang, Q.-S. Hua, D. Yu, H. Jin, and X. Fu, “Core decomposition and maintenance in weighted graph,” World Wide Web, vol. 24, no. 2, pp. 541–561, Mar. 2021.

[191] W. W. Zachary, “An information flow model for conflict and fission in small groups,” J. Anthropol. Res., vol. 33, no. 4, pp. 452–473, 1977.

[192] A. Zareie, A. Sheikhhahmadi, M. Jalili, and M. S. K. Fasaei, “Finding influential nodes in social networks based on neighborhood correlation coefficient,” Knowl.-Based Syst., vol. 194, Apr. 2020, Art. no. 105580.