Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

CSBA-Net: Learning Cross-Scale Structural Dependencies for Influential Node Identification in Complex Networks

Rihab BOUKHECHE
Laboratoire LIRE
Mohammed LALOU
mohammed.lalou@u-bourgogne.fr
Laboratoire d'Informatique de Bourgogne
Nardjes BOUCHEMAL
Laboratoire LISI

Research Article

Keywords: Node Ranking, Positive domination, Machine learning, diffusion model, Deep learning, key players, Influential nodes

Posted Date: November 19th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11432969/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

CSBA-Net: Learning Cross-Scale Structural Dependencies for Influential Node Identification in Complex Networks

Rihab Boukheche[1], Mohammed Lalou[2,*], Nardjes Bouchemal[3]

[1]LIRE Laboratory of Constantine 2, University of Mila, Algeria
rihab.boukheche@centre-univ-mila.dz
[2]LIB UR 7534, F-21000 Dijon, Université Bourgogne Europe, France
**Corresponding author*: *mohammed.lalou@u-bourgogne.fr*
[3]LISI laboratory of Mila, University of Mila, Algeria
n.bouchemal@centre-univ-mila.dz

October 19, 2026

Abstract

Identifying influential nodes is a fundamental task in complex network analysis, with applications in epidemic control, information diffusion, and network robustness. Existing centrality-based and learning-based methods often provide limited integration of structural information across different network scales. To address this limitation, we propose *CSBA-Net*, for *Cross-Scale Structural BiGRU Attention Network*, a multi-scale structural representation learning framework that jointly captures local, community-level, and global structural patterns. The extracted representations are processed by CNNs and integrated through a BiGRU with multi-head self-attention to model dependencies across structural scales. Influential-node scores are learned through regression using SIR-based diffusion simulations as reference targets. We conduct experiments on 24 real-world and 12 synthetic networks from social, biological, technological, and knowledge domains, and show that CSBA-Net achieves Kendall's $\tau = 0.802$, Spearman's $\rho = 0.943$, NDCG@10 = 0.898, and NDCG@20 = 0.916, outperforming the considered centrality-, CNN-, and GNN-based baselines. The results demonstrate the effectiveness of combining complementary structural scales for accurate influential-node ranking across diverse network structures.

Keywords: Node Ranking, Positive domination, Machine learning, diffusion model, Deep learning, key players, Influential nodes.

1 Introduction

Many real-world systems can be modeled as complex network, including biological systems, transportation infrastructures, electrical power grids, and social networks. Each node within these networks is connected to many other nodes, and a small subset of them has a crucial influence over the structural stability and functional efficiency of the system. These nodes, known as *influential nodes*, act as critical drivers of different phenomena, such as the rapid spread of

epidemics [20, 44], cascading failures across interdependent systems [14, 60], and the accelerated dissemination of rumors [51, 28]. Their failure or removal can compromise, or even paralyze, the entire system, potentially triggering severe consequences. Identifying them is therefore crucial to understanding system behavior.

In the literature, much research has focused on traditional centrality metrics to quantify how important a node is within a network, including degree centrality [67], closeness [48], harmonic [43], betweenness [6], and eigenvector [5] centralities, k-shell and coreness centrality [29], gateway score [57], mixed-degree decomposition [72], and entropy-based approaches [69]. Despite their effectiveness in providing structural insights, these approaches often fail to accurately identify node influence in complex or heterogeneous networks because they emphasize either local or global structural characteristics independently. They may also yield unreliable results in the presence of isolated or sparsely connected nodes. Moreover, nodes with similar structural features can be assigned indistinguishable importance scores, thereby reducing the discriminative capability of these measures.

More recently, alternative approaches have emerged using machine learning and deep learning techniques to address these limitations. Many machine learning approaches for node ranking have been proposed, including Logistic Regression [23], K-Nearest Neighbors [74], Decision Trees [73], Support Vector Machines [68], and Random Forests [62]. They enhance modeling flexibility by capturing complex and nonlinear relationships between features, but they still depend on handcrafted feature selection and extraction, which restricts their generalization across different network types. To fill this gap and better learn representations directly from the network structure without human intervention, several deep learning–based approaches have been developed, such as classical Graph Convolutional Networks (GCNs) [9, 75, 22, 3], hybrid models integrating CNNs and GNNs [73, 52], Graph Attention Networks (GATs) [30], and Transformer-based architectures [25].

An analysis of existing deep learning-based work highlights the following main limitations and challenges that remain unaddressed. First, most models formulate influential node identification as a binary classification problem, which fails to capture the continuous and heterogeneous nature of node influence. Second, standard Graph Convolutional Networks (GCNs) assign equal importance (weight) to neighboring nodes, which can reduce their effectiveness in heterogeneous or multi-community networks. Finally, many existing deep learning approaches rely on limited node features, single-scale representations, or fixed feature constructions, particularly in CNN-based methods, which restricts their ability to capture and model multi-scale structural dependencies underlying node influence in networks.

The need for a more flexible architecture capable of effectively integrating heterogeneous and multi-scale structural features is well motivated. To this end, we propose *CSBA-Net*, for *Cross-Scale Structural BiGRU Attention Network*, a novel deep learning framework to identify influential nodes in complex networks that explicitly models cross-scale dependencies using a sequential architecture, addressing the above challenges through the following contributions:

- A new *multi-scale structured representation* approach, in which each node is described through complementary micro-, meso-, and macro-level perspectives, enabling the modeling of hierarchical structural dependencies.
- *Structured neighborhood matrices* that integrate intrinsic node features with first- and second-order neighborhood information, preserving the underlying graph topology while capturing heterogeneous structural patterns across multiple scales.
- A *CNN-BiGRU architecture enhanced with multi-head self-attention*, which jointly learns hierarchical spatial structural patterns and captures inter-scale dependencies, allowing the model to adaptively focus on the most informative structural features.

– A formulation of the problem as a *continuous influence estimation task* based on SIR diffusion simulations rather than a binary classification task. The new formulation enables a more accurate and fine-grained assessment of node influence.
– A *network-adaptive feature balancing strategy* that integrates complementary local, community-level, and global topological descriptors within each structural matrix, preventing feature dominance and ensuring robust multi-scale representations across heterogeneous networks.

The rest of the paper is structured as follows. In Section 2, we review different approaches developed in the literature for identifying influential nodes in complex networks. Section 3 provides the different concepts and definitions needed in the rest of the paper. In Section 4, we describe our proposed approach to deal with the problem and detail the framework architecture. We describe the experimental protocol in Section 5 and present a comprehensive evaluation in Section 6, where we discuss the results obtained on different kinds of networks. Finally, we conclude the paper and outline directions for future work.

2 Related work

We classify the several approaches developed in the literature, to identify influential nodes in complex networks, into three main categories namely, centrality-based methods, machine learning-based methods, and deep learning-based methods. In what follows, we provide an overview of the representative works of each category and mainly focus on deep learning-based methods.

Centrality-based methods. All existing centrality-based methods leverage the topological properties of the network, and can be categorized into four main classes [41]: *(i) Neighborhood-based approaches* that evaluate the node importance based on local connectivity patterns, such as direct neighbors, or multi-step neighbors of a node, which is the case of Degree centrality [67], Coreness centrality [29], and H-index, which gives a more nuanced view of local importance [42], *(ii) Distance-based approaches*, which assess node influence based on distances or paths within the network. This is the case for Betweenness, Closeness and Katz centralities [6, 26, 48]; *(iii) Iterative-based approaches*, where the node influence is continuously updated based on the importance of its neighbors, *i.e.,* they recursively aggregate the influence of neighboring nodes through iterative propagation rules. This is the case for PageRank [7], LeaderRank [40], and eigenvector centrality [5], and *(iv) Structure-perturbation-based approaches* that determine node importance by the impact of its removal on the overall network structure and robustness [64].

These methods often face limitations in terms of performance and flexibility, particularly in complex or heterogeneous settings. They tend to assign identical importance to nodes with the same degree while overlooking structural differences, and remain primarily based on local connectivity without fully accounting for the broader global structure of the network.

Machine learning-based methods. Much research has explored machine learning techniques for identifying influential nodes, introducing a data-driven perspective in which node influence is inferred from extracted features to improve prediction performance [8]. Several models have been developed [23, 62, 68, 74, 73], and a key distinction among many of these approaches lies in how feature vectors are designed to capture node characteristics.

In [74], the authors used a vector comprising nine node features computed based on different centrality methods, and then employed seven machine learning algorithms to capture the relationship between each node's true propagation capability and its structural characteristics. Its dependency on manually crafted features makes it difficult to identify latent structural patterns. Similarly, Ramírez-de-la-Rosa *et al.* [53] identified Twitter opinion leaders through machine learning algorithms by examining users' writing styles and behaviors. Although the approach

captures richer features than structural centrality alone, its reliance on textual and behavioral data limits its applicability to networks where such information is unavailable.

Wen *et al.* [68] proposed a node importance evaluation method based on Least Squares Support Vector Machine (LS-SVM). Four complex indicators and an Analytic Hierarchy Process (AHP) are used to derive importance scores, approximated by LS-SVM after that to reduce computational cost. However, its performance remains bounded by the quality of the AHP scores, and the reliance on a fixed set of manually selected indicators limits generalization to different network structures and diffusion processes. In [54], a Support Vector Regression with an RBF kernel is trained on only 0.5% of nodes selected via cluster sampling, combining degree, extended coreness and connectivity as features. While reducing training data, the method still requires complete global network information and its handcrafted features may fail to capture higher-order dependencies in heterogeneous networks.

Despite generally outperforming centrality measures, ML methods remain limited by handcrafted features, motivating representation learning where node embeddings are learned directly from topology.

Deep learning-based methods. To address the limitations of traditional machine learning and better handle graph-structured data, several deep learning-based approaches have been introduced, including classical Graph Convolutional Networks (GCNs) [75, 3, 70, 50], CNN-based and hybrid CNN-GNN models [71, 49, 1, 73, 52], Graph Attention Networks (GATs) [30, 22], and Transformer-based architectures [25, 65].

For *GCN-based approaches,* Zhao *et al.* [75] proposed InfGCN, which leverages four node features along with neighborhood graph information. It formulates the problem as binary classification rather than continuous influence estimation, and its reliance on a fixed and limited set of features restricts its ability to capture structural patterns beyond immediate neighborhoods. Similarly, Bhattacharya *et al.* [3] introduced DeepInfNode, which integrates degree, betweenness, k-shell, and density to learn influence representations and improve prediction performance. This model also depends on a small predefined set of topological metrics and retains the same binary classification formulation. This limits the model's ability to capture continuous and multi-level structural variation. More recently, Wu *et al.* [70] constructed features from regular equivalence-based similarity with adaptive eigenvalue-based selection. While Patel and Singh [50] derived features from degree sequences across k-hop neighborhoods within a modified GCN with a regression output. The former relies on a single global similarity relation, and the latter aggregates multi-hop information into a single homogeneous signal, rather than explicitly separating local, meso-, and global-scale structure.

For *CNN-based approaches*, Yu *et al.* [71] proposed the *RCNN* model, which constructs a fixed-size neighbor subgraph for each target node via breadth-first search prioritizing higher-degree neighbors. The resulting feature matrix is processed by convolutional layers to output an influence score. However, relying only on degree to construct neighborhoods may cause peripherally located nodes to be misidentified as influential. Ou *et al.* [49] extended this approach in *M-RCNN* by constructing three structured matrices to capture micro-, community-, and macro-level structural information, based on neighbors' degrees, community connectivity, and k-core values, respectively. These matrices are stacked as a three-channel input and processed by a shared CNN, allowing the relative contributions of the three structural levels to be learned automatically. Note that *M-RCNN* treats these levels as a fused input and does not explicitly model their sequential or hierarchical dependencies. Ahmad *et al.* [1] proposed *LCNN*, which constructs a fixed-size adjacency matrix from the target node's one-hop neighborhood. Neighbors are ranked using recursively aggregated degree information, while degree- and H-index-based multi-scale features are embedded into six channels and processed by a CNN to predict node influence. Although *LCNN* reduces computational cost, its explicit neighborhood representation

remains local and does not directly model mesoscopic or global structural dependencies.

Overall, existing GCN-based approaches either rely on predefined structural features or aggregate multi-hop information into a single signal, while CNN-based methods depend on handcrafted features and encode structural information as a flat, non-hierarchical input. Consequently, both families fail to explicitly model dependencies across local, meso-, and global scales.

Hybrid models combining GNNs with other architectures, particularly CNNs, have also been explored. Zhang *et al.* [73] proposed a hybrid CNN–GNN model that generates node feature matrices through a contraction-based operation over first- and second-order neighbors, avoiding predefined network metrics. However, its fixed two-hop operation yields a single-scale representation that cannot capture long-range or community-level dependencies. Ramadhan *et al.* [52] proposed a lightweight model combining 1D convolution with a GraphSAGE layer, using only degree and average neighbor degree as node features, which reduces computational complexity. These hybrid models reduce reliance on handcrafted centrality features but still represent node structure at a single-scale representation, without explicitly capturing distinct local, meso-, and global-scale structural dependencies.

To address the limitations of fixed neighbor weighting in standard GCNs, *Attention and Transformer* mechanisms have been applied. In [30], Kou *et al.* proposed a graph multi-head attention regression model (GMHAR) that replaces binary classification with continuous influence prediction. Its multi-head attention mechanism adaptively weights neighboring nodes, while dense connections enhance feature propagation. However, the model relies only on adjacency information without incorporating structural or node attributes, which may limit its generalization across heterogeneous networks. Similarly, Huang *et al.* [22] employed two Transformer-based encoders to capture structural and semantic information, which are integrated through co-attention. The model's reliance on pre-trained language models and multiple loss terms increases computational cost. Also, its design for attributed knowledge graphs limits direct applicability to plain complex networks. Jiang *et al.* [25] fused local and global structural information through multi-scale features but relied on manually designed metrics and added computational complexity. Tang *et al.* [65] integrated GCNs with Graph Transformers to maximize influence, but treated the task as seed-set selection rather than node influence estimation. Its quadratic complexity limits the scalability.In general, these attention-based approaches overcome fixed neighbor weighting but remain limited by their reliance on a single representation or a specific structural-semantic fusion, without explicitly modeling local, mesoscopic, and global structural roles.

The *CSBA-Net* framework developed in this work is designed to model cross-scale dependencies, and we show, in what follows, how this is achieved.

3 Background and preliminaries

In this section, we introduce the fundamental concepts underlying the proposed approach.

Complex networks. A complex network can be represented as a graph $G = (V, E)$, where V denotes the set of nodes and E the set of edges describing their relationships. Unlike regular graphs, complex networks typically exhibit non-trivial structural properties, such as heterogeneous degree distributions, community structure, small-world characteristics, and high clustering [46, 4]. In this context, influential node identification aims to estimate the relative influence of individual nodes based on their structural role and contribution to network-wide propagation.

Convolutional Neural Networks. CNNs are deep neural architectures that use learnable convolutional filters and parameter sharing to extract local patterns from structured inputs [34, 19]. Although originally developed for grid-structured data such as images [31], CNNs can also process structured matrix representations of network neighborhoods, making them suitable for extracting

local topological patterns.

Gated Recurrent Units. GRU networks [10] use update and reset gates to regulate the integration of past and new information. They enable effective sequence modeling with relatively low computational complexity compared to LSTM [11]. Beyond temporal data, GRUs can model interactions within structured and ordered feature representations [39]. This makes them suitable for capturing dependencies among multi-scale structural representations in complex networks. In this work, a bidirectional GRU (BiGRU) is employed to model the ordered micro-, meso-, and macro-level representations in both directions, enabling information exchange across structural scales.

Multi-head self-attention. Self-attention allows a model to assign different importance to elements of an input representation according to their pairwise relevance [66]. Given an input representation Z, the query (Q), key (K), and value (V) matrices are obtained as:

$$Q = ZW^Q, \qquad K = Z^K, \qquad V = ZW^V$$

where W^Q, W^K, and W^V are learnable projection matrices that transform Z into Q, K, and V, enabling the model to learn relevance across different structural levels.

The scaled dot-product attention is defined as follows:

$$\mathit{Attention}(Q, K, V) = \mathit{Softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$

where d_k is the dimensionality of the key vectors, and *Softmax* produces normalized attention weights that indicate the relative importance of each element.

Multi-head attention performs this operation across multiple projection subspaces, allowing the model to capture complementary interactions among input representations [66]. In the proposed model, multi-head self-attention is applied to the BiGRU outputs to adaptively capture dependencies among different structural levels.

4 Proposed Framework

4.1 Overview

The proposed framework *CSBA-Net* aims to identify influential nodes in complex networks by learning hierarchical structural dependencies through a multi-stage deep learning architecture (see *Figure 1* for a detailed illustration of its architecture).

Given a graph $G = (V, E)$, each node $u \in V$ is represented through multi-scale structural information extracted from its local neighborhood. Specifically, a fixed-size neighborhood subgraph is extracted for each node and encoded into structured representations that capture complementary structural properties at different levels of abstraction. These representations are then encoded by parallel Convolutional Neural Network (CNN) modules that extract spatial structural patterns from each scale. The resulting embeddings are subsequently organized as a sequence and processed by a BiGRU to capture cross-scale dependencies. To further refine the learned representations, a multi-head self-attention mechanism is applied to selectively emphasize the most informative structural features. Finally, the aggregated representation is passed through fully connected layers to predict a continuous influence score for each node. Unlike binary classification approaches, CSBA-Net formulates influential node identification as a regression problem, where node influence is modeled as a continuous value. The ground-truth labels are generated using SIR diffusion simulations, enabling a more realistic characterization of node influence.

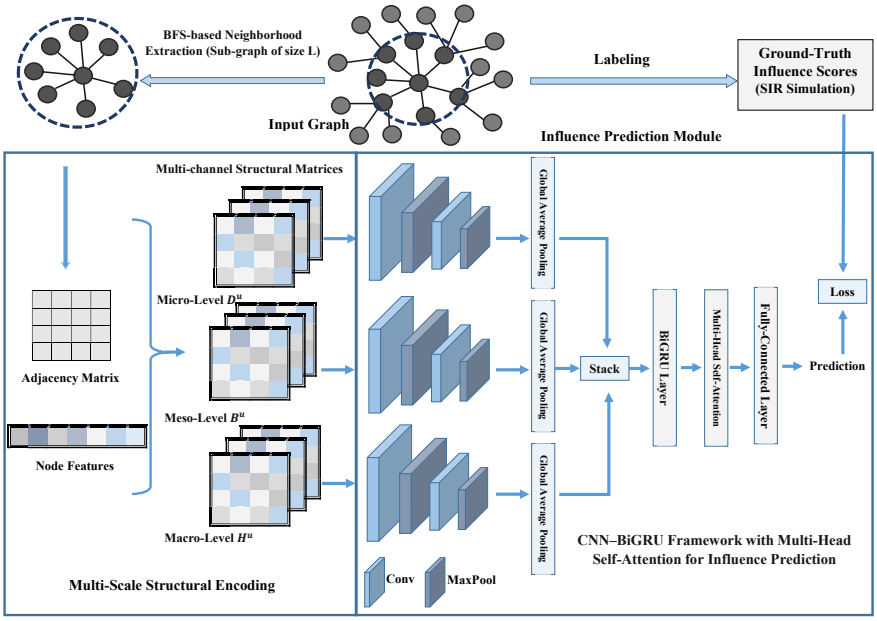


Figure 1: *The proposed CNN–BiGRU framework with multi-head self-attention for influential node identification. BFS-based neighborhood subgraphs are encoded into micro-, meso-, and macro-level matrices (D^u, B^u, H^u), processed by parallel CNN branches, and fused for BiGRU-based inter-level modeling and attention-based feature selection. Fully connected layers then predict influence scores supervised by SIR-based labels.*

This design allows the model to jointly capture local, community-level, and global structural information, while explicitly modeling interactions across these scales in a unified learning framework. In what follows, we detail and motivate each element of the CSBA-Net architecture.

4.2 Multi-Scale Structural Representation

First, and for each node, we extract the neighborhood subgraph and transform it into learnable multi-scale feature representations through the construction of structured feature matrices.

Neighborhood Subgraph Extraction. The influence of a node is closely related to the structural properties of its surrounding neighborhood. To capture this context, we extract, for each target node u, a fixed-size neighborhood subgraph G_u using Breadth-First Search until L nodes are selected (L is a fixed size, which is essential for efficient mini-batch training, see *Section 5*). Candidate neighbors are ranked within each BFS level by the sum of their min-max normalized betweenness and harmonic centralities. This prioritize structurally important and diffusion-relevant nodes rather than relying only on degree (which may bias toward highly connected hubs, as in many existing CNN-based models).

In the case of nodes sharing identical ranking scores, selection is performed randomly to avoid deterministic bias. Also, if fewer than $L-1$ neighbors are available, zero-padding is applied to maintain a fixed-size representation. The adjacency matrix A^u, which represents the resulting subgraph G_u, preserves the local topology around u and serves as the basis for constructing micro-, meso-, and macro-level structural representations.

Structural Feature Construction. To jointly capture intrinsic node properties and neighborhood structure, each target node u is represented by three structured matrices, D^u, B^u, and H^u,

Table 1: Structural properties used for node representation and their diffusion-aware motivations

Property	Scope	Motivation	Normalized
Degree	Local	Measures a node's immediate connectivity and reflects its direct influence potential over neighboring nodes.	Yes
Clustering Coefficient		Captures the cohesiveness of a node's neighborhood, which affects how influence circulates within tightly connected regions.	No
Betweenness Centrality	Inter-community	Identifies nodes that act as intermediaries in information flow, enabling influence to propagate across different regions of the network.	Yes
Gateway Score		Highlights nodes acting as structural bridges across communities, playing a crucial role in cross-community influence dissemination.	No
Community Closeness		Evaluates how efficiently a node interacts within and across communities, reflecting its accessibility and communication reach.	No
Harmonic Centrality	Global	Captures global reachability while remaining robust to disconnected components, reflecting a node's ability to spread influence rapidly across long distances.	Yes
Eigenvector Centrality		Measures influence by assigning higher importance to nodes connected to other influential nodes, emphasizing global propagation strength. This property is adaptively weighted by β_2 to prevent over-amplification of hubs.	No

encoding micro-, meso-, and macro-level topology, respectively. The three matrices use the same three-channel aggregation scheme: the first channel $X^{(1)}$ represents the target node through its intrinsic features $f(u)$, while the second and third, $X^{(2)}$ and $X^{(3)}$, aggregate first- and second-order neighborhood features, denoted $N(u)$ and $N_2(u)$, respectively:

$$X_u^{(1)} = f(u), \quad X_u^{(2)} = \sum_{v \in N(u)} f(v), \quad X_u^{(3)} = \sum_{w \in N_2(u)} f(w)$$

Sum aggregation is used to preserve both neighborhood size and cumulative structural strength, which are key factors in assessing node influence. Note that mean aggregation can reduce structural signals by creating an equivalence between nodes that exhibit inherently different neighborhood strengths. Also, a three-channel design is adopted to balance representational capacity and model robustness, as additional higher-order channels may increase complexity, overfitting, redundancy and noise.

At each scale, the neighborhood adjacency matrix A^u is enriched with its own structural features: degree and clustering coefficient describe local connectivity at the micro level, betweenness, gateway score, and community closeness characterize inter-community roles at the meso level, and harmonic and eigenvector centralities capture global reachability and influence at the macro level. This feature selection is designed to provide complementary structural information across the three network scales (see Table 1).

Let $G_u = (V_u, E_u)$ be the neighborhood subgraph centered at u, with $V_u = \{u_0, u_1, \ldots, u_{L-1}\}$ and $u_0 \equiv u$. The micro-level matrix D^u is defined as:

$$\underset{\forall i,j \in \{0,\ldots,L-1\}}{D^u[i,j]} = \begin{cases} A^u[i,j]\big(\mu_1 D_{u_j} + \mu_2 C_{u_j}\big) & i = 0,\ j \neq 0 \\ A^u[i,j]\big(\mu_1 D_{u_i} + \mu_2 C_{u_i}\big) & i \neq 0,\ j = 0 \\ D_{u_i} & i = j \\ A^u[i,j] & \text{otherwise.} \end{cases} \tag{1}$$

where D_{u_i} and C_{u_i} denote degree and clustering coefficient of node u_i, respectively, and the weighting coefficients μ_1, μ_2 control their relative contributions.

At the meso level, B^u integrates betweenness centrality (B), gateway score (G), and community closeness (CC):

$$B^u[i,j] = \begin{cases} A^u[i,j]\Big(B_{u_j}\big(1+\omega_1 G_{u_j}\big) + \omega_2\, CC_{u_j}\Big), & i=0,\, j\neq 0, \\ A^u[i,j]\Big(B_{u_i}\big(1+\omega_1 G_{u_i}\big) + \omega_2\, CC_{u_i}\Big), & i\neq 0,\, j=0, \\ G_{u_i}, & i=j, \\ A^u[i,j], & \text{otherwise.} \end{cases} \tag{2}$$

Note that $(1+\omega_1 G)$ enhances betweenness centrality without replacing structural mediation (gateway effects vanish when betweenness is zero). Also, $\omega_2 CC$ captures intra-community accessibility, enabling locally well-positioned nodes to contribute even without bridging roles. This balances structural bridging and community influence in B^u.

At the macro level, H^u combines harmonic (H) and eigenvector (E) centralities:

$$H^u[i,j] = \begin{cases} A^u[i,j]\big(\beta_1 H_{u_j} + \beta_2 E_{u_j}\big), & i=0,\, j\neq 0, \\ A^u[i,j]\big(\beta_1 H_{u_i} + \beta_2 E_{u_i}\big), & i\neq 0,\, j=0, \\ E_{u_i}, & i=j, \\ A^u[i,j], & \text{otherwise.} \end{cases} \tag{3}$$

The coefficients μ_1, μ_2, β_1, β_2, ω_1, and ω_2 are data-driven coefficients that control the relative contributions of the selected structural features and adapt the representation to each network's characteristics (ses *Section 4.7* for more details on how thy are computed). In fact, by integrating these features with A^u, the three matrices preserve the original neighborhood topology while encoding complementary information at different structural scales.

Note that D^u, B^u, and H^u jointly form a nine-channel representation that enriches the node embedding. After normalization and feature-specific weighting, the matrices are organized into multi-channel tensors and passed to the convolutional encoding stage for hierarchical feature extraction. An example for $L = 5$ is shown in *Figure. 2*.

4.3 CNN-Based Encoding

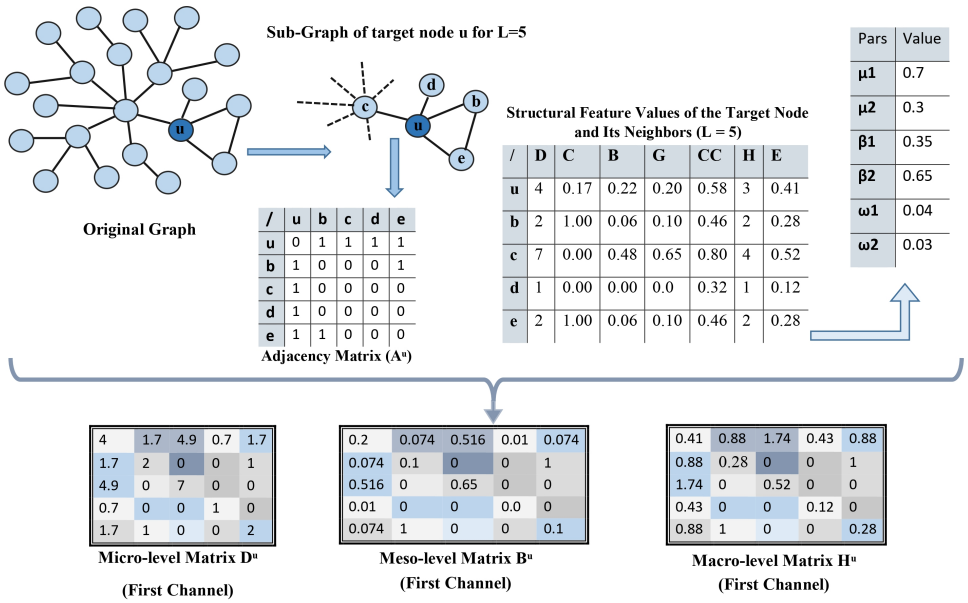
Three parallel CNN branches independently process the structured matrices D^u, B^u, and H^u, corresponding to the micro-, meso-, and macro-level representations. Each matrix is encoded as a $3 \times L \times L$ tensor, whose channels represent the target node and its first- and second-order neighbors.

Each branch consists of two convolutional layers, each followed by a LeakyReLU activation function and a 2×2 max-pooling operation. The first layer maps the 3-channel input to 16 feature maps, while the second increases this to 32 channels. Note that both convolutions use 2×2 kernels with stride and padding set to 1, enabling hierarchical extraction of structural patterns within each scale.

The LeakyReLU activation is defined as:

$$\text{LeakyReLU}(x) = \begin{cases} x, & \text{if } x \geq 0, \\ \kappa x, & \text{if } x < 0, \end{cases} \quad \text{where } \kappa = 0.1$$

LeakyReLU provides stable gradient propagation and avoids the dead-neuron problem of standard ReLU. The max-pooling progressively reduces the spatial dimensions while preserving the most salient structural patterns. After the two convolution–pooling stages, each branch



/	u	b	c	d	e
u	0	1	1	1	1
b	1	0	0	0	1
c	1	0	0	0	0
d	1	0	0	0	0
e	1	1	0	0	0

/	D	C	B	G	CC	H	E
u	4	0.17	0.22	0.20	0.58	3	0.41
b	2	1.00	0.06	0.10	0.46	2	0.28
c	7	0.00	0.48	0.65	0.80	4	0.52
d	1	0.00	0.00	0.0	0.32	1	0.12
e	2	1.00	0.06	0.10	0.46	2	0.28

Pars	Value
μ1	0.7
μ2	0.3
β1	0.35
β2	0.65
ω1	0.04
ω2	0.03

4	1.7	4.9	0.7	1.7
1.7	2	0	0	1
4.9	0	7	0	0
0.7	0	0	1	0
1.7	1	0	0	2

0.2	0.074	0.516	0.01	0.074
0.074	0.1	0	0	1
0.516	0	0.65	0	0
0.01	0	0	0.0	0
0.074	1	0	0	0.1

0.41	0.88	1.74	0.43	0.88
0.88	0.28	0	0	1
1.74	0	0.52	0	0
0.43	0	0	0.12	0
0.88	1	0	0	0.28

Figure 2: *Illustrative construction of D^u, B^u, and H^u for target node u with $L = 5$. A BFS strategy is applied to extract the neighborhood subgraph centered at u, and then its adjacency matrix A^u is constructed. The structured matrices D^u, B^u, and H^u are generated from graph features and the weights $\mu_1, \mu_2, \beta_1, \beta_2, \omega_1, \omega_2$. Only the first channel is shown. The feature and weight values are chosen for illustration only (to show how the matrices are computed in the actual framework).*

produces a $32 \times 10 \times 10$ feature map (for L = 40, as set in *Section 5*). Global average pooling then aggregates each map into a compact 32-dimensional representation, yielding $F_D, F_B, F_H \in \mathbb{R}^{32}$.

These scale-specific vectors summarize the learned structural patterns and are subsequently used for inter-scale modeling. Processing each structural level independently preserves its distinctive information while enabling complementary representations to be integrated in later stages.

4.4 BiGRU-based Fusion

The CNNs extract structural patterns independently at each scale but do not capture dependencies between structural levels. To enable cross-scale modeling, their outputs are stacked into a sequence:

$$X_{\text{seq}} = [F_D, F_B, F_H] \in \mathbb{R}^{3\times 32},$$

This allows the model to treat different structural levels as ordered elements, enabling the learning of interactions across levels. The sequence is then processed by a Bidirectional Gated Recurrent Unit (BiGRU) with hidden size $h = 128$:

$$Z = \text{BiGRU}(X_{\text{seq}}) \in \mathbb{R}^{3\times 256},$$

where 256 results from concatenating the forward and backward hidden states. This sequential fusion captures dependencies among micro-, meso-, and macro-level representations (rather

than treating each scale independently, which is the case for most existing approaches). Since the contribution of different structural levels may vary, a multi-head self-attention mechanism is employed to adaptively emphasize the most informative representations for influence prediction.

4.5 Multi-Head Self-Attention

As not all components contribute equally to the prediction of node influence, a multi-head self-attention mechanism is applied to adaptively weight the cross-scale representations learned by the BiGRU. Following the standard formulation in *Section 3*, the query, key, and value matrices are obtained through learnable projections W^Q, W^K, and W^V, and the scaled dot-product attention is computed as

$$O = \text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V,$$

where d_k denotes the key dimension. Multiple heads operate in parallel to capture complementary interactions among the micro-, meso-, and macro-level representations. Their outputs are concatenated and linearly projected, followed by mean pooling across the sequence dimension to produce a unified embedding:

$$c = \text{Mean}(O) \in \mathbb{R}^{256},$$

The resulting context vector c provides a compact representation of the most informative cross-level dependencies and is passed to the final regression layer for influence prediction.

4.6 Influence Prediction and Training Objective

The final node representation c is passed through two fully connected layers to predict the influence score:

$$\mathbb{R}^{256} \to \mathbb{R}^{1024} \to \mathbb{R}^1 \tag{4}$$

The first layer projects the representation to 1024 dimensions using a LeakyReLU activation, and the second layer outputs a scalar value $\hat{y}_u$, representing the predicted influence of node u.

To train the model, a regression objective is adopted. The ground-truth influence labels are generated using the Susceptible–Infected–Recovered (SIR) diffusion model (for more details, refer to [21]). The SIR model simulates the spreading capability of each node within the network and provides a realistic approximation of its influence. For each node, multiple independent SIR simulations T are performed to ensure statistical stability. The influence score of node u, denoted as y_u, is computed as the average number of infected and recovered nodes across all simulations, following Kitsak *et al.* [29]:

$$y_u = IA(u) = \frac{1}{T}\sum_{j=1}^{T}(|R_j| + |I_j|) \tag{5}$$

where R_j is the set of recovered nodes in the j-th simulation, and I_j is the set of infected nodes remaining at the end of the j-th simulation. This provides a realistic approximation of node influence, reflecting its ability to propagate information through the network.

The model is trained by minimizing the discrepancy between the predicted score $\hat{y}_u$ and the ground-truth score y_u using the Smooth L1 loss function:

$$\text{SmoothL1}(e) = \begin{cases} \frac{e^2}{2}, & |e| < 1 \\ |e| - \frac{1}{2}, & \text{otherwise} \end{cases} \tag{6}$$

Algorithm 1: CNN–BiGRU framework with multi-head self-attention

Input: Graph $G = (V, E)$, neighborhood size L, learning rate lr, epochs N_{ep}
Output: Predicted influence scores $\hat{y}_u$ for all $u \in V$
Preprocessing:
foreach *node* $u \in V$ **do**
  Extract neighborhood subgraph G_u of size L using BFS;
  Construct adjacency matrix A^u;
  Compute structured matrices D^u, B^u, and H^u;
  Encode each matrix as a three-channel tensor;
  Generate influence label y_u using SIR simulations;
Training:
for *epoch* $= 1$ **to** N_{ep} **do**
  Shuffle nodes and create mini-batches;
  foreach *mini-batch* $\mathcal{M}$ **do**
    foreach *node* $u \in \mathcal{M}$ **do**
      $F_D \leftarrow \text{CNN}(D^u)$;
      $F_B \leftarrow \text{CNN}(B^u)$;
      $F_H \leftarrow \text{CNN}(H^u)$;
      $X_{\text{seq}} \leftarrow \text{Stack}(F_D, F_B, F_H)$;
      $\mathcal{Z}_u \leftarrow \text{BiGRU}(X_{\text{seq}})$;
      $\mathcal{O}_u \leftarrow \text{MultiHeadAttention}(\mathcal{Z}_u)$;
      $c_u \leftarrow \text{Mean}(\mathcal{O}_u)$;
      $\hat{y}_u \leftarrow \text{FC}(c_u)$;
    Compute Smooth L1 loss over $\mathcal{M}$ and backpropagate;
    Update parameters using the Adam optimizer with learning rate lr;
return $\hat{y}_u$ *for all* $u \in V$;

where $e = y_u - \hat{y}_u$ denotes the prediction error. This loss function combines the advantages of both L1 and L2 losses, providing robustness to outliers while maintaining sensitivity to small errors.

Algorithm 1 summarizes the different steps of our proposed framework.

4.7 Hyperparameter Definition

The hyperparameters $\mu_1, \mu_2, \beta_1, \beta_2, \omega_1, \omega_2$ are determined using a data-driven strategy that accounts for the scale and distribution of heterogeneous structural features. This calibration ensures balanced feature contributions, prevents large-magnitude features from dominating the representations, and improves their integration during learning and attention-based fusion.

At the micro level, degree centrality is normalized to reduce its dependence on network size, whereas the clustering coefficient already lies in $[0, 1]$. The weights μ_1 and μ_2 are chosen to balance their average contributions:

$$\mu_1 \cdot \mathbb{E}[\deg_{\text{norm}}] \approx \mu_2 \cdot \mathbb{E}[\text{clustering}], \quad \text{under } \mu_1 + \mu_2 = 1$$

Solving this constraint yields:

$$\mu_1 = \frac{\mathbb{E}[\text{clustering}]}{\mathbb{E}[\deg_{\text{norm}}] + \mathbb{E}[\text{clustering}]}, \mu_2 = \frac{\mathbb{E}[\deg_{\text{norm}}]}{\mathbb{E}[\deg_{\text{norm}}] + \mathbb{E}[\text{clustering}]}$$

where $\mathbb{E}[\cdot]$ denotes the empirical mean over all network nodes.

At the community level, ω_1 and ω_2 balance gateway effects, betweenness centrality, and community-level accessibility in B^u. Betweenness centrality is normalized to mitigate its skewed distribution, while gateway score and community closeness remain unnormalized to preserve their structural meaning. The gateway weight is defined as:

$$\omega_1 = \frac{0.1}{\mathbb{E}[\text{gateway}]}$$

providing a moderate, data-adaptive amplification of inter-community connectivity. ω_2 balances bridging and intra-community accessibility by matching their expected contributions:

$$\omega_2 = \frac{\mathbb{E}[\text{betweenness}_{\text{norm}}]}{\mathbb{E}[\text{community_closeness}]}$$

At the macro level, harmonic centrality is normalized to account for variations in network size and path-length distributions, while eigenvector centrality remains unnormalized to preserve its global structural meaning. Following the same principle, β_1 and β_2 satisfy

$$\beta_1 = \frac{\mathbb{E}[\text{eigenvector}]}{\mathbb{E}[\text{harmonic}_{\text{norm}}] + \mathbb{E}[\text{eigenvector}]}, \beta_2 = \frac{\mathbb{E}[\text{harmonic}_{\text{norm}}]}{\mathbb{E}[\text{harmonic}_{\text{norm}}] + \mathbb{E}[\text{eigenvector}]}$$

This network-adaptive formulation of the hyperparameters requires no manual tuning and promotes balanced feature integration across networks with different sizes and topologies.

5 Experimental protocol setup

Problem Formulation and Datasets. We formulate influential node identification as a regression task, using continuous influence scores generated by the SIR model. Unlike binary classification, this formulation provides a more fine-grained representation of node influence. For each node u, we perform $T = 1{,}000$[1] independent SIR simulations, and define its ground-truth influence score as the average number of infected and recovered nodes at the end of the diffusion process. The infection probability is defined relative to the epidemic threshold as $\beta = \alpha\beta_{\text{th}}$, with $\alpha \in [1, 1.9]$, where:

$$\beta_{\text{th}} = \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle}$$

with k denoting node degree, $\langle k \rangle$ the average node degree, and $\langle k^2 \rangle$ the average squared degree over the network. The recovery rate is fixed at $\gamma = 1$ throughout the simulations.

Experiments are conducted on a set of real-world and synthetic networks spanning four domains: social, biological, technological, and knowledge. The datasets exhibit substantial variation in size, density, clustering, and degree distribution, ensuring evaluation across heterogeneous network structures. Their main topological characteristics are reported in *Table 2*.

Evaluation Metrics and Baselines. The ranking performance is evaluated using Kendall's tau (τ) [27], Spearman's rho (ρ) [58], and NDCG [24], which respectively measure pairwise ranking consistency, monotonic agreement, and the accuracy of top-ranked influential nodes. The proposed framework *CSBA-Net* is compared with three groups of baselines:

[1] $T = 1{,}000$ simulations is widely adopted in the literature for influential node identification.

Table 2: *Topological properties of the network datasets.*

Network	Nodes	Edges	Avg. Deg.	Δ	Network	Nodes	Edges	Avg. Deg.	Δ
karate [33]	34	78	4.59	17	E_road [63]	1174	1417	2.41	10
dolphins [33]	62	159	5.13	12	Faa [32]	1226	2408	3.93	34
American_football [33]	115	613	10.66	12	AS_733 [36]	6474	12572	3.88	1458
Hep [37]	9877	25973	5.26	65	Figeys [16]	2239	6432	5.75	314
GrQ [37]	5242	14484	5.53	81	Vidal [56]	3133	6149	3.93	129
Netscience [47]	1461	2742	3.75	34	Bio_Dmela [59]	7393	25569	6.92	190
Stelzl [61]	1706	3155	3.70	95	David [45]	112	425	7.59	49
soc_anybeat [17]	12645	49132	7.77	4800	soc_slashdot [38]	70068	358647	10.24	2507
fb_pages_company [55]	14113	52126	7.39	215	PubMed [35]	19717	44324	4.50	171
web_EPA [12]	4271	8909	4.17	175	maayan_pdzbase [2]	212	242	2.28	21
web_edu [18]	3031	6474	4.27	104	Elegans [15]	453	2025	8.94	237
PowerGrid [13]	4941	6594	2.67	19	tvshow [55]	3892	17239	8.86	126

- *Centrality-based methods*, including Degree Centrality (DC) [67], Betweenness Centrality (BC) [6], k-shell Centrality (KS) [29], and H-index Centrality (HI) [42].
- *CNN-based models*, including RCNN [71], M-RCNN [49], and LCNN [1];
- *Graph learning methods*, including InfGCN [75] and GMHAR [30].

All learning-based baselines are trained using the same SIR-derived ground-truth labels to ensure a fair comparison.

Implementation Details. The proposed framework is implemented in PyTorch and evaluated on a workstation equipped with an NVIDIA GeForce RTX 2070 Max-Q (8 GB VRAM), Intel Core i7-10875H CPU (2.30 GHz), and 16 GB RAM.

The model uses a neighborhood size of $L = 40$, a BiGRU hidden size of 128, and 4 attention heads. Training employs the Adam optimizer with a learning rate of $lr = 5 \times 10^{-4}$, batch size of 32, and 100 epochs ($N_{\text{ep}} = 100$), using mini-batch training for efficient GPU utilization.

For a fair and consistent comparison, we follow the experimental protocol of Yu *et al.* [71], also adopted by M-RCNN [49] and LCNN [1]. The model is trained on a Barabási–Albert (BA) network BA(1000, 4), with $\langle k \rangle = 4$. Yu *et al.* [71] showed, on synthetic BA test networks, that training on a low average degree yields stable generalization across test networks of different densities, whereas training on higher average degrees degrades performance on sparser test networks. The trained model is then evaluated on diverse real-world networks without retraining.

6 Results and Discussion

This section presents a comprehensive evaluation of the proposed framework *CSBA-Net* across 24 benchmark networks. We first compare *CSBA-Net* with different classes of baselines to evaluate its ranking effectiveness. Then, a robustness analysis is performed to evaluate the stability and generalization of the proposed framework across diverse network categories. Finally, we conduct an ablation study to quantify the contribution of each architectural component, followed by a sensitivity analysis to examine the influence of diffusion intensity.

6.1 Overall Performance

We evaluate the ranking performance of CSBA-Net against the considered CNN-based, GNN-based, and centrality-based baselines. Table 3 reports the average Kendall's τ, Spearman's ρ, and NDCG scores over the 24 benchmark networks, with diffusion intensities $\alpha \in [1.0, 1.9]$. CSBA-Net achieves the highest average value for all reported metrics, with $\bar{\tau} = 0.802$, $\bar{\rho} = 0.943$, NDCG@10 $= 0.898$, and NDCG@20 $= 0.916$.

Among the CNN-based approaches, M-RCNN obtains the highest average Kendall's τ of 0.677, followed by LCNN (0.644) and RCNN (0.600), *i.e.*, CSBA-Net improves the average τ by approximately 18.5%, 24.5%, and 33.7%, respectively. The average performance is also higher than that of the GNN-based methods, with $\tau = 0.591$ for InfGCN and $\tau = 0.538$ for GMHAR. Among the centrality baselines, H-index performs best ($\tau = 0.571$) yet still trails the proposed model, while betweenness centrality remains the weakest.

Table 3: *Average ranking scores across 24 datasets and* $\alpha \in [1, 1.9]$. *Best values are shown in bold; NDCG is omitted for InfGCN and GMHAR due to their inductive evaluation protocol.*

Class	Method	Kendall τ	Spearman ρ	NDCG@10	NDCG@20
CNN-based	RCNN	0.600	0.761	0.806	0.811
	M-RCNN	0.677	0.838	0.848	0.846
	LCNN	0.644	0.778	0.841	0.838
	CSBA-Net	**0.802**	**0.943**	**0.898**	**0.916**
GNN-based	InfGCN	0.591	0.713	–	–
	GMHAR	0.538	0.692	–	–
Centrality	DC	0.560	0.731	0.743	0.767
	BC	0.431	0.592	0.628	0.669
	HI	0.571	0.740	0.764	0.778
	KS	0.541	0.710	0.728	0.753

Overall, the high values of both τ and ρ indicate that the ordering predicted by CSBA-Net is well aligned with the reference ranking, *i.e.* a strong ranking agreement with the SIR-derived ground-truth scores. Similarly, the NDCG indicates that the model preserves the most influential nodes near the top of the ranking.

6.2 Performance Across Network Structures and Domains

In this section, we analyze how CSBA-Net performance varies with network structure, size, and sparsity.

Tables 4 and *5* provide the per-network results and give a more detailed view of this aggregate performance. We can see that CSBA-Net obtains the highest Kendall's τ on 23 of the 24 benchmark networks and the highest Spearman's ρ on all 24 networks. It maintains high ranking correlations on both small networks, such as Karate and Dolphins, reaching $\rho = 0.986$ and 0.991, and larger and more complex networks, such as Netscience and AS-733 ($\rho > 0.94$ in both cases). *Figure 3* further illustrates the consistency of the ranking performance across the evaluated datasets.

Performance on sparse and large-scale networks. A larger performance difference is observed on sparse networks, where the average degree is relatively low and local neighborhoods contain fewer directly connected nodes. The results in *Tables 4* and *5* show that CSBA-Net maintains high ranking correlations on such graphs, whereas several CNN-based baselines show a larger degradation. This behavior is particularly visible on E-Road, PowerGrid, and Netscience. On E-Road, for example, CSBA-Net achieves a Kendall's τ of 0.858, compared with 0.056 for RCNN (a relative gain exceeding +1,400%), while on PowerGrid the corresponding values are 0.653 and 0.365 (+78.9%). Netscience shows a similar pattern, with $\tau = 0.911$ for CSBA-Net and 0.511 for RCNN. Similarly, for large-scale networks, CSBA-Net maintains high ranking correlations. For instance, on *Slashdot*, with more than 70,000 nodes, it achieves $\tau = 0.779$ against 0.555 for RCNN (+40.4%) and 0.658 for M-RCNN (+18.4%), while LCNN deteriorates to $\tau = 0.283$.

This behavior can be attributed to the complementary signals added by CSBA-Net. In fact, CSBA-Net incorporates meso- and macro-level structural descriptors, including community-bridging and global-reachability information, which provide useful information to identify influ-

Table 4: Average Kendall's τ per dataset, averaged over $\alpha \in [1.0, 1.9]$.

Network	CNN-based			GNN-based			Centrality			
	CSBA-Net	**RCNN**	**M-RCNN**	**LCNN**	**InfGCN**	**GMHAR**	**DC**	**BC**	**HI**	**KS**
Social networks										
Karate	**0.912**	0.661	0.769	0.798	0.624	0.774	0.656	0.568	0.602	0.533
Dolphins	**0.938**	0.631	0.770	0.864	0.610	0.466	0.701	0.517	0.768	0.510
Am. Football	**0.677**	0.417	0.502	0.150	0.486	0.380	0.532	0.210	0.418	0.105
Anybeat	**0.686**	0.571	0.644	0.198	0.617	0.390	0.415	0.332	0.416	0.417
FB-Company	**0.829**	0.747	0.777	0.695	0.705	0.636	0.544	0.417	0.567	0.566
Slashdot	**0.779**	0.555	0.658	0.283	0.564	0.456	0.670	0.603	0.682	0.683
TVShow	0.712	**0.731**	0.726	0.564	0.638	0.589	0.589	0.327	0.602	0.595
Biological networks										
Stelzl	**0.779**	0.662	0.701	0.724	0.628	0.485	0.463	0.407	0.490	0.491
Figeys	**0.685**	0.589	0.663	0.500	0.567	0.487	0.477	0.440	0.495	0.498
Vidal	**0.827**	0.714	0.760	0.786	0.709	0.573	0.600	0.539	0.637	0.645
Bio_Dmela	**0.896**	0.801	0.832	0.716	0.800	0.629	0.717	0.628	0.747	0.749
PDZBase	**0.875**	0.353	0.704	0.844	0.530	0.606	0.573	0.587	0.618	0.618
Elegans	**0.879**	0.835	0.860	0.812	0.568	0.457	0.649	0.483	0.660	0.667
Technological networks										
Web-EPA	**0.812**	0.717	0.644	0.768	0.649	0.566	0.492	0.460	0.496	0.494
Web_Edu	**0.604**	0.370	0.317	0.375	0.361	0.320	0.286	0.269	0.243	0.268
PowerGrid	**0.653**	0.365	0.499	0.575	0.530	0.550	0.354	0.243	0.372	0.328
E_Road	**0.858**	0.056	0.338	0.649	0.562	0.731	0.461	0.313	0.480	0.486
FAA	**0.847**	0.650	0.722	0.789	0.643	0.582	0.500	0.263	0.564	0.571
AS_733	**0.766**	0.591	0.637	0.569	0.522	0.491	0.589	0.416	0.610	0.612
Knowledge networks										
HEP	**0.858**	0.718	0.784	0.795	0.727	0.621	0.589	0.518	0.621	0.618
GrQc	**0.697**	0.650	0.672	0.630	0.617	0.503	0.627	0.423	0.633	0.624
Netscience	**0.911**	0.511	0.647	0.837	0.520	0.480	0.651	0.351	0.639	0.630
PubMed	**0.788**	0.672	0.696	0.636	0.514	0.648	0.445	0.371	0.478	0.489
David	**0.974**	0.827	0.920	0.888	0.482	0.500	0.853	0.649	0.871	0.793
Average	**0.802**	0.600	0.677	0.644	0.591	0.538	0.560	0.431	0.571	0.541

ential nodes. This is noteworthy for sparse networks, where the immediate neighborhood of a node contains relatively few connections, and local connectivity patterns provide less information to distinguish nodes with different spreading potential. Similarly, for large-scale networks, such representations can retain information about a node's position in the broader network even when the number of nodes and possible diffusion paths increases substantially.

Cross-domain generalization. Using results in *Tables 4* and *5*, we further assess the performance of CSBA-Net across different network domains, which exhibit diverse structural and functional characteristics. For each domain, we compute the average Kendall's τ across the corresponding networks to obtain an overall measure of ranking consistency. CSBA-Net achieves mean Kendall's τ values of 0.790, 0.823, 0.757, and 0.846 for social, biological, technological, and knowledge networks, respectively. It outperforms the best competing method in each category, namely M-RCNN on social and biological domains, and LCNN on technological and knowledge domains, with gains ranging from 9.3% to 21.9%. The variation in average performance across domains highlights the impact of network characteristics on influential-node identification. In particular, the largest gap occurs in technological networks, suggesting they may present more challenging structural properties. RCNN achieves a mean $\tau = 0.458$, versus 0.757 for CSBA-Net (+65.2%). On the challenging Web_Edu network ($\langle k \rangle = 4.27$), CSBA-Net also leads LCNN (0.604 vs. 0.375), a gain of +61.1%.

Scalability and structural variation on Barabási–Albert networks. To examine the effect of network size and density under more controlled conditions, *Tables 6* and *7* report the performance on Barabási–Albert (BA) networks ranging from 1,000 to 6,000 nodes with average degrees of 2, 4, 8, and 10. The sparse series with $\langle k \rangle = 2$ show that as the network size increases from 1,000 to 6,000 nodes, Kendall's τ for CSBA-Net decreases from 0.909 to 0.876, whereas RCNN decreases

Table 5: Average Spearman's ρ per dataset, averaged over $\alpha \in [1.0, 1.9]$.

Network	CNN-based			GNN-based			Centrality			
	CSBA-Net	**RCNN**	**M-RCNN**	**LCNN**	**InfGCN**	**GMHAR**	**DC**	**BC**	**HI**	**KS**
Social networks										
Karate	**0.986**	0.825	0.890	0.926	0.713	0.883	0.821	0.755	0.783	0.740
Dolphins	**0.991**	0.842	0.940	0.968	0.580	0.665	0.863	0.691	0.921	0.686
Am. Football	**0.866**	0.630	0.712	0.220	0.340	0.542	0.712	0.315	0.574	0.148
Anybeat	**0.877**	0.766	0.837	0.299	0.718	0.525	0.569	0.470	0.570	0.571
FB_Company	**0.973**	0.909	0.926	0.842	0.848	0.822	0.730	0.587	0.750	0.749
Slashdot	**0.952**	0.741	0.832	0.444	0.730	0.593	0.849	0.782	0.858	0.859
TVShow	0.927	**0.902**	0.888	0.660	0.825	0.785	0.762	0.473	0.776	0.773
Biological networks										
Stelzl	**0.943**	0.841	0.871	0.889	0.837	0.650	0.640	0.581	0.664	0.664
Figeys	**0.880**	0.785	0.848	0.669	0.788	0.658	0.660	0.620	0.675	0.678
Vidal	**0.977**	0.885	0.921	0.934	0.778	0.762	0.784	0.726	0.819	0.826
Bio_Dmela	**0.989**	0.942	0.956	0.872	0.846	0.803	0.889	0.807	0.908	0.910
PDZBase	**0.995**	0.466	0.881	0.932	0.553	0.760	0.762	0.782	0.804	0.805
Elegans	**0.990**	0.962	0.969	0.946	0.577	0.601	0.830	0.652	0.841	0.852
Technological networks										
Web_EPA	**0.967**	0.878	0.816	0.918	0.837	0.738	0.675	0.641	0.678	0.678
Web_Edu	**0.763**	0.508	0.420	0.485	0.504	0.450	0.393	0.369	0.333	0.369
PowerGrid	**0.835**	0.519	0.688	0.756	0.693	0.720	0.507	0.351	0.532	0.477
E-Road	**0.990**	0.058	0.496	0.827	0.731	0.798	0.630	0.450	0.656	0.675
FAA	**0.989**	0.850	0.897	0.938	0.833	0.719	0.683	0.386	0.748	0.757
AS_733	**0.947**	0.786	0.838	0.752	0.724	0.658	0.780	0.587	0.800	0.802
Knowledge networks										
HEP	**0.984**	0.892	0.934	0.935	0.860	0.775	0.774	0.708	0.801	0.798
GrQc	**0.882**	0.837	0.851	0.738	0.733	0.660	0.804	0.597	0.812	0.805
Netscience	**0.989**	0.623	0.830	0.951	0.702	0.635	0.827	0.505	0.821	0.814
PubMed	**0.953**	0.860	0.870	0.789	0.686	0.733	0.621	0.536	0.656	0.665
David	**0.993**	0.962	0.990	0.979	0.669	0.665	0.969	0.825	0.978	0.943
Average	**0.943**	0.761	0.838	0.778	0.713	0.692	0.731	0.592	0.740	0.710

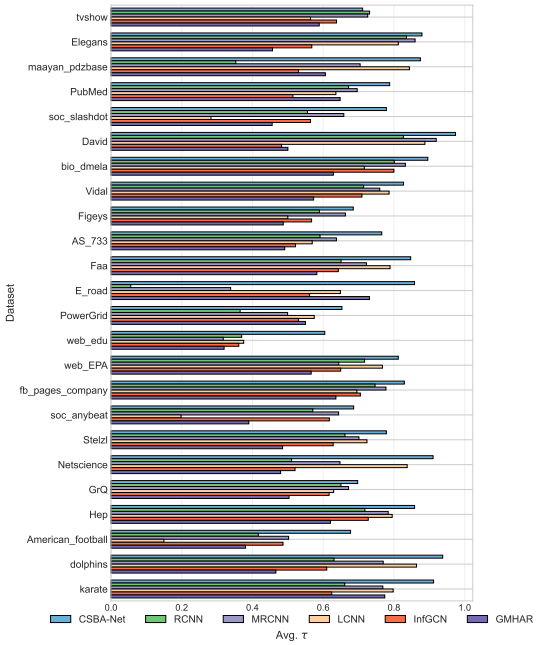
from 0.819 to 0.741 and M-RCNN from 0.820 to 0.722. A similar trend is observed for Spearman's ρ, where CSBA-Net changes from 0.996 to 0.983, compared with a decrease from 0.956 to 0.904 for RCNN and from 0.957 to 0.888 for M-RCNN. Thus, CSBA-Net is not completely insensitive to increasing network size, but its ranking performance exhibits a smaller degradation than the CNN-based baselines. For the remaining BA configurations ($\langle k \rangle \in \{4, 8, 10\}$), the absolute correlation values vary across network sizes and densities for all methods, without a monotonic dependence on density, while CSBA-Net remains the best-performing method in every configuration.

On the other hand, for some BA networks, the performance of CSBA-Net depends on the scale-free level of the network. In fact, when a relatively small number of hubs coexist with many low-degree nodes, local structural descriptors can provide highly heterogeneous information across nodes. The multi-scale representations used by CSBA-Net provide additional global and community-level descriptors that may help distinguish nodes whose local connectivity patterns are similar. The BA experiments show that CSBA-Net achieves the highest Kendall's τ and Spearman's ρ across different configurations, and hence retains high ranking agreement as both network size and average degree vary.

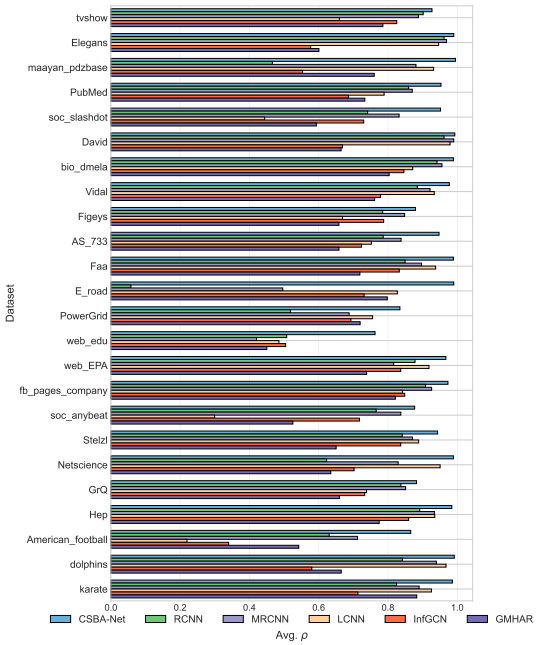
These consistent gains over baselines support the ability of CSBA-Net's multi-scale structural representations to capture topology relevant to node influence across diverse network contexts.

6.3 Ablation Study

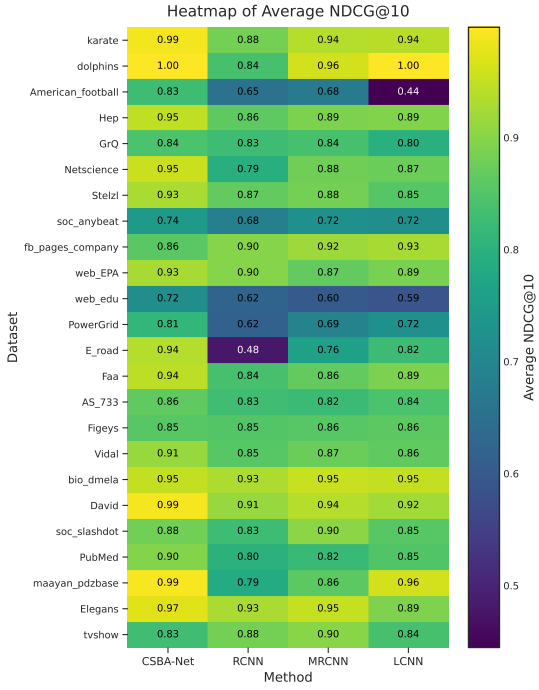
To better understand the contribution of each architectural component to the overall performance of our model, we conduct a controlled ablation study by systematically removing or replacing key modules from the full architecture. Four ablated configurations are evaluated alongside the



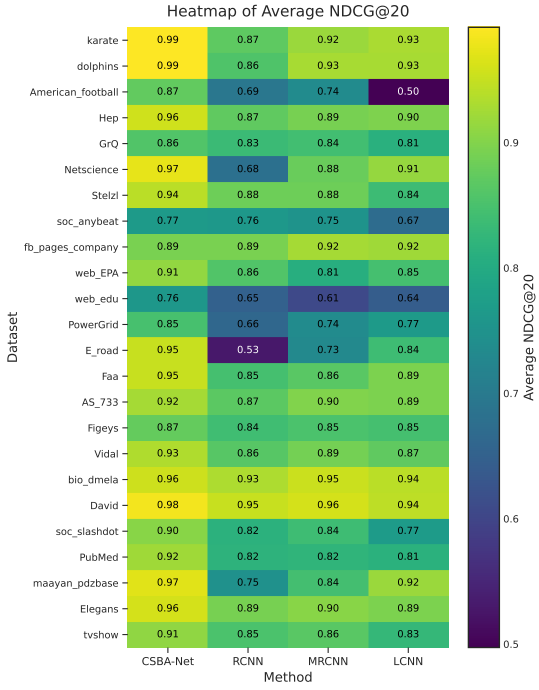
(a) Average Kendall τ



(b) Average of Spearman ρ



(c) Average NDCG@10



(d) Average NDCG@20

Figure 3: *Performance comparison across datasets (top) and heatmap of average NDCG@10 (left) and NDCG@20 (right) per dataset, averaged over* $\alpha \in [1.0, 1.9]$. *Higher values indicate better top-k ranking quality. CSBA-Net achieves the highest scores across the majority of datasets.*

Table 6: *Average Kendall τ on synthetic Barabási–Albert (BA) networks of increasing size and varying average degree.* $\Delta X = (\tau_{CSBA\text{-}Net} - \tau_X)/\tau_X \times 100$. *Results averaged over* $\alpha \in [1.0, 1.9]$. *Best value per row in bold.*

	Average Kendall τ						Relative gain of CSBA-Net (%)				
Network	**CSBA-Net**	**RCNN**	**M-RCNN**	**LCNN**	**InfGCN**	**GMHAR**	**ΔRCNN**	**ΔM-RCNN**	**ΔLCNN**	**ΔInfGCN**	**ΔGMHAR**
1 000 nodes											
BA-1k-2	**0.909**	0.819	0.820	0.866	0.343	0.295	+11.0%	+10.9%	+5.0%	+165.0%	+207.9%
BA-1k-4	**0.323**	0.275	0.274	0.283	0.243	0.038	+17.7%	+18.0%	+14.1%	+32.9%	+755.7%
BA-1k-8	**0.836**	0.807	0.807	0.760	0.410	0.407	+3.5%	+3.5%	+9.9%	+103.9%	+105.4%
BA-1k-10	**0.530**	0.457	0.470	0.434	0.427	0.283	+15.9%	+12.7%	+22.2%	+24.1%	+87.4%
2 000 nodes											
BA-2k-2	**0.896**	0.805	0.810	0.843	0.735	0.319	+11.3%	+10.5%	+6.3%	+21.9%	+180.6%
BA-2k-4	**0.907**	0.805	0.818	0.850	0.759	0.312	+12.7%	+10.9%	+6.7%	+19.5%	+190.4%
3 000 nodes											
BA-3k-2	**0.902**	0.806	0.813	0.848	0.749	0.314	+11.9%	+10.9%	+6.3%	+20.5%	+187.5%
BA-3k-4	**0.303**	0.263	0.266	0.276	0.255	0.180	+15.2%	+14.0%	+10.0%	+18.7%	+68.5%
BA-3k-10	**0.467**	0.360	0.378	0.365	0.413	0.234	+29.7%	+23.6%	+27.8%	+13.0%	+99.8%
4 000–6 000 nodes (sparse, $\langle k \rangle = 2$)											
BA-4k-2	**0.901**	0.799	0.810	0.843	0.662	0.356	+12.8%	+11.3%	+6.9%	+36.1%	+153.5%
BA-5k-2	**0.890**	0.772	0.762	0.838	0.673	0.325	+15.2%	+16.8%	+6.2%	+32.2%	+173.7%
BA-6k-2	**0.876**	0.741	0.722	0.823	0.673	0.305	+18.2%	+21.3%	+6.4%	+30.0%	+187.6%

Table 7: *Average Spearman ρ on synthetic Barabási–Albert (BA) networks of increasing size and varying average degree.* $\Delta X = (\rho_{CSBA\text{-}Net} - \rho_X)/\rho_X \times 100$. *Results averaged over* $\alpha \in [1.0, 1.9]$. *Best value per row in bold.*

	Average Spearman ρ						Relative gain of CSBA-Net (%)				
Network	**CSBA-Net**	**RCNN**	**M-RCNN**	**LCNN**	**InfGCN**	**GMHAR**	**ΔRCNN**	**ΔM-RCNN**	**ΔLCNN**	**ΔInfGCN**	**ΔGMHAR**
1 000 nodes											
BA-1k-2	**0.996**	0.956	0.957	0.956	0.603	0.475	+4.2%	+4.1%	+4.2%	+65.2%	+109.8%
BA-1k-4	**0.449**	0.399	0.398	0.410	0.343	0.057	+12.6%	+12.8%	+9.5%	+30.8%	+693.6%
BA-1k-8	**0.981**	0.949	0.950	0.919	0.610	0.569	+3.4%	+3.3%	+6.8%	+60.9%	+72.3%
BA-1k-10	**0.712**	0.633	0.650	0.607	0.574	0.411	+12.4%	+9.6%	+17.4%	+24.0%	+73.4%
2 000 nodes											
BA-2k-2	**0.992**	0.947	0.951	0.968	0.795	0.453	+4.8%	+4.3%	+2.5%	+24.8%	+118.9%
BA-2k-4	**0.996**	0.949	0.956	0.970	0.865	0.447	+5.0%	+4.2%	+2.6%	+15.1%	+122.9%
3 000 nodes											
BA-3k-2	**0.994**	0.949	0.954	0.970	0.865	0.446	+4.7%	+4.2%	+2.5%	+14.9%	+123.0%
BA-3k-4	**0.422**	0.381	0.386	0.399	0.345	0.119	+10.6%	+9.3%	+5.7%	+22.3%	+253.8%
BA-3k-10	**0.637**	0.510	0.533	0.515	0.590	0.344	+24.9%	+19.4%	+23.6%	+8.0%	+85.3%
4 000–6 000 nodes (sparse, $\langle k \rangle = 2$)											
BA-4k-2	**0.994**	0.945	0.951	0.946	0.891	0.505	+5.1%	+4.4%	+5.1%	+11.5%	+96.6%
BA-5k-2	**0.989**	0.927	0.920	0.945	0.905	0.464	+6.8%	+7.6%	+4.7%	+9.3%	+113.2%
BA-6k-2	**0.983**	0.904	0.888	0.939	0.910	0.433	+8.8%	+10.8%	+4.8%	+8.1%	+127.2%

complete model:

(i) *CSBA-Net_NoBiGRU*, in which the BiGRU-based cross-scale fusion is replaced by direct concatenation;
(ii) *CSBA-Net_NoAttention*, in which the multi-head self-attention layer is removed;
(iii) *CSBA-Net_SingleScale*, which uses only the micro-level representation D^u; and
(iv) *CSBA-Net_RandomNeighbor*, in which the proposed structured neighborhood selection is replaced by uniform random sampling. All other components and the evaluation protocol are kept unchanged.

From *Table 8*, removing the BiGRU module causes the largest performance degradation among the evaluated variants, with average Kendall's τ decreasing from 0.8018 to 0.4216, corresponding to a relative drop of 47.4%. The degradation is particularly pronounced on several networks, where the ranking performance approaches low or even negative correlation with the SIR-based ground truth. As the BiGRU is used to capture the cross-scale dependencies, these results show the usefulness of explicitly modeling interactions between the three structural scales, and that a simple concatenation of the scale-specific features is insufficient to reproduce the performance of the full model.

The single-scale variant, which relies only on D^u, produces the second-largest degradation, with τ falling to 0.5878 (-26.7%). This drop is particularly pronounced on networks where global structural position carries substantial information about diffusion potential, such as *GrQc*, *PubMed*, and *HEP*. This is because in these large, sparsely connected networks, micro-level

connectivity patterns alone are less discriminative as many nodes share similar local degree and clustering profiles. On the other hand, these nodes differ substantially in their global reachability and bridging roles, properties that are only captured by B^u and H^u.

Replacing the structured neighborhood selection with random sampling reduces the average τ to 0.6901 (-13.9%), especially on networks with skewed degree distributions or strong community structure, such as *HEP*, *Elegans*, and *GrQc*. This confirms that the betweenness–harmonic ranking provides more informative local contexts and preserves structural patterns that are more relevant to influence estimation, whereas random sampling weakens this signal and introduces less informative neighborhoods.

Finally, removing the multi-head self-attention layer leads to the smallest, though still consistent, decline, with τ decreasing to 0.7234 (-9.8%). The decline is relatively uniform across different networks. Without attention, the BiGRU output is passed forward, with each scale contributing according to a fixed, learned weighting rather than being adaptively reweighted according to the characteristics of each node's neighborhood. The smaller drop compared with removing the BiGRU indicates that cross-scale sequential modeling has a larger contribution to the overall ranking performance, while the attention mechanism provides an additional refinement of the learned representation.

Table 8: *Ablation study: average Kendall's τ per dataset over $\alpha \in [1.0, 1.9]$. Best values per row are in bold. w/o BiGRU replaces the BiGRU with direct concatenation; w/o Attn removes multi-head self-attention; Single-Scale uses only the micro-level matrix D^u; Random Nbr replaces BFS-based structural selection with random sampling.*

Network	**CSBA-Net (Full)**	**w/o Attn**	**w/o BiGRU**	**Single-Scale**	**Random Nbr**
Social networks					
Karate	**0.9120**	0.8178	0.7655	0.7977	0.8092
Dolphins	**0.9378**	0.8465	0.8152	0.7669	0.8552
Am. Football	**0.6773**	0.6093	0.3817	0.5264	0.5895
Soc_anybeat	**0.6861**	0.6158	−0.2679	0.4985	0.5159
FB_Company	**0.8294**	0.7488	0.4644	0.6064	0.7560
Soc_slashdot	**0.7786**	0.7233	−0.3522	0.6369	0.6031
TVShow	**0.7115**	0.6259	0.0471	0.3535	0.6531
Biological networks					
Stelzl	**0.7786**	0.7255	0.1921	0.4972	0.6967
Figeys	**0.6850**	0.6274	0.1610	0.5809	0.5711
Vidal	**0.8271**	0.7575	0.6745	0.6460	0.7343
Bio_Dmela	**0.8958**	0.8321	0.3756	0.7783	0.8011
PDZBase	**0.8747**	0.7525	0.7302	0.7108	0.7251
Elegans	**0.8792**	0.8094	0.1827	0.5216	0.7504
Technological networks					
Web_EPA	**0.8122**	0.7102	0.6687	0.6615	0.7204
Web_Edu	**0.6044**	0.4519	0.2819	0.3333	0.4849
PowerGrid	**0.6530**	0.6076	0.5721	0.5441	0.6062
E_Road	**0.8582**	0.7721	0.7053	0.6749	0.7146
FAA	**0.8473**	0.7655	0.7146	0.7119	0.7214
AS_733	**0.7655**	0.6921	−0.2660	0.5840	0.5476
Knowledge networks					
Hep	**0.8583**	0.7865	0.5742	0.4866	0.7160
GrQc	**0.6974**	0.6163	0.3504	0.2096	0.6215
Netscience	**0.9107**	0.8415	0.7578	0.7126	0.7637
PubMed	**0.7879**	0.7228	0.7488	0.4201	0.7310
David	**0.9741**	0.9037	0.8399	0.8483	0.8753
Average	**0.8018**	0.7234	0.4216	0.5878	0.6901
Δ (%)	—	−9.8	−47.4	−26.7	−13.9

All these results, together, support the complementary roles of multi-scale structural representations, structured neighborhood selection, sequential cross-scale fusion, and attention in the

proposed architecture. In other words, D^u, B^u, and H^u encode complementary but individually incomplete views of a node's structural context.

Figure 4 illustrates these differences more clearly and shows that they are not confined to a single dataset.

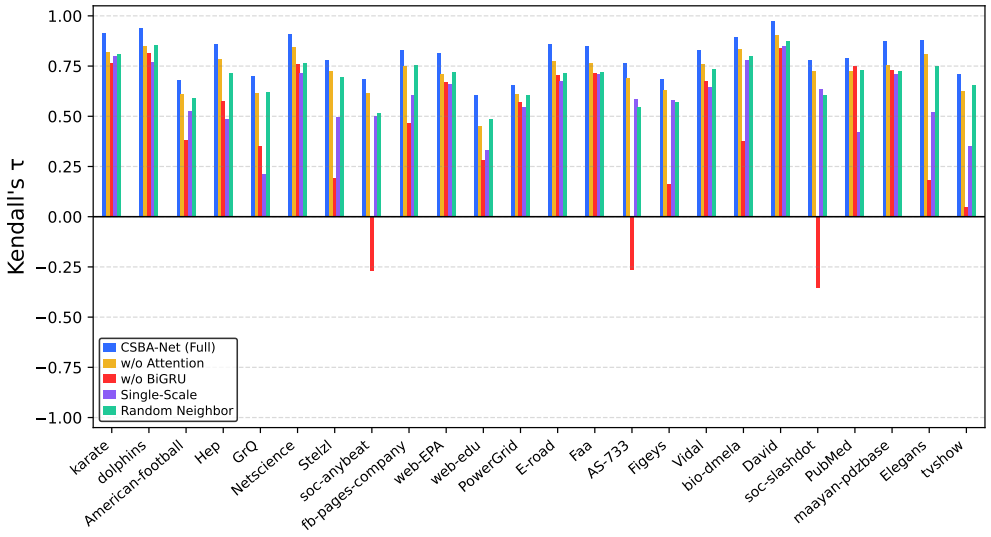


Figure 4: *Illustration of the average Kendall's τ per dataset over $\alpha \in [1.0, 1.9]$ according to the Ablation study results in Table 8.*

6.4 Sensitivity and Stability to Diffusion Intensity

Assessing model stability across diffusion conditions is essential for establishing its practical reliability. We evaluate sensitivity of CSBA-Net to the diffusion intensity by varying the SIR transmission parameter through the scaling factor α. Specifically, the transmission probability is defined as $\beta = \alpha\beta_{\text{th}}$, where α controls the diffusion intensity and is varied from 1.0 to 1.9. We therefore analyze the evolution of Kendall's τ as a function of α. Note that, unlike threshold-based binary classification approaches, our regression framework produces continuous influence scores, requiring rankings to remain consistent with the ground truth across diffusion conditions.

We conduct this analysis on all 24 benchmark real-world networks, and the obtained results are reported in *Figures 5* and *6*.

Performance trend with α. CSBA-Net achieves the highest Kendall's τ on 19 of 24 networks at $\alpha = 1.0$ and on 23 of 24 for all $\alpha \geq 1.2$, without α-specific tuning. The mean Kendall's τ across the 24 networks increases from 0.776 at $\alpha = 1.0$ to 0.822 at $\alpha = 1.8$, then stabilizes at 0.821 at $\alpha = 1.9$. This suggests that increasing infection probability makes influential-node rankings more discriminative, benefiting methods that capture structural spreading capacity. In fact, CSBA-Net's multi-scale representations, encoded in D^u, B^u, and H^u, enable it to capture structural properties relevant to diffusion across multiple propagation scales.

Behavior on sparse networks. Baseline methods show greater instability on sparse and irregular networks. On PowerGrid ($\langle k \rangle = 2.67$), RCNN and LCNN exhibit substantial declines in Kendall's τ, from 0.425 to 0.320 and from 0.715 to 0.485, respectively, whereas CSBA-Net remains within $[0.620, 0.686]$. On E_Road ($\langle k \rangle = 2.41$), the performance of CSBA-Net varies from 0.908 at $\alpha = 1.3$ to 0.733 at $\alpha = 1.9$, reflecting the challenges of ranking nodes in highly sparse

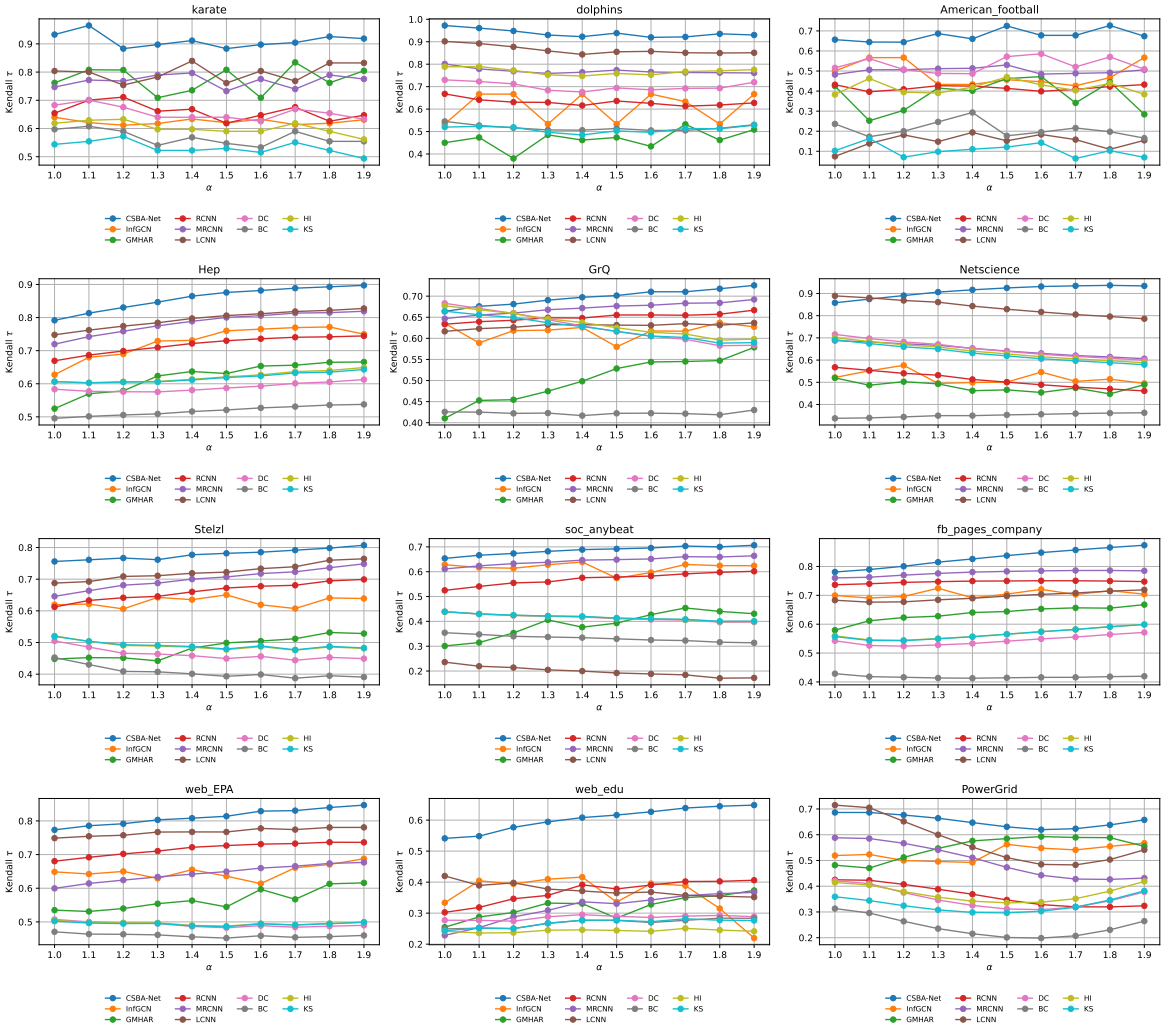


Figure 5: Ranking performance comparison on complex networks: Kendall's τ versus the infection parameter $\alpha \in [1, 1.9]$. Higher τ indicates better agreement with SIR spreading influence.

networks under strong diffusion. Nevertheless, it outperforms RCNN, whose τ remains below 0.068.

Stability across α. Based on the deviation of Kendall's τ across α, we see that CSBA-Net remains stable, with a mean standard deviation of 0.024 across the 24 networks and a maximum of 0.052 on E_Road. LCNN and M-RCNN reach higher deviations of 0.084 and 0.064, respectively, on PowerGrid. Across network categories, CSBA-Net exhibits similar variability in social (0.022), biological (0.021), and knowledge (0.025) networks, with slightly higher variation in technological networks (0.029), reflecting the difficulty posed by sparse, geographically constrained infrastructure graphs such as PowerGrid and E_Road [41, 29].

Saturation at high diffusion intensity. CSBA-Net's mean Kendall's τ increases with α, peaking at $\alpha = 1.8$ and remaining nearly stable over $\alpha \in [1.5, 1.9]$, with a variation below 0.013. This plateau suggests that ranking quality saturates under moderate-to-high diffusion, where structural spreading capacity becomes more informative. Performance is slightly lower near $\alpha = 1.0$,

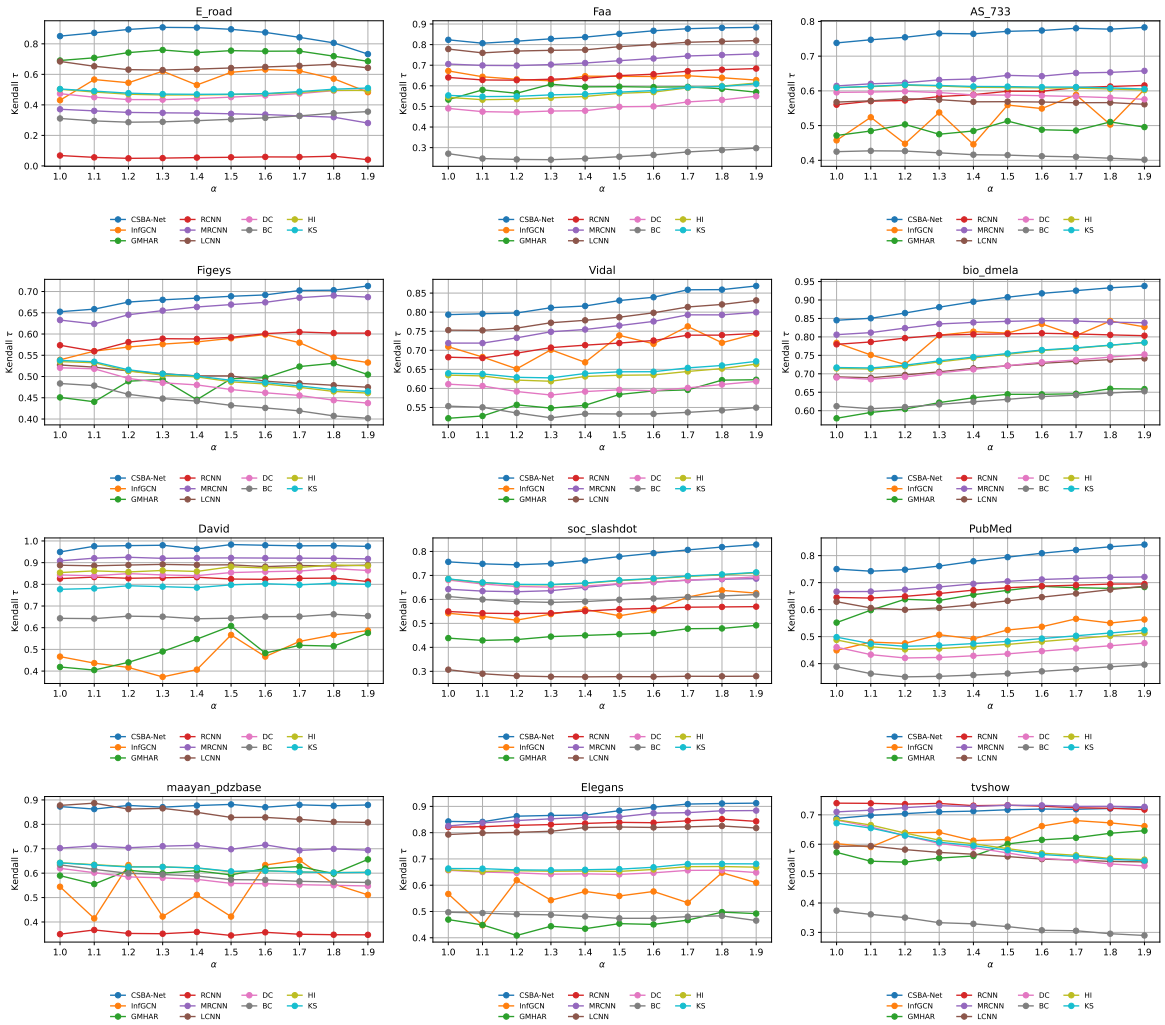


Figure 6: Ranking performance comparison on complex networks: Kendall's τ versus the infection parameter $\alpha \in [1, 1.9]$. Higher τ indicates better agreement with SIR spreading influence.

where stochastic effects around the epidemic threshold introduce greater variability.

Overall, the results show that the proposed framework remains robust across diffusion intensities and does not rely on a specific α. Its stable ranking performance, including on sparse networks, supports the ability of its multi-scale structural representations to capture topology-driven properties relevant to spreading potential. Also, the $T = 1{,}000$ independent SIR simulations provide sufficiently stable ground-truth labels for robust regression.

7 Conclusions and Future Work

In this paper, we introduced CSBA-Net (*Cross-Scale Structural BiGRU Attention Network*), a novel deep learning framework for influential node identification that addresses key limitations of existing approaches, including single-scale structural representations, binary formulations, and limited modeling of dependencies across different network scales. CSBA-Net integrates comple-

mentary micro-, meso-, and macro-level representations that capture local connectivity, inter-community bridging, and global reachability, respectively. These representations are processed independently by CNN branches and subsequently fused through a BiGRU and a multi-head self-attention mechanism to model dependencies across structural scales. In addition, the problem is formulated as a continuous regression task using SIR-based influence scores, enabling more fine-grained estimation of node influence.

Extensive experiments on 24 real-world networks from four domains, together with 12 synthetic Barabási–Albert networks, demonstrate the effectiveness and generalization of the proposed framework. CSBA-Net consistently outperforms centrality-based, CNN-based, and GNN-based baselines across Kendall's τ, Spearman's ρ, NDCG@10, and NDCG@20, achieving average scores of $\bar{\tau} = 0.802$, $\bar{\rho} = 0.943$, NDCG@10 $= 0.898$, and NDCG@20 $= 0.916$. It ranks first in Spearman's ρ on all 24 real-world networks and in Kendall's τ on 23 of them. Notably, the model is trained only on a single synthetic BA network and transferred directly to real-world networks without retraining, demonstrating strong cross-domain generalization. The largest gains are observed on structurally challenging networks, particularly sparse and heterogeneous graphs, where conventional methods degrade more substantially.

Ablation results confirm the importance of the proposed components, with BiGRU-based inter-scale fusion providing the largest contribution, followed by the multi-scale representation, structured neighborhood selection, and multi-head attention. Sensitivity and robustness analyses further show that performance remains stable across diffusion conditions, network domains, and network sizes. All these results support CSBA-Net as a generalizable framework that combines multi-scale structural encoding, inter-scale dependency modeling, and continuous influence estimation within a unified architecture.

Limitations and Future Work. Despite these results, several limitations remain. First, CSBA-Net is currently designed for static, unweighted, and undirected networks. Extending the framework to temporal, directed, and weighted graphs would enable it to model evolving structures and weighted interactions in a broader range of real-world applications. Second, the computational cost of the current pipeline may limit its application to very large networks, particularly due to centrality computations such as betweenness, harmonic, and eigenvector centrality. Future work should therefore investigate scalable approximations and sampling-based strategies to reduce the cost of structural feature extraction and enable its application to larger graphs. Third, the framework relies exclusively on topological information and does not incorporate node attributes such as semantic representations or behavioral information. Integrating these signals with the proposed multi-scale structural representations could improve influence estimation, particularly in social and knowledge networks. Finally, extending CSBA-Net to temporal and attributed settings raises additional challenges in dynamically updating structural representations and modeling the evolution of influence over time.

References

[1] W. Ahmad, B. Wang, and S. Chen. Learning to rank influential nodes in complex networks via convolutional neural networks. *Applied Intelligence*, 54(4):3260–3278, 2024.

[2] T. Beuming, L. Skrabanek, M. Y. Niv, P. Mukherjee, and H. Weinstein. Pdzbase: a protein–protein interaction database for pdz-domains. *Bioinformatics*, 21(6):827–828, 2005.

[3] R. Bhattacharya, N. K. Nagwani, and S. Tripathi. Detecting influential nodes with topological structure via graph neural network approach in social networks. *International Journal of Information Technology*, 15(4):2233–2246, 2023.

[4] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4-5):175–308, 2006.

[5] P. Bonacich. Some unique properties of eigenvector centrality. *Social networks*, 29(4):555–564, 2007.

[6] U. Brandes. A faster algorithm for betweenness centrality. *Journal of mathematical sociology*, 25(2):163–177, 2001.

[7] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[8] J. Cai, J. Luo, S. Wang, and S. Yang. Feature selection in machine learning: A new perspective. *Neurocomputing*, 300:70–79, 2018.

[9] J. Chen, T. Ma, and C. Xiao. Fastgcn: fast learning with graph convolutional networks via importance sampling. *arXiv preprint arXiv:1801.10247*, 2018.

[10] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[11] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[12] W. De Nooy, A. Mrvar, and V. Batagelj. *Exploratory social network analysis with Pajek*, volume 27. Cambridge University Press, 2011.

[13] W. DJ. Collective dynamics of'small-world'networks. *Nature*, 393(6684):409–410, 1998.

[14] Z. Dong, M. Tian, Y. Lu, J. Lai, R. Tang, and X. Li. Impact of core-periphery structure on cascading failures in interdependent scale-free networks. *Physics Letters A*, 383(7):607–616, 2019.

[15] J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 72(2):027104, 2005.

[16] R. M. Ewing, P. Chu, F. Elisma, H. Li, P. Taylor, S. Climie, L. McBroom-Cerajewski, M. D. Robinson, L. O'Connor, M. Li, et al. Large-scale mapping of human protein–protein interactions by mass spectrometry. *Molecular systems biology*, 3(1):MSB4100134, 2007.

[17] M. Fire, R. Puzis, and Y. Elovici. Link prediction in highly fractional data sets. In *Handbook of computational approaches to counterterrorism*, pages 283–300. Springer, 2012.

[18] D. Gleich, L. Zhukov, and P. Berkhin. Fast parallel pagerank: A linear system approach. *Yahoo! Research Technical Report YRL-2004-038, available via http://research. yahoo. com/publication/YRL-2004-038. pdf*, 13:22, 2004.

[19] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[20] H. Guo, Z. Wang, S. Sun, and C. Xia. Interplay between epidemic spread and information diffusion on two-layered networks with partial mapping. *Physics Letters A*, 398:127282, 2021.

[21] H. W. Hethcote. The mathematics of infectious diseases. *SIAM review*, 42(4):599–653, 2000.

[22] H. Huang, L. Sun, B. Du, C. Liu, W. Lv, and H. Xiong. Representation learning on knowledge graphs for node importance estimation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 646–655, 2021.

[23] O. A. Hussain and F. Zaidi. Influence maximization in complex networks through supervised machine learning. In *International Conference on Complex Networks and Their Applications*, pages 217–228. Springer, 2021.

[24] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM*

Transactions on Information Systems (TOIS), 20(4):422–446, 2002.

[25] T. Jiang, Y. Ruan, T. Yu, L. Bai, and Y. Yuan. Identifying influential nodes in complex networks via transformer with multi-scale feature fusion. *Big Data and Cognitive Computing*, 9(5):129, 2025.

[26] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[27] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1-2):81–93, 1938.

[28] M. Kimura, K. Saito, R. Nakano, and H. Motoda. Extracting influential nodes on a social network for information diffusion. *Data Mining and Knowledge Discovery*, 20:70–97, 2010.

[29] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse. Identification of influential spreaders in complex networks. *Nature physics*, 6(11):888–893, 2010.

[30] J. Kou, P. Jia, J. Liu, J. Dai, and H. Luo. Identify influential nodes in social networks with graph multi-head attention regression model. *Neurocomputing*, 530:23–36, 2023.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[32] J. Kunegis. Konect: the koblenz network collection. In *Proceedings of the 22nd international conference on world wide web*, pages 1343–1350, 2013.

[33] J. Kunegis. KONECT – the koblenz network collection. http://konect.cc/networks/, Dec. 2013. Accessed: 2026-02-21.

[34] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.

[35] E. W. Lee and J. C. Ho. Pgb: A pubmed graph benchmark for heterogeneous network representation learning. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 5331–5335, 2023.

[36] J. Leskovec and A. Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[37] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM transactions on Knowledge Discovery from Data (TKDD)*, 1(1):2–es, 2007.

[38] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

[39] Y. Li, D. Tarlow, M. Brockschmidt, and R. Zemel. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*, 2015.

[40] L. Lü, Y.-C. Zhang, C. H. Yeung, and T. Zhou. Leaders in social networks, the delicious case. *PloS one*, 6(6):e21202, 2011.

[41] L. Lü, D. Chen, X.-L. Ren, Q.-M. Zhang, Y.-C. Zhang, and T. Zhou. Vital nodes identification in complex networks. *Physics reports*, 650:1–63, 2016. ISSN 0370-1573.

[42] L. Lü, T. Zhou, Q.-M. Zhang, and H. E. Stanley. The h-index of a network node and its relation to degree and coreness. *Nature communications*, 7(1):10168, 2016.

[43] M. Marchiori and V. Latora. Harmony in the small-world. *Physica A: Statistical Mechanics and its Applications*, 285(3-4):539–546, 2000.

[44] A. S. Mata. An overview of epidemic models with phase transitions to absorbing states running on top of complex networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 31(1), 2021.

[45] N. Meghanathan. Spectral radius as a measure of variation in node degree for complex network graphs. *Proceedings of the 2015 International Conference on Data Science and Advanced Analytics*, 2015.

[46] M. E. Newman. The structure and function of complex networks. *SIAM review*, 45(2): 167–256, 2003.

[47] M. E. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 74(3):036104, 2006.

[48] K. Okamoto, W. Chen, and X.-Y. Li. Ranking of closeness centrality for large-scale social networks. In *International workshop on frontiers in algorithmics*, pages 186–195. Springer, 2008.

[49] Y. Ou, Q. Guo, J.-L. Xing, and J.-G. Liu. Identification of spreading influence nodes via multi-level structural attributes based on the graph convolutional network. *Expert Systems with Applications*, 203:117515, 2022. ISSN 0957-4174.

[50] A. Patel and B. Singh. Graph convolutional network for structural equivalent key nodes identification in complex networks. *Chaos, Solitons & Fractals*, 196:116376, 2025.

[51] X. Qiu, L. Zhao, J. Wang, X. Wang, and Q. Wang. Effects of time-dependent diffusion behaviors on the rumor spreading in social networks. *Physics Letters A*, 380(24):2054–2063, 2016.

[52] M. A. Ramadhan and A. O. Mohammed. A lightweight deep learning-based model for ranking influential nodes in complex networks. *arXiv preprint arXiv:2507.19702*, 2025.

[53] G. Ramírez-de-la Rosa, E. Villatoro-Tello, H. Jiménez-Salazar, and C. Sánchez-Sánchez. Towards automatic detection of user influence in twitter by means of stylistic and behavioral features. In *Human-Inspired Computing and Its Applications*, pages 245–256. Springer, Springer International Publishing, 2014. ISBN 978-3-319-13647-9.

[54] A. A. Rezaei, J. Munoz, M. Jalili, and H. Khayyam. A machine learning-based approach for vital node identification in complex networks. *Expert Systems with Applications*, 214: 119086, 2023.

[55] B. Rozemberczki, R. Davies, R. Sarkar, and C. Sutton. Gemsec: Graph embedding with self clustering. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 65–72, 2019.

[56] J.-F. Rual, K. Venkatesan, T. Hao, T. Hirozane-Kishikawa, A. Dricot, N. Li, G. F. Berriz, F. D. Gibbons, M. Dreze, N. Ayivi-Guedehoussou, et al. Towards a proteome-scale map of the human protein–protein interaction network. *Nature*, 437(7062):1173–1178, 2005.

[57] E. Ruiz Vargas and L. M. Wahl. The gateway coefficient: a novel metric for identifying critical connections in modular networks. *The European Physical Journal B*, 87(7):161, 2014.

[58] P. Sedgwick. Spearman's rank correlation coefficient. *Bmj*, 349, 2014.

[59] R. Singh, J. Xu, and B. Berger. Global alignment of multiple protein interaction networks with application to functional orthology detection. *PNAS*, 105(35):12763–12768, 2008.

[60] O. Smith, J. Crowe, E. Farcot, R. O'Dea, and K. Hopcraft. Cascading failures in networks of heterogeneous node behavior. *Physical Review E*, 101(2):020301, 2020.

[61] U. Stelzl, U. Worm, M. Lalowski, C. Haenig, F. H. Brembeck, H. Goehler, M. Stroedicke, M. Zenkner, A. Schoenherr, S. Koeppen, et al. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–968, 2005.

[62] M. Stolarski, A. Piróg, and P. Bródka. Identifying key nodes for the influence spread using a machine learning approach. *Entropy*, 26(11):955, 2024.

[63] L. Šubelj and M. Bajec. Robust network community detection using balanced propagation. *The European Physical Journal B*, 81(3):353–362, 2011.

[64] Y. Tan, J. Wu, H. Z. Deng, et al. Evaluation method for node importance based on node contraction in complex networks. *Systems Engineering-Theory & Practice*, 11(11):79–83, 2006.

[65] J. Tang, J. Qu, S. Song, Z. Zhao, and Q. Du. Gcnt: Identify influential seed set effectively in social networks by integrating graph convolutional networks with graph transformers. *Journal of King Saud University-Computer and Information Sciences*, 36(8):102183, 2024.

[66] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[67] D. Wei, Y. Li, Y. Zhang, and Y. Deng. Degree centrality based on the weighted network. In *2012 24th Chinese Control and Decision Conference (CCDC)*, pages 3976–3979, 2012.

[68] X. Wen, C. Tu, M. Wu, and X. Jiang. Fast ranking nodes importance in complex networks based on ls-svm method. *Physica A: Statistical Mechanics and its Applications*, 506:11–23, 2018. ISSN 0378-4371.

[69] Y. Wu, A. Dong, Y. Ren, and Q. Jiang. Identify influential nodes in complex networks: A k-orders entropy-based method. *Physica A: Statistical Mechanics and its Applications*, 632: 129302, 2023.

[70] Y. Wu, Y. Hu, S. Yin, B. Cai, X. Tang, and X. Li. A graph convolutional network model based on regular equivalence for identifying influential nodes in complex networks. *Knowledge-Based Systems*, 301:112235, 2024.

[71] E.-Y. Yu, Y.-P. Wang, Y. Fu, D.-B. Chen, and M. Xie. Identifying critical nodes in complex networks via graph convolutional networks. *Knowledge-Based Systems*, 198:105893, 2020.

[72] A. Zeng and C.-J. Zhang. Ranking spreaders by decomposing complex networks. *Physics Letters A*, 377(14):1031–1035, 2013.

[73] M. Zhang, X. Wang, L. Jin, M. Song, and Z. Li. A new approach for evaluating node importance in complex networks via deep learning methods. *Neurocomputing*, 497:13–27, 2022.

[74] G. Zhao, P. Jia, C. Huang, A. Zhou, and Y. Fang. A machine learning based framework for identifying influential nodes in complex networks. *IEEE Access*, 8:65462–65471, 2020.

[75] G. Zhao, P. Jia, A. Zhou, and B. Zhang. Infgcn: Identifying influential nodes in complex networks with graph convolutional networks. *Neurocomputing*, 414:18–26, 2020.