World Wide Web (2023) 26:1667–1683
https://doi.org/10.1007/s11280-022-01110-6

GoMIC: Multi-view image clustering via self-supervised contrastive heterogeneous graph co-learning

Uno Fang[1] · Jianxin Li[1] · Naveed Akhtar[2] · Man Li[1] · Yan Jia[3]

Received: 5 August 2022 / Revised: 9 September 2022 / Accepted: 24 September 2022 /
Published online: 12 October 2022

Abstract
Graph learning is being increasingly applied to image clustering to reveal intra-class and inter-class relationships in data. However, existing graph learning-based image clustering focuses on grouping images under a single view, which under-utilises the information provided by the data. To address that, we propose a self-supervised multi-view image clustering technique under contrastive heterogeneous graph learning. Our method computes a heterogeneous affinity graph for multi-view image data. It conducts Local Feature Propagation (LFP) for reasoning over the local neighbourhood of each node and executes an Influence-aware Feature Propagation (IFP) from each node to its influential node for learning the clustering intention. The proposed framework pioneeringly employs two contrastive objectives. The first targets to contrast and fuse multiple views for the overall LFP embedding, and the second maximises the mutual information between LFP and IFP representations. We conduct extensive experiments on the benchmark datasets for the problem, i.e. COIL-20, Caltech7 and CASIA-WebFace. Our evaluation shows that our method outperforms the state-of-the-art methods, including the popular techniques MVGL, MCGC and HeCo.

Keywords Multi-view clustering · Contrastive graph learning · Feature propagation · Heterogeneous graph learning

✉ Jianxin Li
jianxin.li@deakin.edu.au

Uno Fang
uno.fang@deakin.edu.au

Naveed Akhtar
naveed.akhtar@uwa.edu.au

Man Li
amanda.li@deakin.edu.au

Yan Jia
jiayan2020@hit.edu.cn

[1] School of IT, Deakin University, 221 Burwood Highway, 3125 Burwood, VIC, Australia

[2] School of Physics, Mathematics and Computing, The University of Western Australia, 35 Stirling Highway, 6009 Crawley, WA, Australia

[3] Department of Computer Science and Technology, Harbin Institute of Technology, 518055 Shenzhen, Guangdong, China

1 Introduction

Clustering is typically thought of as a single view problem in computer vision, where an algorithm groups individual data samples based on their overall qualities. These samples, however, may be the outcome of various interpretations or representations of the underlying data. For instance, we can generate different sets of samples as Gabor [1], CLD [2] and HOG [3] descriptors of the images. These representations may hold complementary properties that can be leveraged for improved clustering. This fact has recently piqued interest of the computer vision community, resulting in an emerging topic of multi-view clustering (MVC) [4–12].

Another contemporary line of research for image clustering favors graph-based methods [13–17]. The main benefit of graphs for the clustering problem is that they naturally have the capacity to encode data structure information. For instance, methods like [13, 14, 18–22] leverage trained Graph Convolutional Networks (GCNs) for images to reason about the linkage likelihoods between a given node and its neighbours for graph completion, thereby achieving more accurate clusters.

In general, graph-based methods are known to benefit from Contrastive Learning (CL) [23], which induces models using self-supervision. During training, it maximises the agreement between its predictions and the transformed samples of the original sample. For graphs, the analogous Contrastive Graph Learning (CGL) paradigm aims to maximise the prediction agreement on different views of the same underlying graph [4–7, 24]. These views are created by applying random operations, e.g., adding/deleting nodes/edges and dropping features, to an original graph. In line with the negative sample creation in CL, the CGL considers other original graphs as the negative samples. It learns node-level (intra-view) or graph-level (inter-view) representations - illustrated in Fig. 1(a) - with a graph neural network and a contrastive loss function.

The self-supervised CGL paradigm naturally suites to the multi-view perspective. For instance, [25] and [26] created different graph views and then utilised node-level and graph-level representations for multi-view contrastive learning. These methods consider structural semantics as global information for learning the node-level embeddings, neglecting the fact that each node can also have various features to provide more information. Coming back to our main problem of multi-view image clustering, existing methods generally first compute a data affinity matrix for raw features or learned representation under multiple views, and then perform clustering using the affinity matrix [27–34]. These methods concatenate multiple views to construct a denoised homogeneous graph for image clustering. We provide a simple illustration of a multi-view homogeneous graph for image data in Fig. 1(b), where views are defined using compositional properties. The graph denoising operations, however, can lead to the loss of important semantic information. Additionally, the heterogeneous properties of multi-view data may become meaningless if several views are combined into a homogeneous graph. Theoretically, by treating images as nodes in heterogeneous graphs, it is possible to use more complementary information for multi-view image clustering - Fig. 1(c).

Considering the above narrative, in this work, we propose an inductive **M**ulti-view **I**mage **C**lustering framework with self-supervised contrastive heterogeneous **G**raph c**o**-learning (GoMIC). In GoMIC, we maintain the relationships between different views as a heterogeneous affinity graph, while preserving the uniqueness and independence of each view. Our heterogeneous graph consists of several homogeneous affinity graphs - Fig. 1(d). Each node can readily get the local neighbourhood data from each view by creating the

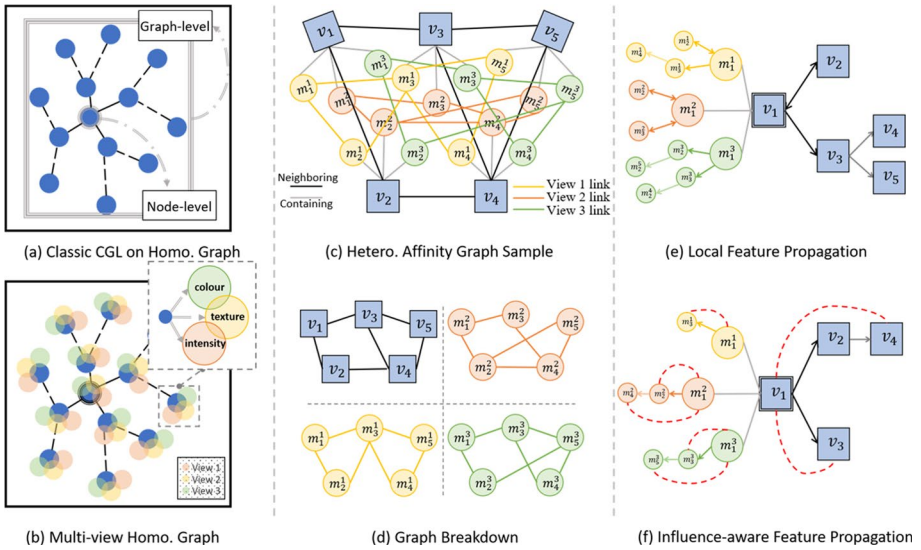

Fig. 1 Illustrations of concepts used in the text. (**a**) Classic Contrastive Graph Learning (CGL) - learns and contrasts homogeneous graphs at node and graph level. (**b**) Nodes in a homogeneous graph can have multiple views. (**c**) Relations among multiple views constitute a heterogeneous affinity graph. v_i indicates a random original node in the dataset, and m_i^j is the j-th view of v_i. Different colors of nodes indicate different views. (**d**) A heterogeneous affinity graph can be broken down into multiple affinity homogeneous graphs. (**e**) Our Local Feature Propagation (LFP) explores local neighbourhood relations. (**f**) Our Influence-aware Feature Propagation (IFP) explores relationships from a target node to the influential node in each view

heterogeneous affinity graph. To understand the propagation of node features, we created two encoding schemes. In the first, we propagate feature from a node to its neighbourhood in its own and other views through several hops - Fig. 1(e). The second strategy is influence-aware propagation that learns how each node feature propagates towards the densest nodes - Fig. 1(f). Both encoding strategies employ a proposed attention mechanism to control the feature influence of different nodes. Furthermore, to better exploit the learned embeddings under CGL, we explicitly contrast each pair of nodes for mutual information maximisation. We also customise the contrastive loss function to fit our contrastive objective.

Our key contributions are summarised below.

1. To the best of our knowledge, this is a pioneering multi-view image clustering method using heterogeneous information networks that leverages contrastive graph learning.
2. We devise two novel heterogeneous information network encoder strategies and an influence attention mechanism to learn the embedding of each node according to its local feature propagation and influence-aware feature propagation, respectively.
3. We enhance the loss function for contrastive graph learning to consider the mutual first neighbouring nodes as the mutual positive samples.
4. We conduct comprehensive experiments on three benchmark datasets, not only demonstrating large improvements over the existing self-supervised heterogeneous graph methods, but achieving better results than popular supervised methods across the datasets.

We discuss related work in Section 2. Section 3 demonstrates our proposed framework, GoMIC. In Section 4, we introduce three open multi-view image datasets and evaluate our proposed framework with comparing to 6 state-of-the-art MVC methods. Finally, Section 5 concludes this paper.

2 Prior art & background

We discuss the related work below. This discussion also includes analytical details that are later utilised in discussing the methodology.

2.1 Heterogeneous graph neural network

In recent years, heterogeneous graphs are becoming increasingly popular in neural network research [35, 36]. For instance, [37] studied the use of hierarchical attentions to depict node-level and semantic-level structures in heterogeneous graphs. Similarly, [38] incorporated intermediate nodes of meta-paths in the networks. [39] developed GTN to automatically identify useful graph connections. A technique dubbed MAHINDER is proposed in [40, 41] to employ and encode meta-paths over different views with attention on the importance of attributes and data views. In an unsupervised setting, a heterogeneous graph neural network is proposed in [42], which samples a fixed size of neighbours and fuses their features using LSTMs [43]. [44, 45] focused on network schema and preserved pairwise and network schema proximity simultaneously. [46] devised node- and edge-type dependent parameters to characterise heterogeneous attention over graph edges. The above methods rely strongly on supervised signals of data to encode graphs, whereas graph structures among the nodes are neglected. In [47], a heterogeneous network HeCo is proposed, which generates meta-paths and network schemas and exploits contrastive learning to further use signals of data in a self-supervised manner. [48] and [49] created item clusters and entity clusters to organise the objects and their nearby entities in the knowledge graph. After that, the hierarchically combining the heterogeneity data derived from the clusters with the weights produced by the hierarchical attention layers yields the representations. Nevertheless, encoding graphs and nodes while comprehensively considering node relations and graph structures still remains largely unresolved for the method.

2.2 Feature propagation

Considering that we devise feature propagation scheme in our technique, it is imperative to discuss related research in this direction in more detail. Expanding a node's feature by propagation is commonly conducted under a generalisation of pagerank equation [50, 51], which can be expressed as

$$\widetilde{\boldsymbol{X}} = \boldsymbol{X}\mathcal{W}_1 + A\widetilde{\boldsymbol{X}}\mathcal{W}_2, \tag{1}$$

where $\boldsymbol{X}$ contains the original features, A encodes the adjacency, $\mathcal{W}_1$ and $\mathcal{W}_2$ are coefficient matrices. However, (1) is not naturally invertible. Therefore, [52] modified it with the degree matrix D as follows

$$\widetilde{X} = X\mathcal{W}_1 + D^{-1}A\widetilde{X}\mathcal{W}_2. \tag{2}$$

The above is convergent if $\mathcal{W}_2$ is non-negative along other conditions. Still, this can only allow feature propagation on homogeneous graphs.

[53] attempted to extend feature propagation to heterogeneous graphs. In a heterogeneous graph, they assumed that there are two different sorts of nodes. With a threshold sparsifying the feature similarities, they first created a learnt feature similarity network for each type. Next, they generated the feature propagation graph for each type. Finally, the overall feature graph is obtained via channel attention [39]. Incidentally, there is a huge computational footprint of this method when dealing with a heterogeneous graph with large number of relations. More importantly, this method is more directed to heterogeneous graph sparsification rather than heterogeneous feature propagation.

2.3 Contrastive loss function

Contrastive Graph Learning (CGL) is derived from contrastive learning (CL) [23] for graph learning. CGL has been increasingly researched recently [4–7, 25] and has achieved excellent performance on graph or node classification by generating and contrasting positive and negative graph view pairs. Here, we organise our review by mainly focusing on contrastive loss function of the related CGL studies, which is helpful in understanding our contribution in Section 3.5. In [4, 7, 25, 54–56], the authors adopt the learning objective of CL rather straightforwardly. In doing so, they focused on node-level representations, and neglected the graph-level information. In [5, 6], for any node v_i, its embedding generated in one view v_i' and the embedding in the other view v_i'', form a positive pair, whereas embeddings of other nodes are negative samples. The pairwise objective for each positive pair (v_i', v_i'') is defined as

$$\ell\left(v_i', v_i''\right) = -\log \underbrace{\frac{\exp(\text{sim}\left(v_i', v_i''\right)/\tau)}{\exp(\text{sim}\left(v_i', v_i''\right)/\tau)_{\text{positive pair}} + \mathbb{E} + \mathbb{A}}}_{} \tag{3}$$

where sim denotes the function computing cosine similarity, τ is the temperature parameter, $\mathbb{E}$ identifies contrasting of inter-view negative pairs as $\mathbb{E} = \sum_{k=1}^{N} \mathbb{1}_{[k\neq i]} \exp(\text{sim}\left(v_i', v_k''\right)/\tau)$, and $\mathbb{A}$ denotes the contrasting of intra-view negative pairs as $\mathbb{A} = \sum_{k=1}^{N} \mathbb{1}_{[k\neq i]} \exp(\text{sim}\left(v_i', v_k'\right)/\tau)$, where $\mathbb{1}_{[k\neq i]} \in \{0, 1\}$ is an indicator function.

Connecting the above back to the heterogeneous graph neural networks, [47] proposed collaboratively contrastive optimization to expand the scope of defining positive samples and used it for self-supervised learning for heterogeneous graphs. However, in this state-of-the-art contrastive objective for heterogeneous graph learning, there is still a lack of consideration on feature propagation in graphs. Also, the frequent use of thresholds in the existing techniques decreases the feasibility of proposed models.

3 Proposed approach

In this section, we discuss the proposed multi-view image clustering with self-supervised contrastive heterogeneous graph co-learning (GoMIC), illustrated in Fig. 2. Our method encodes nodes from the local neighbourhood context and influence-oriented context, which

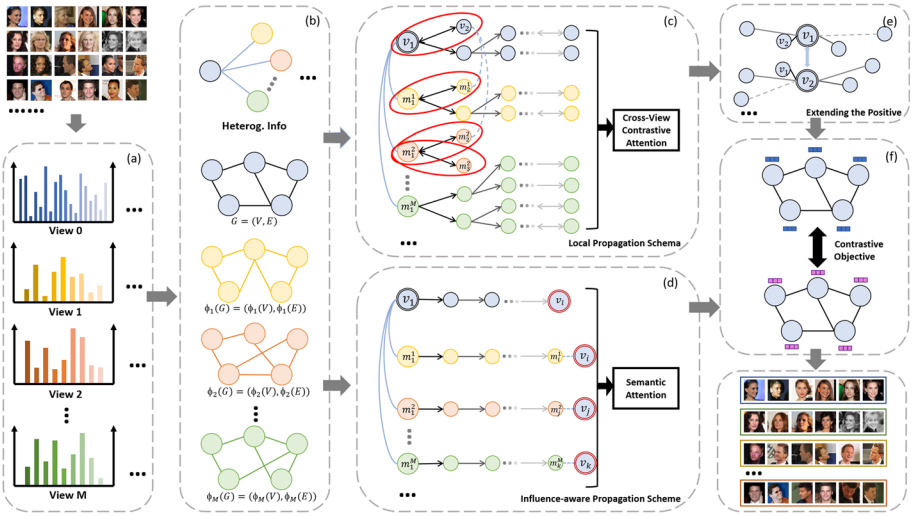

Fig. 2 Overview of the proposed GoMIC framework. For a single-view image dataset, we utilise M feature descriptors to generate M views. Then, the heterogeneous affinity graph $G(.)$ with $M+1$ views (including the original) is constructed, where $M+1$ homogeneous affinity graphs are connected with the relations between the original feature view and each descriptor-based view. Next, we propose two feature propagation encoders, i.e. Local Feature Propagation (LFP) and Influence-aware Feature Propagation (IFP), for generating contrasting representations with cross-view contrastive attention and semantic attention respectively. Before the final contrastive objective, we extend the definition of positive samples to include the first neighbour of each node, which are discovered with the help of LFP

fully captures contrastive structure of the heterogeneous affinity graph. This ensures that our approach well involves clustering boundary nodes (i.e., vague nodes) in the computations. During the encoding, we design innovative attention mechanisms, which learn feature propagation embeddings naturally. Moreover, in our method, a novel contrastive graph learning framework enables embeddings to supervise each other informatively.

3.1 Preliminaries

In order to describe our method for multi-view image clustering with contrastive heterogeneous graph learning, we first formalise the following pertinent notions for better understanding.

Definition 1 (Multi-view Image Data) Given an image dataset $\boldsymbol{V} = \{v_i\}_{i=1}^{N}$ and its feature set $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^{N}$, where N denotes the number of data samples. Each node v_i can be represented in multiple views $\{m_i^n\}_{n=1}^{M}$ with multi-view features $\{\boldsymbol{x}_i^n\}_{n=1}^{M}$, where M is the number of views.

Definition 2 (Heterogeneous Affinity Graph) A heterogeneous affinity graph $G = (V, E, \phi, \psi)$ is constructed from given and multi-view data. V and E are the node and the edge sets of the original $\boldsymbol{X}$, ϕ is the view-based node type mapping function, and ψ is the view-based edge type mapping function. We let $\phi_u(V)$ denote the node embedding of the u-th view, and $\phi_0(V) = V$, i.e., the original node features are the 0-th view. By analogy,

$\psi_0(E) = E$. Also, we let $\phi_u(v_i) = m_i^u$ and $\psi_u(e_i) = p_i^u$, where p_i^u indicates an edge e_i in the u-th view.

Definition 3 (Feature Propagation) Given the node feature set $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^N$ and the edge weight set $\boldsymbol{W} = \{w_{i,j}\}$, where $i, j \in [1, N]$ and $i \neq j$, the propagated feature of a node v_i is governed by

$$\widetilde{\boldsymbol{x}}_i = \mathcal{P}(\boldsymbol{x}_i, \{\widetilde{\boldsymbol{x}}_j\}, \{w_{i,j}\}; \theta_p), \tag{4}$$

where $\widetilde{\boldsymbol{x}}_j$ indicates the propagated feature set of v_i neighbours, $\{w_{i,j}\}$ is the edge weight set of edges between v_i and its neighbours, $\mathcal{P}$ is feature propagation function and θ_p is the propagation parameter.

Definition 4 (Local Propagation Network) The local propagation network $G'(v_i) = (V'(v_i), E'(v_i), \phi, \psi)$ of a node v_i is a directed graph, which consists of several local directed subgraphs. A local directed subgraph starts from v_i though l hops, each of which finds k_l nearest neighbours extracting the neighbourhood of v_i. Let v_a be in the $(h-1)$-th hop and v_b in the h-th hop. If v_a and v_b are the mutual first neighbours, this path stops walking at v_b, i.e., a path will stop walking when a node in it meets its mutual first neighbour in the next step, or when the path reaches the l-th hop. From the first to l-th hop, the feature propagation influence decreases.

Definition 5 (Influential Node) The target node v_i can walk l steps to find its influential node v_{i+l}, which identifies a node with maximum degrees (i.e., degree centrality) and/or maximum density (i.e., density centrality) in the view-based subgraph of v_i.

Definition 6 (Influence-aware Propagation Network) The influence-aware propagation network $G''(v_i) = (V''(v_i), E''(v_i), \phi, \psi)$ of a node v_i is a directed graph, which consists of several paths. A path starts from v_i to a influential node v_{i+l} through the shortest distance in the M-th view, which is in the form of $m_i^M \longrightarrow m_{i+1}^M \longrightarrow \cdots \longrightarrow m_{i+l}^M$. From m_i^M to m_{i+l}^M, the feature propagation influence decreases.

3.2 Heterogeneous affinity graph construction

We construct the heterogeneous affinity graph $G = (V, E, \phi, \psi)$ from multiple views of images based on feature similarity. An edge in our graph indicates the possibility of two nodes having the same label. The graph G consists of $M+1$ homogeneous affinity graphs, which are related by the connections between the original view and each descriptor view, i.e., there are $M+2$ edge types in G. For each node v_i and the original feature $\mathbf{x}_i$, its u-th view feature is $\phi_u(\mathbf{x}_i)$. According to these features, we adjust instance pivot subgraph (IPS) [13] to build the heterogeneous affinity graph G following the steps mentioned below.

Step 1: Feature extraction. In a single-view image dataset, given a node v_i, we utilise M different descriptors (e.g., Gabor [1], HOG [3]) to generate multiple views of v_i - Fig. 2(a). This results in $M+1$ views of v_i and different view-based feature vectors, including the original $\boldsymbol{x}_i$, where the n-th view-based feature vector of v_i is denoted as $\boldsymbol{x}_i^n$. We note that, for benchmark multi-view image datasets [57, 58], standard multiple views of images are already available.

Step 2: Neighbourhood construction. In each view, we utilise h-hop kNN to build its neighbourhood-based subgraph. Let k_t denote the k nearest neighbours at the t-th hop, where

$t = 1, 2, ..., h$. As t increases, the neighbourhood influence towards v_i decreases. Hence, the number of connecting nearest neighbours k_t decreases as well. We add graph edges and their weights along with the neighbourhood discovery. For instance, for an edge $e_{i,j}$ between v_i and v_j, their distance $d_{i,j}$ is computed using their sparse construction error $c_{i,j}$ as

$$d_{i,j} = \left\| \boldsymbol{x}_i - c_{i,j}\boldsymbol{x}_j \right\|_2^2. \tag{5}$$

Then, the weight between v_i and v_j, i.e. $w_{i,j}$ is defined as

$$w_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 1 - \left(s_{ij} + s_{ji}\right)/2 & \text{if } i \neq j \end{cases}, \tag{6}$$

where s_{ij} is the similarity score (i.e., Euclidean distance) between node v_i and v_j, and $s_{ij} = s_{ji}$. Thus, we get the homogeneous affinity graph of each view. To constitute these affinity graphs as a heterogeneous affinity graph $G = (V, E, \phi, \psi)$ - Fig. 2(b), we connect each node v_i with its corresponding other view-based nodes, where each edge weight is kept 1.

Step 3: Node density calculation. Based on the constructed heterogeneous affinity graph, we define the density for a node v_i in the graph as

$$\rho_{v_i} = \frac{1}{|\mathcal{N}_{k_1}(v_i)|} \sum_{v_j \in \mathcal{N}_{k_1}(v_i)} \tilde{\boldsymbol{x}}_i^{\mathsf{T}} \tilde{\boldsymbol{x}}_j, \tag{7}$$

where $\mathcal{N}_{k_1}(v_i)$ is the first hop k nearest neighbours of the node v_i, and $\tilde{\boldsymbol{x}}_i$, $\tilde{\boldsymbol{x}}_j$ are the ℓ_2-normalised feature embeddings of node v_i and v_j. According to this formula, the density of v_i is equal to the average of its similarity with its neighbours. Higher density nodes are considered more discriminative and are more influential in identifying cluster centres.

3.3 Local feature propagation encoder

The Local Feature Propagation (LFP) in our technique governs the interaction among the neighbourhood nodes. We aim to learn the propagated feature embedding of node v_i through its neighbourhood, i.e. to find its neighbours with similar features. Here, we conduct feature propagation on each view-based homogeneous affinity graph, then process them with cross-view contrastive learning - Fig. 2(c). General aggregation strategies like mean-pooling and max-pooling cannot identify if nodes are important, i.e., their mutual first neighbours cannot be emphasised. We employ the following two steps to obtain the embedding of each node v_i from the sight of local neighbourhood.

Step 1: Feature propagation. Based on (2) and discussion in Section 2.2, we incorporate influence of the first neighbours in feature propagation. To explain the concept, we take the n-th view-based node m_i^n as an example. Its feature propagated embedding, which considers feature information and structural information simultaneously by emphasising the importance of mutual first neighbours, is computed as

$$\begin{aligned} \widetilde{\boldsymbol{X}(m_i^n)} =& \mathcal{W}_1 \boldsymbol{X}(m_i^n) + \mathcal{W}_2 \sum_{m_j^n \in \mathcal{N}(m_i^n)} \frac{1}{d_i} \widetilde{\boldsymbol{X}(m_j^n)} \cdot \gamma_{[\sum_{k=1}^{\mathcal{N}}(v_i)=v_j]} \\ &+ \mathcal{W}_3 \sum_{m_j^n \in \mathcal{N}(m_i^n)} \boldsymbol{X}(m_j^n) \cdot \gamma_{[\sum_{k=1}^{\mathcal{N}}(v_i)=v_j]}, \end{aligned} \tag{8}$$

where $\gamma_{[\sum_{k=1}^{\mathcal{N}(v_i)=v_j}]}$ is the indicator function for the mutual first neighbour enhancement during feature propagation. It governs the weight of the first neighbour's influence towards the target node m_i^n. In (8) $d_i \in D$, where D is the degree matrix. Other notations follow from Section 2.2. For conciseness, the text below also avoids repeating explanation of other notations.

Step 2: Cross-view contrastive learning. After having the feature propagated embeddings of nodes in different M views, we feed them to a shared MLP which has a hidden layer to prepare them for the contrastive loss. In this case, we employ the conventional tactic of designating positive and negative samples. In other words, some view-based nodes that are formed from the same originating node form positive node pairs, whilst others form negative node pairs. However, our method aims at multi-view contrastive learning. Therefore, for the propagated n-th view-based node $\widetilde{m_i^n}$, we define the following contrastive loss for LFP

$$\ell(\widetilde{m_i^n}) = -\log \frac{\sum_{k=0}^{M} \exp(\mathrm{sim}\left(\widetilde{m_i^n}, \widetilde{m_i^k}\right)/\tau)}{\sum_{k=0}^{M} \sum_{v_b \in \mathcal{N}(v_i)} \mathbb{1}_{[i \neq b]} \exp(\mathrm{sim}\left(\widetilde{m_i^n}, \widetilde{m_b^k}\right)/\tau)}, \tag{9}$$

where $\mathcal{N}(v_i)$ indicates the nodes in the v_i-oriented subgraph, and $\mathbb{1}_{[i \neq b]} \in \{0, 1\}$ is an indicator function that equals to 1 if $i \neq b$. Then, the overall cost objective is given as

$$\mathcal{J} = \frac{1}{M \cdot |\mathcal{N}(v_i)|} \sum_{n=0}^{M} \ell(\widetilde{m_i^n}). \tag{10}$$

Through the cross-view contrastive objective, we optimise the encoder via back-propagation and learn the embeddings z_i^{LPF} of nodes for v_i.

3.4 Influence-aware feature propagation encoder

We also aim to learn the embedding of node v_i under influence-aware feature propagation (IFP) - Fig. 2(d). For the target node v_i, different views contributes differently to its embedding. The embedding can not only be affected by the nearest neighbours, but also by the influential nodes in its neighbourhood of various views. Thus, we devise the influence-aware feature propagation encoder at node-level and view-level to hierarchically aggregate underlying information through the shortest paths from the target node v_i towards the influential node v_{i+l} (excluding the edge between v_i and v_{i+l}) in different view-based subgraphs (see Def. 6). Take the n-th view as an example, we define a path from m_i^n to its influential node m_{i+1}^n as $p(m_i^n) = \{m_i^n, \cdots, m_{i+l}^n\}$ (see Def. 6).

Each path is processed via a GCN-based encoder to learn feature propagation. As shown Fig. 2(d), for each node (as a target node), its corresponding paths in various views will be the input. Having the path $p(m_i^n) = \{m_i^n, \cdots, m_{i+l}^n\}$, we define the affinity matrix $\boldsymbol{A}(m_i^n) \in \mathbb{R}^{|p(m_i^n)| \times |p(m_i^n)|}$ with the initial feature matrix denoted as $\boldsymbol{X}(m_i^n)$. In the k-th layer of GCN, we update the feature matrix as

$$\begin{aligned} \boldsymbol{X}^k\left(m_i^n\right) =& \sigma(\alpha \cdot \boldsymbol{X}^{k-1}\left(m_i^n\right) \\ &+(1-\alpha)\boldsymbol{D}^{-1}\left(m_i^n\right)\boldsymbol{A}\left(m_i^n\right)\boldsymbol{X}^{k-1}\left(m_i^n\right)\mathcal{W}^{k-1}), \end{aligned} \tag{11}$$

where $\boldsymbol{X}^{k-1}\left(m_i^n\right)$ denotes the updated features of the k-1-th GCN layer for all nodes on the path of $\left(m_i^n\right)$, $\boldsymbol{D}^{-1}\left(m_i^n\right)$ is the diagonal matrix with $\boldsymbol{D}_{i,i}\left(m_i^n\right) = \sum_j \boldsymbol{A}_{i,j}\left(m_i^n\right)$, σ is the ReLU activation, α is a learnable parameter that balances the importance of the updated features, and $\mathcal{W}^{k-1}$ is the transformation parameter. Through GCN, we can receive the embedding of m_i^n as h_i^n. Next, we employ attention mechanism in view-level to hierarchically aggregate context information from other views to the target node v_i. We firstly compute the importance of the n-th view as

$$\beta_n = \frac{\exp\left(\frac{1}{|V|}\sum_{v_i \in V} \boldsymbol{A}_{\mathrm{IFP}}^{\top} \cdot \tanh\left(\mathcal{W}_{\mathrm{IFP}} h_i^n + \mathcal{B}_{\mathrm{IFP}}\right)\right)}{\sum_{j=1}^{M} \exp\left(\frac{1}{|V|}\sum_{v_i \in V} \boldsymbol{A}_{\mathrm{IFP}}^{\top} \cdot \tanh\left(\mathcal{W}_{\mathrm{IFP}} h_i^j + \mathcal{B}_{\mathrm{IFP}}\right)\right)} \tag{12}$$

where $\boldsymbol{A}_{\mathrm{IFP}}^{\top}$ indicates the transpose of affinity graph $\boldsymbol{A}(m_i^n)$, $\tanh(\cdot)$ is the hyperbolic tangent function, $\mathcal{W}_{\mathrm{IFP}}$ are learnable parameters for IFP, and $\mathcal{B}_{\mathrm{IFP}}$ denotes view-level attention. Then, we compute the final embedding as follows

$$z_i^{\mathrm{IPF}} = \sum_{n=1}^{M} \beta_n \cdot h_i^n. \tag{13}$$

3.5 Dual-context contrastive graph learning

To maximise the mutual information between each pair of embeddings generated from LFP and IFP, we design a dual-context contrastive loss. To that end, we extend the definition of positive samples - Fig. 2(e). That is, for a node v_i, not only its LFP and IFP embeddings (i.e., z_i^{LFP} and z_i^{IFP}) are mutually positive, but also the embeddings of its mutual first neighbour v_j (i.e., z_j^{LFP} and z_j^{IFP}) would be considered as its positive samples. We denote the positive sample set of v_i as $\mathbb{P}_i$. This aims to contribute to nailing down clustering centres. According to the extension of positive sample definition, we formulate the dual-context contrastive loss function as

$$\mathfrak{L}(v_i) = \mathfrak{L}(z_i^{\mathrm{LFP}}) + \mathfrak{L}(z_i^{\mathrm{IFP}}). \tag{14}$$

We describe the computation of $\mathfrak{L}(z_i^{\mathrm{LFP}})$ below. The $\mathfrak{L}(z_i^{\mathrm{IFP}})$ is computed analogously. We let

$$\mathfrak{L}(z_i^{\mathrm{LFP}}) = -\log \frac{\sum_{z_j \in \mathbb{P}_i} \exp(\mathrm{sim}\left(z_i^{\mathrm{LFP}}, z_j\right)/\tau)}{\mathrm{Neg.}}. \tag{15}$$

In (15), Neg. refers to contrasting against negative samples, which is defined as

$$\begin{aligned} \mathrm{Neg.} = &\sum_{k=1}^{N} \mathbb{1}_{[i \neq k]} \mathbb{1}_{[z_k^{\mathrm{LFP}} \notin \mathbb{P}_i]} \exp(\mathrm{sim}\left(z_i^{\mathrm{LFP}}, z_k^{\mathrm{LFP}}\right)/\tau) \\ &+ \sum_{a=1}^{N} \mathbb{1}_{[i \neq a]} \mathbb{1}_{[z_a^{\mathrm{LFP}} \notin \mathbb{P}_i]} \exp(\mathrm{sim}\left(z_i^{\mathrm{LFP}}, z_a^{\mathrm{IFP}}\right)/\tau), \end{aligned} \tag{16}$$

where z_a^{IFP} represents the IFP embedding of other nodes for contrasting, and two indicators separate out the negative samples. The overall contrastive objective of maximising mutual information is then

$$\mathcal{J} = \frac{1}{N} \sum_{i=1}^{N} \left[\theta \cdot \mathfrak{A}(z_i^{\text{LFP}}) + (1 - \theta) \cdot \mathfrak{A}(z_i^{\text{IFP}}) \right], \tag{17}$$

where hyper-parameter θ controls the relative importance of the two embeddings.

4 Evaluation

4.1 Experimental Setup

Datasets: To establish the effectiveness of our technique, we perform bechmarking of our method on popular standard multi-view image datasets.

1. **COIL-20** [57] has 1,440 gray-scale images of 20 classes and each class contains 72 images. Each image has been extracted under 3 views, where the first is a 1024-dimensional intensity feature, the second is a 3304-dimensional LBP feature, and the third is a 6750-dimensional Gabor feature.
2. **Caltech7** [58] contains 1,474 images of 7 classes. Each image has 6 views of features extracted. These views include 48-dimensional Gabor feature, 40-dimensional WM feature, 254-dimensional CENTRIST feature, 1984 dimensional HOG feature, 512-dimensional GIST feature, and 928-dimensional LBP feature.
3. **CASIA-WebFace (CWF)** [59] is cleaned to contain 466,169 single-view face images of 10,575 real identities collected from the web. To obtain a relatively large-scale dataset while constraining the image numbers for the available hardware, we select 18 classes of 10,791 images to conduct our experiments. To generate multi-view data, we utilise the 512-dimensional feature vectors extracted by CNN architecture to obtain 3 views (512-dimensional color histogram, 26-dimensional LBP and 512-dimensional Gabor) from each image.

Baseline Methods: For benchmarking, we compare with the following six existing techniques that use a variety of approaches for multi-view image clustering.

1. **MIC** [60] adopts the same strategy used by best single view (BSV) [61] to fill missing instances and then learns a non-negative low-dimensional consensus representation for all views. K-means is applied to the learned consensus representation for final clustering.
2. **DCCAE** [62] extends the method of combining the deep encoder with Canonical Correlation Analysis by introducing a deep decoder, which has deep canonically correlated auto-encoders coordinating and extracting graph representations in a pairwise manner.
3. **MVGL** [27] learns individual graphs and then integrate the multiple learned graphs into a global graph with exactly k components.
4. **MCGC** [28] learns a consensus graph by minimising the disagreement between different views and constraining the rank of the Laplacian matrix.
5. **RHLC-CAGL** [31] automatically captures a latent common Laplacian that is shared by all views.

6. **HeCo** [47] is a self-supervised heterogeneous GNN method, which leverages meta-path to extract two views for contrastive learning.

Implementation Details: In GoMIC, for the single-view image dataset (i.e., CASIA-WebFace), we utilise commonly used feature descriptors, i.e., intensity, LBP and Gabor, to generate multiple views of each image. There are 2 hyperparameters for heterogeneous affinity graph construction in our method, namely, the number of hops $\mathbb{h}$, and the number of each node's nearest neighbours in each hop $\{k_i\}, i = 1, 2, ..., \mathbb{h}$. $\mathbb{h}$ and $\{k_i\}$ are defined according to the used dataset. We set $\mathbb{h} = 4, \{k_i\} = \{10, 5, 1, 1\}$ for COIL-20, $\mathbb{h} = 4, \{k_i\} = \{20, 5, 1, 1\}$ for Caltech7, and $\mathbb{h} = 4, \{k_i\} = \{50, 3, 1, 1\}$ for CASIA-WebFace. Also, while setting up the GCN-based contrastive model, we optimise the learning rate in the range $[1e-4, 5e-3]$. For the dropout function in encoding, we used the range [0.1, 0.5] with a step size 0.05, and τ is tuned in the range [0.5, 0.9] with a step size 0.05, and α and θ are both tuned in the range [0.1, 0.9] with a step size 0.1. Moreover, encoders only conduct aggregation once, i.e., we use 2-layer GCN on LFP and IFP. At the end of GoMIC, the images to be clustered are represented as a graph. Each edge in the graph is associated with a similarity weight in [0, 1]. To generate the clusters, we visit every node and only preserve its neighborhood nodes with the largest weight, i.e., the other neighbor nodes are disconnected from the node. Thus, the clusters get formed in an efficient manner.

4.2 Comparison with the existing state-of-the-art

Table 1 reports the results of several popular multi-view clustering methods and a state-of-the-art self-supervised heterogeneous contrastive graph learning technique, HeCo. The results reveal that multi-view clustering methods which make use of more views usually achieve higher performance. This explains why MIC [60] and DCCAE [62] generally have inferior clustering results. Since MIC relies on the best single view to conduct representation learning, and DCCAE correlates view-based graphs for embeddings pair-by-pair, they are not able to perform as well as other methods. In contrast, methods proposed more recently (MVGL [28], MCGC [28] and RHLC-CAGL [31]) aim to leverage

Table 1 Comparison of clustering performance on three datasets. NMI, PUR and ACC respectively stand for normalised mutual information, purity and accuracy. Self-supervised graph learning-based methods are green highlighted. The best results in each column are bold faced

Dataset	COIL-20 [57]			Caltech7 [58]			CWF [59]		
Method	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR	ACC
MIC [60]	72.42	80.49	58.64	37.99	81.29	58.64	31.84	29.44	24.37
DCCAE [62]	70.58	78.00	55.51	59.14	-	41.89	33.61	36.12	30.59
MVGL [27]	89.20	85.84	66.08	58.81	84.47	62.95	37.21	46.26	32.28
MCGC [28]	83.78	70.84	88.21	59.47	85.06	64.51	47.78	48.13	33.68
RHLC-CAGL [31]	79.81	-	72.10	**79.60**	-	69.30	52.34	50.41	37.46
HeCo [47]	69.11	63.50	56.31	55.32	73.76	51.75	32.45	37.22	21.03
GoMIC (Ours)	**89.83**	**87.35**	**91.12**	79.49	**87.10**	**76.52**	**58.72**	**67.55**	**48.06**

more information from multiple views, which achieves better performance. In the table, HeCo [47] deals with natural heterogeneous networks instead of directly dealing with multi-view images. Nevertheless, it performs reasonably well on this benchmark due to the suitability of heterogeneous graphs to the problem. This is inline with our intuition of exploiting heterogeneous properties of image views for clustering. It is therefore not surprising that the performance of our method, GoMIC, outperforms these powerful baselines. Our approach not only contrasts numerous views but also makes use of two recently developed feature propagation encoding schemes for enhanced contrastive learning.

4.3 Ablation study and parameter analysis

GoMIC encodes multi-view based graphs with two innovative encoding schemes, namely, LFP and IFP. Also, for better clustering centres, we adjust the contrastive loss function by extending the positive sample definition. To understand the impact of these factors in the overall performance of our technique, we introduce three variants of GoMIC to conduct an ablation study. These three variants are respectively denoted as: (1) GoLFP, which contains only LFP as the encoder; (2) GoIFP, which only contains IFP as the encoder and (3) GoMIC-n/e, which has no extension of the positive sample definition. The results of these variants on all three datasets are summarised in Table 2. In the table, we can observe that GoMIC eventually outperforms all these variants by a considerable margin, establishing the benefits of synergising these proposed components. Furthermore, the performances of GoLFP and GoIFP decrease differently when applied to different datasets. This highlights that, for different cases, LFP and IFP are able to contribute differently to the overall performance. A consistent gain of GoMIC over GoMIC-n/e also ascertains the importance of our positive sample definition extension. From the parameters viewpoint, GoMIC has two major hyper-parameters, α and θ. We show the influence of adjusting their values on performance in Fig. 3. The chosen range values are $\{0.1, 0.3, 0.6, 0.9\}$.

5 Conclusions

In order to understand and exploit relationships within image datasets from each node's local neighbourhood and influence-aware context, we introduced an innovative multi-view image clustering method called GoMIC. GoMIC takes advantage of the heterogeneous properties of multi-view image data under contrastive graph learning. To

Table 2 Ablation study results for GoMIC. GoLFP only uses LFP encoder. GoIFP uses only the IFP encoder and GoMIC-n/e does not use the extended definition of positive samples, but uses both LFP and IFP. Difference of these variants to GoMIC highlights the contribution of the components

Dataset	COIL-20 [57]			Caltech7 [58]			CWF [59]		
Method	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR	ACC
GoLFP	79.10	56.10	48.16	63.99	71.29	58.64	29.83	43.14	19.82
GoIFP	40.19	74.10	25.51	32.14	79.60	21.89	11.21	31.29	13.78
GoMIC-n/e	81.25	83.46	85.60	76.34	81.79	68.63	45.80	52.93	43.67
GoMIC	**89.83**	**87.35**	**91.12**	**79.49**	**87.10**	**76.52**	**58.72**	**67.55**	**48.06**

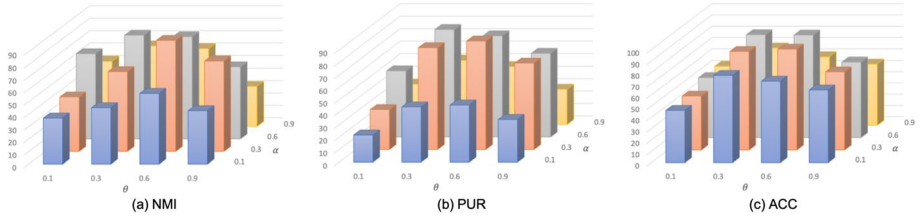

Fig. 3 Parameter analysis on Caltech7 [58]. The best performance is achieved when both α and θ are chosen in the range $\{0.3, 0.6\}$

extract and exploit more underlying information, we devised two strategies to encode the graphs, Local Feature Propagation (LFP) and Influence-aware Feature Propagation (IFP), to represent each node-based subgraph in two contrasting contexts. Also, we employed two contrastive loss functions, and adjusted them to fit the use of LFP and IFP. The first loss function aims to integrate multiple view-based LFP embeddings, and the second nails down the clustering centres with an extended positive sample definition for contrastive graph learning. Experimental results show that our proposed method consistently outperforms the state-of-the-art methods on multi-view clustering benchmarks. Also, our ablation study demonstrates explicit contribution of each novel aspect in our overall technique. Currently, the framework works with the common assumption of balanced data. In the future, we will extend it to also handle imbalanced data.

Author Contributions Uno Fang: Conceptualisation of this study, Methodology, Writing - Original Draft, Software. Jianxin Li: Project administration,Supervision, Writing - Review & Editing. Naveed Akhtar: Writing - Review & Editing, Validation. Man Li: Writing - Review & Editing. Yan Jia: Writing - Review & Editing. All authors reviewed the manuscript.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions

Authors' information Uno Fang is a Ph.D. candidate, and Jianxin Li is an Associate Professor at School of IT, Deakin University. Naveed Akhtar a Senior Lecturer of Machine Learning, AI and Data Science at the Department of Computer Science & Software Engineering at the University of Western Australia. Man Li is pursuing her the Ph.D. degree with School of IT, Deakin University. Yan Jia is a Professor at the Department of Computer Science and Technology, Harbin Institute of Technology.

Data Availability The data that support the findings of this study are openly available in **COIL-20** [57] at https://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php, **Caltech7** [58] at https://data.caltech.edu/, and **CASIA-WebFace** [59] at https://mldta.com/dataset/casia-webface/.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest. The authors have no relevant financial or non-financial interests to disclose. The authors have no conflicts of interest to declare that are relevant to the content of this article. All authors certify that they have no affiliations with or involvement in any organisation or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript. The authors have no financial or proprietary interests in any material discussed in this article.

Consent for publication The authors confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. The

authors further confirm that the order of authors listed in the manuscript has been approved. The authors confirm that they have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing the authors confirm that they have followed the regulations of our institutions concerning intellectual property. The authors are consent for publication.

References

1. Lades, M., Vorbruggen, J.C., Buhmann, J., Lange, J., Von Der Malsburg, C., Wurtz, R.P., et al.: Distortion invariant object recognition in the dynamic link architecture. IEEE Trans. Comput. **42**(3), 300–311 (1993)
2. Manjunath, B.S., Ohm, J.R., Vasudevan, V.V., Yamada, A.: Color and texture descriptors. IEEE Trans. Circuits Syst. Video Technol. **11**(6), 703–715 (2001)
3. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). vol. 1. pp. 886–893. Ieee (2005)
4. Wan, S., Pan, S., Yang, J., Gong, C.: Contrastive and generative graph convolutional networks for graph-based semi-supervised learning. Proceedings of the AAAI Conference on Artificial Intelligence **35**(11), 10049–10057 (2021)
5. Zeng, J., Xie, P.: Contrastive self-supervised learning for graph classification. Proceedings of the AAAI Conference on Artificial Intelligence **35**(12), 10824–10832 (2021)
6. Zhu, Y., Xu, Y., Yu, F., Liu, Q., Wu, S., Wang, L.: Graph contrastive learning with adaptive augmentation. In: Proceedings of the Web Conference vol. 2021. p. 2069–2080 (2021)
7. Hafidi, H., Ghogho, M., Ciblat, P., Swami, A.: Negative sampling strategies for contrastive self-supervised learning of graph representations. Signal Process. **190**, 108310 (2022)
8. Sharma, K.K., Seal, A.: Outlier-robust multi-view clustering for uncertain data. Knowl. Based Syst. **211**, 106567 (2021)
9. Bansal, M., Sharma, D.: A novel multi-view clustering approach via proximity-based factorization targeting structural maintenance and sparsity challenges for text and image categorization. Inf. Process. Manag. **58**(4), 102546 (2021)
10. Ueda, I., Shishido, H., Kitahara, I.: Spatio-temporal aggregation of skeletal motion features for human motion prediction. Array **15**, 100212 (2022). https://doi.org/10.1016/j.array.2022.100212
11. Ntelemis, F., Jin, Y., Thomas, S.A.: Information maximization clustering via multi-view self-labelling. Knowledge-Based Systems 109042 (2022)
12. Yuan, C., Zhu, Y., Zhong, Z., Zheng, W., Zhu, X.: Robust Self-Tuning Multi-View Clustering. World Wide Web **25**(2), 489–512 (2022). https://doi.org/10.1007/s11280-021-00945-9
13. Wang, Z., Zheng, L., Li, Y., Wang, S.: Linkage based face clustering via graph convolution network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition p. 1117–1125 (2019)
14. Yang, L., Zhan, X., Chen, D., Yan, J., Loy, C.C., Lin, D.: Learning to cluster faces on an affinity graph. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. p. 2298–2306 (2019)
15. Li, S., Liu, B., Chen, D., Chu, Q., Yuan, L., Yu, N.: Density-aware graph for deep semi-supervised visual recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. p. 13400–13409 (2020)
16. Yang, L., Chen, D., Zhan, X., Zhao, R., Loy, C.C., Lin, D.: Learning to cluster faces via confidence and connectivity estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. p. 13369–13378 (2020)

17. Yin, H., Song, X., Yang, S., Li, J.: Sentiment analysis and topic modeling for COVID-19 vaccine discussions. World Wide Web **25**(05), 1–17 (2022). https://doi.org/10.1007/s11280-022-01029-y
18. Yang, Y., Guan, Z., Li, J., Zhao, W., Cui, J., Wang, Q.: Interpretable and Efficient Heterogeneous Graph Convolutional Network. IEEE Transactions on Knowledge and Data Engineering p. 1–1 (2021). https://doi.org/10.1109/TKDE.2021.3101356.
19. Zhang, M., Wang, G., Ren, L., Li, J., Deng, K., Zhang, B.: METoNR: A meta explanation triplet oriented news recommendation model. Knowledge-Based Systems **238**, 107922 (2022). https://doi.org/10.1016/j.knosys.2021.107922
20. Mitra, S., Banerjee, S., Naskar, M.K.: Remodelling correlation: A fault resilient technique of correlation sensitive stochastic designs. Array **15**, 100219 (2022). https://doi.org/10.1016/j.array.2022.100219
21. Myllyaho, L., Nurminen, J.K., Mikkonen, T.: Node co-activations as a means of error detection-Towards fault-tolerant neural networks. Array **15**, 100201 (2022). https://doi.org/10.1016/j.array.2022.100201
22. Song, X., Li, J., Lei, Q., Zhao, W., Chen, Y., Mian, A.: Bi-CLKT: Bi-Graph Contrastive Learning Based Knowledge Tracing. Know-Based Syst **241**(C) (2022). https://doi.org/10.1016/j.knosys.2022.108274.
23. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. p. 1597–1607. PMLR (2020)
24. Liao, J., Zhao, X., Li, X., Tang, J., Ge, B.: Contrastive Heterogeneous Graphs Learning for Multi-Hop Machine Reading Comprehension. World Wide Web **25**(3), 1469–1487 (2022). https://doi.org/10.1007/s11280-021-00980-6
25. Hassani, K., Khasahmadi, A.H.: Contrastive multi-view representation learning on graphs. In: International Conference on Machine Learning. p. 4116–4126. PMLR (2020)
26. Wang Y, Min Y, Chen X, Wu J. Multi-view Graph Contrastive Representation Learning for Drug-Drug Interaction Prediction. In: WWW '21: The Web Conference 2021. ACM / IW3C2. p. 2921–2933 (2021)
27. Zhan, K., Zhang, C., Guan, J., Wang, J.: Graph learning for multiview clustering. IEEE Trans. Cybern. **48**(10), 2887–2895 (2017)
28. Zhan, K., Nie, F., Wang, J., Yang, Y.: Multiview consensus graph clustering. IEEE Trans. Image Process. **28**(3), 1261–1270 (2018)
29. Xie, Y., Tao, D., Zhang, W., Liu, Y., Zhang, L., Qu, Y.: On unifying multi-view self-representations for clustering by tensor multi-rank minimization. Int. J. Comput. Vis. **126**(11), 1157–1179 (2018)
30. Chen, Y., Wang, S., Peng, C., Hua, Z., Zhou, Y.: Generalized Nonconvex Low-Rank Tensor Approximation for Multi-View Subspace Clustering. IEEE Trans. Image Process. **30**, 4022–4035 (2021)
31. Jing, P., Su, Y., Li, Z., Nie, L.: Learning robust affinity graph representation for multi-view clustering. Inf. Sci. **544**, 155–167 (2021)
32. Liu, J., Teng, S., Fei, L., Zhang, W., Fang, X., Zhang, Z., et al.: A novel consensus learning approach to incomplete multi-view clustering. Pattern Recognit. **115**, 107890 (2021)
33. Shi, S., Nie, F., Wang, R., Li, X.: Multi-View Clustering via Nonnegative and Orthogonal Graph Reconstruction. IEEE Transactions on Neural Networks and Learning Systems (2021)
34. Xia, W., Wang, Q., Gao, Q., Zhang, X., Gao, X.: Self-supervised Graph Convolutional Network for Multi-view Clustering. IEEE Transactions on Multimedia (2021)
35. Haldar, N., Li, J., Ali, M., Cai, T., Chen, Y., Sellis, T., et al.: Top-k Socio-Spatial Co-engaged Location Selection for Social Users. IEEE Transactions on Knowledge and Data Engineering. Publisher Copyright: IEEE (2022). https://doi.org/10.1109/TKDE.2022.3151095.
36. Xue, G., Zhong, M., Li, J., Chen, J., Zhai, C., Kong, R.: Dynamic network embedding survey. Neurocomputing **472**, 212–223 (2022). https://doi.org/10.1016/j.neucom.2021.03.138
37. Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., et al.: Heterogeneous graph attention network. In: The World Wide Web Conference. p. 2022–2032 (2019)
38. Fu, X., Zhang, J., Meng, Z., King, I.: Magnn: Metapath aggregated graph neural network for heterogeneous graph embedding. In: Proceedings of The Web Conference vol. 2020. p. 2331–2341 (2020)
39. Yun, S., Jeong, M., Kim, R., Kang, J., Kim, H.J.: Graph transformer networks. Adv. Neural Inf. Process. Syst. **32**, 11983–11993 (2019)
40. Zhong, Q., Liu, Y., Ao, X., Hu, B., Feng, J., Tang, J., et al.: Financial defaulter detection on online credit payment via multi-view attributed heterogeneous information network. In: Proceedings of The Web Conference **2020** p. 785–795 (2020)
41. Zheng, S., Guan, D., Yuan, W.: Semantic-aware heterogeneous information network embedding with incompatible meta-paths. World Wide Web **25**(1), 1–21 (2022)

42. Zhang, C., Song, D., Huang, C., Swami, A., Chawla, N.V.: Heterogeneous graph neural network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. p. 793–803 (2019)
43. Dong, Y., Fu, Y., Wang, L., Chen, Y., Dong, Y., Li, J.: A Sentiment Analysis Method of Capsule Network Based on BiLSTM. IEEE Access **02**PP, 1–1 (2020). https://doi.org/10.1109/ACCESS.2020.2973711.
44. Kong, X., Xia, F., Li, J., Hou, M., Li, M., Xiang, Y.: A shared bus profiling scheme for smart cities based on heterogeneous mobile crowdsourced data. IEEE Transactions on Industrial Informatics **10**PP, 1–1 (2019). https://doi.org/10.1109/TII.2019.2947063.
45. Zhao, J., Wang, X., Shi, C., Liu, Z., Ye, Y.: Network schema preserved heterogeneous information network embedding. In: 29th International Joint Conference on Artificial Intelligence (IJCAI) (2020)
46. Hu, Z., Dong, Y., Wang, K., Sun, Y.: Heterogeneous graph transformer. In: Proceedings of The Web Conference vol. 2020. p. 2704–2710 (2020)
47. Wang, X., Liu, N., Han, H., Shi, C.: Self-supervised heterogeneous graph neural network with co-contrastive learning. In: KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. p. 1726–1736. ACM (2021)
48. Wang, J., Shi, Y., Li, D., Zhang, K., Chen, Z., Li, H.: McHa: a multistage clustering-based hierarchical attention model for knowledge graph-aware recommendation. World Wide Web. **25**(3), 1103–1127 (2022)
49. Cai, W., Wang, Y., Mao, S., Zhan, J., Jiang, Y.: Multi-heterogeneous neighborhood-aware for Knowledge Graphs alignment. Inf. Process. Manag. **59**(1), 102790 (2022)
50. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: Bringing order to the web. Stanford InfoLab (1999)
51. Xiang, B., Liu, Q., Chen, E., Xiong, H., Zheng, Y., Yang, Y.: Pagerank with priors: An influence propagation perspective. In: Twenty-Third International Joint Conference on Artificial Intelligence (2013)
52. Dornaika, F.: On the use of high-order feature propagation in Graph Convolution Networks with Manifold Regularization. Inf. Sci. **584**, 467–478 (2022)
53. Zhao, J., Wang, X., Shi, C., Hu, B., Song, G., Ye, Y.: Heterogeneous graph structure learning for graph neural networks. In: 35th AAAI Conference on Artificial Intelligence (AAAI) (2021)
54. Wang, R., Li, L., Tao, X., Dong, X., Wang, P., Liu, P.: Trio-based collaborative multi-view graph clustering with multiple constraints. Inf. Process. Manag. **58**(3), 102466 (2021)
55. Li, J., Zeng, H., Peng, L., Zhu, J., Liu, Z.: Learning to rank method combining multi-head self-attention with conditional generative adversarial nets. Array **15**, 100205 (2022). https://doi.org/10.1016/j.array.2022.100205
56. Zhong, G., Shu, T., Huang, G., Yan, X.: Multi-view spectral clustering by simultaneous consensus graph learning and discretization. Knowledge-Based Systems **235**, 107632 (2022)
57. Nayar, M.H.: Columbia Object Image Library: COIL-100. Department of Computer Science, Columbia University. CUCS-006-96 (1996)
58. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In: 2004 conference on computer vision and pattern recognition workshop. p. 178–178. IEEE (2004)
59. Banerjee, S., Scheirer, W., Bowyer, K., Flynn, P.: On hallucinating context and background pixels from a face mask using multi-scale gans. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. p. 300–309 (2020)
60. Shao, W., He, L., Philip, S.Y.: Multiple incomplete views clustering via weighted nonnegative matrix factorization with $l_{2,1}$ regularization. In: Joint European conference on machine learning and knowledge discovery in databases. p. 318–334. Springer (2015)
61. Xu, W., Liu, X., Gong, Y.: Document clustering based on non-negative matrix factorization. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval. p. 267–273 (2003)
62. Wang, W., Arora, R., Livescu, K., Bilmes, J.: On deep multi-view representation learning. In: International conference on machine learning. p. 1083–1092. PMLR (2015)