A Comprehensive Survey on Deep Clustering: Taxonomy, Challenges, and Future Directions

SHENG ZHOU, Zhejiang University, Hangzhou, China
HONGJIA XU, Zhejiang University, Hangzhou, China
ZHUONAN ZHENG, Zhejiang University, Hangzhou, China
JIAWEI CHEN, Zhejiang University, Hangzhou, China
ZHAO LI, Zhejiang University, Hangzhou, China
JIAJUN BU, Zhejiang University, Hangzhou, China
JIA WU, Macquarie University, Sydney, Australia
XIN WANG, Tsinghua University, Beijing, China
WENWU ZHU, Tsinghua University, Beijing, China
MARTIN ESTER, Simon Fraser University, Burnaby, Canada

Clustering is a fundamental machine learning task, which aim at assigning instances into groups so that similar samples belong to the same cluster while dissimilar samples belong to different clusters. Shallow clustering methods usually assume that data are collected and expressed as feature vectors within which clustering is performed. However, clustering high-dimensional data, such as images, texts, videos, and graphs, poses significant challenges for clustering tasks, such as indiscriminate representation and intricate relationships among instances. Over the past decades, deep learning has achieved remarkable success in effective representation learning and modeling complex relationships. Motivated by these advancements, **Deep Clustering** seeks to improve clustering outcomes through deep learning techniques, garnering considerable interest from both academia and industry. Despite many contributions to this vibrant area of research, the lack of systematic analysis and a comprehensive taxonomy has hindered progress in this field. In this survey, we first explore how deep learning can be integrated into deep clustering and identify two fundamental components: the representation learning module and the clustering module. Then, we summarize and analyze the representative design of these two modules. Furthermore, we introduce a novel

This work is supported in part by the National Natural Science Foundation of China (Grant No.62106221, 62372408), Zhejiang Provincial Natural Science Foundation of China (Grant No. LTGG23F030005), and Ningbo Natural Science Foundation (Grant No. 2022J183).
Authors' Contact Information: Sheng Zhou, School of Software Technology, Zhejiang University, Zhejiang, China; email: zhousheng_zju@zju.edu.cn; Hongjia Xu, College of Computer Science and Technology, Zhejiang University, Zhejiang, China; email: xu_hj@zju.edu.cn; Zhuonan Zheng, College of Computer Science and Technology, Zhejiang University, Zhejiang, China; email: zhengzn@zju.edu.cn; Jiawei Chen, College of Computer Science and Technology, Zhejiang University, Zhejiang, China; email: sleepyhunt@zju.edu.cn; Zhao Li, Zhejiang Lab, Zhejiang University, Zhejiang, China; email: lzjoey@gmail.com; Jiajun Bu (Corresponding author), Zhejiang Key Laboratory of Accessible Perception and Intelligent Systems, College of Computer Science, Zhejiang University, Zhejiang, China; email: bjj@zju.edu.cn; Jia Wu, School of Computing, Macquarie University, New South Wales, Australia; email: jia.wu@mq.edu.au; Xin Wang, Department of Computer Science and Technology, Tsinghua University, Beijing, China; email: xin_wang@tsinghua.edu.cn; Wenwu Zhu, Department of Computer Science and Technology, Tsinghua University, Beijing, China; email: wwzhu@tsinghua.edu.cn; Martin Ester, School of Computing Science, Simon Fraser University, British Columbia, Canada; email: ester@cs.sfu.ca.

ACM 0360-0300/2024/11-ART69
https://doi.org/10.1145/3689036

taxonomy of deep clustering based on how these two modules interact, specifically through multistage, generative, iterative, and simultaneous approaches. In addition, we present well-known benchmark datasets, evaluation metrics, and open-source tools to clearly demonstrate different experimental approaches. Finally, we examine the practical applications of deep clustering and propose challenging areas for future research.

CCS Concepts: • **Theory of computation → Unsupervised learning and clustering;**

Additional Key Words and Phrases: Deep learning, clustering, representation learning

ACM Reference Format:
Sheng Zhou, Hongjia Xu, Zhuonan Zheng, Jiawei Chen, Zhao Li, Jiajun Bu, Jia Wu, Xin Wang, Wenwu Zhu, and Martin Ester. 2024. A Comprehensive Survey on Deep Clustering: Taxonomy, Challenges, and Future Directions. *ACM Comput. Surv.* 57, 3, Article 69 (November 2024), 38 pages. https://doi.org/10.1145/3689036

1 Introduction

Clustering is a fundamental technique in machine learning, often used as a crucial pre-processing step in various data mining tasks. The primary purpose of clustering is to assign the instances into groups so that similar samples belong to the same cluster while dissimilar samples belong to different clusters. The clusters of samples provide a global characterization of data instances, which can significantly benefit further analysis of the whole dataset, such as anomaly detection [1, 2], domain adaptation [3, 4], community detection [5, 6], and discriminative representation learning [7–9], and so on.

Early clustering works typically assume that data is represented in the form of feature vectors and primarily focus on discovering clustering structures within the vector spaces, which is also known as **shallow clustering**. With the rapid development of the Internet and Web services, the research community is showing increasing interest in clustering more complex data such as text, images, videos, and graphs. However, the raw high-dimensional features of these data pose significant challenges for shallow clustering due to the "curse of dimensionality" [10]. Meanwhile, the past decades have also witnessed the success of deep learning in learning representation for high-dimensional data, especially unsupervised ones. Recently, **Deep Clustering**, which aims to enhance the clustering task through deep learning, has attracted significant interest from both the research community and the industry.

To accomplish this objective, a simplistic approach might involve using deep unsupervised representation learning to learn representations for each data instance, and subsequently input these representations into shallow clustering methods to achieve the ultimate clustering results. However, this straightforward method fails to recognize the fundamental link between representation learning and clustering: clustering offers crucial insights for representation learning, while discriminative representation significantly improves the clustering procedure. Given these motivations, deep clustering approaches are dedicated to addressing these research questions:

(1) How to learn discriminative representations with deep learning that can yield better clustering performance?
(2) How to acquire accurate clustering results given discriminative representations with deep learning?
(3) How to efficiently conduct clustering and representation learning in a unified framework and enhance each other?

In light of the three challenges mentioned above, numerous deep clustering methods have emerged, incorporating diverse deep architectures and data types, and have achieved significant

Table 1. Comparison with Related Surveys

Comparisons \ References		Shallow Clustering					Deep Clustering						
		[13]	[19]	[14]	[12]	[15]	[16]	[17]	[18]	[20]	[21]	[22]	**Ours**
Deep representation learning design	Auto Encoder	–	–	–	–	–	✓	✓	✓	✓	✓	✓	✓
	Generative	–	–	–	–	–	✓	✓	–	✓	✓	✓	✓
	Contrastive	–	–	–	–	–	–	–	–	–	✓	✓	✓
	Cluster Based	–	–	–	–	–	✓	✓	–	–	✓	–	✓
Clustering with DNN		✓	✓	✓	–	✓	✓	✓	✓	✓	✓	✓	✓
Interaction between Representation and Clustering	Multistage	✓	✓	✓	✓	✓	–	✓	✓	✓	✓	✓	✓
	Iterative	✓	✓	–	–	–	–	✓	✓	–	–	–	✓
	Generative	–	–	–	–	–	✓	–	–	✓	✓	✓	✓
	Simultaneously	–	–	–	–	–	✓	✓	✓	–	✓	✓	✓
Application		✓	✓	✓	–	✓	–	–	–	–	–	✓	✓
Various types of Dataset		–	–	–	–	–	–	–	–	–	–	–	✓
Evaluation		–	–	–	✓	✓	✓	–	–	–	–	✓	✓
Implementation		–	–	–	–	–	–	–	–	–	–	✓	✓

*Each column indicates a survey paper which is being compared, "✓" means the term of the corresponding row has been surveyed or analyzed in this paper while '-' means not. Note that surveys published later than ours are also included.

success. However, the existing literature has not sufficiently synthesized the current research landscape, limiting the formation of a detailed understanding that effectively integrates and assesses the varied results in this area. To address this deficiency, this article presents a comprehensive survey of deep clustering, introducing a novel taxonomy of existing methods, examining unresolved challenges, and suggesting essential directions for future research.

Related Surveys. As a classic task, clustering has experienced extensive development in the past decades, with numerous surveys focusing on shallow clustering [11–15]. Typically, shallow clustering methods discussed in these surveys utilize feature representations as input and produce cluster assignments for each data instance. Subsequently, as deep learning has evolved, several studies have begun to explore clustering techniques that incorporate deep learning approaches.

Min et al. [16] state that the essence of deep clustering is to learn clustering-oriented representations, so the literature should be classified according to the network architecture of representation learning. Aljalbout et al. [17] designed the taxonomy mainly based on the representation learning architecture and the loss function, discussing the comparisons between methods only on the MNIST and COIL20 datasets. Nutakki et al. [18] review some clustering methods with deep learning, while the most recent advanced techniques for representation learning and clustering were excluded. Furthermore, all these works [16–18] merely cover outdated methods and miss more recent advances in this active research area.

Nevertheless, previous surveys fail to provide a thorough analysis of how deep learning is integrated into deep clustering approaches, or to offer a detailed classification of the methods involved. This deficiency has impeded the advancement of deep clustering and urgently requires rectification. In this article, we focus on clustering techniques that utilize deep learning, particularly focusing on how deep representation learning interacts with clustering using deep neural networks. A summary of how our survey compares to the previous ones is presented in Table 1. Note that some references were cited after the initial pre-print date of this article.

Contributions. In summary, this article seeks to offer a comprehensive survey of the latest deep clustering techniques and help potential readers understand the broad landscape of deep clustering with respect to the following aspects:

- **Cornerstones of Deep Clustering.** We summarize two cornerstones of deep clustering, namely the representation learning module and the clustering module. For each module, we highlight the representative and universal designs summarized from existing methods, which can be easily generalized to new models.
- **Systematic Taxonomy.** We propose a systematic taxonomy of existing deep clustering methods based on the ways of interactions between the representation learning module and the clustering module providing four representative branches of methods. We also compare and analyze the properties of each branch in different scenarios.
- **Abundant Resources and References.** We collect various types of benchmark datasets, evaluation metrics, and open-source implementations of the latest publications on deep clustering, which are organized together with references on Github.[1] We also provide an open source and easy-to-use package that can effectively deploy and develop deep clustering methods, with the aim of further advancing this active research area.[2]
- **Future Directions.** Based on the properties of the representation learning module and the clustering module, as well as their interactions, we discuss the limitations and challenges of existing methods, followed by our insights and thoughts on promising research directions that deserve future investigations.

Organization. The rest of this survey is organized as follows: Section 2 introduces the basic definitions and notation used in this article. Section 3 summarizes the representative design of the representation module, along with different data types. Section 4 summarizes the representative design of the clustering module, which focuses mainly on the basic modules defined in the deep clustering methods. Section 5 summarizes the representative ways of interactions between the two modules, which cover most of the existing literature. Section 6 introduces the widely used benchmark datasets and evaluation metrics. Section 7 discusses the applications of deep clustering. Section 8 discusses limitations, challenges and suggests future research directions that deserve further exploration. The overall organization of this survey is illustrated in Figure 1.

2 Preliminary

In this section, we first briefly introduce some definitions in deep clustering that need to be clarified, then illustrate the notation used in this article in Table 2.

Deep Clustering and Shallow Clustering. Given a set of data instances $\mathcal{X} = \{x_i\}_{i=1}^N$, the objective of clustering is to automatically assign each instance x to groups so that instances in the same group are similar while instances from different groups are dissimilar. The *shallow (non-deep) clustering* takes the feature vectors of the instances as input and outputs the clustering result without deep neural networks. The *deep clustering* aims to cluster unstructured data or high-dimensional data with deep neural networks. It is worth noting that deep clustering is not narrowly defined as applying deep learning techniques in representation learning. Instead, clustering itself can be conducted by deep neural networks and benefits from the interaction with deep representation learning.

Hard Clustering and Soft Clustering. The clustering methods can be categorized into hard and soft clustering according to the type of output they generate. The output of hard clustering is a discrete one-hot cluster label $\tilde{y}_i$ for each instance x_i, while the output of soft clustering is a

[1] https://github.com/zhoushengisnoob/DeepClustering
[2] https://github.com/zhoushengisnoob/OpenDeepClustering

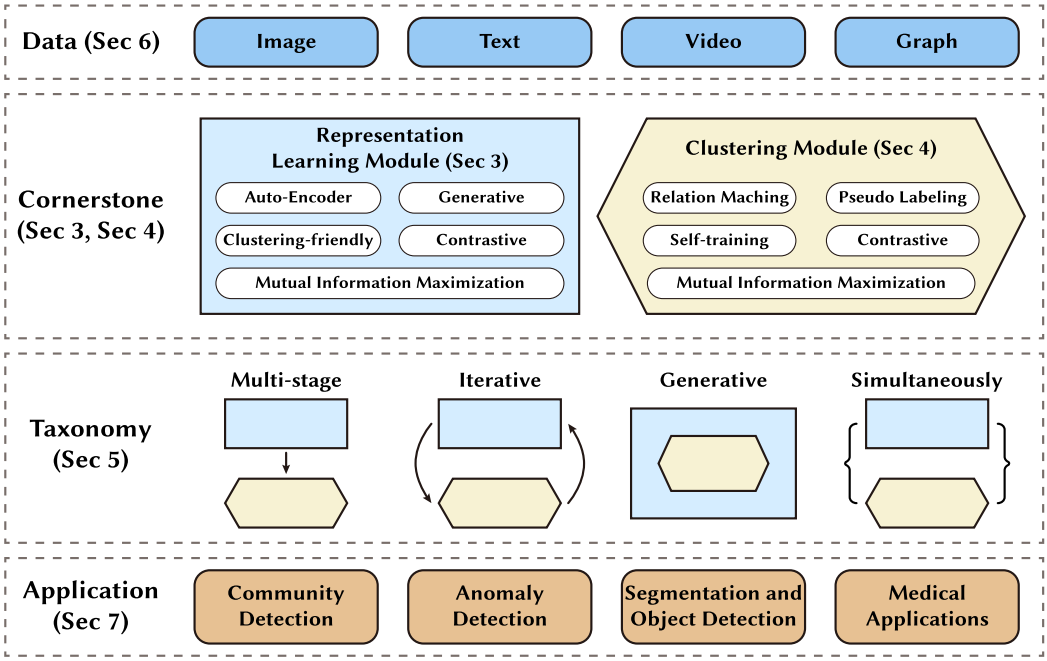

Fig. 1. Overall organization of this survey. We use blue and yellow to highlight the representation of learning module and clustering module.

Table 2. Important Notations Used in this Article

Notations	Explanations	Notations	Explanations
N	The number of data instances	K	The number of clusters
x	Data instance	$\hat{x}$	Data reconstruction
$x^{\mathcal{T}}$	Augmented instance	$.^T$	Transpose of a matrix (vector)
h	Representation	z	Soft assignment
$\tilde{y}$	Predicted hard label	y	Ground truth label
c	Representation (centroid) of cluster	τ	Temperature parameter
$\mathbb{P}, \mathbb{Q}$	Probability distribution	$\mathcal{X} = \{x_i\}_{i=1}^N$	Data instances set
$\|\cdot\|_F$	Frobenius norm of a matrix (vector)	$\|\cdot\|$	The number of elements in a set

continuous cluster assignment probabilistic vector $z_i \in \mathcal{R}^K$. The primary objective of clustering is to assign samples into distinct groups, hence the outcomes of clustering are typically presented with discrete labels. However, the discrete assignment of an instance is usually hard to optimize, especially for deep neural networks with backpropagation. As a result, most of the existing deep clustering methods fall into the soft clustering category. In this approach, the clustering results are produced by a deep neural network f, which produces logits of K dimensional activated by a softmax function. It should be noted that the soft clustering results can be easily transformed to hard clustering by selecting the cluster with the highest probability.

Partitioning Clustering and Overlapping Clustering. The previously described hard clustering, which maximizes the probability vector, is intended for the *partitioning clustering*

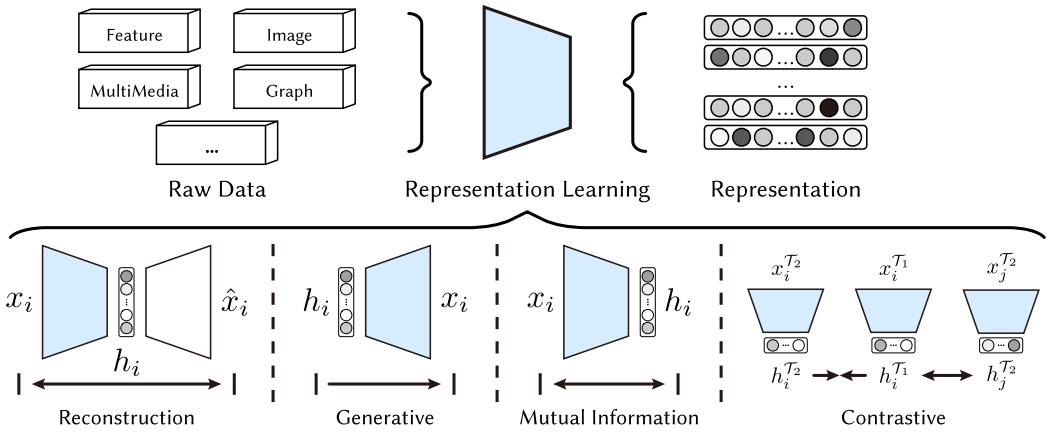

Fig. 2. Representative representation learning modules that transform raw data into lower-dimensional representations.

scenario, where each data instance is assigned to a single cluster only. Consequently, this clustering approach divides the entire dataset into K separate groups. Most current deep clustering techniques are non-overlapping, reflecting the nature of the primarily non-overlapping datasets they are tested on. In the context of overlapping clustering, a data instance can be part of multiple clusters, introducing additional challenges to clustering methods, which will be explored further in Section 8.

3 Representation Learning Module

The past few decades have seen significant advancements in deep representation learning [23–25], particularly in unsupervised methods. All unsupervised learning approaches can function as input generators and are seamlessly integrated into the deep clustering framework, with more details provided in Section 4. However, these methods are generally not designed for clustering tasks and do not incorporate essential clustering data to enhance representation learning. This section presents the *representation learning module* used in deep clustering, which processes raw data to produce a compact, low-dimensional representation. Figure 2 shows the key representation learning modules discussed in this section.

3.1 Auto-Encoder Based Representation Learning

Auto-Encoder [26] is one of the most widely adopted unsupervised deep representation learning methods due to its simplicity and effectiveness. The auto-encoder is a linear stack of two deep neural networks named encoder and decoder. The encoder $\mathbf{f}_e$ encodes the input data x in a low-dimensional representation $h = \mathbf{f}_e(x)$, and the decoder $\mathbf{f}_d$ decodes the low-dimensional representation h to the input data space $\hat{x} = \mathbf{f}_d(h)$. A good auto-encoder is expected to reconstruct the input data without dropping any information, thus the optimization target of the auto-encoder is usually formulated as Equation (1):

$$\mathcal{L}_{AE} = \sum_{i}^{N} \|x_i - \hat{x}_i\|_2^2 = \sum_{i}^{N} \|x_i - \mathbf{f}_d\left(\mathbf{f}_e\left(x\right)\right)\|_2^2, \tag{1}$$

where $\|\cdot\|_2$ is the L2-norm, $\hat{x}_i$ is the reconstructed data. The auto-encoder is a general structure and can be customized for different data types. For example, deep neural networks for vectorized

features [27], convolutional networks for images and graph neural networks for graphs [28], 3D convolutional networks and LSTM auto-encoder for videos [29] and so on.

Analysis. The representation learning framework based on auto-encoders is appreciated for its straightforward implementation and effective training, and has been extensively used in initial deep clustering studies. However, this approach learns representations on an individual instance basis, often overlooking the interconnections between different instances. Consequently, in the embedding space, instances may not be distinctly separated, leading to suboptimal clustering results.

3.2 Deep Generative Representation Learning

Another line of deep unsupervised representation learning lies in generative methods which assume that the data x are generated from a latent representation h and then reversely deduce the posterior of the representation $p(h|x)$ from the data. Among these, the most typical method is the **Variational Auto-Encoder** (**VAE**) [30] and **Generative Adversarial Networks** (**GAN**) [31]. Both methods focus on capturing the data distribution, which is used to map the input data into the latent embedding space. VAE resorts to the variance inference technique and maximizes the **evidence lower bound** (**ELBO**) on the data likelihood:

$$\log p(x) \geq \mathbb{E}_{q(h|x)}[\log p(x|h)] - D_{KL}(q(h|x)\|p(h)), \tag{2}$$

where $D_{KL}(\cdot\|\cdot)$ denotes the KL-divergence between two distributions, $p(h)$ is the prior distribution of the latent representation, $q(h|x;\varphi)$ is the variational posterior of the representation to approximate the true posterior (*i.e.,* $q(h|x;\varphi) \approx p(h|x)$), which can be modeled with the recognition networks φ. Using the reparameterization trick [30] and the Monte Carlo approximation [32], the posterior can be efficiently learned from the Equation (2) via backpropagation. GANs learn representations via a competitive mechanism in which a generator network generates samples to deceive a discriminator network. In response, this discriminator network learns to differentiate between real and generated data, forcing the generator to progressively improve its representations. By modeling the data distribution in an adversarial way, the GANs can learn more discriminative representation, thereby enhancing their clustering capabilities.

Analysis. Deep generative representation learning exhibits advantages such as flexibility, interpretability, and the ability to replicate data points. However, in comparison to other methods of representation learning, they struggle to incorporate clustering information effectively during the learning phase. The widely used strategy involves making assumptions at the cluster level about the data distribution, such as using Gaussian mixture models. However, these assumptions often do not suffice for handling complex data distributions. Consequently, exploring ways to embed category information within the distribution modeling phase continues to be an essential question for deep generative representation learning in clustering contexts.

3.3 Mutual Information Maximization Representation Learning

Mutual information (**MI**) [33] is a fundamental quantity to measure the dependence between random variables X and Y, which is formulated as Equation (3):

$$\mathcal{I}(X;Y) = \int \log \frac{d\mathbb{P}_{XY}}{d\mathbb{P}_X \otimes \mathbb{P}_Y} d\mathbb{P}_{XY}, \tag{3}$$

where $\mathbb{P}_{XY}$ is the joint distribution, $\mathbb{P}_X = \int_Y d\mathbb{P}_{XY}$ and $\mathbb{P}_Y = \int_X d\mathbb{P}_{XY}$ are the marginal distribution, $\mathbb{P}_X \otimes \mathbb{P}_Y$ is the product of the marginal distributions. Traditional mutual information estimations [34] are only tractable for discrete variables or known probability distributions. Recently, MINE [35] was proposed to estimate mutual information with deep neural networks. The widely

used mutual information estimation is the **Jensen-Shannon divergence** (**JSD**) [36] formulated as Equation (4):

$$\mathcal{I}_{JSD}(X;H) = \mathbb{E}_{\mathbb{P}_{XH}}[-\text{sp}(-D(x,h))] - \mathbb{E}_{\mathbb{P}_X \times \mathbb{P}_H}[\text{sp}(D(x,h))], \tag{4}$$

where $\text{sp}(x) = \log(1+e^x)$ is the softplus function. D is a discriminator function modeled by a neural network. Another popular mutual information estimation is InfoNCE [37], which will be introduced in Section 3.4. Taking advantage of neural estimation, mutual information has been widely adopted in unsupervised representation learning [38, 39]. More specifically, the representation is learned by maximizing mutual information between different layers [39] or different parts of the data instances [38], so that the consistency of the representation can be guaranteed. This can be viewed as an early attempt at self-supervised learning which has an extensive impact on the later work.

Analysis. The significant benefit of this branch of methods is that the variables assessed through mutual information are not confined to identical dimensions or semantic spaces, like instances and clusters. Similar to auto-encoder based and deep generative representation learning, the mutual information maximization approaches are also focused on individual instances, potentially facing the same issues in modeling the relationships among instances. However, the marginal distribution used in mutual information estimation is influenced by all observed samples and the relationships among instances are indirectly established.

3.4 Contrastive Representation Learning

Contrastive learning is one of the most popular unsupervised representation learning techniques in recent years. The basic idea is to pull a positive pair of instances close while pushing a negative pair of instances far away, which is also known as instance discrimination. The representative target of contrastive learning is the InfoNCE loss [37] formulated as Equation (5):

$$\mathcal{L}_{InfoNCE} = -\frac{1}{|N|}\sum_{i=1}^{N}\log\frac{\exp(f(h_i^{\mathcal{T}_1}, h_i^{\mathcal{T}_2})/\tau)}{\sum_{j=1}^{N}\exp(f(h_i^{\mathcal{T}_1}, h_j^{\mathcal{T}_2})/\tau)}, \tag{5}$$

where $h_i^{\mathcal{T}_1}$ and $h_i^{\mathcal{T}_2}$ is a pair of positive samples and $h_j^{\mathcal{T}_2}$ are the negative sample representations, f is a similarity function, τ is the temperature parameter [40]. The positive samples are usually conducted by data augmentation which may vary from different data types. For example, flip, rotate, and crop augmentation for image data [41], node dropping, edge perturbation, attribute masking, and subgraph sampling for graph data [42, 43]. Negative samples are selected from an augmented view of other instances in the dataset [41] or from a momentum-updated memory bank of old negative representations [44], which can be viewed as an approximation of noise.

Analysis. Numerous theoretical studies [45–47] have been conducted on contrastive learning, demonstrating that contrastive learning tends to pull instances with the same label close while pushing instances with different labels away in the low-dimensional embedding space. Consequently, this type of learning can greatly improve the effectiveness of clustering by enhancing discriminative representation. In addition, it has been proved that reducing InfoNCE loss corresponds to an increase in the lower limit of mutual information [37]. Thus, contrastive learning can also be used to establish explicit connections between instances and clusters.

3.5 Clustering Friendly Representation Learning

Although the aforementioned representation learning methods have somehow implicitly boosted the performance of clustering, they are not explicitly designed for the clustering task. In this

subsection, we will introduce some representation learning methods that explicitly support the clustering task.

K-means [48] friendly representation is first defined in DCN [49] where samples in the low-dimensional space are expected to scatter around their corresponding cluster centroids. This can well support the assumption of K-means that each sample is assigned to the cluster with the minimum distance to the centroid. The objective $\mathcal{L}_{KF}$ can be formulated as Equation (6):

$$\mathcal{L}_{KF} = \sum_{i=1}^{N} \|\mathbf{f}(x_i) - \tilde{y}_i \mathbf{C}\|_2^2, \tag{6}$$

where $\mathbf{f}(\cdot)$ is the deep neural network for representation learning, $\tilde{y}_i$ is the hard assignment vector of data instance x_i which has only one non-zero elements, $\mathbf{C}$ is the cluster representation matrix where kth column of $\mathbf{C}$, i.e., c_k denotes the centroid of the kth cluster.

Spectral clustering friendly representation learning is inspired by the eigen decomposition in spectral clustering [50] that projects instances into a space with orthogonal bases. In deep clustering, the orthogonal basis is modeled explicitly by reducing correlations within features [51], which can be formulated as Equation (7):

$$\mathcal{L}_{SF} = \sum_{m=1}^{d} \left(-h_m^T h_m / \tau + \log \sum_{n}^{d} \exp\left(h_n^T h_m / \tau \right) \right), \tag{7}$$

where d is the number of feature dimensions, h_m is the mth dimension feature vector, τ is the temperature parameter. The objective is to learn the independent features so that redundant information is reduced.

Analysis. The clustering friendly representation learning benefits from direct optimization for clustering, which may significantly improve the corresponding cluster performance. However, such simplicity also limits the generalization to other clustering methods. Currently, the research community has made more efforts to inspire clustering methods and express them from a deep learning perspective, rather than learning specific representations for each clustering method.

3.6 Subspace Representation Learning

Subspace representation learning is the early stage of subspace clustering [52], which aims to map the data instances to a low-dimensional subspace where instances can be separated. Basically, current subspace representation learning methods [53–58] have relied on the self-expressiveness assumption, where a data instance can be expressed as a linear combination of other data instances from the same subspace, i.e., $\mathbf{X} = \mathbf{X}\mathbf{\Theta}_\mathbf{c}$, where $\mathbf{X}$ is the data matrix and $\mathbf{\Theta}_\mathbf{c}$ is the matrix of self-expression coefficients. For representation learning, the self-expression property leads to the following objective as in Equation (8):

$$\min_{\mathbf{\Theta}_\mathbf{c}} \|\mathbf{\Theta}_\mathbf{c}\|_p + \frac{\lambda}{2}\|\mathbf{H} - \mathbf{H}\mathbf{\Theta}_\mathbf{c}\|_F^2 \quad \text{s.t.} \quad \text{diag}(\mathbf{\Theta}_\mathbf{c}) = \mathbf{0}, \tag{8}$$

where $\|\cdot\|_p$ is a matrix norm, λ controls the weight balance. $\mathbf{H}$ denotes the sample representations learned by a network. In addition, $\mathbf{\Theta}_\mathbf{c}$ can be implemented as parameters of an additional network layer [55].

Analysis. The effectiveness of subspace representation learning is based on the premise that data can be represented through various subspaces characterized by unique feature sets, capable of managing datasets with diverse dimensions and intricate relationships. However, it is challenged by computational difficulties, such as determining the best subspaces and allocating data points to these subspaces, particularly in the context of extensive, high-dimensional datasets.

3.7 Data Type Specific Representation Learning

The representation learning techniques mentioned above are suitable for various data types, but in practical applications, some methods are designed specifically for certain data types. This subsection offers a summary of four widely studied data types and their corresponding representation learning strategies in the context of deep clustering.

3.7.1 Image Representation Learning. Learning representations of images using CNN [59] and ResNet [60] as a backbone has achieved great success in the past decades. In the image deep clustering, they still play active roles as feature extractors or backbones in the representation learning module. Beyond the above two methods, recent advances have been made by introducing modern representation learning techniques such as vision transformer [61] to deep clustering. As one of the most popular directions, unsupervised representation learning for image data will play a central role in deep clustering and affect the other data types.

3.7.2 Text Representation Learning. The early attempts of text representation learning have utilized statistical-based methods such as TF-IDF [62], Word2Vec [63], and Skip-Gram [64]. Later, some works focus on topic modeling [65] and semantic distances [66, 67] for text representation learning, and more [68] on unsupervised scenarios. Recently, pre-trained language models like BERT [69] and GPT-3 [70] are gradually dominating the area of text representation learning. However, the fine-tuning [71] of these methods in the deep clustering task remains an open question. Furthermore, while **large language models** (**LLMs**) [72] have shown remarkable linguistic prowess in various NLP applications, their utilization is restricted to APIs. The use of LLMs for sideway information for clustering has shown great potential [73].

3.7.3 Video Representation Learning. Video representation learning presents a complex challenge, integrating spatial-temporal learning, multimodal learning involving audio [74], and natural language processing tasks such as video abstracts and subtitles, all within a single framework. The early methods utilize the LSTM Autoencoder [29], 3D-ResNets [75], and 3D-U-Net [76] as feature extractor. Recent methods have focused on spatial-temporal modeling [29, 77–80] and Qian et al. [75] in particular incorporates contrastive learning for self-supervision.

3.7.4 Graph Representation Learning. The classic graph representation learning aim at learning low-dimensional representation for nodes so that the proximity among nodes can be preserved in the embedding space. **Graph Neural Networks** (**GNNs**) [81, 82] are widely used, including GCN [83], GraphSAGE [84] and GAT [85], and it gives infinite possibilities of graph node representation learning combining node features and graph topology [86–89]. Furthermore, graph-level information also has great potential in tasks such as protein classification [90], which has prompted increasing attention to graph-level representation learning [91–94].

Analysis. The data type specific representation learning mentioned above can be a naive backbone for feature extraction or end-to-end unsupervised representation learning, which are the most active research directions in deep learning. With more types of data being collected and the fast development of deep learning, we believe that deep clustering will grow along with the data type-specific representation learning techniques.

4 Clustering Module

In this section, we present the key clustering modules used in deep clustering, which utilize low-dimensional representations as input, and produce either cluster labels for hard clustering or probabilities of cluster assignments for soft clustering. Although numerous shallow clustering techniques can be applied directly to clustering tasks, integrating them with deep representation

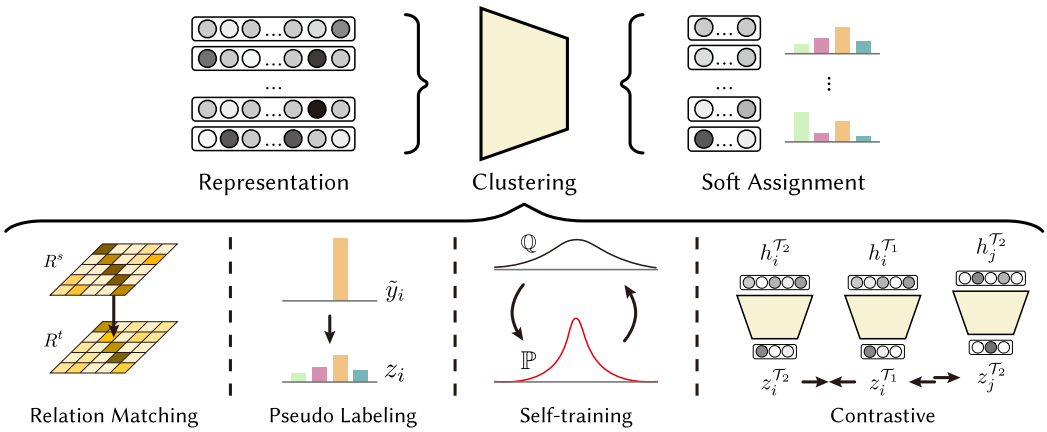

Fig. 3. Representative deep clustering modules that produce soft assignments for samples based on their representations.

learning within a single framework is challenging. Furthermore, these methods often fail to effectively engage the representation learning module to achieve mutual improvement. For further information on shallow clustering methods, we suggest previous surveys [13, 19]. In deep clustering, the objective is to adhere to a deep learning framework, where the clustering results are obtained by optimizing the neural network. As depicted in Figure 3, the subsequent sections will discuss five prevalent approaches to constructing clustering modules in deep clustering.

4.1 Relation Matching Deep Clustering

Relation matching techniques are those that enhance clustering outcomes by aligning connections among instances (instance to instance, I2I) with respect to the clustering outcomes (instance to cluster centers, I2C). To obtain the relationship between instance and cluster centers, the direct approach is to measure the distance or similarity between instance and cluster centers in the low-dimensional space. A closer relation implies a higher probability of belonging to a particular cluster. The I2C can be directly optimized by

$$\mathcal{L}_{I2C} = \sum_{i=1}^{N} \|h_i - Ms_i\|_2 \,, \ \text{s.t.} 1^T s_i = 1 \quad \forall i, j, \tag{9}$$

where s_i is the assignment vector, and the k th column of M, i.e., m_k, denotes the centroid of the k th cluster, h_i is the representation of instance in the low-dimensioanl space. By fixing x and M, optimizing Equation (9) outputs the cluster assignment; by only fixing x, optimizing Equation (9) is equal to the result of running K-means.

Also, the clustering outcomes can be optimized by matching relationships among instances from different views, which can be formulated as

$$\mathcal{L}_{RM} = \sum_{i}^{N} \sum_{j}^{N} \ell \left(R_{ij}^{s}, R_{ij}^{t} \right), \tag{10}$$

where ℓ is a measure of relation matching, e.g., cosine similarity or Euclidean distance, R^s and R^t are the relations in source and target space. Here, the relationship based on I2C is typically treated as the source space, and the target space may come from the similarity between the embedding or raw features of instances.

Analysis. The relation matching deep clustering explicitly connects the representation learning and clustering by matching the relation from embedding spaces to label spaces, which is straightforward and easy to implement. However, calculating N^2 pairs of instances for I2I formulations is computationally inefficient. To tackle this challenge, some methods only preserve the k-nearest-neighbor relations [95, 96] for each instance or the relations with high confidence [95]. Although this can somehow improve the training efficiency, the extra hyper-parameter is hard to set in an unsupervised manner. Furthermore, among all pairs of relations, many of them are noisy especially in the early training phases with limited capability. How to filter out the clean relations to boost the performance while discarding the noisy relations is still an open research question.

4.2 Pseudo Labeling Deep Clustering

Pseudo labeling, a common approach in semi-supervised learning [97], has been recently applied to deep clustering. This method can be considered a form of relation matching, characterized by discrete relations that depend on the consistency of the labels. According to the way of using pseudo labels, existing methods can be largely divided into two groups: instance-wise pseudo labeling [95, 98, 99] and relation-wise pseudo labeling [100, 101].

The instance-wise pseudo labeling filters out a subset of instances with high confidence and trains the network in a supervised manner with cross-entropy loss as Equation (11):

$$\mathcal{L}_{IPL} = -\frac{1}{|\mathcal{X}^c|}\sum_{i}^{\mathcal{X}^c}\sum_{k=1}^{K}\tilde{y}_{ik}\log\left(z_{ik}\right), \tag{11}$$

where $\mathcal{L}_{IPL}$ denotes the loss of instance-wise pseudo labeling, $\mathcal{X}^c$ denotes the subset of instances with high confidence, $\tilde{y}_{ik}$ and z_{ik} are the predicted hard label and soft cluster assignment. The confidence is usually estimated by the entropy or maximum of the probabilistic distribution of assignment.

The general idea of relation-wise pseudo labeling is to enforce instances with same pseudo labels closer while instances with different pseudo labels away from each other in the embedding space. Given the filtered instances, relation-wise pseudo labeling constructs the discrete relations among instances to guide the representation learning: the *must-link* for pairs of instances with same pseudo labels and *cannot-link* for pairs of instances with different pseudo labels as Equation (12):

$$\mathcal{L}_{RPL} = \frac{1}{|C|}\sum_{\{i,k\}\in C} R_{ik} - \frac{1}{|\mathcal{M}|}\sum_{\{i,j\}\in\mathcal{M}} R_{ij}, \tag{12}$$

where $\mathcal{M}$ is the set of must-link relations and C is the set of cannot-link relations, R_{ij} is the similarity of instance x_i and x_j in the low-dimensional embedding space.

Analysis. Pseudo labeling introduces the advantages of semi-supervised learning to unsupervised clustering tasks. However, its effectiveness largely depends on the quality of the pseudo labels, which are prone to the influence of model performance and the adjustment of hyperparameters, particularly in unsupervised settings. Existing approaches [95, 98] employ pretraining as an initial step prior to pseudo labeling to address these issues, yet this area still requires further focus.

4.3 Self-Training Deep Clustering

The self-training strategy is introduced to the deep clustering task [102] and opens up an active branch of methods named self-training deep clustering [9, 102–107]. More specifically, the cluster assignment distribution is optimized by minimizing the KL-divergence with an auxiliary

distribution as Equation (13):

$$\mathcal{L} = D_{KL}(\mathbb{P}\|\mathbb{Q}) = \sum_{i}^{N}\sum_{k}^{K} p_{ik}\log\frac{p_{ik}}{q_{ik}}, \tag{13}$$

where $\mathbb{Q}$ is the cluster assignment distribution and $\mathbb{P}$ is the auxiliary distribution. q_{ik} and p_{ik} denote the probability that instance x_i belongs to the group k. The assignment distribution $\mathbb{Q}$ follows the assumption of K-means and is produced by the embedding distance between the instance and cluster centroids as in Equation (14):

$$q_{ik} = \frac{(1 + \|h_i - c_k\|_2^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_j^K (1 + \|h_i - c_j\|_2^2/\alpha)^{-\frac{\alpha+1}{2}}}, \tag{14}$$

where h_i is the representation of data instance x_i and c_k is the representation of cluster k, α is the freedom degree of the Student's t-distribution [108]. The auxiliary distribution $\mathbb{P}$ is a variant of the assignment distribution $\mathbb{Q}$ with both instance-wise and cluster-wise normalization as in Equation (15):

$$p_{ik} = \frac{q_{ik}^2/f_k}{\sum_j^K q_{ij}^2/f_j}, \tag{15}$$

where $f_k = \sum_i^N q_{ik}$ are soft cluster frequencies.

Analysis. The success of self-training deep clustering relies on the following properties: First, the square of cluster assignment probability with cluster-wise normalization will encourage the model to pay more attention (gradient) to the instances with higher confidence, which in turn reduces the impact of low-confidence ones. As a result, the cluster assignment vector tends to be one-hot. Second, the soft cluster frequencies f_k can be viewed as the sum of the probability that the instance belongs to the kth cluster. This can prevent the degenerate solution that all instances belong to the same cluster. Despite its effectiveness, this objective is prone to class imbalance issues and requires additional investigation.

4.4 Contrastive Deep Clustering

Similar to contrastive representation learning, the target of contrastive deep clustering is to pull the positive pairs close while pushing the negative pairs far away. The major difference lies in the definition of positive pairs and negative pairs, which can be further divided into three groups:

4.4.1 Instance-Instance Contrast. The instance-instance contrast treats the cluster assignment of each instance as the representation and directly reuses the contrastive representation learning loss as Equation (16):

$$\mathcal{L}_{IIC} = -\frac{1}{|N|}\sum_{i=1}^{N}\log\frac{\exp(f(z_i^{\mathcal{T}_1}, z_i^{\mathcal{T}_2})/\tau)}{\sum_{j=1}^{N}\exp(f(z_i^{\mathcal{T}_1}, z_j^{\mathcal{T}_2})/\tau)}, \tag{16}$$

where $z_i^{\mathcal{T}_1}$ is the cluster assignment of augmented instance $x_i^{\mathcal{T}_1}$ predicted by the clustering module.

4.4.2 Cluster-Cluster Contrast. The cluster-cluster contrast treats each cluster as an instance in the embedding space, the target is pulling the cluster and its augmented version close while pushing different clusters far away, which can be formulated as Equation (17):

$$\mathcal{L}_{CCC} = -\frac{1}{|N|}\sum_{k=1}^{K}\log\frac{\exp(f(c_k^{\mathcal{T}_1}, c_k^{\mathcal{T}_2})/\tau)}{\sum_{j=1}^{K}\exp(f(c_k^{\mathcal{T}_1}, c_j^{\mathcal{T}_2})/\tau)}, \tag{17}$$

where $c_k^{\mathcal{T}_1}$ and $c_k^{\mathcal{T}_2}$ are the representations of cluster k in different augmented views. It should be noted that the cluster-cluster contrast satisfies the basic requirement of clustering that each cluster should be dissimilar, which is in agreement with the clustering-friendly representation learning described in Section 3.5.

4.4.3 Instance-Cluster Contrast. The instance-cluster contrast is similar to the K-means, which utilizes the cluster centroid as explicit guidance. Given the representation of each instance and cluster centroid in the same low-dimensional space, each instance is expected to be close to the corresponding cluster centroid while far from the other cluster centroids. Such similarity and dissimilarity can be naturally modeled by the contrastive learning as Equation (18):

$$\mathcal{L}_{ICC} = -\frac{1}{|N|}\sum_{i=1}^{N}\log\frac{\exp\left(f\left(h_i, c_i'\right)/\tau\right)}{\sum_{j=1}^{K}\exp\left(f\left(h_i, c_j\right)/\tau\right)}, \tag{18}$$

where c_i' is the corresponding cluster centroid of x_i which is usually estimated by an alternative clustering method. This can also be understood as maximizing the mutual information between the representation and cluster assignment with data augmentations.

Analysis. Besides the advantages inherited from mutual information maximization clustering, the primary advantages of contrastive deep clustering are that data augmentation helps improve the robustness of clustering with data augmentation, which has been ignored by most existing methods. For detailed advantages of contrastive learning, we suggest the former contrastive learning surveys [109, 110].

5 Taxonomy of Deep Clustering

In this section, we expand upon the core modules of representation learning and clustering as outlined in previous discussions on deep clustering frameworks. Given the inputs and outputs of these modules, various strategies are implemented in existing deep clustering methods to enhance their interaction and collaborative efforts, thus increasing the effectiveness of deep clustering. Therefore, we will omit the intricate details of each module's design and instead concentrate on their interconnections, classifying the existing approaches into four specific categories:

(1) Multi-stage deep clustering: the representation learning module is sequentially connected with the clustering module.
(2) Iterative deep clustering: the representation learning module and the clustering module are iteratively updated.
(3) Generative deep clustering: the clustering module is modeled as a prior of representation module.
(4) Simultaneous deep clustering: the representation learning module and the clustering module are updated simultaneously.

We select the most representative methods from each category within our taxonomy and provide a detailed presentation of their specific designs under our taxonomy in Table 3.

5.1 Multistage Deep Clustering

Given the modules for representation learning and clustering, it is evident that the output of representation learning smoothly transitions into the input of the clustering module, thus directly helping to produce the clustering results. Multistage deep clustering refers to the methods in which the two modules are optimized separately and connected sequentially. With the evolving of both representation learning and clustering modules, multistage methods continue to be an active area in the field of deep clustering.

Table 3. Specific Designs of Representation and Clustering Modules of Methods Following Our Taxonomy

	Representation Module	Clustering Module	Method
Multi Stage	Auto-Encoder	k-means	SAE [111]
	Auto-Encoder	k-means + Relation Matching	DEN [112]
	Subspace	k-means	PARTY [57]
	Contrastive	k-means	IDFD [51]
	Auto-Encoder + Subspace	spectral clustering	DSC-Nets [55], SENet [56]
	Generative + Subspace	spectral clustering	DASC [58]
	Data Specific	Relation Matching	GTL [113]
	Data Specific	spectral clustering	SCPSO [114]
	Data Specific	k-means	ClusterLLM [73]
Generative	Generative (VAE)	Relation Matching	GMVAE [115], VaDE [116]
	Generative (VAE)	k-means + Pseudo Labeling	DGG [117]
	Generative (GAN)	Relation Matching	ClusterGAN_1 [118], GM-GAN [119], GANMM [120]
	Generative (GAN)	k-means	ClusterGAN_2 [121]
	Data Specific + Generative(GAN)	Relation Matching	CommunityGAN [122]
Iterative	Pseudo Labeling	Relation Matching	DeepCluster [99], DeepMCAT [123], CCFC [124]
	Auto-Encoder + Clustering Friendly	Relation Matching	DCN [49], AEDC [125]
	Auto-Encoder + Subspace	Relation Matching	NCSC [54]
	Auto-Encoder + Subspace	Pseudo Labeling	$\text{S}^2\text{ConvSCN}$ [53], PSSC [126]
	Contrastive	Pseudo Labeling + Relation Matching	JULE [7], GATCluster [101]
	Contrastive	Pseudo Labeling	SPICE [98]
	Contrastive	Self Training	DAC [100]
	Contrastive + Clustering Friendly	Pseudo Labeling	SCAN [95]
	Data Specific	Pseudo Labeling	XDC [74]
	Pseudo Labeling	Pseudo Labeling	POT [127]
Simultaneously	Auto-Encoder	Self-training	DEC [102], DBC [106]
	Auto-Encoder	Relation Matching + Self-training	IDEC [103]
	Contrastive + Auto-Encoder	Self-training	DEPICT [105], DEC-DA [104]
	Data Specific + Auto-Encoder	Self-training	SDCN [128]
	Data Specific + Auto-Encoder	Self-training	scDeepCluster [129]
	Pseudo Labeling	Pseudo Labeling + Relation Matching	CCNN [130]
	Clustering Friendly	Pseudo Labeling	PICA [131]
	Contrastive	Pseudo Labeling + Relation Matching	DCCM [132]
	Contrastive	Mutual Information	IIC [133]
	Contrastive	Contrastive	CC [134], DCDC [135]
	Contrastive	Self-training	SCCL [136]
	Contrastive	Relation Matching	DRC [137]
	Contrastive + Auto-Encoder	Self Training + Relation Matching	IcicleGCN [138]
	Contrastive + Clustering Friendly	Contrastive	GCC [139]

* The term (I2I) and (I2C) following "Relation Matching" refer to "Instance to Instance" and "Instance to Cluster" respectively.

Early multistage deep clustering methods [111, 112] have trained a deep auto-encoder to learn representations, which can be directly packed as input of the K-means method to obtain clustering results. Despite their simplicity, these methods had already surpassed earlier approaches in effectiveness and spurred the advancement of deep clustering. Later, deep subspace clustering was proposed to learn an affinity matrix and instance representations first, and then perform clustering by spectral clustering on the affinity matrix [55, 56, 58] or the K-means on the instance representations [57]. Thanks to the contribution of scikit-learn [140] and many other open-source machine learning libraries, clustering algorithms have been applied to many fields and data types with a limited cost of programming. For example, in the scenario of textual/video/graph data clustering, relation (similarity) matching was used in [113, 141, 142], K-Means in [143], Spectral Clustering in [76, 114, 144] and Hierarchical Agglomerative Clustering in [145], so as many other clustering algorithms being directly applied. In graph data, graph cut based node clustering like Metis [146], Graclus [147] and **Balance Normalized Cut** (**BNC**) [148] were used in graph clustering applications [149–151].

The most recent multistage methods have explicitly incorporated the clustering prior into the representation learning, then conducted clustering on the target friendly representations. For example, IDFD [51] learn representations with two aims: learning similarities between instances and reducing correlations within features. With the above explicit purposes, a naive K-means on the learned representations can also achieve competitive clustering results over many existing deep clustering methods.

Summary. Multistage methods enjoy the property of fast deployment, programming friendly, and intuitive understanding, and thus and be easily employed for applications with various data types [152–159]. However, such a simple combination of deep representation learning and shallow clustering has the following weaknesses:

(1) The majority of representation learning approaches are not specifically tailored for clustering tasks, lacking the necessary discriminative capability for such tasks.
(2) The clustering results reveal fundamental connections between the data instances, while they are not used effectively to inform the learning of discriminative representation.

To conclude, such a straightforward cascade connection will cut off the information flow and interactions between representation learning and clustering, thus the limitations of both sides will influence the final performance together.

5.2 Iterative Deep Clustering

To address the shortcomings of the previously described multistage deep clustering, iterative deep clustering improves the process by utilizing the advantages of robust representations to support clustering, while simultaneously the results of clustering inform the representation learning process. In summary, the typical iterative deep clustering framework is iteratively updated through two main phases: (1) deriving clustering outcomes from current representations and (2) refining the representations based on the latest clustering outcomes. As the representation module serves only to feed the clustering module in iterative deep clustering, this subsection categorizes the existing iterative deep clustering approaches based on the feedback from the clustering module.

5.2.1 Iterative Deep Clustering with Individual Supervision. The individual supervision in iterative deep clustering depends on the pseudo labels generated by the clustering module, which can be used to train the representation learning module in a supervised manner.

In early works [49, 125], the cluster centroids and assignments are updated in a K-means way. S^2ConvSCN [53] and PSSC [126] combine subspace clustering and pseudo labeling, which obtain pseudo labels by spectral clustering or partitioning the pseudo similarity graph. Later, many works

[95, 98, 99] tend to utilize neural networks for both representation learning and clustering, where these two parts are combined together as one neural network. The clustering module is usually a **multilayer perceptron** (**MLP**) that produces soft clustering assignments. In this way, the hard pseudo labels can guide both clustering and representation learning through gradient backpropagation with proper constraints. The representative method is DeepCluster [99], which alternates between K-means clustering and updating the backbone along with the classifier by minimizing the gap between predicted clustering assignments and pseudo labels. In fact, DeepCluster has already been applied as a mature clustering algorithm in video clustering [74].

Later, SCAN [95] follows a pretraining-with-finetuning framework. The clustering results are fine-tuned with self-labeling, which selects the highly confident instances by threshold the soft assignment probability, and updates the whole network by minimizing the cross-entropy loss on the selected instances. SPICE [98] is another representative iterative deep clustering method, where the classification model is first trained under the guidance of pseudo labels and then retrained by the semi-supervised training on the set of reliably labeled instances.

5.2.2 Iterative Deep Clustering with Relational Supervision. The relational supervision in iterative deep clustering refers to the relationship based on the pseudo labels, which provides pairwise guidance to the representation learning module. More specifically, the relationship is usually modeled by whether two instances have same discrete pseudo labels [7, 100] and the model is trained as a binary classification task. Another popular branch of methods [54, 101] models the relationship by the similarity of cluster assignment probabilities, which trains representation learning as a regression task.

Summary. Iterative deep clustering methods [123, 124, 127, 160–162] benefit from mutual promotion between representation learning and clustering. However, they also suffer from the error propagation in the iterative process. More specifically, inaccurate clustering results can lead to chaotic representations in which performance is limited by self-labeling effectiveness. Furthermore, this will in turn affect clustering results, especially in the early stage of training. Therefore, existing iterative clustering methods rely heavily on the pretraining of the representation module. Despite these challenges, iterative deep clustering methods enjoy rapid improvements in both representation learning and clustering capabilities, showing considerable potential in practical applications.

5.3 Generative Deep Clustering

While the previously discussed methods have established a connection between the representation learning and clustering modules through gradient propagation, this connection relies solely on the transfer of gradients between the modules. In an effort to enhance the natural collaborative capabilities of these modules, leveraging the broad utility of deep generative networks in both areas, approaches based on generative deep clustering have been developed and extensively explored.

Specifically, this group of methods makes hypotheses about the latent cluster structure and then infers the clustering assignment by estimation of data density. The most representative model is the **Gaussian mixture model** (**GMM**) [163], which assumes that the data points are generated from a mixture of Gaussians. Specifically, suppose that there are K clusters, and an observed sample x is generated from the following process:

(1) Choose a cluster: $c \sim \text{Mult}(\pi)$
(2) Draw a sample: $\mathbf{x}|c \sim \mathcal{N}\left(\mu_c, \sigma_c^2 \mathbf{I}\right)$

where π denotes the prior probability for clustering; $\text{Mult}(\pi)$ is the multinomial distribution with the parameter π; μ_c and σ_c are the mean and variance parameters of the Gaussian distribution

corresponding to the cluster c. The well-known expectation maximization algorithm can be used to learn optimal parameters and clustering assignments.

Although GMM has gained successful applications, its shallow structure is usually insufficient to capture the nonlinear patterns of the data, adversely affecting its performance on complex data (e.g., images, texts, graphs, etc.). To address this problem, deep generative models have been proposed to combine the generative model with powerful deep neural networks, which have enough capacity to model complex and nonlinear data. This kind of method can be classified into two types: methods based on VAE and methods based on **Generative Adversarial Networks (GAN)**.

5.3.1 Deep Generative Clustering Based on Variational Auto-Encoder. For clustering of high-dimensional and complex data, one promising solution is to directly stack GMM with a deep neural network — GMM generates a latent vector z, and the deep neural network further transforms the latent vector z into the complex data instance x. In this way, the stacked model can enjoy the merits of the latent cluster structure and meanwhile has sufficient capacity to model the complex data. For example, the representative models, VaDE [116] and GMVAE [115], assume the following generative process for each instance:

(1) Choose a cluster: $c \sim \text{Mult}(\pi)$
(2) Draw a latent vector: $\mathbf{z}|c \sim \mathcal{N}\left(\mu_{\mathbf{z}}(c;\beta), \sigma_z^2((c;\beta))\mathbf{I}\right)$
(3) Draw a sample: $\mathbf{x}|\mathbf{z} \sim \mathcal{N}\left(\mu_{\mathbf{x}}(\mathbf{z};\theta), \sigma_{\mathbf{x}}^2(\mathbf{z};\theta)\mathbf{I}\right)$

where $\mu_{\mathbf{z}}(.;\beta)$, $\sigma_{\mathbf{z}}^2(.;\beta)$, $\mu_{\mathbf{x}}(.;\theta)$ and $\sigma_{\mathbf{x}}^2(.;\theta)$ are given by neural networks with parameters β and θ, which determinize the mean and variance of Gaussian distributions, respectively. Given the above generative process, the optimal parameters and cluster assignment can be obtained by maximizing the likelihood of the given data points as in Equation (19):

$$\log p(\mathbf{x}) = \log \int_{\mathbf{z}} \sum_{c} p(\mathbf{x}|\mathbf{z}) p(\mathbf{z}|c) p(c) \mathbf{dz}. \tag{19}$$

However, directly optimizing the above likelihood is intractable as it involves integration and complex neural networks. VAE [30] sheds light on how to tackle this problem so that the parameters and posterior can be efficiently estimated via backpropagation. Specifically, the generative model is trained with the following variational inference objective as Equation (20), a.k.a. the ELBO:

$$\mathcal{L}_{\text{ELBO}}(\mathbf{x}) = \mathbb{E}_{q(\mathbf{z},c|\mathbf{x})}\left[\log \frac{p(\mathbf{x},\mathbf{z},c)}{q(\mathbf{z},c|\mathbf{x})}\right], \tag{20}$$

where $q(\mathbf{z},\mathbf{c}|\mathbf{x};\varphi)$ is the variational posterior to approximate the true posterior, which can be modeled with the recognition networks φ. Monte Carlo [32] and the reparameterization trick [30] can be used to learn the parameters.

More recently, upon VaDE and GMVAE, some improved variants have been proposed. For example, Prasad et al. [164] introduced a data augmentation technique that constrains an input instance (e.g., image) to share a similar clustering distribution with its augmented one; Li et al. [165] employed Monte Carlo objective and the Std Annealing track for optimizing mixture models, which would generate better-separated embeddings than the basic VAE-based methods; Ji et al. [166] proposed to replace decoder in VAE with an improved mutual-information-based objective; Wang et al. [167] proposed to separate the latent embeddings into two parts which capture the particularity and commonality of the clusters, respectively.

5.3.2 Deep Generative Clustering Based on Generative Adversarial Network. Recent years have witnessed great success of the GAN in estimating complex data distribution [122, 168–171]. A

standard GAN contains two components: a generator G that aims at synthesizing "real" samples to fool the discriminator, and a discriminator D tries to discriminate the real data from the generated samples. With the adversary between the two components, the generator could generate samples that have a distribution similar to the data. Inspired by such a great ability, it would be promising to integrate GAN into generative clustering models. Specifically, Ben-Yosef et al. [119] proposed to stack a GMM with a GAN, where a GMM serves as a prior distribution for generating data instances. Formally, they optimized the following objective function as Equation (21):

$$\min_G \max_D V(D,G) = \mathop{\mathbb{E}}_{\mathbf{x}\sim p_{\mathcal{X}}(\mathbf{x})}[\log D(\mathbf{x})] + \mathop{\mathbb{E}}_{\mathbf{z}\sim p_{\mathcal{Z}}(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))], \tag{21}$$

where $p_{\mathcal{X}}(\mathbf{x})$ denotes the training data distribution; $p_{\mathcal{Z}}(\mathbf{z})$ is a prior distribution of G and defined as mixture of Gaussians as Equation (22):

$$p_{\mathcal{Z}}(\mathbf{z}) = \sum_{k=1}^{K} \pi_k * \mathcal{N}\left(\mu_c, \sigma_c^2 \mathbf{I}\right). \tag{22}$$

By equipping GAN with such a multi-modal probability distribution, the model could provide a better fit to the complex data distribution especially when the data includes many different clusters.

There are also some improved variants. For example, Yu et al. [120] proposed to directly replace the Gaussian distribution of the GMM with a GAN and developed a ϵ-expectation- maximization learning algorithm to forbid early convergence issues; Ntelemis et al. [172] proposed to employ Sobel operations prior to the discriminator of the GAN; Mukherjee et al. [118] proposed to sample the latent vector z from a mixture of one-hot encoded variables and continuous latent variables. An inverse network with clustering-specific loss is introduced to make the model more friendly to the clustering task. Analogously, an inverse network is introduced in [121, 173] for the feature-level (i.e., latent vector) adversary.

Summary. Although deep generative clustering models can generate samples while completing clustering, they also have some weaknesses: (1) Training a generative model usually involves Monte Carlo sampling, which may incur training unstable and high computational complexity; (2) The mainstream generative models are based on VAE and GAN, and inevitably inherit the same disadvantages of them. VAE-based models usually require prior assumptions on the data distributions, which may not be held in real cases; although GAN-based algorithms are more flexible and diverse, they usually suffer from mode collapse and slow convergence, especially for the data with multiple clusters.

5.4 Simultaneous Deep Clustering

In contrast to generative deep clustering methods that strengthen the connection between representation learning and clustering through probability distributions, a more effective technique to leverage the advantages of deep learning is to optimize both modules simultaneously under a single optimization goal. As a result, simultaneous deep clustering has developed rapidly and has achieved significant success in recent times. Although most iterative deep clustering methods also optimize both two modules with a single objective, the two modules are optimized explicitly iteratively and cannot be updated simultaneously. In this subsection, we introduce the representative architectures of simultaneously deep clustering.

5.4.1 Auto-Encoder with Self-Training. Auto-encoder is a powerful tool to learn data representations in an unsupervised way, and has been utilized since the first attempts of simultaneously deep clustering [112].

The representative method is DEC [102] which combines the auto-encoder with the self-training strategy. This simple but effective strategy has deeply influenced the follow-up works.

The auto-encoder is pre-trained, and only the encoder is utilized as the initialization of the representation learning module. The self-training strategy mentioned in Section 4.3 is then introduced to optimize clustering and representation learning simultaneously.

Based on the vanilla DEC method, many variants and improvements are proposed. To preserve the local structure of each instance, IDEC [103] further integrates the reconstruction loss into the autoencoder. The general formulation of auto-encoder with self-training can be summarized as Equation (23):

$$\mathcal{L}_{AEST} = \mathcal{L}_{AE} + \mathcal{L}_{ST}, \tag{23}$$

where $\mathcal{L}_{AE}$ is the loss of auto-encoder and $\mathcal{L}_{ST}$ is the loss of clustering oriented self-training, e.g., the neighborhood constraint in DEC. To improve the capability of auto-encoder, some efforts are made to adapt different data types. In [104–106], the linear layer of auto-encoder is replaced with fully convolutional layers so that the image feature can be captured well. In CCNN [130], the clustering convolutional neural network is proposed as a new backbone to extract the representations that are friendly to the clustering task. In DEPICT [105], an additional noisy encoder is introduced, and the robustness of the auto-encoder is improved by minimizing the reconstruction error of every layer between the noisy decoder and the clean encoder.

Although the self-training strategy has achieved success, later work also makes attempts to solve a specific problem. To increase the robustness of the clustering, self-training is applied between two branches: clean and augmented [104] (noisy [105]). Specifically, the target distribution $\mathbb{P}$ is computed using the clean branch to guide the soft assignments $\mathbb{Q}$ of the augmented or noisy branch. The self-training strategy can be combined with subspace clustering. CSC [107] introduces the assumption named invariance of distribution, i.e., the target distribution $\mathbb{P}$ should be invariant to different distance metrics in subspace space. Therefore, two metrics (Euclidean and Cosine distance) are used to calculate the target distributions $\mathbb{P}_E$ and $\mathbb{P}_C$, with the KL divergence between them minimized.

According to the analysis aforementioned, self-training is similar to the K-means clustering and suffers from the unbalance problem [174] between different clusters. To solve the problem of unbalanced data and out-of-distribution samples, StatDEC [9] improves the target distribution by adding the normalized instance frequency of the clusters. In this way, the model can preserve discrimination of small groups and form a local clustering boundary which is insensitive to unbalanced clusters.

5.4.2 Mutual Information Maximization Based Clustering. As illustrated in Section 3.3 the mutual information maximization has been successfully applied in both representation and clustering. The unified form of mutual information in both modules has provided convenience for understanding and implementation.

The representative mutual information maximization based clustering method is DCCM [132]. For each data instance, the mutual information between the deep and shallow layer representations is maximized so that the consistency of the representation can be guaranteed. This consistency is further extended to the cluster assignment space by encouraging instances with the same pseudo-labels to share similar representations.

Many later works can be regarded as variants of this method. In VCAMI [175] and IIC [133], the **augmented mutual information** (**AMI**) is introduced to improve robustness. Such an augmentation invariant has inspired the later contrastive deep clustering methods, which will be introduced in the next subsection. In ImC-SWAV [176], the mutual information between the integrated discrete representation and a discrete probability distribution is maximized, which improves the vanilla SWAV [177] method.

5.4.3 Contrastive Deep Clustering. Similar to mutual information maximization based deep clustering, contrastive learning has also been successfully applied in both the representation learning module and the clustering module. The main idea of contrastive learning is pulling similar instances closer while pushing different instances away, which is in the spirit of clustering that instances from the same cluster should be close while instances from different clusters should be apart.

The representative contrastive deep clustering method is CC [134]. The basic idea is to treat each cluster as the data instance in the low-dimensional space. The instance discrimination task in contrastive representation learning can be migrated to the clustering task by discriminating different clusters, which is the fundamental requirement of clustering. Furthermore, the advantage of augmentation invariant and local robustness can be preserved in the clustering task.

Taking CC as the fundamental architecture, many contrastive deep clustering can be viewed as variants of it. PICA [131] can be viewed as the degeneration of CC without augmentation, it directly separates different clusters by minimizing the cosine similarity between the cluster assignment statistic vectors. In DCCM [132], the augmentation is introduced to guarantee the local robustness of the learned representation. DRC [137] has the same contrastive learning as CC where the cluster representation is called assignment probability. The difference lies in the regularization of the cluster, which is inspired by group lasso [178]. In CRLC [179], contrastive learning is performed between the cluster assignments of two augmented versions of the same instance, rather than the cluster representations. Also, the dot product in contrastive learning is replaced by the log-dot product, which is more suitable for probabilistic contrastive learning. SCCL [136] extends this approach with textual data augmentation, which proves that this contrastive learning based self-training framework is universally applicable.

The later works further adopt the contrastive clustering in the semantic space. In SCL [180], the negative samples are limited by different pseudo labels, so that instances from different clusters can be further distinguished. MiCE [181] follows a divide-and-conquer strategy in which the gating function divides the whole dataset into clusters and the experts in each cluster aim at discriminating the instances in the cluster. However, compared with InfoNCE which implicitly models alignment and uniformity implicitly [45], MiCE models these two properties in a more explicit way. Recently, TCC [182] further improved efficiency with the reparametrization trick and explicitly improved the discriminability of the cluster.

Some other methods attempt to overcome the problems of vanilla contrastive learning. In GCC [139], the positive pair and negative pair are selected by the KNN graph constructed on the instance representation. This may be related to overcoming the "false negative" problem in contrastive learning. In NCC [183], the contrastive learning module is replaced from SimCLR to BYOL [184], so that the overreliance on negative samples can be solved.

5.4.4 Hybrid Simultaneous Deep Clustering. The aforementioned simultaneous deep clustering methods have remarkable characteristics and advantages, some other works are hybrids of the above techniques. SCCL [136] and Sundareswaran et al. [185] combine contrastive representation learning and self-training. DDC [186] and RCC [187] combine relation matching clustering and pseudo labeling to boost the clustering performance. DCC [188] combines autoencoder-based representation and relation matching clustering. The auto-encoder based representation learning and spectral clustering are combined in [117] by incorporating the augmentations from the contrastive learning.

Summary. The simultaneous deep clustering [128, 129, 135, 138, 189–198] has attracted the most attention for its unified optimization. Intuitively, the learned representation is clustering oriented, and the clustering is conducted on the discriminative space. However, it may arise from

an undesired prejudice of the optimization focus between the representation learning module and the clustering module, which can only be mitigated by manually setting the balanced parameter for now. Also, the model is easy to sink into degenerate solutions where all instances are assigned to a single cluster.

6 Datasets and Evaluation Metrics

In this section, we introduce benchmark datasets and evaluation metrics that are widely used in existing deep clustering methods.

6.1 Datasets

6.1.1 Image Datasets. Image is the most commonly used data type in real-world deep clustering. The early attempts of deep clustering are applied to image datasets including COIL-20,[3] CMU PIE,[4] Yale-B,[5] MNIST,[6] CIFAR[7] and STL-10.[8] Recently, efforts has been paid to perform clustering on large volume vision datasets (e.g., ImageNet). Although existing methods have achieved promising performance on ImageNet-10 and ImageNet-Dogs dataset, clustering on Tiny-ImageNet (200 clusters) or full-size ImageNet is still challenging.

6.1.2 Textual Datasets. The widely used textual datasets in early applications of textual data clustering include Reuters-21578[9] and 20 Newsgroups[10] datasets, which have already been vectorized and little feature engineering is needed. Currently, raw textual datasets including IMDB,[11] stackOverflow,[12] and more in nlp-datasets github repository[13] is still challenging for deep textual clustering.

6.1.3 Video Datasets. The ultimate task of video clustering varies from action classification [76] to video anomaly detection [153]. Kinetics-400 and Kinetics-600[14] are two of the most famous video datasets. The others include UCF-101 dataset[15] and HMDB-51 dataset.[16]

6.1.4 Graph Datasets. Commonly used graph datasets for node clustering can be referred from the following papers [149, 150, 199] and Stanford Network Analysis Project.[17] And there are also graph-level classification datasets like PROTINS [90] and MUTAG [200], which can be used to perform and evaluate graph-level clustering.

6.2 Evaluation Metrics

Evaluation metrics aim at evaluating the validity of methods. In the field of Deep Clustering, three standard clustering performance metrics are widely used: Accuracy(ACC), **Normalized Mutual Information** (**NMI**) and **Adjusted Rand Index** (**ARI**).

[3] http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php
[4] http://www.cs.cmu.edu/afs/cs/project/PIE/MultiPie/Multi-Pie/Home.html
[5] http://vision.ucsd.edu/~leekc/ExtYaleDatabase/Yale%20Face%20Database.htm
[6] http://yann.lecun.com/exdb/mnist/index.html
[7] http://www.cs.toronto.edu/~kriz/cifar.html
[8] https://cs.stanford.edu/~acoates/stl10/
[9] https://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection
[10] http://qwone.com/~jason/20Newsgroups/
[11] http://ai.stanford.edu/~amaas/data/sentiment/
[12] https://github.com/jacoxu/StackOverflow
[13] https://github.com/niderhoff/nlp-datasets
[14] https://www.deepmind.com/open-source/kinetics
[15] https://www.crcv.ucf.edu/research/data-sets/ucf101/
[16] https://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/#dataset
[17] http://snap.stanford.edu/

6.2.1 Accuracy. ACC indicates the average correct classification rate of clustering samples. Given the ground truth labels $Y = \{y_i | 1 \leq i \leq N\}$ and the predicted hard assignments $\tilde{Y} = \{\tilde{y}_i | 1 \leq i \leq N\}$, ACC can be computed as Equation (24):

$$ACC(\tilde{Y}, Y) = \max_{g} \frac{1}{N} \sum_{i=1}^{N} \mathbf{1} \{y_i = g(\tilde{y}_i)\}, \tag{24}$$

where g is the set of all possible one-to-one mappings between the predicted labels and ground truth labels. The optimal mapping can be efficiently obtained by the Hungarian algorithm [201].

6.2.2 Normalized Mutual Information. NMI quantifies the mutual information between the predicted labels and ground truth labels into $[0, 1]$ as Equation (25):

$$NMI(\tilde{Y}, Y) = \frac{\mathcal{I}(\tilde{Y}; Y)}{\frac{1}{2}\left[H(\tilde{Y}) + H(Y)\right]}, \tag{25}$$

where $H(Y)$ is the entropy of Y and $\mathcal{I}(\tilde{Y}; Y)$ is the mutual information between $\tilde{Y}$ and Y.

6.2.3 Adjusted Rand Index. ARI comes from the **Rand Index** (**RI**), which regards the clustering result as a series of pair-wise decisions and measures it according to the rate of correct decisions as Equation (26):

$$RI = \frac{TP + TN}{C_N^2}, \tag{26}$$

where TP and TN denote the number of true positive pairs and true negative pairs, C_N^2 is the number of possible sample pairs. However, the RI value of two random partitions is not a constant approaching 0, thus ARI was introduced as Equation (27):

$$ARI = \frac{RI - \mathbb{E}(RI)}{\max(RI) - \mathbb{E}(RI)}. \tag{27}$$

Both ACC and NMI $\in [0, 1]$ while ARI $\in [-1, 1]$, in which higher values indicate better performance.

7 Applications

Despite the success of deep clustering in mining global patterns among instances, it has also benefited various downstream tasks. In this section, we discuss some typical applications of deep clustering.

7.1 Community Detection

Community Detection [6, 202, 203] aim at partitioning the graph network into several subgraphs mainly according to connection density, which can be treated as a node-level graph clustering task. Early works are mainly based on modularity measurement [204, 205], maximum flows [206], graph cut [207] and its extension, spectral methods [208]. With the development of GNNs [209, 210], nodes are represented as individual instances in the low-dimensional space. As a result, the border between modern community detection [5] and graph clustering [211–213] is gradually blurring, and GNN-based graph clustering [214, 215] have already been applied to many applications. However, unlike early community detection, which focuses on the network topology, graph clustering usually incorporates the node attributes and other side information. How to release the power of GNNs while preserving the topology characteristic is still under study.

7.2 Anomaly Detection

Anomaly detection (a.k.a. Outlier Detection, Novelty Detection) is a technique for identifying abnormal instances or patterns among data. Early before deep clustering, density-based clustering methods [216–218] have specifically mentioned and addressed the problem of noise during clustering, which has enlightened a group of density-based anomaly detection methods [219, 220]. The later anomaly detection methods [153, 221–223] have utilized the clustering results and identified anomaly instances as those far from the cluster centroids or the border of each cluster. Currently, deep clustering has shown great potential in forming a better clustering space for anomaly detection. Instead of performing anomaly detection after deep clustering, recent efforts have been put into conducting them in a unified framework: identify and remove anomaly instances to reduce the impact on clustering [224], and anomaly detection can be further improved with better clustering results.

7.3 Segmentation and Object Detection

Image Segmentation is one of the most important approaches to simulating human understanding of images that aim at dividing pixels into disjoint regions. Generally speaking, image segmentation is performing pixel classification in a supervised manner and pixel clustering in an unsupervised manner [225, 226]. Currently, deep clustering has been successfully applied in segmentation using clustered regions to generate a Scene Graph [227]. Yi et al. [228] surveyed graph cut based image segmentation, where graph cut is one of the most fundamental solutions to perform clustering (Spectral Clustering). 3D clustering can be a solution to 3D Object Detection, like in [229], 3D points were clustered to represent an object with geometric consistency. But such clustering based segmentation and object detection have no guarantee for small regions and objects, where the expected clustering result is highly unbalanced. The global positional information of a pixel may be ignored when clustering is performed.

7.4 Medical Applications

The convolutional neural network has successfully promoted the development of medical image processing in a supervised manner. However, the manual dataset labeling process is often labor intensive and requires expert medical knowledge [123], which is hard to realize in real-world scenarios. Recently, deep clustering has been introduced to automatically categorize large-scale medical images [123]. Mittal et al. [160] introduce medical image clustering analysis for a faster diagnosis of COVID-19. In the field of biological sciences, **single-cell RNA sequencing** (**scRNA-seq**) [230] provides a cell gene matrix in output for the analysis of cell population and behavior and even new cell discovery. For this purpose, ScDeepCluster [129] and ItClust [189] develop their models based on DEC [102] to cluster scRNA-seq data and MARS [231] that combine transfer learning and clustering to discover novel cell types. More applications can be found in the gene data clustering area [232].

Beyond the successful applications of deep clustering previously mentioned, clustering holds significant promise in various other fields such as financial analysis [233–235], trajectory analysis [236, 237], and understanding social media [238–240]. Although many of these methods have not yet adopted deep learning techniques, given the growing volumes and complexity of data, deep clustering is expected to significantly influence these areas.

8 Future Directions

In this section, we conduct some future directions of deep clustering based on the above cornerstone, taxonomy, and real-world applications.

8.1 Initialization of Deep Clustering Module

The initialization of deep neural networks usually plays an important role in the efficiency and stability of training [241]. This is more critical in deep clustering, where both the representation learning module and the clustering module are modeled by deep neural networks. Recently, model pre-training [242] has been a popular network initialization technique that has also been introduced to deep clustering [95]. However, the pre-training based initialization is appropriate for the representation learning module but has not been well studied for the clustering module. Although there have been some initialization schemes on shallow clustering [243], the initialization for the cluster module with deep neural networks is still under investigation.

8.2 Overlapping Deep Clustering

The deep clustering methods discussed in this article largely focus on partitioning clustering where each instance belongs to a single cluster. Meanwhile, in the real-world scenario, each instance may belong to multiple clusters, e.g., users in a social network [244] may belong to several communities, and video/audio on social media may have several tags [245]. Among the deep clustering methods discussed in this article, if the clustering constraints are conducted on the cluster assignment probabilistic matrix, they can be directly adapted to the overlapping clustering setting. However, if the training relies on the pseudo hard label of data instances, they may fail in the overlapping clustering setting. Although multi-label classification has been widely studied in the literature [246, 247], how to adapt to unsupervised clustering is still an open research question.

8.3 Boosting Representation with Deep Clustering

Throughout this article, we can find that a good representation is essential for clustering. Although clustering-friendly representation learning has been studied in the literature, they are designed for the specific shallow clustering method. In contrast, the clustering structure denotes the high-order pattern of the dataset, which should be preserved in the comprehensive representation [248]. The deep clustering methods discussed in this article focus on how to incorporate representation learning to boost the clustering, meanwhile, how to in turn boost the representation learning by the clustering remains to be studied.

8.4 Deep Clustering Explanation

As an unsupervised task, the clustering process usually lacks human priors such as label semantics, number of clusters [249], which makes the clustering results difficult to explain or understand. Some methods [250] have already combined the tags provided by the users to increase the explanation of the clustering results. However, it relies on accurate human tagging, which may not be realized in practice. With the development of causal inference in deep learning, the explanation of clustering among instances will hopefully be improved. Both the research and industry community are expecting a generalized explanation framework for clustering, especially on high-dimensional data. To conclude, how to utilize the casual inference techniques in clustering is of great importance and deserves more attention.

8.5 Transfer Learning with Deep Clustering

Transfer learning [251] aims to bridge the gap between training and test datasets with a distribution shift. The general idea is to transfer the knowledge from the known data to the unknown test data. Recently, deep clustering has played an increasingly important role in unsupervised transfer learning [3, 4, 252–254], where the target domain is unsupervised. For example, ItClust [189] and MARS [231] have achieved success in scRNA-seq clustering (Section 7.4), AD-Cluster [255] that

use clustering has improved domain adaptive person re-identification. In contrast, unsupervised transfer learning methods can also benefit deep clustering. Taking UCDS [256] as an example, the unsupervised domain adaptation is used to perform clustering among variant domains. The distribution shift is one of the key factors that affect the performance of machine learning models, including deep clustering. The performance of cluster analysis can give further understanding of the unsupervised target domain, but how to transfer the clustering result to knowledge and effectively minimize the distribution shift based on clustering results can be further explored.

8.6 Clustering with Anomalies

In Section 7.2, we have discussed the applications of deep clustering in anomaly detection where the instances are well clustered. Concerning the existence of anomaly instances in the dataset, the clustering may also be influenced and mutually restricted, since most existing deep clustering methods have no specific response to the outliers. The classic K-means method is known to be sensitive to outliers, although there have been a few works on overcoming this problem [224], they are designed for shallow clustering methods. To this end, how to improve the robustness of deep clustering for anomaly instances and gradually improve the clustering performance by reducing the detected anomaly instances is still an open research question.

8.7 Degenerate Solution versus Unbalanced Data

The issue of degenerate solutions [99] has emerged as a significant challenge in deep clustering, characterized by the potential assignment of all instances to a single cluster. Numerous deep clustering techniques have implemented additional restrictions to address this issue [95, 99, 112, 131, 133, 134, 257], with the entropy of the cluster size distribution being the most widely adopted strategy. Through maximization of entropy, it is anticipated that instances will be distributed uniformly across each cluster, thereby preventing a collapse into a single cluster. Importantly, the effectiveness of this approach depends on a uniform distribution of actual labels, a condition met by standard datasets such as CIFAR10 and CIFAR100. However, in practical environments, this assumption may be too rigid as many datasets exhibit unbalance or have a long-tailed distribution [258]. The conflicting goals of achieving uniform distribution and handling unbalanced data can significantly undermine the effectiveness of deep clustering. Recently, the focus on unbalanced data clustering [259, 260] has intensified, which is expected to improve the effectiveness of clustering in practical scenarios.

8.8 Efficient Training versus Global Modeling

To improve training efficiency and scalability, most existing deep clustering methods have utilized the mini-batch training strategy, where instances are separated into batches and the model is updated with each batch. This is suitable for the task where the instances are independent on each other, such as classification and regression. However, as deep clustering heavily relies on the complicated relationship among instances, such mini-batch training may lose the ability of global modeling. Although some existing methods have used cluster representations or prototypes [248] to store global information, it is still worth studying how to balance training efficiency and model capability.

9 Conclusion

In this review, we provide a detailed and current summary of the deep clustering research area. We initially outline the foundational elements of deep clustering, specifically the representation learning and clustering modules, along with their typical configurations. We categorize the existing approaches into a taxonomy based on the interaction types between the representation learning

and clustering modules, specifically: multi-stage deep clustering, iterative deep clustering, generative deep clustering, and simultaneous deep clustering. Subsequently, we compile the standard datasets, evaluation metrics, and uses of deep clustering. Finally, we explore prospective future trends in deep clustering that offer promising opportunities.

References

[1] Hossein Saeedi Emadi and Sayyed Majid Mazinani. 2018. A novel anomaly detection algorithm using DBSCAN and SVM in wireless sensor networks. *Wireless Personal Communications* 98, 2 (2018), 2025–2035.

[2] S. Wibisono, M. T. Anwar, A. Supriyanto, and IHA Amin. 2021. Multivariate weather anomaly detection using DBSCAN clustering algorithm. In *Journal of Physics: Conference Series*, Vol. 1869. IOP Publishing, 012077.

[3] Hui Tang, Ke Chen, and Kui Jia. 2020. Unsupervised domain adaptation via structurally regularized deep clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8725–8735.

[4] Qiang Zhou, Wen'an Zhou, and Shirui Wang. 2021. Cluster adaptation networks for unsupervised domain adaptation. *Image and Vision Computing* 108 (April 2021). DOI: https://doi.org/10.1016/j.imavis.2021.104137

[5] Xing Su, Shan Xue, Fanzhen Liu, Jia Wu, Jian Yang, Chuan Zhou, Wenbin Hu, Cecile Paris, Surya Nepal, Di Jin, Quan Z. Sheng, and Philip S. Yu. 2024. A Comprehensive survey on community detection with deep learning. *IEEE Transactions on Neural Networks and Learning Systems* 35, 4 (April 2024), 4682–4702. DOI: https://doi.org/10.1109/TNNLS.2021.3137396

[6] Fanzhen Liu, Shan Xue, Jia Wu, Chuan Zhou, Wenbin Hu, Cecile Paris, Surya Nepal, Jian Yang, and Philip S Yu. 2020. Deep learning for community detection: progress, challenges and opportunities. arXiv:2005.08225. Retrieved from https://arxiv.org/abs/2005.08225

[7] Jianwei Yang, Devi Parikh, and Dhruv Batra. 2016. Joint unsupervised learning of deep representations and image clusters. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5147–5156.

[8] Yu Meng, Yunyi Zhang, Jiaxin Huang, Yu Zhang, Chao Zhang, and Jiawei Han. 2020. Hierarchical topic mining via joint spherical tree and text embedding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1908–1917.

[9] Mina Rezaei, Emilio Dorigatti, David Ruegamer, and Bernd Bischl. 2021. Learning statistical representation with joint deep embedded clustering. *arXiv preprint arXiv:2109.05232*, 1 (2021).

[10] Richard Bellman. 1966. Dynamic programming. *Science* 153, 3731 (1966), 34–37.

[11] Itziar Frades and Rune Matthiesen. 2010. Overview on techniques in cluster analysis. In *Bioinformatics Methods in Clinical Research*, Rune Matthiesen (Ed.). Humana Press, Totowa, NJ, 81–107. DOI: https://doi.org/10.1007/978-1-60327-194-3_5

[12] Dongkuan Xu and Yingjie Tian. 2015. A comprehensive survey of clustering algorithms. *Annals of Data Science* 2, 2 (2015), 165–193.

[13] Anil K. Jain, M. Narasimha Murty, and Patrick J. Flynn. 1999. Data clustering: A review. *ACM Computing Surveys (CSUR)* 31, 3 (1999), 264–323.

[14] Pavel Berkhin. 2006. A survey of clustering data mining techniques. In *Grouping Multidimensional Data*. Springer, 25–71.

[15] Hui Yin, Amir Aryani, Stephen Petrie, Aishwarya Nambissan, Aland Astudillo, and Shengyuan Cao. 2024. A rapid review of clustering algorithms. *arXiv preprint arXiv:2401.07389* (2024).

[16] Erxue Min, Xifeng Guo, Qiang Liu, Gen Zhang, Jianjing Cui, and Jun Long. 2018. A survey of clustering with deep learning: From the perspective of network architecture. *IEEE Access* 6 (2018), 39501–39514. DOI: https://doi.org/10.1109/ACCESS.2018.2855437

[17] Elie Aljalbout, Vladimir Golkov, Yawar Siddiqui, Maximilian Strobel, and Daniel Cremers. 2018. Clustering with deep learning: Taxonomy and new methods. *arXiv preprint arXiv:1801.07648* (2018).

[18] Gopi Chand Nutakki, Behnoush Abdollahi, Wenlong Sun, and Olfa Nasraoui. 2019. An introduction to deep clustering. In *Clustering Methods for Big Data Analytics*. Springer, 73–89.

[19] Rui Xu and Donald Wunsch. 2005. Survey of clustering algorithms. *IEEE Transactions on Neural Networks* 16, 3 (2005), 645–678.

[20] Artúr István Károly, Róbert Fullér, and Péter Galambos. 2018. Unsupervised clustering for deep learning: A tutorial survey. *Acta Polytechnica Hungarica* 15, 8 (2018), 29–53.

[21] Yazhou Ren, Jingyu Pu, Zhimeng Yang, Jie Xu, Guofeng Li, Xiaorong Pu, Philip S. Yu, and Lifang He. 2024. Deep Clustering: A comprehensive survey. *IEEE Transactions on Neural Networks and Learning Systems* (July 2024). DOI: https://doi.org/10.1109/TNNLS.2024.3403155

[22] Yue Liu, Jun Xia, Sihang Zhou, Xihong Yang, Ke Liang, Chenchen Fan, Yan Zhuang, Stan Z Li, Xinwang Liu, and Kunlun He. 2022. A survey of deep graph clustering: Taxonomy, challenge, application, and open Resource. *arXiv preprint arXiv:2211.12875* (2022).
[23] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 8 (2013), 1798–1828.
[24] Jingya Zhou, Ling Liu, Wenqi Wei, and Jianxi Fan. 2022. Network representation learning: From preprocessing, feature extraction to node embedding. *ACM Computing Surveys (CSUR)* 55, 2 (2022), 1–35.
[25] Muhammad Usman, Mian Ahmad Jan, Xiangjian He, and Jinjun Chen. 2019. A survey on representation learning efforts in cybersecurity domain. *ACM Computing Surveys (CSUR)* 52, 6 (2019), 1–28.
[26] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. 1985. *Learning Internal Representations by Error Propagation.* Technical Report. California Univ San Diego La Jolla Inst for Cognitive Science.
[27] Junhai Zhai, Sufang Zhang, Junfen Chen, and Qiang He. 2018. Autoencoder and its various variants. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC'18)*. IEEE, 415–419.
[28] Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. 2018. Adversarially regularized graph autoencoder for graph embedding. In *Procesdings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, Int Joint Conf Artifical Intelligence.
[29] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. 2015. Unsupervised learning of video representations using lstms. In *International Conference on Machine Learning*. PMLR, 843–852.
[30] Diederik P. Kingma. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).
[31] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.
[32] Malvin H. Kalos and Paula A. Whitlock. 2009. *Monte Carlo Methods*. John Wiley & Sons.
[33] Justin B. Kinney and Gurinder S. Atwal. 2014. Equitability, mutual information, and the maximal information coefficient. *Proceedings of the National Academy of Sciences* 111, 9 (2014), 3354–3359.
[34] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. 2004. Estimating mutual information. *Physical Review E* 69, 6 (2004), 066138.
[35] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. 2018. Mutual information neural estimation. In *International Conference on Machine Learning*. PMLR, 531–540.
[36] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. 2016. f-gan: Training generative neural samplers using variational divergence minimization. *Advances in Neural Information Processing Systems* 29 (2016).
[37] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).
[38] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2018. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670* (2018).
[39] Philip Bachman, R. Devon Hjelm, and William Buchwalter. 2019. Learning Representations by Maximizing Mutual Information Across Views. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2019/file/ddf354219aac374f1d40b7e760ee5bb7-Paper.pdf
[40] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the Knowledge in a Neural Network. *CoRR* abs/1503.02531 (2015). Retrieved from http://arxiv.org/abs/1503.02531
[41] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*. PMLR, 1597–1607.
[42] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph Contrastive Learning with Augmentations. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 5812–5823. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2020/file/3fe230348e9a12c13120749e3f9fa4cd-Paper.pdf
[43] Haoyang Li, Xin Wang, Ziwei Zhang, Zehuan Yuan, Hang Li, and Wenwu Zhu. 2021. Disentangled contrastive learning on graphs. In *Advances in Neural Information Processing Systems* 34 (2021).
[44] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9729–9738.
[45] Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research)*, PMLR, 9929–9939. Retrieved from http://proceedings.mlr.press/v119/wang20k.html

[46] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in Neural Information Processing Systems* 33 (2020), 18661–18673.
[47] Yifei Wang, Qi Zhang, Yisen Wang, Jiansheng Yang, and Zhouchen Lin. 2021. Chaos is a ladder: A new understanding of contrastive learning. In *International Conference on Learning Representations*.
[48] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE Transactions on Information Theory* 28, 2 (1982), 129–137.
[49] Bo Yang, Xiao Fu, Nicholas D. Sidiropoulos, and Mingyi Hong. 2017. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *International Conference on Machine Learning*. PMLR, 3861–3870.
[50] Andrew Ng, Michael Jordan, and Yair Weiss. 2001. On Spectral Clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, MIT Press. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2001/file/801272ee79cfde7fa5960571fee36b9b-Paper.pdf
[51] Yaling Tao, Kentaro Takagi, and Kouta Nakata. 2021. Clustering-friendly representation learning via instance discrimination and feature decorrelation. In *International Conference on Learning Representations*.
[52] Ehsan Elhamifar René Vidal. 2009. Sparse subspace clustering. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 00*, Vol. 6. 2790–2797.
[53] Junjian Zhang, Chun-Guang Li, Chong You, Xianbiao Qi, Honggang Zhang, Jun Guo, and Zhouchen Lin. 2019. Self-supervised convolutional subspace clustering network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5473–5482.
[54] Tong Zhang, Pan Ji, Mehrtash Harandi, Wenbing Huang, and Hongdong Li. 2019. Neural collaborative subspace clustering. In *International Conference on Machine Learning*. PMLR, 7384–7393.
[55] Pan Ji, Tong Zhang, Hongdong Li, Mathieu Salzmann, and Ian Reid. 2017. Deep subspace clustering networks. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2017/file/e369853df766fa44e1ed0ff613f563bd-Paper.pdf
[56] Shangzhi Zhang, Chong You, René Vidal, and Chun-Guang Li. 2021. Learning a self-expressive network for subspace clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12393–12403.
[57] Xi Peng, Shijie Xiao, Jiashi Feng, Wei-Yun Yau, and Zhang Yi. 2016. Deep subspace clustering with sparsity prior. In *IJCAI*. 1925–1931.
[58] Pan Zhou, Yunqing Hou, and Jiashi Feng. 2018. Deep adversarial subspace clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1596–1604.
[59] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.
[60] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.
[61] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*. Retrieved from https://openreview.net/forum?id=YicbFdNTTy
[62] Michael Steinbach. 2000. *A Comparison of Document Clustering Techniques*. Technical Report# 00_034/University of Minnesota.
[63] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*. PMLR, 1188–1196.
[64] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*. Retrieved from http://arxiv.org/abs/1301.3781
[65] Thomas Hofmann and others. 1999. Probabilistic latent semantic analysis. In *UAI*, 289–296.
[66] Robert Dzisevič and Dmitrij Šešok. 2019. Text classification using different feature extraction approaches. In *2019 Open Conference of Electrical, Electronic and Information Sciences (eStream'19)*. IEEE, 1–4.
[67] K Rajendra Prasad, Moulana Mohammed, and RM Noorullah. 2019. Hybrid topic cluster models for social healthcare data. *Int J Adv Comput Sci Appl* 10, 11 (2019), 490–506.
[68] Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James Glass. 2019. An unsupervised autoregressive model for speech representation learning. In *INTERSPEECH 2019* (Interspeech), 146–150. DOI: https://doi.org/10.21437/Interspeech.2019-1473
[69] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *2019 Conference of The North Amercian Chapter of the Association for Computational Linguistics: Hhuman Language Technologies (NAACL HLT 2019)*, VOL. 1, Assoc Computat Linguist, N Amer Chapter; Comp Res Assoc, Comp Community Consortium; Natl Sci Fdn; Natl Rees Council Canada; Google, 4171–4186.

[70] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 1877–1901. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[71] Jiong Zhang, Wei-cheng Chang, Hsiang-fu Yu, and Inderjit Dhillon. 2021. Fast multi-resolution transformer fine-tuning for extreme multi-label text classification. *Advances in Neural Information Processing Systems* 34 (2021).

[72] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

[73] Yuwei Zhang, Zihan Wang, and Jingbo Shang. 2023. Clusterllm: Large language models as a guide for text clustering. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

[74] Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. 2020. Self-supervised learning by cross-modal audio-video clustering. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 9758–9770. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2020/file/6f2268bd1d3d3ebaabb04d6b5d099425-Paper.pdf

[75] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. 2021. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6964–6974.

[76] Bo Peng, Jianjun Lei, Huazhu Fu, Yalong Jia, Zongqian Zhang, and Yi Li. 2021. Deep video action clustering via spatio-temporal feature learning. *Neurocomputing* 456 (2021), 519–527. DOI: https://doi.org/10.1016/j.neucom.2020.05.123

[77] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. 2021. A large-scale study on unsupervised spatiotemporal representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3299–3309.

[78] Qin Zhang, Jia Wu, Hong Yang, Yingjie Tian, and Chengqi Zhang. 2016. Unsupervised feature learning from time series. In *IJCAI*. New York 2322–2328.

[79] Qin Zhang, Jia Wu, Peng Zhang, Guodong Long, and Chengqi Zhang. 2018. Salient subsequence learning for time series clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41, 9 (2018), 2193–2207.

[80] Sirine Taleb and Nadine Abbas. 2022. Hybrid machine learning classification and inference of stalling events in mobile videos. In *2022 4th IEEE Middle East and North Africa COMMunications Conference (MENACOMM'22)*. IEEE, 209–214.

[81] Ziwei Zhang, Peng Cui, and Wenwu Zhu. 2022. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering* 34, 1 (January 2022), 249–270. DOI: https://doi.org/10.1109/TKDE.2020.2981333

[82] Daixin Wang, Peng Cui, and Wenwu Zhu. 2016. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1225–1234.

[83] Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*. Retrieved from https://openreview.net/forum?id=SJU4ayYgl

[84] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2017/file/5dd9db5e033da9c6fb5ba83c7a7ebea9-Paper.p

[85] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*. Retrieved from https://openreview.net/forum?id=rJXMpikCZ

[86] Xin Wang, Shuyi Fan, Kun Kuang, and Wenwu Zhu. 2021. Explainable automated graph representation learning with hyperparameter importance. In *International Conference on Machine Learning*. PMLR, 10727–10737.

[87] Zhen Zhang, Jiajun Bu, Martin Ester, Jianfeng Zhang, Chengwei Yao, Zhao Li, and Can Wang. 2020. Learning temporal interaction graph embedding via coupled memory networks. In *Proceedings of the Web Conference 2020*. 3049–3055.

[88] Wenwu Zhu, Xin Wang, and Peng Cui. 2020. Deep learning for learning graph representations. In *Deep Learning: Concepts and Architectures*. Springer, 169–210.

[89] Sheng Zhou, Xin Wang, Jiajun Bu, Martin Ester, Pinggang Yu, Jiawei Chen, Qihao Shi, and Can Wang. 2020. DGE: Deep generative network embedding based on commonality and individuality. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 6949–6956.

[90] Karsten M. Borgwardt, Cheng Soon Ong, Stefan Schönauer, SVN Vishwanathan, Alex J. Smola, and Hans-Peter Kriegel. 2005. Protein function prediction via graph kernels. *Bioinformatics* 21, suppl_1 (2005), i47–i56.

[91] Zhen Zhang, Jiajun Bu, Martin Ester, Jianfeng Zhang, Zhao Li, Chengwei Yao, Huifen Dai, Zhi Yu, and Can Wang. 2023. Hierarchical multi-view graph pooling with structure learning. *IEEE Transactions on Knowledge and Data Engineering* 35, 1 (2023), 545–559. DOI: https://doi.org/10.1109/TKDE.2021.3090664

[92] Zhen Zhang, Hongxia Yang, Jiajun Bu, Sheng Zhou, Pinggang Yu, Jianwei Zhang, Martin Ester, and Can Wang. 2018. ANRL: Attributed network representation learning via deep neural networks.. In *Ijcai*, Vol. 18. 3155–3161.

[93] Jia Wu, Xingquan Zhu, Chengqi Zhang, and S Yu Philip. 2014. Bag constrained structure pattern mining for multi-graph classification. *IEEE Transactions on Knowledge and Data Engineering* 26, 10 (2014), 2382–2396.

[94] Ting Guo, Jia Wu, Xingquan Zhu, and Chengqi Zhang. 2017. Combining structured node content and topology information for networked graph clustering. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 11, 3 (2017), 1–29.

[95] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. 2020. Scan: Learning to classify images without labels. In *European Conference on Computer Vision*. Springer, 268–285.

[96] Zhiyuan Dang, Cheng Deng, Xu Yang, Kun Wei, and Heng Huang. 2021. Nearest neighbor matching for deep clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13693–13702.

[97] Dong-Hyun Lee and others. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, Atlanta, 896.

[98] Chuang Niu, Hongming Shan, and Ge Wang. 2022. Spice: Semantic pseudo-labeling for image clustering. *IEEE Transactions on Image Processing* 31 (2022), 7264–7278.

[99] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV'18)*. 132–149.

[100] Jianlong Chang, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. 2017. Deep adaptive image clustering. In *Proceedings of the IEEE International Conference on Computer Vision*. 5879–5887.

[101] Chuang Niu, Jun Zhang, Ge Wang, and Jimin Liang. 2020. Gatcluster: Self-supervised gaussian-attention network for image clustering. In *European Conference on Computer Vision*. Springer, 735–751.

[102] Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *International Conference on Machine Learning*. PMLR, 478–487.

[103] Xifeng Guo, Long Gao, Xinwang Liu, and Jianping Yin. 2017. Improved deep embedded clustering with local structure preservation.. In *Ijcai*. 1753–1759.

[104] Xifeng Guo, En Zhu, Xinwang Liu, and Jianping Yin. 2018. Deep embedded clustering with data augmentation. In *Asian Conference on Machine Learning*. PMLR, 550–565.

[105] Kamran Ghasedi Dizaji, Amirhossein Herandi, Cheng Deng, Weidong Cai, and Heng Huang. 2017. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *Proceedings of the IEEE International Conference on Computer Vision*. 5736–5745.

[106] Fengfu Li, Hong Qiao, and Bo Zhang. 2018. Discriminatively boosted image clustering with fully convolutional auto-encoders. *Pattern Recognition* 83 (2018), 161–173. DOI: https://doi.org/10.1016/j.patcog.2018.05.019

[107] Xi Peng, Jiashi Feng, Jiwen Lu, Wei-Yun Yau, and Zhang Yi. 2017. Cascade subspace clustering. In *Thirty-First AAAI Conference on Artificial Intelligence*.

[108] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, 11 (2008), 2579–2605.

[109] Longlong Jing and Yingli Tian. 2020. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43, 11 (2020), 4037–4058.

[110] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. 2020. A survey on contrastive self-supervised learning. *Technologies* 9, 1 (2020), 2.

[111] Fei Tian, Bin Gao, Qing Cui, Enhong Chen, and Tie-Yan Liu. 2014. Learning deep representations for graph clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 28.

[112] Peihao Huang, Yan Huang, Wei Wang, and Liang Wang. 2014. Deep embedding network for clustering. In *2014 22nd International Conference on Pattern Recognition*. IEEE, 1532–1537.

[113] Yuen-Hsien Tseng. 2010. Generic title labeling for clustered documents. *Expert Systems with Applications* 37, 3 (2010), 2247–2254.

[114] R. Janani and S. Vijayarani. 2019. Text document clustering using spectral clustering algorithm with particle swarm optimization. *Expert Systems with Applications* 134 (2019), 192–200. DOI: https://doi.org/10.1016/j.eswa.2019.05.030

[115] Nat Dilokthanakul, Pedro A. M. Mediano, Marta Garnelo, Matthew C. H. Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. 2017. Deep unsupervised clustering with gaussian mixture variational autoencoders. Retrieved from https://openreview.net/forum?id=SJx7Jrtgl

[116] Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. 2016. Variational deep embedding: An unsupervised and generative approach to clustering. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 1965–1972.

[117] Linxiao Yang, Ngai-Man Cheung, Jiaying Li, and Jun Fang. 2019. Deep clustering by gaussian mixture variational autoencoders with graph embedding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6440–6449.

[118] Sudipto Mukherjee, Himanshu Asnani, Eugene Lin, and Sreeram Kannan. 2019. Clustergan: Latent space clustering in generative adversarial networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 4610–4617.

[119] Matan Ben-Yosef and Daphna Weinshall. 2018. Gaussian mixture generative adversarial networks for diverse datasets, and the unsupervised clustering of images. *arXiv preprint arXiv:1808.10356* (2018).

[120] Yang Yu and Wen-Ji Zhou. 2018. Mixture of GANs for clustering. In *IJCAI*. 3047–3053.

[121] Kamran Ghasedi, Xiaoqian Wang, Cheng Deng, and Heng Huang. 2019. Balanced self-paced learning for generative adversarial clustering network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4391–4400.

[122] Yuting Jia, Qinqin Zhang, Weinan Zhang, and Xinbing Wang. 2019. Communitygan: Community detection with generative adversarial nets. In *The World Wide Web Conference*. 784–794.

[123] Turkay Kart, Wenjia Bai, Ben Glocker, and Daniel Rueckert. 2021. DeepMCAT: Large-scale deep clustering for medical image categorization. In *Deep Generative Models, and Data Augmentation, Labelling, and Imperfections*. Springer, 259–267.

[124] Jie Yan, Jing Liu, and Zhong-Yuan Zhang. 2024. Ccfc: Bridging federated clustering and contrastive learning. *arXiv preprint arXiv:2401.06634* (2024).

[125] Chunfeng Song, Feng Liu, Yongzhen Huang, Liang Wang, and Tieniu Tan. 2013. Auto-encoder based data clustering. In *Iberoamerican Congress on Pattern Recognition*. Springer, 117–124.

[126] Juncheng Lv, Zhao Kang, Xiao Lu, and Zenglin Xu. 2021. Pseudo-supervised deep subspace clustering. *IEEE Transactions on Image Processing* 30 (2021), 5252–5263. DOI: https://doi.org/10.1109/TIP.2021.3079800

[127] Chuyu Zhang, Hui Ren, and Xuming He. 2024. P2OT: Progressive partial optimal transport for deep imbalanced clustering. In *The Twelfth International Conference on Learning Representations*. Retrieved from https://openreview.net/forum?id=hD3sGVqPsr

[128] Deyu Bo, Xiao Wang, Chuan Shi, Meiqi Zhu, Emiao Lu, and Peng Cui. 2020. Structural deep clustering network. In *Proceedings of The Web Conference 2020*. 1400–1410.

[129] Tian Tian, Ji Wan, Qi Song, and Zhi Wei. 2019. Clustering single-cell RNA-seq data with a model-based deep learning approach. *Nature Machine Intelligence* 1, 4 (2019), 191–198.

[130] Chih-Chung Hsu and Chia-Wen Lin. 2017. Cnn-based joint clustering and representation learning with feature drift compensation for large-scale image data. *IEEE Transactions on Multimedia* 20, 2 (2017), 421–429.

[131] Jiabo Huang, Shaogang Gong, and Xiatian Zhu. 2020. Deep semantic clustering by partition confidence maximisation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8849–8858.

[132] Jianlong Wu, Keyu Long, Fei Wang, Chen Qian, Cheng Li, Zhouchen Lin, and Hongbin Zha. 2019. Deep comprehensive correlation mining for image clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8150–8159.

[133] Xu Ji, Joao F. Henriques, and Andrea Vedaldi. 2019. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9865–9874.

[134] Yunfan Li, Peng Hu, Zitao Liu, Dezhong Peng, Joey Tianyi Zhou, and Xi Peng. 2021. Contrastive clustering. In *2021 AAAI Conference on Artificial Intelligence (AAAI'21)*.

[135] Zhiyuan Dang, Cheng Deng, Xu Yang, and Heng Huang. 2021. Doubly contrastive deep clustering. *arXiv preprint arXiv:2103.05484* (2021).

[136] Dejiao Zhang, Feng Nan, Xiaokai Wei, Shangwen Li, Henghui Zhu, Kathleen McKeown, Ramesh Nallapati, Andrew Arnold, and Bing Xiang. 2021. Supporting clustering with contrastive learning. In *2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2021)*, Assoc Computat Linguist, 5419–5430.

[137] Huasong Zhong, Chong Chen, Zhongming Jin, and Xian-Sheng Hua. 2020. Deep robust clustering by contrastive learning. *arXiv preprint arXiv:2008.03030* (2020).

[138] Yuankun Xu, Dong Huang, Chang-Dong Wang, and Jian-Huang Lai. 2024. Deep image clustering with contrastive learning and multi-scale graph convolutional networks. *Pattern Recognition* 146 (February 2024). DOI: https://doi.org/10.1016/j.patcog.2023.110065

[139] Huasong Zhong, Jianlong Wu, Chong Chen, Jianqiang Huang, Minghua Deng, Liqiang Nie, Zhouchen Lin, and Xian-Sheng Hua. 2021. Graph contrastive clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9224–9233.

[140] Fabian Pedregosa, Gaeel Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research* 12 (October 2011), 2825–2830.

[141] Hsi-Cheng Chang and Chiun-Chieh Hsu. 2005. Using topic keyword clusters for automatic document clustering. *IEICE TRANSACTIONS on Information and Systems* 88, 8 (2005), 1852–1860.

[142] Weize Kong, Michael Bendersky, Marc Najork, Brandon Vargo, and Mike Colagrosso. 2020. Learning to cluster documents into workspaces using large scale activity logs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2416–2424.

[143] Tania Cerquitelli, Evelina Di Corso, Francesco Ventura, and Silvia Chiusano. 2017. Data miners' little helper: Data transformation activity cues for cluster analysis on document collections. In *Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics*. 1–6.

[144] Nabil Alami, Mohammed Meknassi, Noureddine En-nahnahi, Yassine El Adlouni, and Ouafae Ammor. 2021. Unsupervised neural networks for automatic Arabic text summarization using document clustering and topic modeling. *Expert Systems with Applications* 172 (June 2021). DOI: https://doi.org/10.1016/j.eswa.2021.114652

[145] Ying Shen, Qiang Zhang, Jin Zhang, Jiyue Huang, Yuming Lu, and Kai Lei. 2018. Improving medical short text classification with semantic expansion using word-cluster embedding. In *International Conference on Information Science and Applications*. Springer, 401–411.

[146] George Karypis and Vipin Kumar. 1998. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing* 20, 1 (1998), 359–392.

[147] Inderjit S. Dhillon, Yuqiang Guan, and Brian Kulis. 2007. Weighted graph cuts without eigenvectors a multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 11 (2007), 1944–1957.

[148] Kai-Yang Chiang, Joyce Jiyoung Whang, and Inderjit S. Dhillon. 2012. Scalable clustering of signed networks using balance normalized cut. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*. 615–624.

[149] Wei-Lin Chiang, Xuanqing Liu, Si Si, Yang Li, Samy Bengio, and Cho-Jui Hsieh. 2019. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 257–266.

[150] Mina Nasrazadani, Afsaneh Fatemi, and Mohammadali Nematbakhsh. 2022. Sign prediction in sparse social networks using clustering and collaborative filtering. *The Journal of Supercomputing* 78, 1 (2022), 596–615.

[151] Yao Wu, Xudong Liu, Min Xie, Martin Ester, and Qing Yang. 2016. CCCF: Improving collaborative filtering via scalable user-item co-clustering. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. 73–82.

[152] Yutong Li, Juanjuan Cai, and Jingling Wang. 2020. A text document clustering method based on weighted Bert model. In *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC'20)*, Vol. 1. IEEE, 1426–1430.

[153] Ziming Wang, Yuexian Zou, and Zeming Zhang. 2020. Cluster attention contrast for video anomaly detection. In *Proceedings of the 28th ACM International Conference on Multimedia*. 2463–2471.

[154] A. Bryar Hassan, A. Tarik Rashid, and K. Hozan Hamarashid. 2021. A novel cluster detection of COVID-19 patients and medical disease conditions using improved evolutionary clustering algorithm star. *Computers in Biology and Medicine* 138 (November 2021). DOI: https://doi.org/10.1016/j.compbiomed.2021.104866

[155] Phuong Dao, Recep Colak, Raheleh Salari, Flavia Moser, Elai Davicioni, Alexander Schönhuth, and Martin Ester. 2010. Inferring cancer subnetwork markers using density-constrained biclustering. *Bioinformatics* 26, 18 (2010), i625–i631.

[156] Qiang Wang, Yixin Deng, Francisco Roldan Sanchez, Keru Wang, Kevin McGuinness, Noel O'Connor, and Stephen J. Redmond. 2024. Dataset clustering for improved offline policy learning. *arXiv preprint arXiv:2402.09550* (2024).

[157] Georgios Vardakas, Ioannis Papakostas, and Aristidis Likas. 2024. Deep clustering using the soft silhouette score: Towards compact and well-separated clusters. *arXiv preprint arXiv:2402.00608* (2024).

[158] Lukas Miklautz, Andrii Shkabrii, Collin Leiber, Bendeguz Tobias, Benedict Seidl, Elisabeth Weissensteiner, Andreas Rausch, Christian Böhm, and Claudia Plant. 2023. Non-redundant image clustering of early medieval glass beads. In *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA'23)*. IEEE, 1–12.

[159] Guang-Yu Zhang, Dong Huang, and Chang-Dong Wang. 2024. Unified and tensorized incomplete multi-view kernel subspace clustering. *IEEE Transactions on Emerging Topics in Computational Intelligence* 8, 2 (April 2024), 1550–1566. DOI: https://doi.org/10.1109/TETCI.2024.3353576

[160] Himanshu Mittal, Avinash Chandra Pandey, Raju Pal, and Ashish Tripathi. 2021. A new clustering method for the diagnosis of CoVID19 using medical images. *Applied Intelligence* 51, 5 (2021), 2988–3011.

[161] Wenqi Guo, Weixiong Zhang, Zheng Zhang, Ping Tang, and Shichen Gao. 2022. Deep temporal iterative clustering for satellite image time series land cover analysis. *Remote Sensing* 14, 15 (2022), 3635.

[162] Haiyan Yu, Xiaoyu Xu, Honglei Li, Yuting Wu, and Bo Lei. 2024. Semi-supervised possibilistic c-means clustering algorithm based on feature weights for imbalanced data. *Knowledge-Based Systems* 286 (February 2024). DOI: https://doi.org/10.1016/j.knosys.2024.111388

[163] Douglas Reynolds. 2015. Gaussian mixture models. In *Encyclopedia of Biometrics*, Stan Z. Li and Anil K. Jain (Eds.). Springer US, Boston, MA, 827–832. DOI: http://dx.doi.org/10.1007/978-1-4899-7488-4_196

[164] Vignesh Prasad, Dipanjan Das, and Brojeshwar Bhowmick. 2020. Variational clustering: Leveraging variational autoencoders for image clustering. In *2020 International Joint Conference on Neural Networks (IJCNN'20)*. IEEE, 1–10.

[165] Zhihan Li, Youjian Zhao, Haowen Xu, Wenxiao Chen, Shangqing Xu, Yilin Li, and Dan Pei. 2020. Unsupervised clustering through gaussian mixture variational autoencoder with non-reparameterized variational inference and std annealing. In *2020 International Joint Conference on Neural Networks (IJCNN'20)*. IEEE, 1–8.

[166] Qiang Ji, Yanfeng Sun, Junbin Gao, Yongli Hu, and Baocai Yin. 2022. A decoder-free variational deep embedding for unsupervised clustering. *IEEE Transactions on Neural Networks and Learning Systems* 33, 10 (October 2022), 5681–5693. DOI: https://doi.org/10.1109/TNNLS.2021.3071275

[167] Wenqing Wang, Junpeng Bao, and Siyao Guo. 2022. Neural generative model for clustering by separating particularity and commonality. *Information Sciences* 589 (April 2022), 813–826. DOI: https://doi.org/10.1016/j.ins.2021.12.037

[168] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Communications of the ACM* 63, 11 (November 2020), 139–144. DOI: https://doi.org/10.1145/3422622

[169] Abdul Jabbar, Xi Li, and Bourahla Omar. 2021. A survey on generative adversarial networks: Variants, applications, and training. *ACM Computing Surveys (CSUR)* 54, 8 (2021), 1–49.

[170] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*. PMLR, 214–223.

[171] Deepak Raina, Dimitrios Ntentia, SH Chandrashekhara, Richard Voyles, and Subir Kumar Saha. 2023. Expert-agnostic ultrasound image quality assessment using deep variational clustering. In *2023 IEEE International Conference on Robotics and Automation (ICRA'23)*. IEEE, 2717–2723.

[172] Foivos Ntelemis, Yaochu Jin, and Spencer A. Thomas. 2022. Image clustering using an augmented generative adversarial network and information maximization. *IEEE Transactions on Neural Networks and Learning Systems* 33, 12 (December 2022), 7461–7474. DOI: https://doi.org/10.1109/TNNLS.2021.3085125

[173] Yanhai Gan, Xinghui Dong, Huiyu Zhou, Feng Gao, and Junyu Dong. 2022. Learning the precise feature for cluster assignment. *IEEE Transactions on Cybernetics* 52, 8 (August 2022), 8587–8600. DOI: https://doi.org/10.1109/TCYB.2021.3079914

[174] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks* 106 (October 2018), 249–259. DOI: https://doi.org/10.1016/j.neunet.2018.07.011

[175] Qiang Ji, Yanfeng Sun, Yongli Hu, and Baocai Yin. 2021. Variational deep embedding clustering by augmented mutual information maximization. In *2020 25th International Conference on Pattern Recognition (ICPR'21)*. IEEE, 2196–2202.

[176] Foivos Ntelemis, Yaochu Jin, and Spencer A. Thomas. 2021. Information maximization clustering via multi-view self-labelling. *Knowledge-Based Systems* 250 (August 2022). DOI: https://doi.org/10.1016/j.knosys.2022.109042

[177] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances In Neural InformatioN Processing Systems 33, NEURIPS 2020 (Advances in Neural Information Processing Systems)*.

[178] Lukas Meier, Sara Van De Geer, and Peter Bühlmann. 2008. The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 70, 1 (2008), 53–71.

[179] Kien Do, Truyen Tran, and Svetha Venkatesh. 2021. Clustering by maximizing mutual information across views. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9928–9938.

[180] Jiabo Huang and Shaogang Gong. 2022. Deep clustering by semantic contrastive learning. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, BMVA Press, 39. Retrieved from https://bmvc2022.mpi-inf.mpg.de/39/

[181] Tsung Wei Tsai, Chongxuan Li, and Jun Zhu. 2020. Mice: Mixture of contrastive experts for unsupervised image clustering. In *International Conference on Learning Representations*.

[182] Yuming Shen, Ziyi Shen, Menghan Wang, Jie Qin, Philip H. S. Torr, and Ling Shao. 2021. You never cluster alone. In *Advances in Neural Information Processing Systems (NEURIPS 2021)* 34 (Advances in Neural Information Processing Systems).

[183] Zhizhong Huang, Jie Chen, Junping Zhang, and Hongming Shan. 2022. Learning representation for clustering via prototype scattering and positive sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 6 (2022), 7509–7524.

[184] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. 2020. Bootstrap your own latent a new approach to self-supervised learning. In

Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS'20), Curran Associates Inc., Vancouver, BC, Canada.

[185] Ramakrishnan Sundareswaran, Jansel Herrera-Gerena, John Just, and Ali Jannesari. 2021. Cluster analysis with deep embeddings and contrastive learning. *arXiv preprint arXiv:2109.12714* (2021).

[186] Jianlong Chang, Yiwen Guo, Lingfeng Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. 2019. Deep discriminative clustering analysis. *arXiv preprint arXiv:1905.01681* (2019).

[187] Sohil Atul Shah and Vladlen Koltun. 2017. Robust continuous clustering. *Proceedings of the National Academy of Sciences* 114, 37 (2017), 9814–9819.

[188] Sohil Atul Shah and Vladlen Koltun. 2018. Deep continuous clustering. *arXiv preprint arXiv:1803.01449* (2018).

[189] Jian Hu, Xiangjie Li, Gang Hu, Yafei Lyu, Katalin Susztak, and Mingyao Li. 2020. Iterative transfer learning with neural network for clustering and cell type classification in single-cell RNA-seq analysis. *Nature Machine Intelligence* 2, 10 (2020), 607–618.

[190] Karpagalingam Thirumoorthy and Karuppaiah Muneeswaran. 2021. A hybrid approach for text document clustering using Jaya optimization algorithm. *Expert Systems with Applications* 178 (September 2021). DOI: https://doi.org/10.1016/j.eswa.2021.115040

[191] Mohammadreza Sadeghi, Hadi Hojjati, and Narges Armanfard. 2023. C3: Cross-instance guided contrastive clustering. In *34th British Machine Vision Conference 2023, BMVC 2023, Aberdeen, UK, November 20-24, 2023*, BMVA Press, 899–900. Retrieved from http://proceedings.bmvc2023.org/899/

[192] Qiang Feng, Liguo Han, Binghui Zhao, and Qiang Li. 2023. Microseismic events recognition via joint deep clustering with residual shrinkage dense network. *IEEE Transactions on Geoscience and Remote Sensing* 61 (2023). DOI: https://doi.org/10.1109/TGRS.2023.3314149

[193] Ziyou Zheng, Shuzhen Zhang, Hailong Song, and Qi Yan. 2024. Deep clustering using 3D attention convolutional autoencoder for hyperspectral image analysis. *Scientific Reports* 14, 1 (2024), 4209.

[194] Mengling Hu, Chaochao Chen, Weiming Liu, Xinyi Zhang, Xinting Liao, and Xiaolin Zheng. 2023. Learning uniform clusters on hypersphere for deep graph-level clustering. *arXiv preprint arXiv:2311.13953* (2023).

[195] Yue Qian, Shixin Yao, Tianjun Wu, You Huang, and Lingbin Zeng. 2024. Improved selective deep-learning-based clustering ensemble. *Applied Sciences* 14, 2 (2024), 719.

[196] Durgesh Singh, Ahcène Boubekki, Robert Jenssen, and Michael C. Kampffmeyer. 2023. Supercm: Revisiting clustering for semi-supervised learning. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'23)*. IEEE, 1–5.

[197] Yulin Yao, Yu Yang, Linna Zhou, Xinsheng Guo, and Gang Wang. 2023. Multi-CC: A new baseline for faster and better deep clustering. *Electronics* 12, 20 (2023), 4204.

[198] Ying Zhong, Dong Huang, and Chang-Dong Wang. 2023. Deep temporal contrastive clustering. *Neural Processing Letters* 55, 6 (2023), 7869–7885.

[199] Julian Shun, Farbod Roosta-Khorasani, Kimon Fountoulakis, and Michael W. Mahoney. 2016. Parallel local graph clustering. *Proceedings of the VLDB Endowment* 9, 12 (2016).

[200] Asim Kumar Debnath, Rosa L. Lopez de Compadre, Gargi Debnath, Alan J. Shusterman, and Corwin Hansch. 1991. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of Medicinal Chemistry* 34, 2 (1991), 786–797.

[201] Harold W. Kuhn. 1955. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* 2, 1-2 (1955), 83–97.

[202] Santo Fortunato. 2010. Community detection in graphs. *Physics Reports* 486, 3-5 (2010), 75–174.

[203] Di Jin, Zhizhi Yu, Pengfei Jiao, Shirui Pan, Dongxiao He, Jia Wu, Philip S. Yu, and Weixiong Zhang. 2023. A survey of community detection approaches: From statistical modeling to deep learning. *IEEE Transactions on Knowledge and Data Engineering* 35, 2 (February 2023), 1149–1170. DOI: https://doi.org/10.1109/TKDE.2021.3104155

[204] Michelle Girvan and Mark EJ Newman. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* 99, 12 (2002), 7821–7826.

[205] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, 10 (2008), P10008.

[206] Gary William Flake, Steve Lawrence, C Lee Giles, and Frans M. Coetzee. 2002. Self-organization and identification of web communities. *Computer* 35, 3 (2002), 66–70.

[207] Jianbo Shi and Jitendra Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 8 (2000), 888–905.

[208] Ulrike Von Luxburg. 2007. A tutorial on spectral clustering. *Statistics and Computing* 17, 4 (2007), 395–416.

[209] Jianxin Ma, Peng Cui, Kun Kuang, Xin Wang, and Wenwu Zhu. 2019. Disentangled graph convolutional networks. In *International Conference on Machine Learning*. PMLR, 4212–4221.

[210] Sheng Zhou, Jiajun Bu, Xin Wang, Jiawei Chen, and Can Wang. 2019. HAHE: Hierarchical attentive heterogeneous information network embedding. *arXiv preprint arXiv:1902.01475* (2019).

[211] Xiang Li, Yao Wu, Martin Ester, Ben Kao, Xin Wang, and Yudian Zheng. 2017. Semi-supervised clustering in attributed heterogeneous information networks. In *Proceedings of the 26th International Conference on World Wide Web*. 1621–1629.

[212] Xiang Li, Yao Wu, Martin Ester, Ben Kao, Xin Wang, and Yudian Zheng. 2022. Schain-iram: An efficient and effective semi-supervised clustering algorithm for attributed heterogeneous information networks. *IEEE Transactions on Knowledge and Data Engineering* 34, 4 (April 2022), 1980–1992. DOI: https://doi.org/10.1109/TKDE.2020.2997938

[213] Sheng Zhou, Jiajun Bu, Zhen Zhang, Can Wang, Lingzhou Ma, and Jianfeng Zhang. 2020. Cross multi-type objects clustering in attributed heterogeneous information network. *Knowledge-Based Systems* 194 (2020). DOI: https://doi.org/10.1016/j.knosys.2019.105458

[214] Anton Tsitsulin, John Palowitch, Bryan Perozzi, and Emmanuel Mueller. 2023. Graph clustering with graph neural networks. *Journal of Machine Learning Research* 24, (2023).

[215] Filippo Maria Bianchi, Daniele Grattarola, and Cesare Alippi. 2020. Spectral clustering with graph neural networks for graph pooling. In *International Conference on Machine Learning*. PMLR, 874–883.

[216] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, Vol. 96. 226–231.

[217] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. 2017. DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems (TODS)* 42, 3 (2017), 1–21.

[218] Dorin Comaniciu and Peter Meer. 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 5 (2002), 603–619.

[219] Mete Çelik, Filiz Dadaşer-Çelik, and Ahmet Şakir Dokuz. 2011. Anomaly detection in temperature data using DBSCAN algorithm. In *2011 International Symposium on Innovations in Intelligent Systems and Applications*. IEEE, 91–95.

[220] Zhenguo Chen and Yong Fei Li. 2011. Anomaly detection based on enhanced DBScan algorithm. In *CEIS 2011 (Procedia Engineering)*. DOI: https://doi.org/10.1016/j.proeng.2011.08.036

[221] Nadine Abbas, Youssef Nasser, Maryam Shehab, and Sanaa Sharafeddine. 2021. Attack-specific feature selection for anomaly detection in software-defined networks. In *2021 3rd IEEE Middle East and North Africa Communications Conference (menacomm)*. IEEE, 142–146.

[222] Amir Markovitz, Gilad Sharir, Itamar Friedman, Lihi Zelnik-Manor, and Shai Avidan. 2020. Graph embedded pose clustering for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10539–10547.

[223] Xiaoxiao Ma, Jia Wu, Shan Xue, Jian Yang, Chuan Zhou, Quan Z. Sheng, Hui Xiong, and Leman Akoglu. 2023. A comprehensive survey on graph anomaly detection with deep learning. *IEEE Transactions on Knowledge and Data Engineering* 35, 12 (December 2023), 12012–12038. DOI: https://doi.org/10.1109/TKDE.2021.3118815

[224] Hongfu Liu, Jun Li, Yue Wu, and Yun Fu. 2019. Clustering with outlier removal. *IEEE Transactions on Knowledge and Data Engineering* 33, 6 (2019), 2369–2379.

[225] Guy Barrett Coleman and Harry C. Andrews. 1979. Image segmentation by clustering. *Proc. IEEE* 67, 5 (1979), 773–785.

[226] Agus Zainal Arifin and Akira Asano. 2006. Image segmentation by histogram thresholding using hierarchical cluster analysis. *Pattern Recognition Letters* 27, 13 (2006), 1515–1521.

[227] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. 2018. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV'18)*. 670–685.

[228] Faliu Yi and Inkyu Moon. 2012. Image segmentation: A survey of graph-cut methods. In *2012 International Conference on Systems and Informatics (ICSAI'12)*. IEEE, 1936–1941.

[229] Nenglun Chen, Lei Chu, Hao Pan, Yan Lu, and Wenping Wang. 2022. Self-supervised image representation learning with geometric set consistency. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022) (IEEE Conference on Computer Vision and Pattern Recognition)*, IEEE; CVF; IEEE Comp Soc, 19270–19280. DOI: https://doi.org/10.1109/CVPR52688.2022.01869

[230] Gökcen Eraslan, Lukas M. Simon, Maria Mircea, Nikola S. Mueller, and Fabian J. Theis. 2019. Single-cell RNA-seq denoising using a deep count autoencoder. *Nature Communications* 10, 1 (2019), 1–14.

[231] Maria Brbić, Marinka Zitnik, Sheng Wang, Angela O. Pisco, Russ B. Altman, Spyros Darmanis, and Jure Leskovec. 2020. MARS: Discovering novel cell types across heterogeneous single-cell experiments. *Nature Methods* 17, 12 (2020), 1200–1206.

[232] Byron J. Gao, Obi L. Griffith, Martin Ester, and Steven JM Jones. 2006. Discovering significant opsm subspace clusters in massive gene expression data. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 922–928.

[233] Deepak Jaiswal, Vikrant Kaushal, Pankaj Kumar Singh, and Abhijeet Biswas. 2021. Green market segmentation and consumer profiling: A cluster approach to an emerging consumer market. *Benchmarking: An International Journal* 28, 3 (2021), 792–812. DOI: https://doi.org/10.1108/BIJ-05-2020-0247

[234] Ramu Govindasamy, Surendran Arumugam, Jingkun Zhuang, Kathleen M. Kelley, and Isaac Vellangany. 2018. Cluster analysis of wine market segmentation–a consumer based study in the mid-atlantic usa. *Economic Affairs* 63, 1 (2018), 151–157.

[235] Liam Close and Rasha Kashef. 2020. Combining artificial immune system and clustering analysis: A stock market anomaly detection model. *Journal of Intelligent Learning Systems and Applications* 12 (January 2020), 83–108. DOI: https://doi.org/10.4236/jilsa.2020.124005

[236] Tharindu Bandaragoda, Daswin De Silva, Denis Kleyko, Evgeny Osipov, Urban Wiklund, and Damminda Alahakoon. 2019. Trajectory clustering of road traffic in urban environments using incremental machine learning in combination with hyperdimensional computing. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC'19)*. IEEE, 1664–1670.

[237] Xavier Olive and Jérôme Morio. 2019. Trajectory clustering of air traffic flows around airports. *Aerospace Science and Technology* 84 (January 2019), 776–781. DOI: https://doi.org/10.1016/j.ast.2018.11.031

[238] Lin Yue, Weitong Chen, Xue Li, Wanli Zuo, and Minghao Yin. 2019. A survey of sentiment analysis in social media. *Knowledge and Information Systems* 60, 2 (2019), 617–663.

[239] Korawit Orkphol and Wu Yang. 2019. Sentiment analysis on microblogging with K-means clustering and artificial bee colony. *International Journal of Computational Intelligence and Applications* 18, 03 (2019), 1950017.

[240] Linhong Zhu, Aram Galstyan, James Cheng, and Kristina Lerman. 2014. Tripartite graph clustering for dynamic sentiment analysis on social media. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*. 1531–1542.

[241] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 249–256.

[242] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences* 63, 10 (2020), 1872–1897.

[243] M Emre Celebi, Hassan A. Kingravi, and Patricio A. Vela. 2013. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Systems with Applications* 40, 1 (2013), 200–210.

[244] Jierui Xie, Stephen Kelley, and Boleslaw K. Szymanski. 2013. Overlapping community detection in networks: The state-of-the-art and comparative study. *ACM Computing Surveys (csur)* 45, 4 (2013), 1–35.

[245] Jian Wu, Victor S. Sheng, Jing Zhang, Hua Li, Tetiana Dadakova, Christine Leon Swisher, Zhiming Cui, and Pengpeng Zhao. 2020. Multi-label active learning algorithms for image classification: Overview and future promise. *ACM Computing Surveys (CSUR)* 53, 2 (2020), 1–35.

[246] Weiwei Liu, Haobo Wang, Xiaobo Shen, and Ivor W. Tsang. 2022. The emerging trends of multi-label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 11 (November 2022), 7955–7974. DOI: https://doi.org/10.1109/TPAMI.2021.3119334

[247] Min-Ling Zhang and Zhi-Hua Zhou. 2013. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering* 26, 8 (2013), 1819–1837.

[248] Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. 2020. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations*.

[249] Flavia Moser, Rong Ge, and Martin Ester. 2007. Joint cluster analysis of attribute and relationship data withouta-priori specification of the number of clusters. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 510–519.

[250] Hongjing Zhang and Ian Davidson. 2021. Deep descriptive clustering. In *Proceedings of the Thirtieth Inteanational Joint Conference on Artificial Intelligence, IJCAI 2021*, Int Joint Conf Artifical Intelligence, 3342–3348.

[251] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 22, 10 (2009), 1345–1359.

[252] Shan Xue, Jie Lu, Jia Wu, Guangquan Zhang, and Li Xiong. 2016. Multi-instance graphical transfer clustering for traffic data learning. In *2016 International Joint Conference on Neural Networks (IJCNN'16)*. IEEE, 4390–4395.

[253] Zhijie Deng, Yucen Luo, and Jun Zhu. 2019. Cluster alignment with a teacher for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9944–9953.

[254] Sheng Zhou, Yucheng Wang, Defang Chen, Jiawei Chen, Xin Wang, Can Wang, and Jiajun Bu. 2021. Distilling holistic knowledge with graph neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV'21)*. 10387–10396.

[255] Yunpeng Zhai, Shijian Lu, Qixiang Ye, Xuebo Shan, Jie Chen, Rongrong Ji, and Yonghong Tian. 2020. Ad-cluster: Augmented discriminative clustering for domain adaptive person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9021–9030.

[256] Willi Menapace, Stéphane Lathuilière, and Elisa Ricci. 2020. Learning to cluster under domain shift. In *European Conference on Computer Vision*. Springer, 736–752.

[257] Sungwon Park, Sungwon Han, Sundong Kim, Danu Kim, Sungkyu Park, Seunghoon Hong, and Meeyoung Cha. 2021. Improving unsupervised image clustering with robust learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12278–12287.

[258] Shoujin Wang, Wei Liu, Jia Wu, Longbing Cao, Qinxue Meng, and Paul J. Kennedy. 2016. Training deep neural networks on imbalanced data sets. In *2016 International Joint Conference on Neural Networks (IJCNN'16)*. IEEE, 4368–4374.

[259] Xiang Fang, Yuchong Hu, Pan Zhou, and Dapeng Oliver Wu. 2022. Unbalanced incomplete multi-view clustering via the scheme of view Eeolution: Weak views are meat; Strong views do eat. *IEEE Transactions on Emerging Topics in Computational Intelligence* 6, 4 (August 2022), 913–927. DOI: https://doi.org/10.1109/TETCI.2021.3077909

[260] X. W. Liang, A. P. Jiang, T. Li, Y. Y. Xue, and G. T. Wang. 2020. LR-SMOTE–An improved unbalanced data set oversampling based on K-means and SVM. *Knowledge-Based Systems* 196 (2020). DOI: https://doi.org/10.1016/j.knosys.2020.105845

Received 11 June 2022; revised 5 May 2024; accepted 26 July 2024