IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 23, 2021, accepted September 4, 2021, date of publication September 7, 2021, date of current version September 14, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3110845*

# Identifying Protein Complexes in Protein-Protein Interaction Data Using Graph Convolutional Network

**NAZAR ZAKI[1,2], HARSH SINGH[1], AND ELFADIL A. MOHAMED[3]**

[1]Big Data Analytics Center (BIDAC), United Arab Emirates University (UAEU), Al Ain, United Arab Emirates
[2]Department of Computer Science and Software Engineering, College of Information Technology, United Arab Emirates University (UAEU), Al Ain, United Arab Emirates
[3]College of Engineering and Information Technology, Ajman University, Ajman, United Arab Emirates

Corresponding author: Nazar Zaki (nzaki@uaeu.ac.ae)

**ABSTRACT** Protein complexes are groups of two or more polypeptide chains that bind to form noncovalent networks of protein interactions. Over the past decade, researchers have created a number of means of computing the ways in which protein complexes and their members can be identified through these interaction networks. Although most of the existing methods identify protein functional complexes from the protein-protein interaction networks (PPIs) at a fairly decent level, the applicability of advanced graph network methods has not yet been adequately investigated. This paper proposes various graph convolutional network (GCN) methods to improve the detection of protein complexes. We first formulate the protein complex detection problem as a node classification problem. Then, we developed a Neural Overlapping Community Detection (NOCD) model to cluster the nodes (proteins) using a complex affiliation matrix. A representation learning approach, that combines a multi-class GCN feature extractor (to obtain the nodes' features) and a mean shift clustering algorithm (to perform the clustering), is also utilized. We convert the dense-dense matrix operations into dense-sparse or sparse-sparse matrix operations to improve the efficiency of the multi-class GCN network by reducing space and time complexities. The proposed solution significantly improves the scalability of the existing GCN. Finally, we apply clustering aggregation to find the best protein complexes. A grid search is then performed on various detected complexes obtained via three well-known protein detection methods, namely ClusterONE, CMC, and PEWCC, with the help of the Meta-Clustering Algorithm (MCLA) and the Hybrid Bipartite Graph Formulation (HBGF). We test the proposed GCN-based methods on various publicly available datasets and find that they perform significantly better than previous state-of-the-art methods. The code/data are available for free download from https://github.com/Analystharsh/GCN_complex_detection.

**INDEX TERMS** Protein complex detection, graph convolutional network (GCN), protein-protein interaction (PPI), neural overlapping community detection (NOCD), meta-clustering algorithm (MCLA), hybrid bipartite graph formulation (HBGF).

## I. INTRODUCTION

Proteins are key drivers of growth and development in all organisms, yet most of the cellular functions of living systems are not driven by individual proteins. Instead, many protein nodes, also known as protein complexes or protein communities, contribute to cellular function; as these proteins control the overwhelming majority of biological processes within cells, they also control appropriate cell functionality. Cells necessarily respond to several stimuli, and the cellular response is a complex procedure involving the assignation of particular tasks to specific proteins, meaning a certain type and number of proteins are needed for any given function. Therefore, biologists have recently shifted their attention away from the relationships between the structures and functions of individual protein families towards the consideration of cellular networks as a whole [1]. For a complete comprehension of the functions of a protein, it must be examined in

The associate editor coordinating the review of this manuscript and approving it for publication was Hocine Cherifi.

light of its interaction partners and the complex to which it belongs.

It is generally acknowledged that protein complexes comprise groups of two or more interacting polypeptide chains [1]. The ability to detect such complexes is crucial as they are central to biological processes and create the framework for the network of protein-protein interactions (PPIs). For example, protein complex formation is responsible for antigen-antibody interaction and transportation, gene expression control, cell cycle control, signaling, differentiation, protein folding, transcription, translation, and enzyme inhibition [1]. Thus, building or destroying various protein complexes triggers the initialization, modulation, or termination of several biological processes [2]. Meanwhile, gene mutations can lead to substantial protein complex abnormalities [3], [4], which may, in turn, influence how proteins interact with other partners. In particular, they also may modify the interactions among different proteins, and in certain instances, may also initiate self-interaction [3]. These modifications are small, yet they are nonetheless important as they are linked to significant numbers of alterations in self-functionality; thus, in-depth knowledge of these interactions can assist in the development of medical treatments. Additionally, knowledge of protein complexes can provide a greater understanding of various forms of certain diseases. For example, studies have demonstrated that some genetic diseases are caused by proteins with similar functional interactions [1], [5]. Using data extracted from PPI can thus aid researchers in the discovery of inter-gene evolutionary relationships that can guide them to unique protein complexes, leading to the identification of unique genes related to specific diseases [3]. Additionally, as they play a significant role in physiological function, which makes them superior to standard in vitro methods for therapeutic agent analysis, knowledge of protein complexes is making significant contributions to the creation of new drug therapies [6]. Finally, the investigation into protein complexes may reveal previously undiscovered pathways and proteins, new methods for disease control, and new ways to classify genes. All of these elements will enable the development of novel methods for targeting, identifying, and retarding disease progression.

Many previous successful methods have been put forward for the detection of protein complexes from PPI networks which can be divided into the following seven categories:

1) A local neighborhood density search approach, focusing on the discovery of dense subgraphs inside the input network, including MCODE [7], DPClus [6], ProRank [8], [9], ProRank+ [10], CMC [11], PROCOMOSS [12] and PEWCC [13], NCMine [14], Core&Peel [15], SPICi [16] and non-cooperative sequential game [17].
2) Local search approaches based on cost, focusing on the extraction of modules from interaction graphs through the partition of the graphs into linked subgraphs employing cost functions for the guidance of searches towards the optimal partition, including RNSC [18], ModuLand [19], and STM [20].
3) Approaches employing Flow Simulation, focusing on the imitation of ways in which information spreads through a network, including MCL [21] and RW [5].
4) Approaches based on statistics, relying on the employment of statistical concepts for clustering proteins, e.g. how many shared neighbors pair of proteins have, and on notions of referential attachments for module members with other elements within the module; this includes SL [22], idenPC-MIIP [23], idenPC-CAP [24] and Farutin [1].
5) Stochastic search methods based on population employed to develop algorithms used to detect communities and networks including CGA [2], IGA [6], and EHO-MCL [25].
6) Approaches based on modularity, topological structure, overlapping information, and GO annotations, including CFinder [26] and [27], ClusterONE [28], and SE-DMTG [29].
7) Graph-based clustering methods, which includes statistical-based measures methods such as [1] which uses the concept of statistical significance to measure the strength of the relationship between two nodes (proteins), which requires prior estimation of the p-value. Cost-based Local search (CL) [30], Population-based Stochastic search (PS) [2] and [6], Local neighborhood Density search (LD), [7] and [28].

While the methods mentioned above can identify protein complexes with a fair level of accuracy, in this paper, we make the following three major contributions to further improve the detection of protein complexes in a PPI network.

- **Contribution 1**: We employ node classification approaches [31] to classify nodes (proteins) into classes (complexes). First, the interaction matrix (adjacency matrix) and degree matrix are prepared from a given PPI network. Second, an identity matrix is used as a feature representation of the nodes. Using these inputs, three GCN models [32] are employed. (1) a multi-class GCN classification with a $2^N$ label size (where $N$ is the number of complexes), (2) a multi-class GCN classification with label size $K$ (where $K$ is the number of possible combinations of the labels in the respective datasets), and (3) a multi-label GCN classification. These classification methods provide the protein complex labels for all the nodes (proteins) in the network. These models not only detect non-overlapping communities but are also self-sufficient in the definition itself to detect overlapping complexes [33].
- **Contribution 2**: Two representations learning based approaches are proposed for complex detection. The first approach, the "NOCD GCN method [34]" uses a generative model to learn the community affiliation matrix from the node features and adjacency matrix. This requires modeling the loss function in terms of

negative likelihood involving the Bernoulli Poisson (BP) method. The second approach extracts feature from the existing pre-trained GCN model and uses feature embedding to form clusters using the mean shift algorithm [35]. GCN approaches are further advanced by proposing efficient matrix operations inside the GCN layers. The dense matrices involved in the GCN model, such as the feature, adjacency, and degree matrices are converted to the compressed sparse row (CSR) matrix format [36]. This removes the redundant operations from the existing GCN architectures.

- **Contribution 3**: A clustering aggregation process [37] is proposed, which takes all the resulting clusters yielded by applying well-known protein complex detection methods without knowing which is the best performer and produces the optimal clusters. In this case, a grid search is performed in conjunction with the Meta-Clustering Algorithm [38] (MCLA) and the Hybrid Bipartite Graph Formulation (HBGF) [37].

## II. METHODS

### A. DATASETS

The PPI interaction networks of the datasets were extracted from the BioGrid interaction database [39]. These datasets provide PPI networks for two species namely ''Homo sapiens'' and ''Mus musculus'' and are termed the ''Human'' and ''Mouse'' datasets, respectively. These raw datasets were pre-processed by removing duplicate nodes/interaction edges and merging all available PPIs networks for particular species.

The CORUM reference complexes dataset [40] which includes 623 ''Mouse'' related reference complexes and 2,645 ''Human'' related complexes were used to evaluate the performance of the proposed methods.

In addition, two popular datasets were also considered in this study, namely the Collins and Gavin datasets [41]. The Gavin dataset was extracted by computing the socio-affinity index in all yeast PPI networks, as proposed by the original authors. If this term is greater than 5, then that PPI is considered; otherwise, it is excluded. Meanwhile a different metric (purification enrichment test) was used to retain the Collins dataset. The details of the two PPI datasets are summarised in Table 1.

### B. DEVELOPING THE MULTI-CLASS GCN CLASSIFICATION MODEL

Inspired by the recent promising results achieved by applying graph-based learning techniques for detecting communities in graphs [34], we employed the GCN [32] method to classify proteins into classes (complexes). The method is initiated by creating an adjacency matrix $A$ of a shape $N \times N$ (where $N$ is the number of nodes). In this case, if two nodes (proteins) are connected (interact), then the corresponding entry in the adjacency matrix is represented as 1 or 0 otherwise. Therefore, the input feature matrix $F_1$ is considered as an identity matrix of shape $N \times N$, which presents the features for each node in the absence of the explicit node features. The input adjacency matrix is formed by adding the feature matrix $F_1$ and $\hat{A}$ to add self-connection of each node in the adjacency matrix ($\hat{A} = A + F_1$). This input feature matrix is then normalized [32] as $Z = D^{-1/2}\hat{A}D^{-1/2}$ ($D$ is degree matrix). In this case, each graph neural network layer will consist of its weight matrices $W_i$. The $W_1$ for example is the weight matrix of the shape $N \times 512$ for the first GCN layer [32]. We can then obtain the matrix $K$ by multiplying the feature matrix $F_1$ by weight matrix $W_1$ ($K = F_1\ W_1$). Further, the matrix $K$ is multiplied by $Z$ to obtain $P$ ($P = ZK$), which is then passed to the ReLU activation function $F_2 = ReLU(P)$. This output is again passed to another GCN layer with weight matrix $W_2$ of shape $512 \times L$ as a feature matrix for the next layer, where $L$ is the length of each node label. This process is repeated using $F_2$, $W_2$, $\hat{A}$, and $D$.

Finally, the output $F_3$ obtained from the second GCN layer is passed to the row-wise softmax function $Y = \sigma(F_3)$ where $\sigma$ is row-wise Softmax function. With the help of $Y$ and $\hat{Y}$, the presented GCN is trained with the loss function defined as categorical cross-entropy $L = H(Y, \hat{Y})$, where $H(;,;)$ represents categorical cross-entropy. The weighted matrices $W_1$ and $W_2$ are updated during the training. Overview of this approach has been illustrated in Figure 1. Based on the type of label matrices, there are two versions of multi-class GCN approach. In first version, each label has size $2^{N_c}$ where $N_c$ is selected number of complexes. Hence, each label is unique whether node is belonging to one or more complexes. In the second version, the label matrix $Y$ for all the nodes in the input graph is prepared by inspecting the possible number of combinations $K$ for nodes belonging to one or more complexes. In this way, each label is formed as one hot encoder of the length equal to the number of such possible combinations for the particular dataset.

To extend the method to handle multi-label classification, instead of the softmax function, we used the sigmoid function after the second GCN layer, and the loss function was also changed into a class-wise summation of binary cross-entropy. Therefore, the loss function is $L = \sum_{c_i \in C}(BC(Y_{c_i}, \hat{Y}_{c_i}))$ where $C$ is the set of all classes, $c_i$ is class $i$ of set $C$, and $BC$ is the binary cross-entropy. Finally, the output values $>0.5$ are converted into 1, otherwise, they are converted into 0.

#### 1) EXPLICIT NODE FEATURES

In our experiments, we primarily used an identity matrix as feature embedding. Moreover, we also tried a custom feature matrix, which was built using the RNA-RNA interaction networks and RNA-protein interaction networks [24]. For the node classification task, the adjacency matrix and feature matrix were required as input. The following process was used to obtain the feature matrix (feature embedding) from the above mentioned two extra networks:

- The RNA-RNA interaction networks are clustered using greedy modularity [42]. In this case, almost all the RNA nodes can be clustered into three big groups. Therefore, the feature length for each node is assigned to 3,

**TABLE 1.** Details of the PPIs networks used in the study.

| PPI Dataset | Homo sapiens | Mus musculus | Collin | Gavin |
|---|---|---|---|---|
| Number of Proteins | 17737 | 7301 | 1622 | 1430 |
| Number of Interactions | 338923 | 20679 | 9074 | 6531 |
| Average Number of Neighbors | 38.237 | 5.747 | 16.58 | 9.494 |
| Network Diameter | 8 | 7 | 15 | 13 |
| Clustering Coefficient | 0.12 | 0.133 | 0.648 | 0.42 |
| Network Density | 0.002 | 0.001 | 0.017 | 0.007 |
| Network Heterogeneity | 2.212 | 5.29 | 1.124 | 1.049 |
| Characteristic Path Length | 2.969 | 3.696 | 5.542 | 4.485 |

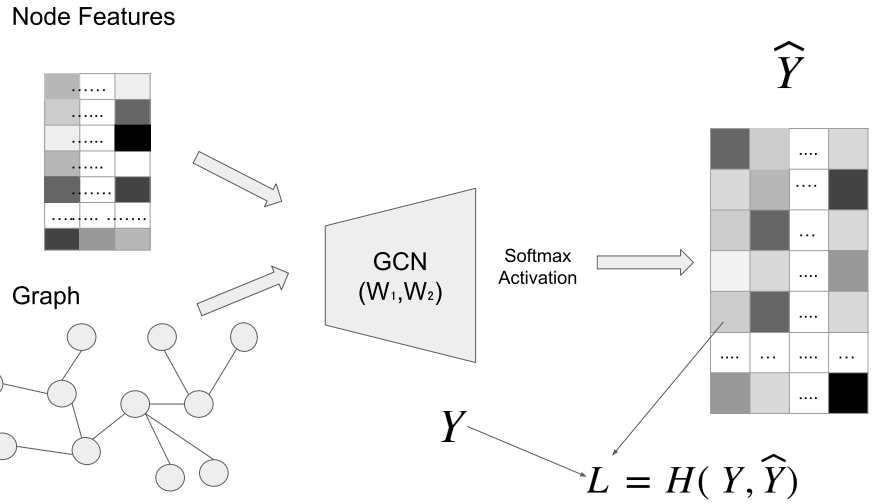


**FIGURE 1.** Overview of the multi-class GCN node classification.

which is one hot encoder with a length of 3. Entry 1 in this feature embedding represents the inclusion of the relationship with corresponding RNA clusters, while entry 0 represents the non-inclusion of that RNA-RNA cluster. The Girvan-Newman [43] yields one big cluster (55600 nodes out of 56000 nodes in total)

- The RNA-RNA clusters are labeled with unique IDs.
- Each protein is labeled as per the RNA clusters to which it is connected in the RNA-protein network. Generally, it is in the sparse matrix form as most of the entries of this feature matrix are zeros.
- These node feature embedding is used as a feature matrix.

### C. REPRESENTATION LEARNING APPROACHES

One of the limitations of GCN approaches is the computational cost. As the process includes large and sparse matrix addition, matrix inversion, matrix multiplication, etc., the use of dense matrices incurs high computational costs. To address this issue, we have converted the input sparse matrices into the compressed sparse row (CSR) matrix format [36].

#### 1) THE NEURAL OVERLAPPING COMMUNITY DETECTION - GCN METHOD

This section, describes the probabilistic generative GCN model, which does not require any label for the training purpose. Rather, it simply takes the graph adjacency matrix and node feature matrix as inputs. It subsequently uses the Neural Overlapping Community Detection (NOCD) model [34] to learn the connectivity among the nodes (protein) and optimize the weights accordingly. It is a completely unsupervised method that relies on the probabilistic modeling of the output. This output is termed the complex affiliation matrix $F$. Thus, the problem boils down to the $p(A|F)$ estimation, where $A$ is the adjacency matrix. If the complex affiliation matrix is prior, the entries in the adjacency matrix can be sampled as, $A_{uv} \sim Bernoulli(1 - e^{F_u F_v^T})$, where $u$ and $v$ are two nodes. In this case, the higher value of $F_u F_v^T$ indicates a higher chances that $u$ and $v$ are connected and in the same community. All the settings are kept the same as for the multi-class GCN classification method, but the row-wise softmax function is changed to the element-wise ReLU activation after the second GCN layer. The other differences in the GCN architectures are as follows:

- The dropout layer is used in the last GCN layer
- $L_2$ regularization is applied to both weight matrices in the network
- The batch normalization layer is also added to the first GCN layer

In this case, the output is the complex affiliation matrix $F$, not $\hat{Y}$, which uses labels to perform the training. Therefore, the negative likelihood of the proposed model can be

written as:

$$-\log p\,(A|F) = -\sum_{(u,v)\in E} \log(1 - e^{-F_u F_v^T}) + \sum_{(u,v)\notin E} F_u F_v^T \tag{1}$$

To reduce the effect of non-edges (sparse effect), we select only a certain number of non-edges to balance the estimation. This new term is written as follows:

$$L = -E_{(u,v)\sim P_E}\left[\log(1 - e^{-F_u F_v^T})\right] + E_{(u,v)\sim P_N}\left[F_u F_v^T\right] \tag{2}$$

where $P_E$ and $P_N$ indicate the uniform distribution over edges and non-edges. By minimizing this loss function, we can optimize the weights of the hidden layer of the GCN. This hidden layer has a size of 512 (Figure 2).

#### 2) GCN FEATURE EXTRACTION AND UNSUPERVISED FEATURE LEARNING

In this step, we extracted features for all the nodes in the datasets from the last layer of the second version of multi-class GCN classification model (before applying the row-wise softmax function) pre-trained on the Human/Mouse dataset.

Once these features are retrieved, we have applied mean shift algorithm, which does not require prior information about the number of clusters.

This algorithm accepts the node features, estimates the number of clusters, and assigns nodes to clusters. The kernel that used in this experimental work was a flat/uniform kernel. In this case, the kernel $K(u) = \frac{1}{2h}\begin{cases}1, & ||u|| \le 1\\ 0, & \text{otherwise}\end{cases}$, where $u$ is the data point and $h$ is the bandwidth of the kernel. This algorithm identifies dense regions by using the following kernel density estimation (KDE) function:

$$f_k\,(u) = \frac{1}{nh^d}\sum_{i=1}^{n} K\left(\frac{u - u_i}{h}\right) \tag{3}$$

where $n$ is the total number of data points, and $u_i$ is an ith data point. First, this algorithm finds dense regions with a predefined bandwidth. Second, it determines the mean of the data points in that region, and then shifts its centroid toward the mean point. The second step is repeated until the shifting of centroids cases. This is also called the convergence of the mean shift algorithm. In this way, it computes the optimal number of clusters.

### D. CLUSTERING AGGREGATION

#### 1) META-CLUSTERING ALGORITHM (MCLA)

The objective of the MCLA is to combine clusters obtained from different clustering techniques. It also provides the association confidence estimation of all the instances (or data points). The MCLA uses hyperedges [44] as the starting vertices. In this case, the hyperedges are the members of the indicator matrix (consider indicator matrix as a set of column vectors) $H^l$. It maps the labels of all the clustering into corresponding binarized column vectors. The number of hyperedges, in this case, depends on the number of clusters. If the number of clusters is $k^1$, $k^2$, ....., $k^p$ then $\sum_{l=1}^{p} k^l$ presents the total number of hyperedges. Here, $k^l$ denotes the number of clusters in lth clustering, and p denotes the number of clusterings. These indicator matrices can be collectively written, as $H = (H^1, H^2, \ldots, H^p)$. It is also called the adjacency matrix of the hypergraph. Each of the column vectors of the hypergraph adjacency matrix is a specific hyperedge. Each row of the indicator matrix $H^l$ represents the corresponding labels of clustering $l$. The key concept of the MCLA is to combine similar hyperedges and form meta-hyperedges. Later the instances (objects) are assigned to each of these meta-hyperedges based on the association membership values. The steps for the MCLA are as follow:

1) Forming meta-graphs from hyperedges
2) Transforming the meta-graph into meta-clusters
3) Creating meta-hyperedges
4) Object association contest among the meta-hyperedges

#### 2) HYBRID BIPARTITE GRAPH FORMULATION (HBGF)

HBGF method [37] treats clusters and data points as its basic entities. The first step in this algorithm is creating a bipartite graph and then partitioning the graph to obtain optimized clustering with optimal clusters. Each part of the partitioned bipartite graph represents the consensus cluster. For clustering $(C_1, C_2, \ldots, C_n)$, a bipartite graph can be represented by $G\,(V, E)$. Here, $V$ represents the set of instances $(v_1, v_2, \ldots, v_n)$, clustering $(C_1, C_2, \ldots, C_n)$, and E represents the edges between the nodes. The edges are undirected, and every node in the graph has an edge with the other nodes. Each edge has a weight $W\,(i, j)$ associated with it. Here, $i$ and $j$ represent the nodes. The edge weight between nodes $i$ and $j$ is defined as in Equation 4.

$$W\,(i,j) = \begin{cases}0, & \text{if } i \text{ and j both are clusters or instances}\\ 1, & \text{if } i \text{ is cluster and } j \text{ is instance or vice versa}\end{cases} \tag{4}$$

### E. EVALUATION MEASURES

#### 1) TEST ACCURACY AND SUBSET ACCURACY

To measure the node classification accuracy, one hot encoder is used as the label for all the nodes. A value of 1 denotes the activation of the corresponding class, while a value of 0 depicts the deactivation of the corresponding class. The predicted outcome for a sample is called predicted matched only if both the outcome and label have 1 in the same place. If the total number of labels in the test set is $U$, and the number of total predicted matched sample outcomes is $V$, then the test accuracy ($TA$) is defined as $TA = \frac{V}{U}$. However, in the case of multi-label classification, a method such as a subset accuracy [45] can also be used.

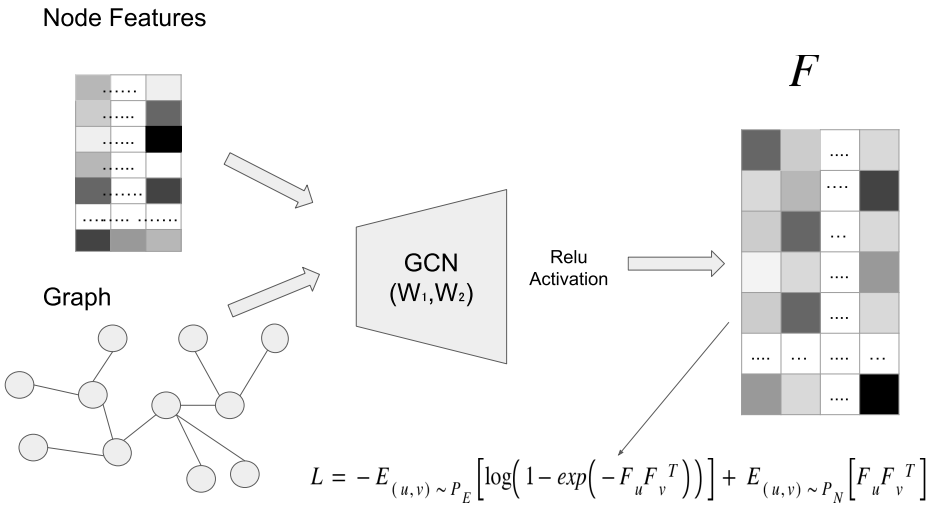


**FIGURE 2.** Overview of the NOCD GCN model.

#### 2) HAMMING LOSS AND HAMMING SCORE

Hamming loss ($HL$) and Hamming score ($HS$) [46] were also used and they are calculated as follows:

$$HL = \frac{1}{nL}\sum_{i=1}^{n}\sum_{j=1}^{L}\left(S\left(y_j^i, \hat{y}_j^i\right)\right) \tag{5}$$

$$S\left(y_j^i, \hat{y}_j^i\right) = \begin{cases} 1, & \text{if } y_j^i \neq \hat{y}_j^i \\ 0, & \text{if } y_j^i = \hat{y}_j^i \end{cases} \tag{6}$$

$$HS = 1 - HL \tag{7}$$

where $y_j^i$ denotes the value of class $j$ of label $i$, while $\hat{y}_j^i$ denotes the class $j$ of predicted outcome of sample $i$. $L$ is the label size and $n$ is the number of labels.

#### 3) PRECISION, RECALL, AND F-MEASURE

The overlapping score ($OS$) between the ground truth complex $P$ and the predicted complex $Q$ is defined as:

$$OS(P, Q) = \frac{|P \cap Q|^2}{|P||Q|} \tag{8}$$

In this case the threshold for $OS(P, Q)$ is set as 0.2 [47]. This denotes that if the value of the overlapping score between $P$ and $Q$ is 0.2, then both match each other. The Precision and Recall [48], [49] are defined as:

$$Precision = \frac{|KMC|}{|TPC|} \tag{9}$$

$$Recall = \frac{|PMC|}{|TKC|} \tag{10}$$

where TKC is the Total Known Complexes, TPC is the Total Predicted Complexes, PMC is the number of Predicted Matched Complexes, and KMC is the number of Known Matched Complexes. The F -Measure is calculated as follows:

$$F - measure = \frac{2 \times Precision \times Recall}{(Precision + Recall)} \tag{11}$$

#### 4) NORMALIZED MUTUAL INFORMATION (NMI)

NMI [50] is another powerful performance metric to evaluate clustering methods given ground truth complexes. Assume Y is the ground truth label matrix for all the nodes, and $\hat{Y}$ is the clustering label matrix for all the nodes obtained using the applied method. NMI is defined as follows.

$$NMI(Y, \hat{Y}) = \frac{2 \times I\left(Y; \hat{Y}\right)}{\left[H\left(Y\right) + H\left(\hat{Y}\right)\right]} \tag{12}$$

where $H(\hat{Y})$ is entropy for the clustering labels and $H(Y)$ is entropy for the ground truth labels. $I(:, :)$ is termed as mutual information [50].

## III. EXPERIMENTAL WORK AND RESULTS

### A. PROTEIN CLASSIFICATION APPROACHES

This experimental work used the Collins, Human, and Mouse datasets. The loss function was minimized through the Adam optimizer, with an initial learning rate of 0.01. The number of epochs was 200. The first version of multi-class GCN classification model had each label of length $2^N$, providing a test accuracy of 84.26% with a train-to-test data ratio of 80:20 for the Collins dataset, where $N$ is the total number of complexes ($N = 10$ in our experiment). The multi-label classification model with top 100 complexes (complexes having maximum number of protein nodes) provided Hamming Loss, Hamming score, and Subset accuracy of 0.0136, 0.9864, and 0.1554, respectively Table 2. To improve the performance,

**TABLE 2.** Overall performance comparison of (GCN + mean-shift) model with other states-of-the-arts algorithms on human and mouse datasets.

| Method | PMC | TPC | KMC | Precision | Recall | F-measure |
|---|---|---|---|---|---|---|
| **Human** | | | | | | |
| GCN + mean-shift (ours) | 844 | 1128 | 1381 | 0.523 | 0.748 | 0.616 |
| idenPC-CAP [24] | 1784 | 5504 | 1767 | 0.324 | 0.668 | 0.436 |
| idenPC-MIIP [23] | 391 | 1019 | 811 | 0.384 | 0.307 | 0.341 |
| NCMine [14] | 2387 | 14401 | 1240 | 0.166 | 0.469 | 0.245 |
| Core&Peel [15] | 2923 | 13084 | 1559 | 0.223 | 0.589 | 0.324 |
| ClusterONE [28] | 373 | 924 | 638 | 0.404 | 0.241 | 0.302 |
| SPICi [16] | 485 | 1899 | 978 | 0.256 | 0.370 | 0.302 |
| CMC [11] | 6443 | 53734 | 1942 | 0.120 | 0.734 | 0.206 |
| MCL [21] | 375 | 2676 | 652 | 0.140 | 0.247 | 0.179 |
| **Mouse** | | | | | | |
| GCN + mean-shift (ours) | 121 | 139 | 167 | 0.357 | 0.871 | 0.506 |
| idenPC-CAP [24] | 421 | 1696 | 324 | 0.248 | 0.520 | 0.336 |
| idenPC-MIIP [23] | 105 | 403 | 146 | 0.261 | 0.234 | 0.247 |
| NCMine [14] | 408 | 3127 | 303 | 0.130 | 0.486 | 0.206 |
| Core&Peel [15] | 368 | 2563 | 238 | 0.144 | 0.382 | 0.209 |
| ClusterONE [28] | 170 | 1006 | 152 | 0.169 | 0.244 | 0.200 |
| SPICi [16] | 121 | 457 | 195 | 0.265 | 0.313 | 0.287 |
| CMC [11] | 143 | 862 | 188 | 0.166 | 0.302 | 0.214 |
| MCL [21] | 120 | 1098 | 192 | 0.109 | 0.308 | 0.161 |

a modified version of the multi-class GCN method was used, as it offers both greater flexibility in terms of the number of communities and reduced space and time complexities (please refer to section II). In this case, the model was able to achieve 0.676 precision, 0.837 recall, and 0.748 F-score for the Human dataset. These approaches have provided good results, but all of them are using labels while training.

### B. REPRESENTATION LEARNING APPROACHES

The NOCD GCN model is implemented on top 50 communities with the hidden size of 512, and, a weight decay of 1e-3, the learning rate was kept at 1e-4, the dropout rate was 0.05, and the batch size was selected as 20,000, and Adam was used as optimizer. The number of epochs, in this case, was 200. The results of the NOCD GCN model for the Mouse dataset are presented in Table 3. Performance starts decreasing as we increase the number of complexes.

As described in section II, customized feature matrix was constructed an NOCD GCN model was applied using this feature matrix. For the top 10 complexes of the Human dataset, the NMI score was recorded as 0.5. the NMI score started decreasing as we increase the number of complexes. For top 100 complexes, NOCD GCN method with a customized feature matrix achieved an NMI score of 0.105 for the Human dataset.

Representation learning with the feature extracted through the pre-trained GCN from the Mouse and Human datasets has been performed with the clustering algorithm mean shift. The results of this feature learning algorithm are provided in Table 2. Results for other methods are obtained from [24].

**TABLE 3.** Performance metrics of the NOCD GCN model for the mouse dataset.

| Metrics | Value |
|---|---|
| NMI | 0.404 |
| PMC | 23 |
| TPC | 50 |
| KMC | 30 |
| Precision | 0.60 |
| Recall | 0.46 |
| F-measure | 0.52 |

The number of epochs was 400, in this case, and a hidden layer sizes 512 were selected for this process with Adam as optimizer. The remaining settings were kept as for the second version of multi-class GCN classification experimental work. It became clear that the proposed approach for clustering proteins (nodes) outperforms all state-of-the-art clustering algorithms, thereby proving the effectiveness of the proposed approach.

### C. CLUSTERING AGGREGATION

Both MCLA and HBGF algorithms require a prior estimation of the number of clusters. Thus, we performed a grid search using the HBGF and MCLA to get the optimal number of clusters. Gavin dataset is used in this case to test this approach. First, three classical protein complex detection techniques (CMC [11], ClusterONE [28], and PEWCC [13]) were used to detect clusters in Gavin PPI dataset. The number of clusters yielded using the three techniques are 124, 243, and 206, respectively. After applying the MCLA and HBGF algorithms on the detected clusters using grid search from 60 complexes to 250 complexes. The best performances

**TABLE 4.** Performance metrics using the MCLA and HBGF models.

| | MCLA | HBGF | MCL [21] | ClusterONE [28] | PEWCC [13] |
|---|---|---|---|---|---|
| **Precision** | 0.82 | 0.6 | 0.122 | 0.83 | 0.71 |
| **Recall** | 0.33 | 0.68 | 0.121 | 0.337 | 0.49 |
| **F-Measure** | 0.47 | 0.64 | 0.121 | 0.48 | 0.58 |

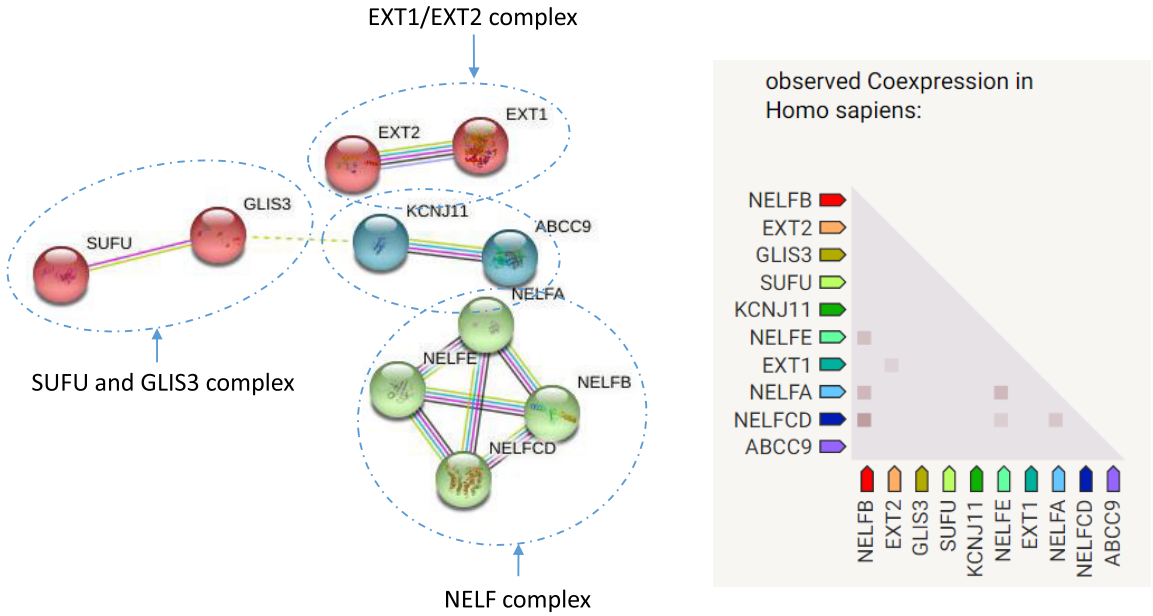


**FIGURE 3.** Samples of small human complexes detected 100% accurately by the proposed approaches which are hard to be detected by other methods. In the figure, the protein whose genes are observed to be correlated with expression in numerous experiments are also shown to highlight the correlation between the proteins in the functional complex. The co-expression scores based on RNA expression patterns and protein co-regulation were calculated using String [53].

(MCLA method with 237 complexes and HBGF method with 66 complexes) are shown in Table 4.

## IV. DISCUSSION AND CONCLUSION

In this paper, we introduced two GCN-based approaches to detect protein complexes in several benchmarked PPI datasets. The first approach is a multi-class GCN classification method, while the second is a multi-label GCN classification method. We also incrementally demonstrated improvements by changing the GCN method from a simple multi-class to a multi-label classification problem and then improvised the multi-class GCN method.

Following the incremental improvement, we proposed a sparse matrix operations-based GCN methodology. The efficiency in terms of the time complexity in the operations between the sparse and dense matrices was also compared. Then, we solved the problem of complex detection in an unsupervised condition. The NOCD model and the GCN feature extractor + mean shift clustering method were proposed, and the performances of both approaches were evaluated. The effectiveness of the proposed representation learning approach was demonstrated on Human and Mouse datasets, which showed that both outperform state-of-the-art methods in the detection of protein complexes. We also showed the effect of including explicit node features in the NOCD GCN method. Furthermore, we found that assembling yielded clusters (complexes) using the MCLA and HBGF has great potential. In the future, more complex detection methods should be added to the three explored in this experimental work to further improve the precision and recall scores.

Besides leveraging the advantages of the GCN, the proposed approaches were able to detect small complexes, which most state-of-the-art methods struggle to detect (as shown in Figure 3). For example, the proposed approaches accurately detected human complexes such as EXT1/EXT2 complex [51], [52], SUFU/GLIS3 complex, and NELF complex.

### A. LIMITATIONS AND FUTURE DIRECTIONS

This work did not conduct a thorough investigation of the feature clustering algorithm. The mean shift algorithm was selected from a pool of algorithms that includes the OPTICS and Affinity Propagation algorithms [54], [55]. Other techniques may perform even better than the mean-shift algorithm, and thus deserve further investigation.

Moreover, this study used an identity matrix and a small length of customized feature vectors to provide the node features. The length of the feature embeddings can be further increased with the help of the topological features present in the neighborhood of the protein nodes. This would form a robust feature matrix for the GCN operation, which might yield even better performance metrics.

Finally, since protein complexes are structured in many graph types, studying the characterization of a bipolar fuzzy detour g- eccentric node (protein) [56], [57], [58] represents one of our future research directions. Furthermore, we also plan to investigate the notion of bipolar fuzzy detour g-boundary nodes and bipolar fuzzy detour g-interior nodes in a bipolar fuzzy graph (PPI).

## COMPETING INTERESTS

The authors declare that they have no competing interests.

## REFERENCES

[1] V. Farutin, K. Robison, E. Lightcap, V. Dancik, A. Ruttenberg, S. Letovsky, and J. Pradines, "Edge-count probabilities for the identification of local protein communities and their organization," *Proteins, Struct., Function, Bioinf.*, vol. 62, no. 3, pp. 800–818, Dec. 2005.

[2] H. Liu and J. Liu, "Clustering protein interaction data through chaotic genetic algorithm," in *Proc. Asia–Pacific Conf. Simulated Evol. Learn.* Hefei, China: Springer, 2006, pp. 858–864.

[3] M. Altaf-Ul-Amin, Y. Shinbo, K. Mihara, K. Kurokawa, and S. Kanaya, "Development and implementation of an algorithm for detection of protein complexes in large interaction networks," *BMC Bioinf.*, vol. 7, no. 1, pp. 1–13, Dec. 2006.

[4] N. Zaki and H. Alashwal, "Improving the detection of protein complexes by predicting novel missing interactome links in the protein-protein interaction network," in *Proc. 40th Annu. Int. Conf. Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 5041–5044.

[5] K. Macropol, T. Can, and A. K. Singh, "RRW: Repeated random walks on genome-scale protein networks for local cluster discovery," *BMC Bioinf.*, vol. 10, no. 1, pp. 1–10, Dec. 2009.

[6] H. Ravaee, A. Masoudi-Nejad, S. Omidi, and A. Moeini, "Improved immune genetic algorithm for clustering protein-protein interaction network," in *Proc. IEEE Int. Conf. Bioinf. BioEng.*, May 2010, pp. 174–179.

[7] G. D. Bader and C. W. Hogue, "An automated method for finding molecular complexes in large protein interaction networks," *BMC Bioinf.*, vol. 4, no. 1, pp. 1–27, 2003.

[8] N. Zaki, J. Berengueres, and D. Efimov, "Detection of protein complexes using a protein ranking algorithm," *Proteins, Struct., Function, Bioinf.*, vol. 80, no. 10, pp. 2459–2468, Oct. 2012.

[9] N. Zaki, J. Berengueres, and D. Efimov, "ProRank: A method for detecting protein complexes," in *Proc. 14th Annu. Conf. Genetic Evol. Comput.*, 2012, pp. 209–216.

[10] E. M. Hanna and N. Zaki, "Detecting protein complexes in protein interaction networks using a ranking algorithm with a refined merging procedure," *BMC Bioinf.*, vol. 15, no. 1, pp. 1–11, Dec. 2014.

[11] G. Liu, L. Wong, and H. N. Chua, "Complex discovery from weighted PPI networks," *Bioinformatics*, vol. 25, no. 15, pp. 1891–1897, Aug. 2009.

[12] S. K. M. M. Hossain, Z. Mahboob, R. Chowdhury, A. Sohel, and S. Ray, "Protein complex detection in PPI network by identifying mutually exclusive protein-protein interactions," *Procedia Comput. Sci.*, vol. 93, pp. 1054–1060, Jan. 2016.

[13] N. Zaki, D. Efimov, and J. Berengueres, "Protein complex detection using interaction reliability assessment and weighted clustering coefficient," *BMC Bioinf.*, vol. 14, no. 1, pp. 1–9, Dec. 2013.

[14] S. Tadaka and K. Kinoshita, "NCMine: Core-peripheral based functional module detection using near-clique mining," *Bioinformatics*, vol. 32, no. 22, pp. 3454–3460, 2016.

[15] M. Pellegrini, M. Baglioni, and F. Geraci, "Protein complex prediction for large protein protein interaction networks with the Core&Peel method," *BMC Bioinf.*, vol. 17, no. S12, pp. 37–58, Oct. 2016.

[16] P. Jiang and M. Singh, "SPICi: A fast clustering algorithm for large biological networks," *Bioinformatics*, vol. 26, no. 8, pp. 1105–1111, Apr. 2010.

[17] U. Maulik, S. Basu, and S. Ray, "Identifying protein complexes in PPI network using non-cooperative sequential game," *Sci. Rep.*, vol. 7, no. 1, Dec. 2017, Art. no. 8410.

[18] A. D. King, N. Pržulj, and I. Jurisica, "Protein complex prediction via cost-based clustering," *Bioinformatics*, vol. 20, no. 17, pp. 3013–3020, 2004.

[19] I. A. Kovács, R. Palotai, M. S. Szalay, and P. Csermely, "Community landscapes: An integrative approach to determine overlapping network module hierarchy, identify key nodes and predict network dynamics," *PLoS ONE*, vol. 5, no. 9, Sep. 2010, Art. no. e12528.

[20] W. Hwang, Y.-R. Cho, A. Zhang, and M. Ramanathan, "A novel functional module detection algorithm for protein-protein interaction networks," *Algorithms Mol. Biol.*, vol. 1, no. 1, pp. 1–11, Dec. 2006.

[21] A. J. Enright, S. V. Dongen, and C. A. Ouzounis, "An efficient algorithm for large-scale detection of protein families," *Nucl. Acids Res.*, vol. 30, no. 7, pp. 1575–1584, 2002.

[22] M. P. Samanta and S. Liang, "Predicting protein functions from redundancies in large-scale protein interaction networks," *Proc. Nat. Acad. Sci. USA*, vol. 100, no. 22, pp. 12579–12583, 2003.

[23] Z. Wu, Q. Liao, and B. Liu, "IdenPC-MIIP: Identify protein complexes from weighted PPI networks using mutual important interacting partner relation," *Briefings Bioinf.*, vol. 22, no. 2, pp. 1972–1983, Mar. 2021.

[24] Z. Wu, Q. Liao, S. Fan, and B. Liu, "IdenPC-CAP: Identify protein complexes from weighted RNA-protein heterogeneous interaction networks using co-assemble partner relation," *Briefings Bioinf.*, vol. 22, no. 4, Dec. 2020, Art. no. bbaa372.

[25] R. R. Rani, D. Ramyachitra, and A. Brindhadevi, "Detection of dynamic protein complexes through Markov clustering based on elephant herd optimization approach," *Sci. Rep.*, vol. 9, no. 1, Dec. 2019, Art. no. 11106.

[26] I. Derényi, G. Palla, and T. Vicsek, "Clique percolation in random networks," *Phys. Rev. Lett.*, vol. 94, nos. 16–29, 2005, Art. no. 160202.

[27] B. Adamcsek, G. Palla, I. J. Farkas, I. Derényi, and T. Vicsek, "CFinder: Locating cliques and overlapping modules in biological networks," *Bioinformatics*, vol. 22, no. 8, pp. 1021–1023, 2006.

[28] T. Nepusz, H. Yu, and A. Paccanaro, "Detecting overlapping protein complexes in protein-protein interaction networks," *Nature Methods*, vol. 9, pp. 471–472, Mar. 2012.

[29] R. Wang, C. Wang, L. Sun, and G. Liu, "A seed-extended algorithm for detecting protein complexes based on density and modularity with topological structure and GO annotations," *BMC Genomics*, vol. 20, no. 1, pp. 1–28, Dec. 2019.

[30] J. Ruan and W. Zhang, "Identifying network communities with a high resolution," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 77, no. 1, Jan. 2008, Art. no. 016104.

[31] S. Abu-El-Haija, A. Kapoor, B. Perozzi, and J. Lee, "N-GCN: Multi-scale graph convolution for semi-supervised node classification," in *Proc. 35th Uncertainty Artif. Intell. Conf.*, 2020, pp. 841–851.

[32] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*. [Online]. Available: http://arxiv.org/abs/1609.02907

[33] T. S. Evans and R. Lambiotte, "Line graphs, link partitions, and overlapping communities," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 80, no. 1, Jul. 2009, Art. no. 016105.

[34] O. Shchur and S. Günnemann, "Overlapping community detection with graph neural networks," 2019, *arXiv:1909.12201*. [Online]. Available: http://arxiv.org/abs/1909.12201

[35] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995.

[36] N. Goharian, A. Jain, and Q. Sun, "Comparative analysis of sparse matrix algorithms for information retrieval," *Computer*, vol. 2, p. 4, Dec. 2003.

[37] X. Z. Fern and C. E. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning," in *Proc. 21st Int. Conf. Mach. Learn. (ICML)*, 2004, p. 36.

[38] R. Caruana, M. Elhawary, N. Nguyen, and C. Smith, "Meta clustering," in *Proc. 6th Int. Conf. Data Mining (ICDM)*, Dec. 2006, pp. 107–118.

[39] R. Oughtred, C. Stark, B.-J. Breitkreutz, J. Rust, L. Boucher, C. Chang, N. Kolas, L. O'Donnell, G. Leung, R. McAdam, F. Zhang, S. Dolma, A. Willems, J. Coulombe-Huntington, A. Chatr-aryamontri, K. Dolinski, and M. Tyers, "The BioGRID interaction database: 2019 update," *Nucleic Acids Res.*, vol. 47, no. D1, pp. D529–D541, Jan. 2019.

[40] A. Ruepp, B. Waegele, M. Lechner, B. Brauner, I. Dunger-Kaltenbach, G. Fobo, G. Frishman, C. Montrone, and H.-W. Mewes, "CORUM: The comprehensive resource of mammalian protein complexes—2009," *Nucleic Acids Res.*, vol. 38, pp. 497–501, Jan. 2007.
[41] U. Güldener, M. Münsterkötter, M. Oesterheld, P. Pagel, A. Ruepp, H.-W. Mewes, and V. Stümpflen, "MPact: The MIPS protein interaction resource on yeast," *Nucleic Acids Res.*, vol. 34, no. 90001, pp. 436–441, 2006.
[42] M. Chen, K. Kuzmin, and B. K. Szymanski, "Community detection via maximization of modularity and its variants," *IEEE Trans. Comput. Social Syst.*, vol. 1, no. 1, pp. 46–65, Mar. 2014.
[43] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 6, Jun. 2004, Art. no. 66133.
[44] P. Bonacich, A. Cody Holdren, and M. Johnston, "Hyper-edges and multidimensional centrality," *Social Netw.*, vol. 26, no. 3, pp. 189–203, Jul. 2004.
[45] R. B. Pereira, A. Plastino, B. Zadrozny, and L. H. Merschmann, "Correlation analysis of performance measures for multi-label classification," *Inf. Process. Manage.*, vol. 54, no. 3, pp. 359–369, May 2018.
[46] S. Destercke, "Multilabel prediction with probability sets: The Hamming loss case," in *Proc. Int. Conf. Inf. Process. Manage. Uncertainty Knowl.-Based Syst.*, 2014, pp. 496–505.
[47] Z. Wu, Q. Liao, and B. Liu, "A comprehensive review and evaluation of computational methods for identifying protein complexes from protein–protein interaction networks," *Briefings Bioinf.*, vol. 21, no. 5, pp. 1531–1548, Sep. 2020.
[48] W. Yang, X. J. Zhu, J. Huang, H. Ding, and H. Lin, "A brief survey of machine learning methods in protein sub-Golgi localization," *Current Bioinf.*, vol. 14, no. 3, pp. 234–240, 2019.
[49] H. N. Chua, K. Ning, W.-K. Sung, H. W. Leong, and L. Wong, "Using indirect protein-protein interactions for protein complex prediction," *J. Bioinf. Comput. Biol.*, vol. 6, no. 3, pp. 435–466, 2008.
[50] L. N. F. Ana and A. K. Jain, "Robust data clustering," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2003, pp. 1–2.
[51] K. Adelman and J. T. Lis, "Promoter-proximal pausing of RNA polymerase II: Emerging roles in metazoans," *Nature Rev. Genet.*, vol. 13, no. 10, pp. 720–731, Oct. 2012.
[52] Y. Aoi, E. R. Smith, A. P. Shah, E. J. Rendleman, S. A. Marshall, A. R. Woodfin, F. X. Chen, R. Shiekhattar, and A. Shilatifard, "NELF regulates a promoter-proximal step distinct from RNA Pol II pause-release," *Mol. Cell*, vol. 78, no. 2, pp. 261–274, Apr. 2020.
[53] B. Snel, G. Lehmann, P. Bork, and M. A. Huynen, "STRING: A web-server to retrieve and display the repeatedly occurring neighbourhood of a gene," *Nucleic Acids Res.*, vol. 28, no. 18, pp. 3442–3444, 2000.
[54] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, Feb. 2007.
[55] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowledg Discovery Data Mining (KDD)*, 1996, pp. 226–231.
[56] S. Poulik and G. Ghorai, "Detour $g$-interior nodes and detour $g$-boundary nodes in bipolar fuzzy graph with applications," *Hacettepe J. Math. Statist.*, vol. 49, no. 3, pp. 106–119, Mar. 2019.
[57] S. Poulik and G. Ghorai, "Certain indices of graphs under bipolar fuzzy environment with applications," *Soft Comput.*, vol. 24, no. 7, pp. 5119–5131, Apr. 2020.
[58] S. Poulik and G. Ghorai, "Determination of journeys order based on graph's Wiener absolute index with bipolar fuzzy information," *Inf. Sci.*, vol. 545, pp. 608–619, Feb. 2016.

**NAZAR ZAKI** received the Ph.D. degree from Universiti Teknologi Malaysia (UTM). He was a recipient of the Dean's recognition for a valuable Ph.D. work, in 2004. He worked as the Chair of the Department of Computer Science and Software Engineering, CIT, United Arab Emirates University (UAEU), for almost ten years introduced new academic programs and contributed significantly to the establishment and success of the department. He is currently a Professor in computer science (AI and machine-learning). He is the Founder and the Director of the Big Data Analytics Center with a mission to ingrain a sustained impact through groundbreaking data analytics research and services. He has published more than 120 scientific results in reputable journals and conferences. He is also a frequent recipient of certificates of achievement for publishing in top journals. His research interests include the fields of data mining, machine learning, and bioinformatics. He mainly focuses on developing intelligent algorithms to solve problems in domains, such as biology and healthcare. He received several scholarship awards, such as the College Recognition Award for excellence in scholarship, in 2007, 2012, and 2016, respectively, the Best Paper Award in leading conferences, such as ACM GECCO, in 2011, and the Chancellor's Annotation Award in Technology, in 2015.

**HARSH SINGH** received the bachelor's degree in electronics and communication engineering from IIIT Naya Raipur, India. He is currently a Research Assistant with the Big Data Analytics Centre (BIDAC), United Arab Emirates University (UAEU). He is also a self-motivated researcher with a strong background in mathematics, statistics, programming, and has both applied and research experience, including but not limited to, computer vision, machine learning, deep learning, and reinforcement learning. Prior to joining the Big Data Analytics Centre (BIDAC), he has worked as a Research Intern with the AI Research Group, University of St. Andrews. Before that, he has worked for several startups from India and abroad, working in computer vision and deep learning. His current research interests include developing machine learning and deep learning algorithms for solving medical problems.

**ELFADIL A. MOHAMED** received the Ph.D. degree in computer science from Universiti Teknologi Malaysia, Malaysia, in 2002. He is currently working as an Assistant Professor with the College of Engineering and Information Technology, Ajman University, United Arab Emirates. He published several scientific results in reputable journals and leading conferences. His research interests include data analysis, data mining, and databases.