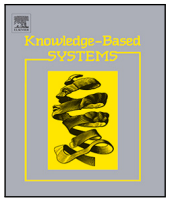
Knowledge-Based Systems 283 (2024) 111163

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



*Olap*GN: A multi-layered graph convolution network-based model for locating influential nodes in graph networks

Yasir Rashid, Javaid Iqbal Bhat *

Department of Computer Science, Islamic University of Science & Technology, Kashmir 192122, J&K, India

ARTICLE INFO

Keywords:
Influential user detection
Deep learning
Overlapping community detection
Social network
Graph convolution networks

ABSTRACT

Complex networks necessitate the identification of key nodes owing to their ubiquity across the network. Traditional methodologies, such as machine learning-based and centrality-based techniques, evaluate node relevance only on network topologies or node properties. Nevertheless, both network topologies and node attributes must be considered at the time of evaluating the relevance of nodes. As a solution to this problem, this study presents *Olap*GN, a deep learning model that uses Graph Convolutional Networks to identify the most significant nodes in a complicated network. By integrating the two modules (deep learning and probabilistic nature), the proposed technique identifies overlapping groups and the most significant nodes within a complex network. The suggested approach determines the most significant individuals of the overlapping community after identifying their overlap. Several experiments have been conducted on actual social networks, such as VAST, Facebook, Medicine, Computer Science, and DBLP to evaluate the efficacy of the proposed model. In locating the overlapping communities and most significant nodes in heterogeneous complex networks, the proposed method has produced far better results than all other prevailing methods used for the purpose.

1. Introduction

The life has become so complex today that the people find it quite difficult to meet one another personally. This can be attributed to several factors, such as time constraint, and day-to-day busy schedules An alternative or prominent solution to this problem lies in meeting the individuals can meet through various online networks. A social network is a group of people, as well as the relationships between those people. The links or edges between the individuals can be relationships, connections, or interactions. Facebook and LinkedIn are the examples of popular online social networks in which individuals share their thoughts and feelings on various issues. In these online social networks, the majority of individuals base their choices on the opinion of others. Their decisions are greatly affected by the ideas of others. Such individuals are sometimes referred to as influential users. With the rapid development of online social networks, social influence is increasingly pervasive and significantly impacts our everyday lives. In this way, our beliefs, behavior, etc. are influenced as a result of another individual [1].

People on social media platforms like Facebook, Instagram, Twitter, etc., maintain tight relationships with their network of friends, and are highly influenced by their interactions. In online social networks, such as sports and political forums, it is often believed that individuals create a community. Many members of a community may belong to two or more communities at the same time, causing some communities to overlap. Consequently, overlapping characteristics are commonly seen in several online social networks. Individuals in these groups are significantly impacted by the most prominent users. These significant users play a crucial role in a variety of community-related activities. Moreover, prominent users play a key role in various facets of online social networks, including viral marketing, [2,3], the search for subject matter experts [4,5], and the dissemination of information [6,7]. Thus, online social network assessment is becoming an increasingly important aspect of social networks [1]. By evaluating the online social network's big data [8] from an influential standpoint, several benefits may be obtained, such as a better understanding of people's social behavior, the promotion of national security, and economic stability among others. Yet, the most challenging aspect of impact analysis is the identification of prominent users. This challenge depends on multiple topological relationships between users, different qualities to be evaluated, and various techniques for discovering influential users.

* Corresponding author.
E-mail addresses: yasir.rashid@iust.ac.in (Y. Rashid), javaid.iqbal@iust.ac.in (J.I. Bhat).

https://doi.org/10.1016/j.knosys.2023.111163
Received 28 March 2023; Received in revised form 30 August 2023; Accepted 3 November 2023
Available online 7 November 2023

Using various machine learning and deep learning techniques, a substantial amount of work has been completed in the past. The study conducted by [9] investigated the dynamics of social impact, combining recurrent neural networks (RNNs) with representation learning, [10] demonstrated an end-to-end predictor. Moreover, despite their faults, which include a disparity between the relative importance of structures and functions, a number of previous papers have utilized the centrality techniques. Various criteria, including designing principles, network structure [11], and subject features [12] have been used to classify these identification strategies or approaches. For a more accurate evaluation of outcomes, one must have a solid understanding of the corresponding field. Support Vector Machine (SVM), Naive Bayes, and logistic regression are among the conventional machine learning approaches explored to handle the problem of effect prediction [13,14]. The issue with these strategies is that they rely heavily on feature engineering, which has a negative impact on the efficacy of machine learning-based methods. In reality, network nodes' influential value is determined not only by their qualities but also by the relationships between them and their neighbors. The existing deep learning models are graph convolution networks (GCNs), which possess both node characteristics and connections between nodes. Their use was successful in a variety of domains including medication creation [15], social networks [16], and natural language processing (NLP) [17]. It expands CNN's capacity to process unordered input, interact directly with graphs, and handle structural information. Graph Convolutional Networks (GCN) automatically learn and extract crucial features and patterns from graph data, using deep learning methods. GCN captures graph structure by considering node-neighbor relationships. Graph convolutional layers convey network signals. Each layer utilizes surrounding node data to modify its node representations. Layering GCN may capture more complex relationships and extract deeper graph information. GCN may find a network representation that encodes node importance or centrality to identify important decision-makers. Learned representations may rank graph network nodes by importance. GCN makes graph analysis more scalable and economical by automating feature extraction and ranking. This may be used in social network research and other fields to discover the most essential graph network nodes. To begin employing a Graph Convolutional Network (GCN) on social network graph data, an adjacency matrix or edge list must be constructed to represent the graph. The social network's nodes are then represented in a meaningful way via the usage of GCN. To do this, we use numerous network convolutional layers to change the node representations based on information gathered from nearby nodes. Non-linearity is introduced through activation functions, and data may be summarized by pooling or aggregation processes. Predictions or probabilities for a given job, such as node classification or link prediction, are generated by the GCN's final layer. To perform different analysis and prediction tasks in social network analysis, the GCN must be trained on such data. This allows it to learn to capture the structural and relational information of the social network. In the development of deep networks, numerous layers are built upon one another. The various benefits of using graph convolution network models include the ability to process graph-structured data, which allows for the representation of a large number of systems across a variety of domains, and the solution to the problem of node classification through the use of a robust architecture for machine learning on graphs.

The current research proposes a deep learning framework that utilizes the characteristics of graph convolution networks for identifying the influential nodes in social networks. For this purpose, both node characteristics as well as inter-node connections were analyzed. The study has been organized into various sections. After introducing to all the aspects and issues of the study in Section 1, the review of literature has been undertaken in Section 2. Section 3 explains the framework of the study. Section 4 is focused on experimental set-up and results. Section 5 concludes the study by justifying he proposed model.

This article contributes to the body of literature in a variety of following ways:

- This research work proposes a deep-learning probabilistic nature model for finding the influential nodes in various complex networks.
- The proposed model consists of two modules: Module 1 finds overlapping communities from various complex networks. whereas Module 2 identifies the most influential users from the overlapping community network identified by Module 1 of the proposed model.
- A simulation over both real-time and synthetic datasets has made the task more challenging and beneficial for those working in the same field.

2. Related works

This section presents the significant results drawn from investigation of the relevance of community discovery and top-K influential user identification from a huge network or data with varying thicknesses and dimensions. The methodologies used for community detection and identification of famous users as separate groups based on a variety of parameters have been reviewed here.

2.1. Community detection methods

The identification of communities in various, complex networks has been a subject of investigation for many studies. The work in this area was started by [18,19]. Azaouz et al. [20] deeply studied the perspective of community structure in various kinds of graphs and examined the numerous methodologies stemming from diverse domains, such as computer science, physics, etc. Typically, these techniques have been categorized into three primary groups: optimization, heuristic, and approximation approaches. Luo et al. [21] demonstrates theoretically that an FNLF model with the SLF-NM^{2}U algorithm converges faster than an NLF model. This acceleration is sustained when the momentum coefficient g lies within the interval [0, 1]. The results, both theoretical and empirical, are highly consistent.

2.1.1. Methodologies based on optimization

These algorithms determine the most effective approach that enhances or reduces the given criteria, such as inter-cluster distance and intra-cluster similarity while satisfying a set of network constraints. Recently, Srinivas et al. [22] suggested a technique based on integer linear programming which discovers the community structure and prominent users inside that community. This model demonstrated its superior performance in several ways. For clustering, Brandes et al. [23], additionally applied an integer linear programming that optimized the modularity metric. Yuan et al. [24] proposed a Kalman-filter-integrated latent factor analysis (KLFA)-based estimator for temporally dynamic QoS data. Its central concept is to make user latent features (LFs) time-dependent while service latent features remain time-consistent. The user LFs are trained using a Kalman filter to accurately represent temporal patterns, while the service LFs are trained using an alternating least squares algorithm to accurately represent historical QoS data. Yuan et al. [25] propose ingeniously a multilayered and random latent factor (MLF) model. (a) adopting randomized learning to train LFs to implement a 'one-iteration' training process to save time; and (b) adopting the principle of a generally multilayered structure as in a deep forest or multilayered extreme learning machine to structure its LFs, thereby enhancing its representative learning ability.

2.1.2. Heuristic methods

From the perspective of community identification, heuristic approaches have been categorized primarily into four categories, viz. divisive, agglomerative, multi-level, and diffusive. Under the divisive approaches, all nodes are first believed to be belonging to the same community; and then, the network is divided according to different

criteria. Newman et al. in their research work [19] have utilized edge-betweenness as a criterion for clustering, and presented a strategy for identifying communities in a network based on a technique of division. Further, the algorithm evaluates the quality of produced communities. Subsequently, Zhang et al. [26] created a community detection search technique based on meta-heuristics. In agglomerative clustering, each unit is regarded as a distinct community; and communities are periodically coupled until the given performance measure no longer improves. A very popular greedy agglomerative approach "Louvain method" for maximizing modularity introduced by [27], examines every node as a cluster initially, then nearly nodes begin merging with neighboring communities until the modularity value no longer improves. In the multilevel idea, a hierarchy is formed in which the size of the graph lowers as one descends the hierarchy, causing the network to contract, and then the clustering technique is used. Among the reinforcement systems for boosting projection performance that supported these approaches was MCCA, proposed by Rhouma et al. [28]. Raghavan et al. [29] introduced a diffusive strategy for label transmission in cluster networks. Initially, labels indicating the many communities to which the nodes belong are given to the nodes. In subsequent iterations, every unit obtains the label that is most often held by its neighbors.

2.1.3. Approximate methods

These methods ensure the quality of the solution, and are mostly used for NP-hard problems. These are assessed using the maximum number of conceivable examples. Tantipathananandh et al. in their research work [30] proposed two techniques for discovering communities in dynamic social networks. LDF is a recommended approximation method [31] for maximizing modularity that acts solely upon the degree of connections, i.e. the structural characteristic of a network. Luo et al. [32] proposes an ASNL model for precise and efficient SHDI matrix representation. It incorporates data density-oriented modeling and ADMM into its learning methodology. It parallelizes its learning process further to improve its computational efficacy.

2.2. Influential user detection methods

Numerous techniques, such as information entropy [33], propagation dynamics [34], etc. have been developed by the researchers for finding the important nodes in a complex network. These methods are categorized into five major categories: centrality-based, greedy and heuristic, node similarity-based, machine learning-based, and GCN-based. Apart from being a prominent topic in the deep learning field, GCN has attracted the attention of many researchers to apply it in other relevant fields also.

2.2.1. Centrality-based approaches

Numerous academics have developed centrality-based approaches to diverse topological structures that restrict performance and adaptability. These centrality-based techniques have been further classified into a variety of categories [35]. The distance-based methods focus on the smallest route between nodes and include betweenness centrality [36] and proximity centrality [37]. According to the Hellinger distance between two nodes in a bipartite network, Taheri et al. [38] intriguing HellRank centrality approach finds critical nodes. Taking into account the number of neighbors and the negative influence of regional grouping on the circuit, Chen et al. [39] produced successful results in this regard. VoteRank [40] has rated nodes based on the voting process using the local structural information of nodes. In large-scale networks, this method produced better results or solutions. They rely on the network size, i.e. degree centrality [41], which is based on neighbor relationships. Iteration-based approaches, such as EigenVector centrality [42] and PageRank centrality [43] evaluate the significance of a node's peers to determine the network node's own relevance.

2.2.2. Greedy and heuristic-based approaches

Several researchers have also created greedy and heuristic-based algorithms for identifying influential people in online social networks. Kempe et al. [44] developed a technique based on a greedy approach, termed as Hill climbing, however, this method is inefficient since it requires thousands of rounds to achieve a certain level. DegGreedy, a technique suggested by Nandi et al. [45], increases the distribution of impact based on node neighbors, resulting in high scalability and efficiency on huge data sets. Cheng et al. [46] eventually identified this issue and provided a technique that estimated influence based on a static propagation graph, resulting in a significant increase in efficiency. Few academics attempted to tackle the challenge of impact maximization using greedy and heuristic-based strategies. Ren et al. [47] explored the Laplacian operator in order to optimize this issue with the fewest number of nodes. Using greedy and heuristic techniques, Zhao et al. [48] further expanded the graph coloring issue to several complicated networks. Sheikhahmadi et al. [49] devised the IMSN approach, which selected parent nodes by taking the provided node's friends into account. This strategy proved effective for avoiding issues that overlapped due to shared acquaintances.

2.2.3. Node similarity-based methods

Some studies have also calculated the strength of edges by determining the information flow between the nodes based on their similarity [50]. The basic goal of these methods is to determine the degree to which similarities exist between two nodes based on their common properties. Common-Neighbor [51] is the most prevalent technique to date. Alshahrani et al. [52] presented a unique approach called PrKatz, for the efficient identification of essential network nodes. This strategy examined the likelihood of accepting information for each edge and eliminated those edges which failed to meet a predetermined threshold value. Katz centrality was then applied to the network to identify important nodes in decreasing order based on their PrKatz score. The main shortcoming of this technique was its temporal complexity; otherwise, it worked quite well.

2.2.4. Machine learning-based methods

The researchers concentrated primarily on feature selection or feature engineering [53] through these methods to improve performance. Several machine learning techniques, including logistic regression, SVM, and KNN have been used to identify important users. Various industries and situations [54] use these techniques for locating key network nodes, setting the groundwork for several advancements in this area. However, a significant disadvantage of using these algorithms is that feature selection is time-consuming and disregards the link between target nodes.

2.2.5. GCN-based methods

As a trending issue, several GCN-based methods for finding key nodes in diverse complex networks [55] have been developed. These approaches have also addressed a variety of other concerns, such as the relative significance of nodes [56]. Cai et al. in [57] proposed a straightforward and efficient method for augmenting the graph contrastive learning framework for recommendation. Specifically, they investigate the fundamental concept of enhancing user–item interaction graph structures by making the singular value decomposition potent enough. The principal findings indicate that the proposed graph augmentation scheme is resistant to data scarcity and popularity bias. Wei et al. in [58] propose a novel Contrastive Graph Structure Learning via Information Bottleneck (CGI) to learn improved multi-view representation augmentation from diverse perspectives. In particular, the authors propose a fully differentiable learner to remove nodes and edges in order to create various forms of augmentation views in conjunction with the recommendation. The authors incorporate information bottleneck in an innovative manner into the multi-view contrastive learning process for recommendation and demonstrate its effectiveness. Most

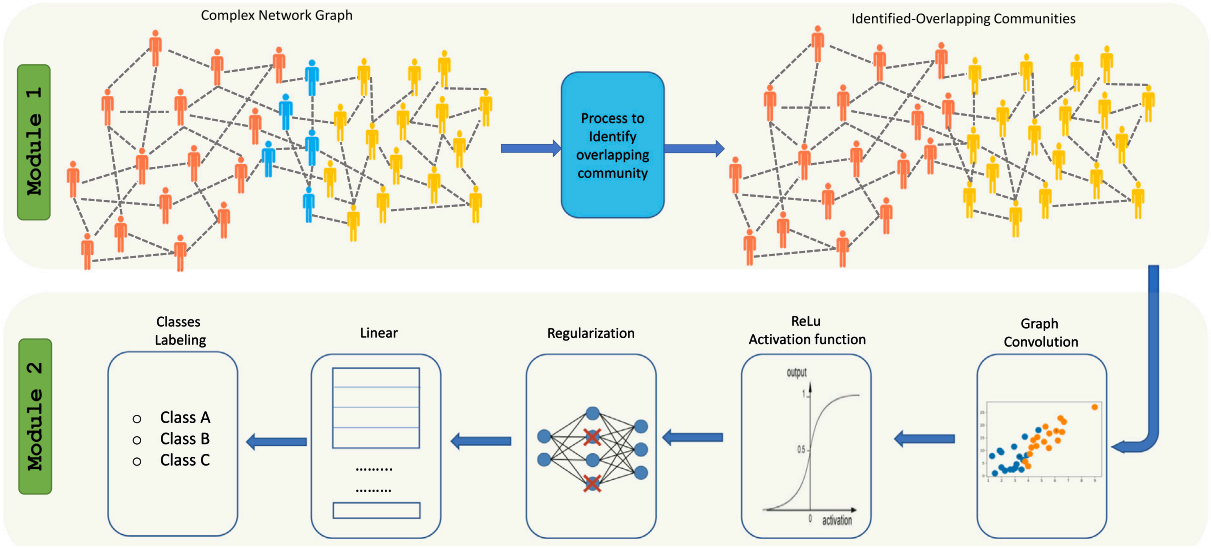


Fig. 1. Two-Layered architecture of the proposed *Olap*GN model.

recently, OU et al. [59] presented a novel application of GCN to identify influence nodes via multi-level structural attributes. Internally, there are two types of GCN-based procedures as explained below:

(a) Spectral-based GCN Methods: As demonstrated by Henaff et al. [60] and Bruna et al. [61], spectral-based GCN algorithms focus mostly on filters for calculating the convolutional kernel from a signal processing aspect.

(b) Spatial-based GCN methods: These techniques define network resulting solution as aggregates of neighboring feature information, such as ChebNet [62], GraphSAGE [63], and so on. These approaches regularly evaluate both the node's characteristics and its neighbors' relationships while aggregating neighbor information.

Social influence is becoming more prevalent and has a great effect on our daily lives as a result of the fast growth of online social networks. In this, beliefs, behavior, etc. are influenced due to another individual [1]. On various social media outlets like Facebook, Twitter, etc. people are closely connected, and remain in constant communication with their network friends; and they are affected in one way or the other by these communications. Further, prominent users play a key role in a variety of features of online social networks, such as viral marketing [2,3], search for subject matter experts [4,5], and dissemination of information [6,7].

3. Mathematical modeling

This section primarily describes the proposed solution approach called *Olap*GN which is required to identify the most influential node in online social networks. Fig. 1 depicts the different modules utilized by the proposed *Olap*GN model. For instance, module 1 supports overlapping communities' identification, whereas Module 2 identifies the most important nodes in a complex graph network using graph neural networks. Moreover, Table 1 depicts the symbolic notations and their respective definitions used frequently throughout this work.

3.1. Module to track the evolution of overlapping communities

In Fig. 1, Module 1 processes data presented as a network of interconnected overlapping communities to produce a final output consisting solely of overlapping communities. The GCN initially learns node embeddings, thereby encoding the structural information and relationships within the graph. A community detection algorithm is then applied to the graph to identify initial communities. These communities are depicted as subgraphs, sets of nodes, and their respective

Table 1
Frequently used symbolic notations.

Notations	Interpretation of notations
M_{pq}	Matrix
R_p, R_q	Row vector
Q	Affiliation matrix
K_E, K_N	Uniform distribution
T	Neighbor network's symmetric normalized Laplacian
ϑ, F	Trainable weights
c	Bias parameters
Λ	Non-linear function

embeddings, which serve as input for subsequent GCN layers. The GCN enables nodes within each community to alter their embeddings based on interactions and connections via message passing and aggregation. To accommodate the overlapping nature of communities, the GCN architecture incorporates a specialized layer or module. This layer facilitates the exchange of information and connections between communities, allowing nodes to simultaneously adhere to numerous communities. The final output of the GCN is a collection of overlapping communities, where each community consists of nodes with significant connections and similarities, following the goal of identifying and characterizing overlapping structures within the network. The subject under study has used the overlapping community detection model which implicitly makes use of the core idea of integrating the power of GNNs with the Bernoulli–Poisson (BP) probabilistic model. This model is a generative graph model that allows for overlapping communities detection [64–66]. graph [67]:

$$M_{pq} \sim \text{BP}\left(1 - \exp\left(-Q_p Q_q^T\right)\right) \tag{1}$$

Here, Q_p is the row vector of community affiliations of node p. The probability of a connection between nodes p and q increases if they share many communities, as measured by the dot product $Q_p Q_q^T$. This model has several desirable properties. It results in a variety of community topologies, which in turn causes dense overlaps to develop between the communities [68]. The BP model makes use of either the maximum likelihood estimation method or the Markov Chain Monte Carlo algorithm with coordinate ascent.

This study constructs Q with a GNN rather than treating the affiliation matrix Q as a free variable over which optimization is carried out as per the following Eq. (2):

$$Q := \text{GNN}_\theta(M, Y) \tag{2}$$

To guarantee F's non-negativity, a ReLU non-linearity function is applied element-by-element to the output layer. The BP model's negative log-likelihood is represented by Eq. (3) as given below.

$$-\log p(M \mid Q) = -\sum_{(p,q)\in S} \log\left(1-\exp\left(-Q_p Q_q^T\right)\right) + \sum_{(p,q)\notin S} Q_p Q_q^T \tag{3}$$

Since real-world graphs are frequently very sparse, the second part in Eq. (3) will contribute significantly more to the loss. As a common method in imbalanced classification, this study responds to this problem by balancing parameters as per Eq. (4) [69]:

$$\mathcal{L}(Q) = -\mathbb{S}_{(p,q)-K_E}\left[\log\left(1-\exp\left(-Q_p Q_q^T\right)\right)\right] + \mathbb{K}_{(p,q)\sim K_N}\left[Q_p Q_q^T\right] \tag{4}$$

where, K_E and K_N stand for uniform distributions over edges and non-edges. Unlike conventional approaches' direct optimization of the affiliation matrix Q, we look for neural network parameters θ^* that reduce the (balanced) negative log-likelihood [70].

$$\omega^\star = \arg\min_{\omega} \mathcal{L}\left(\mathrm{GNN}_\theta(M, Y)\right) \tag{5}$$

For community prediction, a GNN has several benefits. First, compared to simpler models, the GNN produces similar community affiliation vectors for neighboring nodes because of an adequate inductive bias, which enhances the accuracy of predictions. Further, with this formulation, the node attributes can be easily included in the model. We can utilize M as node characteristics if the node properties Y are inaccessible [71]. The formulation from Eq. (2) also makes it possible to forecast communities inductively for nodes that have not been visible during the training process.

3.2. Module to trace the influential nodes

This section aims to describe the structure of a deep learning model, *Olap*GN (see Fig. 1, Module 2), to find the most important nodes in intricate networks. This study first sampled the fixed-size neighbor networks and created feature vectors containing each node's four classic centralities (degree, closeness centrality, betweenness centrality, and clustering coefficient). After that, a 2-dimensional presentation of each node in the graph network, produced by the graph convolution network loaded with feature vector and neighbor network, is represented to highlight the learning process. Later a ReLU function (rectified linear activation function) has been utilized for transforming the summed weighted input to the output of that input. Afterward, a regularization process has been applied to reduce the used model's complexity by preventing overfitting.

3.2.1. Constructing neighbor networks

A node's capacity to influence is, typically, influenced by its surrounding nodes. The earlier research has demonstrated that the (k + 1)th representation of a node in GCN models is connected to the neighborhood's fifth representation to only connect to its neighbor networks, a k- step network [72]. As a result, estimating the connectivity of a node's neighbors, the node's significance is required. Breadth First Search is a valuable technique for creating a node's neighborhood network (BFS), obtaining its neighbors first, then inducing the neighbors by these neighbors. The scale of neighboring networks varies from one to another which is inappropriate for mini-batch learning for most deep learning models. In order to address these problems rightly, the size of each node's neighbor network in a network has been defined as a fixed number F. The ith step neighbor of the desired targeted node has been retrieved for each step i of BFS. After that, we have calculated the total number of neighbors (referred to as ti) not farther than ith step from the target. Move to the next step, if it is less than the preset value F. If it is larger than the required value, then the ith step neighbors are filtered based on their betweenness centrality. The smaller betweenness neighbors have been discarded, while the larger ones have been allowed to stay. As bridge nodes play a more significant role in the information transmission process, betweenness centrality has been chosen as the criteria because it reflects the bridging role of nodes.

3.2.2. GCN-based model

In this study, a robust model based on a Graph Convolutional Network has been developed and used to process the neighbor network of each node by obtaining the data for the neighbor network of each node in the form of a graph. An input layer, a GCN layer, three Fully Connected layers, and an output layer have been included in the proposed model. GCN layer is in-turn multilayered in which authors have used various activation function layers, pooling layers, graph attention layers, and graph Convolutional recurrent layers for respective tasks. Moreover, it is embedded with these layer by layer, incrementally building the model.

The input layer creates a normalized symmetric Laplacian of each node's feature vector and neighbor network. This paper has chosen a few traditional centralities as node features to highlight the structural characteristics of nodes in order to avoid excessive dependence on feature engineering: Degree, Closeness Centrality, Betweenness Centrality, and Clustering Coefficient are listed in that order [73]. In the interim, we normalized the traits to prevent over-fitting. The normalization procedure for each characteristic e is as follows:

$$\alpha_e = \frac{\zeta_e}{\delta} - 0.5 \tag{6}$$

where δ stands for the network's node count; and α_e represents the network's ranking position based on the centrality feature's value of e. As a result of the normalization, each value transformed the feature between −0.5 and 0.5. Layer GCN Semi-supervised methods called Graph Convolutional Networks (GCN) have been used to learn node representation vectors by using the graph structures, and feature vectors [74]. The following equation describes the GCN layer:

$$\vartheta^{i+1} = \Lambda\left(T\vartheta^i F^i + c^i\right) \tag{7}$$

Suppose T is the neighbor network's symmetric normalized Laplacian. In that case, the ith GCN layer's representations of nodes have been indicated by the symbol ϑ^i, whereas F^i and c^i indicate the trainable weights and bias parameters respectively. The feature vectors in ϑ^0 are nearby nodes in the input layer. Further, we included the skip connection in the GCN layer to enhance the utilization of the node functionalities. In the meantime, Dropout technology was used in this layer to avoid over-fitting [75]. The LogSoftMax classifier received the results from the fully linked layers. This research work compared classification results with the ground truth acquired from SIR simulation trials and optimized the loss of negative logarithm probability.

Deep learning requires a considerable quantity of data. Smaller networks with fewer nodes indicate insufficient training data for deep learning algorithms. To address this problem and enhance the model fit with small networks, we learn from the universal principles underlying this by taking technology classes and pre-training our model [76]. Transfer learning is the process of using a previously trained model on some tasks. It is applied to new but connected activities. Deep transfer learning has proved highly successful in computer vision and other fields. We have extended pre-training to graph data in this study. Although there are many sizes and types of networks, pre-training is still possible because all networks have network architecture. First, we pre-trained the proposed model using large network datasets; and then, we fine-tuned it using tiny network training datasets. In tiny networks, the model may also predict nodes.

4. Experiment set-up

To conduct a quantitative analysis of our suggested architecture, we selected specific assessment criteria and five actual networks from the real world to use as experimental networks. We also compared the traditional machine learning-based and centrality-based techniques to find which fitted the framework most.

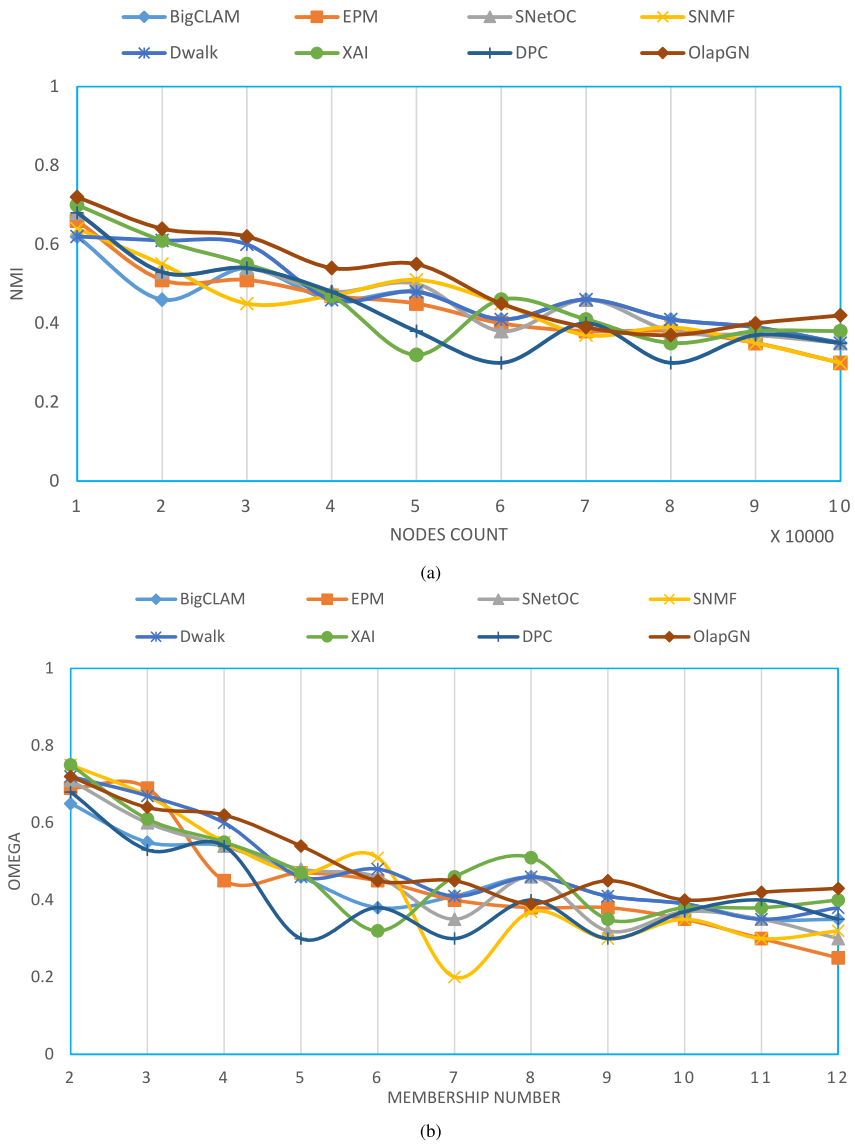


Fig. 2. Computation of different test algorithms, outcomes over NMI and Omega (Fig. 2(b)) Metrices.

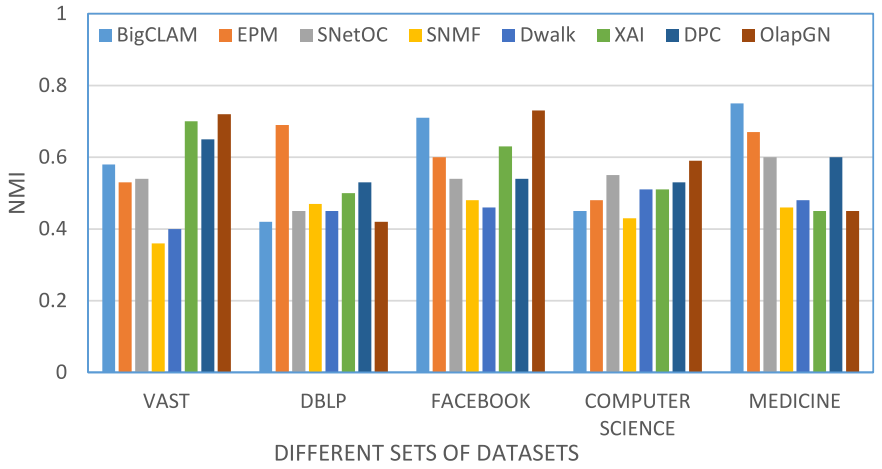
4.1. Details about datasets and parameters

The simulations use two distinct kinds of networks: artificial and real-world networks taken from the physical realm. The LFR (Lancichinetti–Fortunato–Radichi) benchmark generator is used to construct artificial networks with implanted communities [77]. The experiments conducted for the purpose of the current study also leveraged VAST and DBLP dynamic social networks. The VAST network was composed of data from 9834 calls made by 400 members over a 10-day period. The DBLP network is a cooperation graph of authors' scientific contributions towards bibliographies that were produced during the years 1998 and 2013 within the realm of computer science [78]. In all, there are 1,314,050 nodes and 18,986,618 edges. A joint publication is represented by an edge between two authors. Whereas, the publishing date is indicated on the edges. This network has been separated into ten successive snapshots as well. Facebook is a group of small ego networks having nodes between 50 and 800 that come from the Facebook graph [79]. However, larger graph datasets (10K + nodes) with credible ground-truth overlapping community information and node attributes are not publicly available, hindering methodology evaluation and development. Thus, the authors have selected 2 more pre-processed real-world datasets (Computer Science and Medicines) which meet these criteria as future benchmarks.

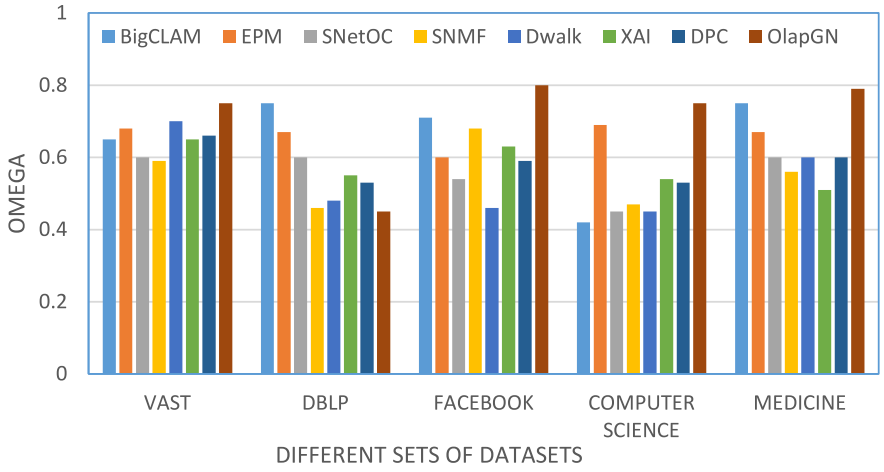
The current study has focused solely on the Computer Science dataset to optimize the proposed architecture and hyperparameters. Similar parameters tuning is implicitly applied for the rest of the datasets. The projected continuous community affiliations need to be converted into community assignments to compare the detected communities to the actual communities. Thus, we have assigned node p to the community C, if its affiliation strength Q_{pc} is above a fixed threshold μ. Like all other hyperparameters, $\mu = 0.5$ was chosen as the best score on the Computer Science dataset and used in subsequent tests without additional tuning. It is prominent to note that all the models including $OlapGN$ models employ the identical hyperparameter setup calibrated exclusively on the Computer Science dataset, where the number of node counts, number of edges, and dimension attribute vector is set to 10K, 23K, and 2.5K respectively. In order to guarantee an accurate comparison, each technique was provided with an exact number of communities. The rest of the hyperparameters were adjusted to their default tuning values [80,81].

4.2. Baseline approaches, and performance metrics

The current study has examined the proposed *Olap*GN model with other similar domain baselines community recovery in graphs and



(a)



(b)

Fig. 3. Computation of different test algorithms, outcomes against some of the rarely used synthetic and real-time datasets over NMI (Fig. 3(a)) and Omega (Fig. 3(b)) metrices.

known ground-truth communities. Probabilistic inference, NMF (Non-negative Matrix Factorization), and deep learning were the baselines used for overlapping community detection. Some approaches used characteristics while others use graph structure. Bernoulli–Poisson is the foundation for BigCLAM [65], EPM [66], and SNetOC [67]. Moreover, SNMF is an NMF technique for detecting overlapping communities [82]. The authors have also constructed Dwalk [83], a neural graph embedding-based algorithm that considered both node features and graph structure. Non-exhaustive Overlapping (NEO) K-Means are used to cluster the graph nodes, allowing them to be assigned to overlapping communities. This study also offers the comparison against the most recent state-of-the-art technology, including Explainable-AI (XAI) [84] and density peak clustering (DPC) [85].

The standard metrics used for evaluating the agreement between genuine and identified communities, such as Jaccard and F1 scores, might offer arbitrarily high scores for entirely uninformative community assignments. This was one of the things discovered in this work. Thus, this study accounts for the use of something called overlapping normalized mutual information (NMI) and the Omega Index, which is more reliable and meaningful as well.

4.3. Efficacy of overlapping community detection

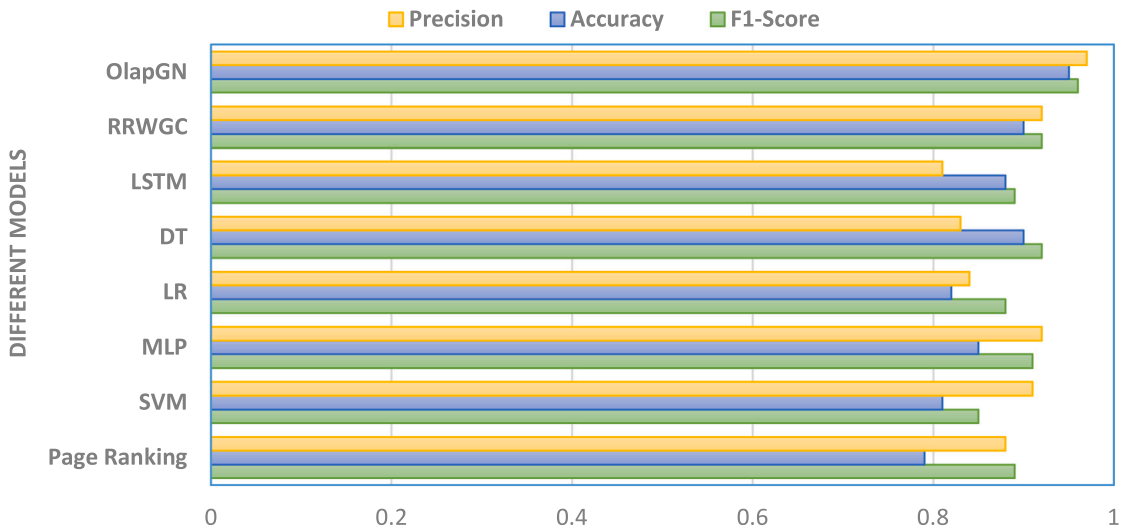
This section analyzes the proposed approach called *Olap*GN in terms of overlapping community detection. The discovery of overlapping communities is the most fundamental and crucial stage in determining the influential node.

4.3.1. Test of artificial network

At first, this study tested the effectiveness of the proposed *Olap*GN technique for detecting overlapping communities on networks with the size ranging from 10,000 nodes to 100,000. Moreover, the primary parameters used to represent the LFR generator were: we considered Intra-community strength of 0.4, 10% to the overlapping density, community size was represented in the range of 100 to 2000, and a value of 5 represented the number of communities to which each overlapping node belonged. The results derived from *Olap*GN were analyzed and compared with those of the other five methods. The NMI was used to quantitatively assess the quality of the community structure determined by each approach, as shown in Fig. 2(a). All the approaches deteriorated moderately with an increase in the number of nodes. The performance of the novel approach proposed in this work i.e., *Olap*GN was closer to that of BigCLAM; it was far superior to that produced by other methods. On the other hand, this study tested the *Olap*GN method's overlapping community detection with membership numbers ranging from 2 to 12 using the Omega method of overlapping community identification (see Fig. 2(b)). It is pertinent to note that *Olap*GN outperformed other approaches when the community size increased, especially when it was greater than 9.

4.3.2. Test on real-world network

The six techniques (five baselines plus the proposed *Olap*GN method), as depicted in Fig. 3, were applied to five real-world social networks, viz VAST, DBLP, Facebook, Computer Science, and Medicine. The proposed method *Olap*GN did not perform uniformly on



	Page Ranking	SVM	MLP	LR	DT	LSTM	RRWGC	OlapGN
Precision	0.88	0.91	0.92	0.84	0.83	0.81	0.92	0.97
Accuracy	0.79	0.81	0.85	0.82	0.9	0.88	0.9	0.95
F1-Score	0.89	0.85	0.91	0.88	0.92	0.89	0.92	0.96

Fig. 4. Comparative experimental results of different working models.

all types of real-world networks, but it exhibits well on the majority of networks. However, the NMI performance of the proposed technique was not relatively impressive when tested on the DBLP and medicine datasets (see Fig. 3(a)). This can be attributed to the topological potential field of these two networks which had no evident peak–valley structure. For instance, in this particular case, local maximum potential nodes from large-scale communities had an impact on certain loose and small-scale communities. Thus, larger communities around them will absorb these loose communities, reducing the identified community number. In contrast (see Fig. 3(b)), the modularity maximization concept underlies the MIEN and SHRINK approaches, which efficiently split the network, and obtained a higher OMEGA value except on the DBLP dataset. Conclusively, the proposed technique guarantees community structure partition accuracy and competitive modularity.

4.4. The efficacy to identify the influential node in graph network

This study employed GCN-based deep learning to identify influential users. The success of the *Olap*GN model was precisely evaluated using findings from various similar domain models such as page ranking, RRWGC(Return Random Walk Gravity Centrality) [86], SVM(Support Vector Machine) [87], MLP(Multilayer Perceptron) [88], LR(Linear Regression), DT(Decision Tree) [89], and LSTM (Long Short-term Memory) [90]. Moreover, each model's acquired precision, accuracy, and F1-score levels were compared GridSearchCV was utilized to analyze the parameters for the current study's deep learning and machine learning models. Five cross-validations were used to overcome the overfitting problem and improve the introduced models' quality. Fig. 4 highlights the performance of different models, including page ranking, SVM, MLP, LR, DT, LSTM, and *Olap*GN precision, accuracy, and F1-score values. It is significant to note that the proposed *Olap*GN model produced more reliable results than the other models. For instance, the proposed model's precision, accuracy, and F1-score are approximately 97%, 95%, and 96% respectively. Thus, the proposed model has proved its superiority in detecting influential users in the network.

5. Conclusion

Several studies have appeared with the result that only a limited number of nodes are significant in most complex networks. Thus, these nodes are crucial for several applications. The benefits can be greater, if the influential node is identified in overlapping communities as it exercises more influence in overlapping communities. This study explored the difficulty of determining the essential users from many communities intersecting in complex networks. By including network architecture or other node features in many obsolete approaches, such as centrality techniques and machine learning methods, the performance was badly hit. However, by combining two modules (Deep learning + probabilistic nature), the proposed novel method discovered overlapping groups and the most essential members in a complicated network. In the first phase of Module 1, overlapping community detection was performed when *Olap*GN (graph convolutional neural network) had received the input. After detecting the overlapping nature, Module 2 identified the most important members of the overlapping community. In order to examine the efficacy of the proposed model, several experiments were conducted in five real-world social networks, viz VAST, Facebook, Medicine, Computer Science, and DBLP. While finding the overlapping communities and most important nodes in diverse complex networks, the proposed approach greatly outperformed its baseline counterparts. In this study, there are several intriguing avenues to investigate in the future. Incorporating additional information, such as node attributes or content, is one option for identifying influential nodes more effectively. To increase the approach's applicability, it could be applied to dynamic social networks where users join and depart over time. In addition, the proposed method can be mathematically reformulated and evaluated across numerous microblogging platforms. Finally, a memory-efficient algorithm could be developed to improve scalability by decreasing memory consumption. These potential future enhancements could substantially improve the method.

CRediT authorship contribution statement

Yasir Rashid: Conceptualization, Methodology, Experimentation, Formal analysis, Data curation, Writing – original draft, Visualization. **Javaid Iqbal Bhat:** Investigation, Supervision, Software, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] S. Peng, G. Wang, D. Xie, Social influence analysis in social networking big data: Opportunities and challenges, IEEE Netw. 31 (1) (2016) 11–17.
[2] H. Ma, H. Yang, M.R. Lyu, I. King, Mining social networks using heat diffusion processes for marketing candidates selection, in: Proceedings of the 17th ACM Conference on Information and Knowledge Management, 2008, pp. 233–242.
[3] J. Hartline, V. Mirrokni, M. Sundararajan, Optimal marketing strategies over social networks, in: Proceedings of the 17th International Conference on World Wide Web, 2008, pp. 189–198.
[4] C.S. Campbell, P.P. Maglio, A. Cozzi, B. Dom, Expertise identification using email communications, in: Proceedings of the Twelfth International Conference on Information and Knowledge Management, 2003, pp. 528–531.
[5] J. Zhang, M.S. Ackerman, L. Adamic, Expertise networks in online communities: structure and algorithms, in: Proceedings of the 16th International Conference on World Wide Web, 2007, pp. 221–230.
[6] Q. Liu, B. Xiang, N.J. Yuan, E. Chen, H. Xiong, Y. Zheng, Y. Yang, An influence propagation view of pagerank, ACM Trans. Knowl. Discov. Data (TKDD) 11 (3) (2017) 1–30.
[7] K. Saito, R. Nakano, M. Kimura, Prediction of information diffusion probabilities for independent cascade model, in: International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, Springer, 2008, pp. 67–75.
[8] S. Yu, M. Liu, W. Dou, X. Liu, S. Zhou, Networking for big data: A survey, IEEE Commun. Surv. Tutor. 19 (1) (2016) 531–549.
[9] Y. Matsubara, Y. Sakurai, B.A. Prakash, L. Li, C. Faloutsos, Rise and fall patterns of information diffusion: model and implications, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2012, pp. 6–14.
[10] C. Li, J. Ma, X. Guo, Q. Mei, Deepcas: An end-to-end predictor of information cascades, in: Proceedings of the 26th International Conference on World Wide Web, 2017, pp. 577–586.
[11] H. Liao, M.S. Mariani, M. Medo, Y.-C. Zhang, M.-Y. Zhou, Ranking in evolving complex networks, Phys. Rep. 689 (2017) 1–54.
[12] M.A. Al-Garadi, K.D. Varathan, S.D. Ravana, E. Ahmed, G. Mujtaba, M.U.S. Khan, S.U. Khan, Analysis of online social network connections for identification of influential users: Survey and open research issues, ACM Comput. Surv. 51 (1) (2018) 1–37.
[13] Y. Mei, Y. Zhong, J. Yang, Finding and analyzing principal features for measuring user influence on twitter, in: 2015 IEEE First International Conference on Big Data Computing Service and Applications, IEEE, 2015, pp. 478–486.
[14] J.-V. Cossu, N. Dugué, V. Labatut, Detecting real-world influence through twitter, in: 2015 Second European Network Intelligence Conference, IEEE, 2015, pp. 83–90.
[15] N. De Cao, T. Kipf, An implicit generative model for small molecular graphs, 2018, arXiv preprint.
[16] J. Qiu, J. Tang, H. Ma, Y. Dong, K. Wang, J. Tang, Deepinf: Social influence prediction with deep learning, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 2110–2119.
[17] J. Bastings, I. Titov, W. Aziz, D. Marcheggiani, K. Sima'an, Graph convolutional encoders for syntax-aware neural machine translation, 2017, arXiv preprint arXiv:1704.04675.
[18] M. Girvan, M.E. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. 99 (12) (2002) 7821–7826.
[19] M.E. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2) (2004) 026113.
[20] M. Azaouzi, D. Rhouma, L. Ben Romdhane, Community detection in large-scale social networks: state-of-the-art and future directions, Soc. Netw. Anal. Min. 9 (1) (2019) 1–32.
[21] X. Luo, Y. Zhou, Z. Liu, M. Zhou, Fast and accurate non-negative latent factor analysis on high-dimensional and sparse matrices in recommender systems, IEEE Trans. Knowl. Data Eng. (2021).
[22] S. Srinivas, C. Rajendran, Community detection and influential node identification in complex networks using mathematical programming, Expert Syst. Appl. 135 (2019) 296–312.
[23] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, D. Wagner, On modularity clustering, IEEE Trans. Knowl. Data Eng. 20 (2) (2007) 172–188.
[24] Y. Yuan, X. Luo, M. Shang, Z. Wang, A kalman-filter-incorporated latent factor analysis model for temporally dynamic sparse data, IEEE Trans. Cybern. (2022).
[25] Y. Yuan, Q. He, X. Luo, M. Shang, A multilayered-and-randomized latent factor model for high-dimensional and sparse matrices, IEEE Trans. Big Data 8 (3) (2020) 784–794.
[26] P. Ji, S. Zhang, Z. Zhou, A decomposition-based ant colony optimization algorithm for the multi-objective community detection, J. Ambient Intell. Humaniz. Comput. 11 (1) (2020) 173–188.
[27] V.D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, J. Stat. Mech.: Theory Exp. 2008 (10) (2008) P10008.
[28] D. Rhouma, L. Ben Romdhane, An efficient multilevel scheme for coarsening large scale social networks, Appl. Intell. 48 (10) (2018) 3557–3576.
[29] U.N. Raghavan, R. Albert, S. Kumara, Near linear time algorithm to detect community structures in large-scale networks, Phys. Rev. E 76 (3) (2007) 036106.
[30] C. Tantipathananandh, T. Berger-Wolf, Constant-factor approximation algorithms for identifying dynamic communities, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, pp. 827–836.
[31] T.N. Dinh, M.T. Thai, Community detection in scale-free networks: approximation algorithms for maximizing modularity, IEEE J. Sel. Areas Commun. 31 (6) (2013) 997–1006.
[32] X. Luo, Y. Zhong, Z. Wang, M. Li, An alternating-direction-method of multipliers-incorporated approach to symmetric non-negative latent factor analysis, IEEE Trans. Neural Netw. Learn. Syst. (2021).
[33] C. Guo, L. Yang, X. Chen, D. Chen, H. Gao, J. Ma, Influential nodes identification in complex networks via information entropy, Entropy 22 (2) (2020) 242.
[34] D.-B. Chen, H.-L. Sun, Q. Tang, S.-Z. Tian, M. Xie, Identifying influential spreaders in complex networks by propagation probability dynamics, Chaos 29 (3) (2019) 033120.
[35] L. Lü, D. Chen, X.-L. Ren, Q.-M. Zhang, Y.-C. Zhang, T. Zhou, Vital nodes identification in complex networks, Phys. Rep. 650 (2016) 1–63.
[36] U. Brandes, A faster algorithm for betweenness centrality, J. Math. Sociol. 25 (2) (2001) 163–177.
[37] K. Okamoto, W. Chen, X.-Y. Li, Ranking of closeness centrality for large-scale social networks, in: International Workshop on Frontiers in Algorithmics, Springer, 2008, pp. 186–195.
[38] S.M. Taheri, H. Mahyar, M. Firouzi, K. E. Ghalebi, R. Grosu, A. Movaghar, et al., Hellrank: a hellinger-based centrality measure for bipartite social networks, Soc. Netw. Anal. Min. 7 (1) (2017) 1–16.
[39] D.-B. Chen, H. Gao, L. Lü, T. Zhou, Identifying influential nodes in large-scale directed networks: the role of clustering, PLoS One 8 (10) (2013) e77455.
[40] J.-X. Zhang, D.-B. Chen, Q. Dong, Z.-D. Zhao, Identifying a set of influential spreaders in complex networks, Sci. Rep. 6 (1) (2016) 1–10.
[41] T. Opsahl, Degree centrality in a weighted network, Phys. Rev. E 77 (4) (2008).
[42] P. Bonacich, Some unique properties of eigenvector centrality, Soc. Netw. 29 (4) (2007) 555–564.
[43] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Comput. Netw. ISDN Syst. 30 (1–7) (1998) 107–117.
[44] D. Kempe, J. Kleinberg, É. Tardos, Maximizing the spread of influence through a social network, in: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003, pp. 137–146.
[45] G. Nandi, U. Sharma, A. Das, A novel hybrid approach for influence maximization in online social networks based on node neighborhoods, in: Advances in Electronics, Communication and Computing, Springer, 2018, pp. 509–520.
[46] S. Cheng, H. Shen, J. Huang, G. Zhang, X. Cheng, Staticgreedy: solving the scalability-accuracy dilemma in influence maximization, in: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, 2013, pp. 509–518.
[47] X.-L. Ren, N. Gleinig, D. Helbing, N. Antulov-Fantulin, Generalized network dismantling, Proc. Natl. Acad. Sci. 116 (14) (2019) 6554–6559.
[48] X.-Y. Zhao, B. Huang, M. Tang, H.-F. Zhang, D.-B. Chen, Identifying effective multiple spreaders by coloring complex networks, Europhys. Lett. 108 (6) (2015) 68005.
[49] A. Sheikhahmadi, M.A. Nematbakhsh, Identification of multi-spreader users in social networks for viral marketing, J. Inf. Sci. 43 (3) (2017) 412–423.
[50] E. Pap, M. Jocić, A. Szakál, D. Obradović, Z. Konjović, Managing big data by directed graph node similarity, in: 2016 IEEE 17th International Symposium on Computational Intelligence and Informatics (CINTI), IEEE, 2016, pp. 000025–000030.
[51] M.S. Granovetter, The strength of weak ties, Am. J. Sociol. 78 (6) (1973) 1360–1380.

[52] M. Alshahrani, Z. Fuxi, A. Sameh, S. Mekouar, S. Huang, Top-k influential users selection based on combined katz centrality and propagation probability, in: 2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), IEEE, 2018, pp. 52–56.
[53] J. Cai, J. Luo, S. Wang, S. Yang, Feature selection in machine learning: A new perspective, Neurocomputing 300 (2018) 70–79.
[54] X. Zheng, Z. Zeng, Z. Chen, Y. Yu, C. Rong, Detecting spammers on social networks, Neurocomputing 159 (2015) 27–34.
[55] E.-Y. Yu, Y.-P. Wang, Y. Fu, D.-B. Chen, M. Xie, Identifying critical nodes in complex networks via graph convolutional networks, Knowl.-Based Syst. 198 (2020) 105893.
[56] S. Qi, W. Wang, B. Jia, J. Shen, S.-C. Zhu, Learning human-object interactions by graph parsing neural networks, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 401–417.
[57] X. Cai, C. Huang, L. Xia, X. Ren, Lightgcl: Simple yet effective graph contrastive learning for recommendation, 2023, arXiv preprint arXiv:2302.08191.
[58] C. Wei, J. Liang, D. Liu, F. Wang, Contrastive graph structure learning via information bottleneck for recommendation, Adv. Neural Inf. Process. Syst. 35 (2022) 20407–20420.
[59] Y. Ou, Q. Guo, J.-L. Xing, J.-G. Liu, Identification of spreading influence nodes via multi-level structural attributes based on the graph convolutional network, Expert Syst. Appl. 203 (2022) 117515, http://dx.doi.org/10.1016/j.eswa.2022.117515.
[60] M. Henaff, J. Bruna, Y. LeCun, Deep convolutional networks on graph-structured data, 2015, arXiv preprint arXiv:1506.05163.
[61] J. Bruna, W. Zaremba, A. Szlam, Y. LeCun, Spectral networks and locally connected networks on graphs, 2013, arXiv preprint arXiv:1312.6203.
[62] M. Defferrard, X. Bresson, P. Vandergheynst, Convolutional neural networks on graphs with fast localized spectral filtering, Adv. Neural Inf. Process. Syst. 29 (2016).
[63] W. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, Adv. Neural Inf. Process. Syst. 30 (2017).
[64] A. Todeschini, X. Miscouridou, F. Caron, Exchangeable random measures for sparse and modular graphs with overlapping communities, 2016, arXiv preprint arXiv:1602.02114.
[65] J. Yang, J. Leskovec, Overlapping community detection at scale: a nonnegative matrix factorization approach, in: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, 2013, pp. 587–596.
[66] M. Zhou, Infinite edge partition models for overlapping community detection and link prediction, in: Artificial Intelligence and Statistics, PMLR, 2015, pp. 1135–1143.
[67] O. Shchur, S. Günnemann, Overlapping community detection with graph neural networks, 2019, arXiv preprint arXiv:1909.12201.
[68] J. Yang, J. Leskovec, Structure and overlaps of ground-truth communities in networks, ACM Trans. Intell. Syst. Technol. 5 (2) (2014) 1–35.
[69] H. He, E.A. Garcia, Learning from imbalanced data, IEEE Trans. Knowl. Data Eng. 21 (9) (2009) 1263–1284.
[70] J. Yang, J. McAuley, J. Leskovec, Community detection in networks with node attributes, in: 2013 IEEE 13th International Conference on Data Mining, IEEE, 2013, pp. 1151–1156.
[71] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2016, arXiv preprint arXiv:1609.02907.
[72] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, M. Sun, Graph neural networks: A review of methods and applications, AI Open 1 (2020) 57–81.
[73] P. Zhang, J. Wang, X. Li, M. Li, Z. Di, Y. Fan, Clustering coefficient and community structure of bipartite networks, Physica A 387 (27) (2008) 6869–6875.
[74] M. Welling, T.N. Kipf, Semi-supervised classification with graph convolutional networks, in: J. International Conference on Learning Representations (ICLR 2017), 2016.
[75] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[76] S. Pan, Q. Yang, A survey on transfer learning, ieee Trans. Knowl. Discov. Data Eng. 22 (10) (2010).
[77] A. Lancichinetti, S. Fortunato, Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities, Phys. Rev. E 80 (1) (2009) 016118.
[78] Y. Liu, H. Gao, X. Kang, Q. Liu, R. Wang, Z. Qin, Fast community discovery and its evolution tracking in time-evolving social networks, in: 2015 IEEE International Conference on Data Mining Workshop (ICDMW), 2015, pp. 13–20, http://dx.doi.org/10.1109/ICDMW.2015.177.
[79] J. Mcauley, J. Leskovec, Discovering social circles in ego networks, ACM Trans. Knowl. Discov. Data (TKDD) 8 (1) (2014) 1–28.
[80] Y. Niu, D. Kong, L. Liu, R. Wen, J. Xiao, Overlapping community detection with adaptive density peaks clustering and iterative partition strategy, Expert Syst. Appl. 213 (2023) 119213, http://dx.doi.org/10.1016/j.eswa.2022.119213, URL https://www.sciencedirect.com/science/article/pii/S095741742202231X.
[81] A. Bouyer, H. Ahmadi Beni, B. Arasteh, Z. Aghaee, R. Ghanbarzadeh, Fip: A fast overlapping community-based influence maximization algorithm using probability coefficient of global diffusion in social networks, Expert Syst. Appl. 213 (2023) 118869, http://dx.doi.org/10.1016/j.eswa.2022.118869, URL https://www.sciencedirect.com/science/article/pii/S0957417422018875.
[82] X. Zhang, N. Guan, W. Zhang, X. Huang, S. Wu, Z. Luo, Symmetric non-negative matrix factorization based link partition method for overlapping community detection, in: 2015 IEEE International Conference on Systems, Man, and Cybernetics, IEEE, 2015, pp. 2198–2203.
[83] A. Lutov, D. Yang, P. Cudré-Mauroux, Bridging the gap between community and node representations: Graph embedding via community detection, in: 2019 IEEE International Conference on Big Data (Big Data), IEEE, 2019, pp. 2681–2690.
[84] A.K. Sangaiah, S. Rezaei, A. Javadpour, W. Zhang, Explainable ai in big data intelligence of community detection for digitalization e-healthcare services, Appl. Soft Comput. 136 (2023) 110119, http://dx.doi.org/10.1016/j.asoc.2023.110119.
[85] Y. Niu, D. Kong, L. Liu, R. Wen, J. Xiao, Overlapping community detection with adaptive density peaks clustering and iterative partition strategy, Expert Syst. Appl. 213 (2023) 119213.
[86] M. Curado, L. Tortosa, J.F. Vicent, A novel measure to identify influential nodes: return random walk gravity centrality, Inform. Sci. 628 (2023) 177–195.
[87] M. Bilal, A. Khan, S. Jan, S. Musa, S. Ali, Roman urdu hate speech detection using transformer-based model for cyber security applications, Sensors 23 (8) (2023) 3909.
[88] A.A. Rezaei, J. Munoz, M. Jalili, H. Khayyam, A machine learning-based approach for vital node identification in complex networks, Expert Syst. Appl. 214 (2023) 119086.
[89] S. Mandal, B.K. Singh, K. Thakur, Epileptic seizure detection using deep learning based long short-term memory networks and time-frequency analysis: a comparative investigation in machine learning paradigm, Braz. Arch. Biol. Technol. 65 (2022).
[90] S. Kumar, A. Mallik, B. Panda, Influence maximization in social networks using transfer learning via graph-based lstm, Expert Syst. Appl. 212 (2023) 118770.