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

SHENG ZHOU, HONGJIA XU, ZHUONAN ZHENG, JIAWEI CHEN, ZHAO LI, JIAJUN BU, Zhejiang University, China

JIA WU, Macquarie University, Australia

XIN WANG, WENWU ZHU, Tsinghua University, China

MARTIN ESTER, Simon Fraser University, Canada

Clustering is a fundamental machine learning task which has been widely studied in the literature. Classic clustering methods follow the assumption that data are represented as features in a vectorized form through various representation learning techniques. As the data become increasingly complicated and complex, the shallow (traditional) clustering methods can no longer handle the high-dimensional data type. With the huge success of deep learning, especially the deep unsupervised learning, many representation learning techniques with deep architectures have been proposed in the past decade. One straightforward way to incorporate the benefit of deep learning is to first learn the deep representation before feeding it into shallow clustering methods. However, this is suboptimal due to: i) the representation is not directly learned for clustering which limits the clustering performance; ii) the clustering relies on the relationship among instances which is complicated rather than linear; iii) the clustering and representation learning is dependent on each other which should be mutually enhanced. To tackle the above challenges, the concept of **Deep Clustering**, i.e., jointly optimizing the representation learning and clustering, has been proposed and hence attracted growing attention in the community. Motivated by the tremendous success of deep learning in clustering, one of the most fundamental machine learning tasks, and the large number of recent advances in this direction, in this paper we conduct a comprehensive survey on deep clustering by proposing a new taxonomy of different state-of-the-art approaches. We summarize the essential components of deep clustering and categorize existing methods by the ways they design interactions between deep representation learning and clustering. Moreover, this survey also provides the popular benchmark datasets, evaluation metrics and open-source implementations to clearly illustrate various experimental settings. Last but not least, we discuss the practical applications of deep clustering and suggest challenging topics deserving further investigations as future directions.

1   INTRODUCTION

Clustering is a fundamental problem in machine learning and frequently serves as an important preprocessing step in many data mining tasks. The primary purpose of clustering is to assign the instances into groups so that the 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 the further analysis on the whole dataset, such as anomaly detection [166, 201], domain adaptation [180, 240], community detection [121, 178] and discriminative representation learning [133, 164, 214], etc.

Although shallow clustering methods have achieved tremendous success, they assume that instances are already represented in a latent vectorized space with a good shape. With the rapid development of internet and web services in the past decades, the research community is showing an increasing interest in discovering new machine learning models capable of handling unstructured data without explicit features, such as images, and high-dimensional data with thousands of dimensions, etc. As such, shallow clustering methods can no longer be directly applied to deal with such data. Recent years have witnessed the success of representation learning with deep learning, especially in the unstructured and high-dimensional data [166, 201]. However, the deep learning technique was not explored in the clustering process. The complex relations among instances can not be well captured, which results in sub-optimal clustering results.

To address the issues, **Deep Clustering**, which aims at joint optimization of deep representation learning and clustering, arises and has attracted increasing attention recently in the community. More specifically, the deep clustering methods focus on the following research challenges:

(1) How to learn discriminative representations that can yield better clustering performance?
(2) How to efficiently conduct clustering and representation learning in a unified framework?
(3) How to break down the barriers between clustering and representation learning, enabling them to enhance each other in an interactive and iterative manner?

To tackle the above challenges, numerous deep clustering methods have been proposed with variant deep architectures and data types. Motivated by the tremendous success of deep learning in clustering, one of the most fundamental machine learning tasks, and the large number of recent advances in this direction, in this paper we conduct a comprehensive survey on deep clustering by proposing a new taxonomy of various state-of-the-art approaches.

Related Surveys Overview papers on clustering tend to have close relations to our work, however, we clearly state their distinctions from our survey as follows. There have been surveys on classic clustering [56, 208], particularly summarizing classic clustering methods into the following main groups: i) partition based methods [125, 152] that assign the data instances to the closest cluster centers, ii) hierarchy based methods [66, 96, 231] that construct the hierarchical relationships among data instances, iii) density based methods [20, 37, 50, 51, 167] that take the instances in the region with high density as a cluster, and iv) generative methods [162, 210] that assume data belonging to the same cluster are generated from the same distribution. Generally, these methods take the feature representations as input and output the cluster assignment of each instance. To the best of our knowledge, very few works have been proposed to survey the deep clustering approaches. Min et al. [135] in their survey paper states that the essence of deep clustering is to learn clustering-oriented representations, so the literature should be classified according to network architecture. However, we discover that the importance of clustering constraint and the interaction between representation and clustering play central roles in modern deep clustering. Nutakki et al. [146] review some basic methods, excluding the most recent advanced techniques for representation learning and clustering. Aljalbout et al. [3] design the taxonomy mainly based on the DNN architecture and the loss function, discussing the comparisons among methods only
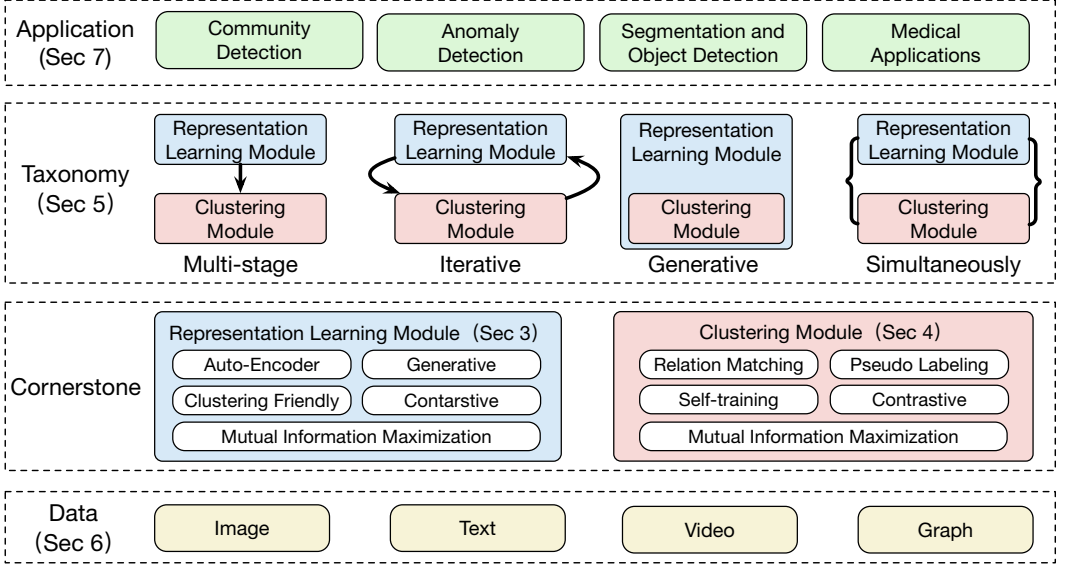
Fig. 1. Overall organization of this survey.

on the MNIST and COIL20 datasets. Furthermore, all these works [3, 135, 146] merely cover methods prior to the year of 2018, missing more recent advances in the past four years. Therefore, we strongly believe that this survey with a new taxonomy of literature and more than 200 scientific works will strengthen the focus and promote the future research on deep clustering.

Contributions. In summary, this paper aims at supporting the potential readers understanding the panorama 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 representation learning module and clustering module through providing four representative branches of methods. We also compare and analyse the properties of each branch within different scenarios.
- **Abundant Resources and References.** We collect various types of benchmark datasets, evaluation metrics and open-source implementations of latest publications on deep clustering, which are organized together with references on the Github (1.8K Star)[1].
- **Future Directions.** Based on the properties of representation learning module and 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 deserving future investigations.

Scope. In this survey, we focus on the clustering with deep learning techniques, especially the interactions between deep representation learning and clustering with deep neural networks. For

---

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

the other fundamental research problems such as initialization of clustering, automatic identifying number of clusters, etc., we provide discussions in Section 8 and leave them in the future work. The comparisons between surveys on shallow clustering, deep clustering and representation learning can be found in Table 1.

Table 1. Comparison with related surveys.

Comparisons	References	Deep Clustering					Shallow Clustering				Unsupervised Learning				
		Ours	[135]	[3]	[146]	[98]	[208]	[209]	[84]	[12]	[1]	[109]	[11]	[95]	[124]
Deep representation learning design	Auto Encoder	✓	✓	✓	✓	✓	-	-	-	-	✓	-	✓	✓	✓
	Generative	✓	✓	✓	-	✓	-	-	-	-	✓	✓	✓	✓	✓
	Contrastive	✓	-	-	-	-	-	-	-	-	-	✓	-	✓	✓
	Cluster Based	✓	✓	✓	-	-	-	-	-	-	-	✓	-	✓	✓
Clustering with DNN		✓	✓	✓	✓	✓	-	✓	✓	✓	-	-	-	-	-
Interaction between Representation and Clustering	Multistage	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-	-
	Iterative	✓	-	✓	✓	-	-	✓	✓	-	-	✓	-	✓	✓
	Generative	✓	✓	-	-	✓	-	-	-	-	-	-	-	-	-
	Simultaneously	✓	✓	✓	✓	-	-	-	-	-	-	-	-	-	-
Application		✓	-	-	-	-	-	✓	✓	✓	✓	✓	✓	✓	-
Dataset		✓	-	-	-	-	-	-	-	-	-	✓	-	-	-
Evaluation		✓	✓	-	-	-	✓	-	-	-	-	-	-	✓	-
Implementation		✓	-	-	-	-	-	-	-	-	-	-	-	-	-

* Each column indicates a survey paper being compared, '✓' means the term of corresponding row has been surveyed or analysed in this paper while '-' means not.

Organization. The rest of this survey is organized as follows: Section 2 introduces the basic definitions and notations used in this paper. Section 3 summarizes the representative design of representation module, along with different data types. Section 4 summarizes the representative design of clustering module, which mainly focuses on the basic modules defined in the deep clustering methods. Section 5 summarizes the representative ways of interactions between the two modules, which covers most 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 explorations. 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 we illustrate the notations used in this paper in Table 2.

Deep Clustering and Shallow (Non-deep) Clustering. Given a set of data instances $X = \{x_i\}_{i=1}^N$, clustering aims to automatically assign each instance x into 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 instances as input and outputs the clustering result without deep neural networks. The *deep clustering* aims at clustering 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, the 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 clustering and soft clustering according to the output type. 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 continuous cluster assignment probabilistic vector $z_i \in \mathcal{R}^K$. The discrete assignment of instance is usually hard

Table 2. Important notations used in this paper

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

to optimize, especially for deep neural networks with backpropagation. As a result, most existing deep clustering methods belong to the soft clustering category where the clustering results are produced by a deep neural network f whose outputs are softmax activated K-dimension logits. For the final evaluation purpose and discrete clustering category, the hard label can be obtained by selecting the dimension with maximum probability.

Partitioning Clustering and Overlapping Clustering. The above-defined hard clustering by maximizing the probability vector is designed for the *partitioning clustering* task where each data instance belongs to only one cluster. This means the clustering method participates the whole dataset into K disjoint groups. The majority of existing deep clustering methods are disjoint since the major evaluated datasets are disjoint ones. For the overlapping clustering setting, each data instance may belong to more than one cluster. This has brought extra difficulty to the clustering methods which will be discussed in section 8.

3 REPRESENTATION LEARNING MODULE

Recent decades have witnessed the rapid development of deep representation learning [11, 188, 238], especially the unsupervised ones. Intuitively, all unsupervised representation learning methods can serve as an input generator and be directly incorporated into deep clustering framework (further discussion on this issue in section 4). However, most existing methods are not intently designed for clustering task, and unable to integrate the potential clustering information to learn better representations. In this section, we introduce the *representation learning module* in deep clustering, which takes the raw data as input and outputs the low-dimensional representation (a.k.a. embedding). Figure 2 illustrates the representative representation learning modules described in this section.

3.1 Auto-Encoder based Representation Learning.

Auto-Encoder [165] is one of the most widely adopted unsupervised deep representation learning methods for its simplicity and effectiveness. The auto-encoder is a linear stack of two deep neural networks named encoder and decoder. The encoder f_e encodes the input data x into low-dimensional representation $h = f_e(x)$, and the decoder f_d decodes the low-dimensional representation h to the input data space $\hat{x} = f_d(h)$. A good auto-encoder is expected to reconstruct the input data without dropping any information, thus the optimization target of auto-encoder is usually formulated as:

$$\mathcal{L}_{AE} = \sum_{i}^{N} \|x_i - \hat{x}_i\|_2^2 = \sum_{i}^{N} \|x_i - f_d\left(f_e\left(x\right)\right)\|_2^2 \tag{1}$$
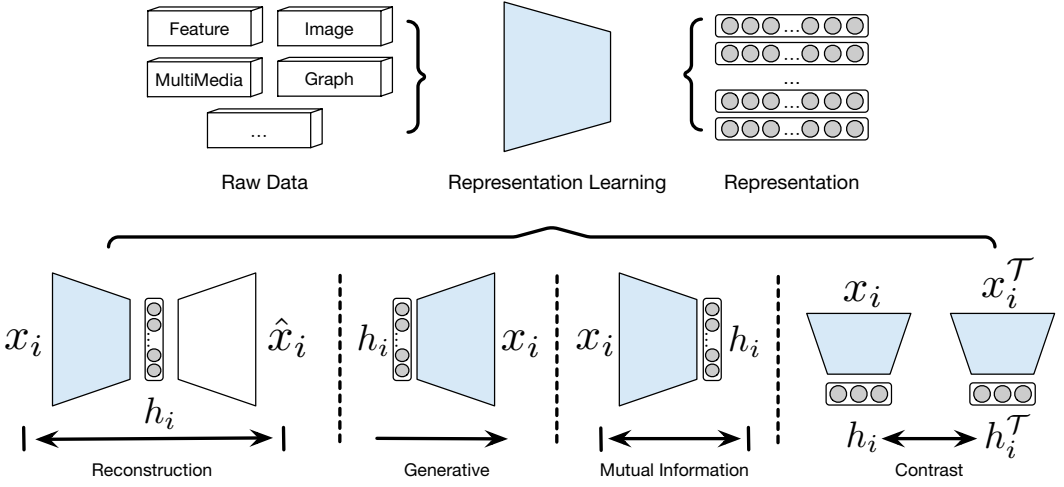
Fig. 2. Representative representation learning modules

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 [220], convolutional networks for images and graph neural networks for graphs [150], 3D convolutional networks and LSTM auto-encoder for videos [176] etc..

Analysis. The auto-encoder based representation learning framework enjoys the property of easy implementation and efficient training, and has been widely adopted in the early works of deep clustering. However, the representations are learned in an instance-wise manner while the relationships among different instances are largely ignored. As a result, the instances in the embedding space may not be well discriminated from each other, resulting in poor clustering performance.

3.2 Deep Generative Representation Learning.

Another line of deep unsupervised representation learning lies in the generative model. Generative methods assume that the data x are generated from 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 variational auto-encoder (VAE) [102]. 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 modelled with the recognition networks φ. By utilizing the reparameterization trick [102] and Monte Carlo approximation [97], the posterior can be efficiently learned from the equation (2) via backpropagation.

Analysis. Deep generative models enjoy some merits such as being flexible, explainable and capable to recreate data points. It would be promising to transform the generative representation models for deep clustering task so that the clustering models could inherit these advantages.

3.3 Mutual Information Maximization Representation Learning.

Mutual information (MI) [103] is a fundamental quantity for measuring the dependence between random variables X and Y, which is formulated as:

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

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 [106] are only tractable for discrete variables or known probability distributions. Recently, MINE [9] is proposed to estimate the mutual information with deep neural networks. The widely used mutual information estimation is the Jensen-Shannon divergence (JSD) [143] formulated as:

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

where $\text{sp}(x) = \log(1 + e^x)$ is the softplus function. D is a discrimintor function modeled by a neural network. Another popular mutual information estimation is InfoNCE [148] which will be introduced in subsection 3.4. Benefiting from the neural estimation, the mutual information has been widely adopted in unsupervised representation learning [7, 75]. More specifically, the representation is learned by maximizing the mutual information between different layers [7] or different parts of the data instances [75], so that the consistency of representation can be guaranteed. This can be viewed as an early attempt at self-supervised learning which has an extensive impact on the later works.

Analysis. As a fundamental measure of correlation and dependence, the mutual information has several advantages. The major advantage taken by the deep clustering task is that variables measured by mutual information are not restricted to same dimension and semantic space, such as instances and clusters. The detailed applications will be introduced in subsection 4.4 and 5.4.2. Similar to auto-encoder based and deep generative representation learning, the objective of mutual information maximization methods is also instance-wise, which may also have the aforementioned problems in capturing the relationships among instances. However, the marginal distribution in mutual information estimation depends on all the observed samples. In other words, the relationships among instances are captured in an implicit way, which has also boosted the performance in deep clustering.

3.4 Contrastive Representation Learning

Contrastive learning is one of the most popular unsupervised representation learning techniques in the past few years. The basic idea is to pull positive pair of instances close while push negative pair of instances far away, which is also known as instance discrimination. The representative target of contrastive learning is the InfoNCE loss [148] formulated as:

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

where h_i is the representation of the anchor sample, $h_i^{\mathcal{T}}$ is the positive sample representation and h_j, $h_j^{\mathcal{T}}$ are the negative sample representations, f is a similarity function, τ is the temperature parameter [74]. The positive samples are usually conducted by data augmentation which may vary from different data types. For example, the flip, rotate and crop augmentation for image data [30], the node dropping, edge perturbation, attribute masking and subgraph sampling for graph data [113, 217]. The negative samples are selected from augmented view of other instances in the

dataset [30] or a momentum updated memory bank of old negative representations [72], which can be viewed as approximation of noise.

Analysis. There has been several theoretical analyses on contrastive learning, and substantial evidences have shown that the representation learned by contrastive learning can benefit the clustering task. In [196], contrastive learning is explained with two properties: alignment of features from positive pairs and uniformity of the feature distribution on the hypersphere. The alignment property encourages the samples with similar features or semantic categories to be close in the low-dimensional space, which is essential for clustering. Such discriminative power has also been proved in the supervised manner [101].

The former work [148] has proved that minimizing the InfoNCE loss is equivalent to maximizing the lower bound of mutual information, while the data augmentation is not considered in the mutual information maximization. Recent study [199] has shown that the augmentation has played the role of 'ladder' in connecting instances within same category. As a result, instances from same cluster may be pulled closer which can benefit the clustering in the discriminative space.

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 [125] friendly representation is first defined in DCN [212] 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:

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

where $\mathrm{f}(\cdot)$ 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, C is the cluster representation matrix where k-th column of C, i.e., c_k denotes the centroid of the k-th cluster.

Spectral clustering friendly representation learning is inspired by the eigen decomposition in spectral clustering [140] that projects instances into the space with orthogonal bases. In deep clustering, the orthogonal basis is modeled explicitly by reducing correlations within features [181], which can be formulated as:

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

where d is the number of feature dimensions, h_m is the m-th 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 the direct optimization for clustering, which may significantly boost the corresponding clustering performance. However, such simplicity also limits the generalization to other clustering methods. Currently, the research community has put more efforts on the inspirations of clustering methods and express them in 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 [192], which aims at mapping the data instances into a low-dimensional subspace where instances can be separated. Basically, current subspace representation learning methods [88, 157, 225, 229, 230, 239] 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., $X = X\Theta_c$, where X is the data matrix and Θ_c is the self-expression coefficient matrix. For representation learning, the self-expression property leads to the following objective:

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

where $\|\cdot\|_p$ is a matrix norm, λ controls the weight balance. $\mathbf{H}$ denotes the sample representations learned by a network. Further, Θ_c can be implemented as the parameters of an additional network layer [88].

Analysis. The success of subspace representation learning relies on the theoretical assurance, which provides explicit modeling of relationships among data instances. However, it suffers from the limitation of solving the coefficient matrix of size $N \times N$, which is computationally difficult for large-scale data.

3.7 Data type specific representation learning

The former subsections have summarized the universal architectures of representation learning module. In real-world scenarios, representation learning for different data types can be viewed as the variants of the aforementioned architectures. In this subsection, we summarize four widely studied data types and their corresponding representation learning methods in deep clustering.

3.7.1 Image Representation Learning. Learning representations of images using CNN [110] and ResNet [73] as 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 representation learning module. Beyond the above two methods, recent advances have been made by introducing modern representation learning techniques such as vision transformer [47] to deep clustering. As one of most popular directions, the 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 the statistical based methods like TF-IDF [177], Word2Vec[108] and Skip-Gram [134]. Later, some works focus on the topic modeling [76] and semantic distances [48, 158] for text representation learning, and more [34] on unsupervised scenarios. Recently, the pre-trained language models like BERT [43] and GPT-3 [18] are gradually dominating the area of text representation learning. However, the fine tuning [224] of these methods in deep clustering task is still an open question.

3.7.3 Video Representation Learning. The video representation learning is a challenging task which combines the spatial-temporal learning, multi-model learning (with audio) [4], and natural language processing (with video abstract and subtitles) into one place. The early methods utilize LSTM Autoencoder [176], 3D-ResNets [160] and 3D-U-Net [155] as feature extractor. The recent methods have focused on spatial-temporal modeling [53, 176, 227, 228] and Qian et al. [160] in particular incorporates contrastive learning for self-supervision.

3.7.4 Graph Representation Learning. The classic graph representation learning aims at learning low dimensional representation for nodes so that the proximity among nodes can be preserved in the embedding space. Graph Neural Networks (GNNs) [194, 234] are widely used including GCN
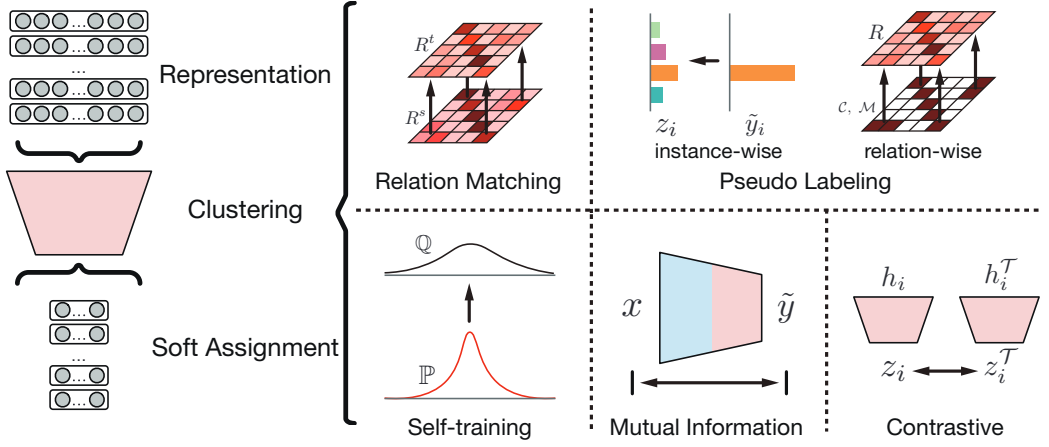
Fig. 3. Representative clustering modules

[104], GraphSAGE [70] and GAT [191], brings infinite possibility of graph node representation learning combining node features and graph topology[198, 232, 243, 246]. Furthermore, the graph-level information also has great potential in tasks like proteins classification [16], which has drown increasing attention in graph-level representation learning [67, 204, 233, 235].

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

4 CLUSTERING MODULE

In this section, we introduce the representative clustering modules in deep clustering, which takes the low-dimensional representations as input, and outputs the cluster labels for hard clustering or cluster assignment probabilities for soft clustering. Although many shallow clustering methods can be directly employed for clustering, they are hard to be trained with deep representation learning in a unified framework. More importantly, they can not well interact with representation learning module and mutually enhanced. For more shallow clustering methods, please refer to the former surveys [84, 209]. In deep clustering, a more 'deep' way of clustering is as follows: the vectorized features are directly fed forward through deep neural networks and reduce the dimension to the cluster number k, then the softmax function is applied on the last layer so that the assignment distribution can be established.

Although the k-dimensional representations are in the form of probability distribution, they may not represent the cluster distribution without explicit constraints and intensify the *degenerate problem* where all instances are assigned to the same cluster. In this section, we assume the clustering process is conducted by the deep neural networks and focus on the clustering targets. Figure 3 illustrates the representative clustering module described in this section.

4.1 Relation Matching Deep Clustering

In deep clustering, each data instance can be represented in two spaces, namely the d-dimensional *embedding space* and K-dimensional *label space*. The relationships among instances are expected

to be consistent during dimension reduction, which has been utilized to bridge the semantic gap between representation learning and clustering. Borrowing the idea from the domain adaptation [126], relation matching can be realized in bidirectional ways:

$$\mathcal{L}_{RM} = \sum_i^N \sum_j^N \ell(R_{ij}^s, R_{ij}^t) \tag{9}$$

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, which can be either embedding space or label space. It is worth noting that the relations defined here narrowly refers to the continuous ones and we introduce the discrete relations as pseudo labeling in subsection 4.2.

Analysis. The relation matching deep clustering explicitly connect the representation learning and clustering, which is straightforward and easy to implement. However, calculating N^2 pairs of instances is computational inefficient. To tackle this challenge, some methods only preserve the k-nearest-neighbor relations [39, 190] for each instance or the relations with high confidence [190]. Although this can somehow improve the training efficiency, the extra hyper-parameter is hard to set in the 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 discard the noisy relations is still an open research question.

4.2 Pseudo Labeling Deep Clustering

The pseudo labeling can be viewed as another type of relation matching where the relations are discrete based on the consistency of labels. It has been widely studied in semi-supervised learning [111], and recently being introduced to deep clustering. According to the way of utilizing pseudo labels, existing methods can be largely divided into two groups: instance-wise pseudo labeling [21, 141, 190] and relation-wise pseudo labeling [28, 142].

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:

$$\mathcal{L}_{IPL} = -\frac{1}{|\mathcal{X}^c|} \sum_i^{\mathcal{X}^c} \sum_{k=1}^K \tilde{y}_{ik} \log(z_{ik}) \tag{10}$$

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 entropy or maximum of the assignment probabilistic distribution.

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 construct 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:

$$\mathcal{L}_{RPL} = \frac{1}{|C|} \sum_{\{i,k\} \in C} R_{ik} - \frac{1}{|\mathcal{M}|} \sum_{\{i,j\} \in \mathcal{M}} R_{ij} \tag{11}$$

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. The pseudo labeling has brought the power of semi-supervised learning into the unsupervised clustering task. However, the performance highly relies on the quality of filtered pseudo labels which is susceptible to model capability and hyper-parameter tuning, especially in

the unsupervised manner. The current methods [141, 190] have taken the pre-training as the early stage before pseudo labeling to tackle this challenge, but more attention deserved to be paid on it.

4.3 Self-training Deep Clustering

The self-training strategy is introduced to the deep clustering task [206] and open up an active branch of methods named self-training deep clustering. More specifically, the cluster assignment distribution is optimized by minimizing the KL-divergence with an auxiliary distribution:

$$\mathcal{L} = D_{KL}(\mathbb{P}\|\mathbb{Q}) = \sum_{i}^{N} \sum_{k}^{K} p_{ik} \log \frac{p_{ik}}{q_{ik}} \tag{12}$$

where $\mathbb{Q}$ is the cluster assignment distribution and $\mathbb{P}$ is the auxiliary distribution. q_{ik} and p_{ik} denotes the probability of instance x_i belong to cluster k. The assignment distribution $\mathbb{Q}$ follows the assumption of K-means and is produced by the embedding distance between instance and cluster centroids:

$$q_{ik} = \frac{\left(1 + \|h_i - c_k\|_2^2 / \alpha\right)^{-\frac{\alpha+1}{2}}}{\sum_{j}^{K} \left(1 + \|h_i - c_j\|_2^2 / \alpha\right)^{-\frac{\alpha+1}{2}}} \tag{13}$$

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 [189]. The auxiliary distribution $\mathbb{P}$ is a variant of assignment distribution $\mathbb{Q}$ with both instance-wise and cluster-wise normalization:

$$p_{ik} = \frac{q_{ik}^2 / f_k}{\sum_{j}^{K} q_{ij}^2 / f_j} \tag{14}$$

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

Analysis. The self-training deep clustering has been widely adopted in several existing works [60, 68, 69, 112, 156, 164, 206]. The success relies on the following properties: First, the square of cluster assignment probability with cluster-wise normalization will guide the model put more attention (gradient) on the instances with higher confidence, which in turn reduce 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 instance belongs to the k-th cluster. This can prevent the degenerate solution that all instances belong to the same cluster. The above two properties have a profound effect on the later works and many deep clustering methods can be viewed as variants of it.

4.4 Mutual Information Maximization based Clustering

As introduced in subsection 3.3, mutual information maximization has achieved tremendous success in representation learning. Benefiting from its unrestriction to the feature dimension and semantic meaning, it has also been introduced to the clustering module for measuring the dependence between instance and cluster assignment. The mutual information maximization based clustering can be formulated as:

$$\mathcal{I}(X, \tilde{Y}) = D_{KL}(p(x, \tilde{y})\|p(x)p(\tilde{y})) = \sum_{x \in X} \sum_{\tilde{y} \in \mathcal{Y}} p(x, \tilde{y}) \log \frac{p(x, \tilde{y})}{p(x)p(\tilde{y})}. \tag{15}$$

where X is the original data, $\tilde{Y}$ is the predicted labels, $p(x, \tilde{y})$ is the joint distribution and $p(x), p(\tilde{y})$ are the marginal distribution. The estimation of mutual information is same as that in representation learning.

Analysis. The mutual information maximization based clustering and representation learning are quite similar despite the dimension of optimization object. The major advantage is that it overcomes the gap between the representation learning and clustering. As a consequence, the rapid development of deep representation learning techniques can be naturally introduced to the clustering task and optimized in a unified framework.

4.5 Contrastive Deep Clustering

Inspired by the success of contrastive representation learning and mutual information maximization based deep clustering, contrastive learning has also been introduced to 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:

4.5.1 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:

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

where z_i is the cluster assignment of instance x_i predicted by the clustering module.

4.5.2 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:

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

where c_k is the representation of cluster k. It is worth noting that cluster-cluster contrast satisfy the basic requirement of clustering that each cluster should be dissimilar, which agrees with the clustering friendly representation learning described in subsection 3.5.

4.5.3 Instance-Cluster contrast. The instance-cluster contrast is similar to the K-means which utilizes the cluster centroid as an 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:

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

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

Analysis. Besides the advantages inherited from the mutual information maximization clustering, the primary advantages of contrastive deep clustering is that the data augmentation helps improve the robustness of clustering, which has been ignored by most existing methods. For detailed advantages of contrastive learning, please refer to the former contrastive learning surveys [85, 95].

5 TAXONOMY OF DEEP CLUSTERING

In this section, we summarize existing deep clustering methods into four branches based on the interaction between the representation learning module and the clustering module:

(1) Multi-stage deep clustering: the representation learning module is sequential 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 representation module.
(4) Simultaneous deep clustering: the representation learning module and the clustering module are simultaneously updated.

5.1 Multistage Deep Clustering

Multistage deep clustering refers to the methods where two modules are separately optimized and sequentially connected. One straightforward way is to employ the deep unsupervised representation learning techniques to learn the representations (embedding) for each data instance first, and then feed the learned representations into the classic clustering models to obtain the final clustering results. Such detachment of data processing and clustering facilitate researchers to perform clustering analysis. More specifically, all existing clustering algorithms can be of service to any research scenarios.

Early multi-stage deep clustering methods [81, 183] have trained a deep auto-encoderto learn the representations, which can be directly packed as input of the K-means method to obtain the clustering results. Later, deep subspace clustering was proposed to learn an affinity matrix and instance representations first, and then performs clustering by spectral clustering on the affinity matrix [88, 229, 239] or the K-means on the instance representations [157]. Thanks to the contribution of scikit-learn [154] and many other open-source machine learning libraries, clustering algorithms has been applied to many fields with a limited cost of programming. For example, in the scenario of textual/video/graph data clustering, relation (similarity) matching was used in [26, 105, 186], K-Means in [25], Spectral Clustering in [2, 87, 155] and Hierarchical Agglomerative Clustering in [172], so as many other clustering algorithms being directly applied. In special, graph cut based node clustering like Metis [100], Graclus [44] and Balance Normalized Cut (BNC) [32] were used in graph clustering applications [33, 139, 205].

The most recent multistage methods have explicitly incorporated the clustering prior into the representation learning, then conduct clustering on the target friendly representations. For example, IDFD [181] learn representations with two aims: learning similarities among 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. However, such a simple combination of deep representation learning and shallow (traditional) clustering has the following weakness: 1) Most representation learning methods are not intently designed for clustering tasks, which can not provide sufficient discriminative power for clustering. Specifically, clustering reflects global patterns among data instances, while existing representation learning methods have largely focused on the individual patterns. 2) The clustering results can not be further utilized to guide the representation learning, which is of great importance for comprehensive representation. In particular, the cluster structure implies

the inherent relationships among data instances, and can in turn serve as critical guidance for improving representation learning. To conclude, such straightforward cascade connection will cut off the information flow and interactions between representation learning and clustering, thus the limitations of both side will influence the final performance together.

5.2 Iterative Deep Clustering

The key motivation of iterative deep clustering is that good representations can benefit clustering while clustering results reversely provide supervisory to representation learning. Briefly speaking, most existing iterative deep clustering pipeline is iteratively updated between two steps: 1) calculating clustering results given current representations and 2) updating the representations given the current clustering results. Since the representation module only provides input for the clustering module in iterative deep clustering, in this subsection, we classify existing iterative deep clustering methods according to the information provided by 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 utilized to train the representation learning module in a supervised manner.

In early works [175, 212], the cluster centroids and assignments are updated in a K-means way. S^{2}ConvSCN [225] and PSSC [127] combine subspace clustering and pseudo labeling, which obtain pseudo labels by spectral clustering or partitioning the pseudo similarity graph. Later, a lot of works [21, 141, 190] 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 [21], 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 [4].

Recently, SCAN [190] follows a pretraining-with-finetuning framework. The clustering results are fine-tuned with self-labeling, which selects the highly confident instances by thresholding the soft assignment probability, and updates the whole network by minimizing the cross-entropy loss on the selected instances. SPICE [141] is another representative iterative deep clustering method, where the classification model is first trained with 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 pesudo 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 [28, 214] and the model is trained as a binary classification task. Another popular branch of methods [142, 230] model the relationship by the similarity of cluster assignment probabilities, which train the representation learning as a regression task.

Summary. Iterative deep clustering methods benefit from the mutual promotion between representation learning and clustering. However, they also suffer from the error propagation in the iterative process. More specifically, the inaccurate clustering results can lead to chaotic representations where the performance are limited by the self-labeling effectiveness. Furthermore, this will in turn affect the clustering results especially in the early stage of training. Therefore, existing iterative clustering methods heavily rely on the pretraining of representation module.

5.3 Generative Deep Clustering

Generative models are able to capture, represent and recreate data points, and thus are drawing increasing attention from both academia and industry. They would make the hypotheses about the latent cluster structure and then infer the clustering assignment by estimation of data density. The most representative model is Gaussian Mixture Model (GMM) [163], which assumes that the data points are generated from a Mixture-of-Gaussians. Specifically, suppose there are K clusters, and an observed sample x is generated from the following process:

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

where π denotes the prior probability for clustering; $\mathrm{Mult}(\pi)$ 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 employed to learn the optimal parameters and clustering assignment.

While GMM has gained successful applications, its shallow structure is usually insufficient for capturing the nonlinear patterns of the data, adversely affecting its performance on complex data (e.g., images, texts, graphs and etc). To address this problem, deep generative models have been proposed to combine the generative model with the powerful deep neural networks, which have enough capacity to model the non-linear and complex data. This kind of methods can be classified into two types: the methods based on Variational Auto-Encoder (VAE) and the 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 merit of the latent cluster structure and meanwhile has sufficient capacity to model the complex data. For example, the representative models, VaDE [93] and GMVAE [45], assume the following generative process for each instance:

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

where $\mu_z(.; \beta)$, $\sigma_z^2(.; \beta)$, $\mu_x(.; \theta)$ and $\sigma_x^2(.; \theta)$ 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:

$$\log p(x) = \log \int_z \sum_c p(x|z)p(z|c)p(c)\mathrm{d}z \tag{19}$$

However, directly optimizing the above likelihood is intractable as it involves integration and complex neural network. Variational Auto-Encoder (VAE)[102] sheds a light to tackle this problem such that the parameters and the posterior can be efficiently estimated via backpropagation. Specifically, the generative model is trained with the following variational inference objective, a.k.a. the evidence lower bound (ELBO):

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

where $q(z, c|x; \varphi)$ is the variational posterior to approximate the true posterior, which can be modelled with the recognition networks φ. Monte Carlo [97] and the reparameterization trick [102] can be employed to learn the parameters.

More recently, upon VaDE and GMVAE, some improved variants have been proposed. For example, Prasad et al. [159] 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. [119] 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. [89] proposed to replace decoder in VAE with an improved mutual-information-based objective; Wang et al. [197] 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 the great success of Generative Adversarial Network (GAN) in estimating complex data distribution [6, 63, 83]. A standard GAN contains two components: a generator G that targets 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 similar distribution to the data. Inspired by such outstanding ability, it would be promising to integrate GAN into generative clustering models. Specifically, Ben-Yosef et al. [10] proposed to stack a GMM with a GAN, where GMM serves as a prior distribution for generating data instances. Formally, they optimized the following objective function:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_X(x)} [\log D(x)] + \mathbb{E}_{z \sim p_Z(z)} [\log(1 - D(G(z)))] \tag{21}$$

where $p_X(x)$ denotes the training data distribution; $p_Z(z)$ is a prior distribution of G and defined as mixture of Gaussians:

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

By equipping GAN with such 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. [218] proposed to directly replace the Gaussian distribution of GMM with a GAN and developed a ϵ-expectation- maximization learning algorithm to forbid early convergence issues; Ntelemis et al. [144] proposed to employ Sobel operations prior to the discriminator of the GAN; Mukherjee et al. [138] proposed to sample the latent vector z from a mixture of one-hot encoded variables and continuous latent variables. An inverse network with a clustering-specific loss is introduced to make the model more friendly to the clustering task. Analogously, an inverse network is introduced in [57, 59] 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 multiply clusters.

5.4 Simultaneous Deep Clustering

Simultaneously deep clustering is the most active direction in current deep clustering, where representation learning module and the clustering module are simultaneously optimized in an end-to-end manner. Although most iterative deep clustering methods also optimize both two modules with a single objective, the two modules are optimized in an explicit iterative manner and can not 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 [81].

The representative method is DEC [206] which combines the auto-encoder with the self-training strategy. Such 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 subsection 4.3 is then introduced to optimize the 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 [68] further integrates the reconstruction loss to the auto-encoder. The general formulation of auto-encoder with self-training can be summarized as:

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

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 [60, 69, 112], the linear layer of auto-encoder is replaced with fully convolutional layers so that the image feature can be well captured. In CCNN [77], the clustering convolutional neural network is proposed as a new backbone to extract the representations which are friendly to the clustering task. In DEPICT [60], an additional noisy encoder is introduced, and the robustness of 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 works also make attempts for specific problem. To boost the robustness of clustering, self-training is applied between two branches: clean and augmented [69] (noisy [60]). Specifically, the target distribution $\mathbb{P}$ is computed via the clean branch to guide the soft assignments $\mathbb{Q}$ of the augmented or noisy branch. Self-training strategy can be combined with subspace clustering. CSC [156] 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 compute the target distributions $\mathbb{P}_E$ and $\mathbb{P}_C$, with the KL divergence between them minimized.

According to aforementioned analysis, self-training is similar to the K-means clustering and suffers from the unbalance problem [19] between different clusters. To solve the problem of the unbalanced data and out-of-distribution samples, StatDEC [164] improves target distribution by adding normalized instance frequency of 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 subsection 3.3 and 4.4, the mutual information maximization has been successfully applied in both representation and clustering. The unified form of mutual information in both modules has brought convenience for understanding and implementation.

The representative mutual information maximization based clustering method is DCCM [202]. For each data instance, the mutual information between deep and shallow layer representation is maximized so that the consistency of representation can be guaranteed. Such consistency is further extended to the cluster assignment space by encouraging the instances with the same pseudo labels to share similar representations. Many later works can be regarded as variants of this method. In VCAMI [90] and IIC [91], the augmented mutual information (AMI) is introduced to improve the robustness. Such augmentation invariant has inspired the later contrastive deep clustering methods, which will be introduced in the next subsection. In ImC-SWAV [145], the mutual information between the integrated discrete representation and a discrete probability distribution is maximized, which improves the vanilla SWAV [22] method.

5.4.3 Contrastive Deep Clustering. Similar to mutual information maximization based deep clustering, contrastive learning has also been successfully applied in both representation learning module and 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 [118]. 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.

Take CC as the fundamental architecture, many contrastive deep clustering can be viewed as variants of it. PICA [80] can be viewed as the degeneration of CC without augmentation, it directly separates different clusters by minimizing the cosine similarity between the cluster-wise assignment statistic vectors. In DCCM [202], the augmentation is introduced to guarantee the local robustness of the learned representation. DRC [236] has the same contrastive learning as CC where the cluster representation is called assignment probability. The difference lies in the cluster regularization which is inspired by group lasso [131]. In CRLC [46], 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 the probabilistic contrastive learning. SCCL [222] 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 [79], the negative samples are limited by different pseudo labels, so that instances from different clusters can be further distinguished. MiCE [185] follows a divide-and-conquer strategy where 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 that models alignment and uniformity implicitly[196], MiCE model these two properties in a more explicit way. Recently, TCC [171] further improve the efficiency with the reparametrization trick and explicitly improve the cluster discriminability.

Some other methods make attempts on overcoming the problems of vanilla contrastive learning. In GCC [237], 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 [82], the contrastive learning module is replaced from SimCLR to BYOL [65], so that the over-reliance on the 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 [222] and Sundareswaran et al. [179] combine contrastive representation learning and self-training. DDC [27] and RCC [169] combine relation matching clustering and pseudo labeling to boost the clustering performance. DCC [170] combines the auto-encoder based representation and the relation matching clustering. The auto-encoder based representation learning and spectral clustering are combined in [215] by incorporating the augmentations from the contrastive learning.

Summary. The simultaneous deep clustering 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 incur from undesired prejudice of optimization focus between representation learning module and clustering module, which can only be mitigated by the manually setting of the balanced parameter for now. Also, the model is easy to sink into degenerate solutions where all instances are assigned into one 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[2], CMU PIE[3], Yale-B[4], MNIST[5], CIFAR[6] and STL-10[7]. 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 texutal data clustering include Reuters-21578[8] and 20 Newsgroups[9] datasets, which have already been vectorized and little feature engineering is needed. Currently, raw textual datasets including IMDB[10], stack-Overflow[11], and more in nlp-datasets github repository[12] is still challenging for deep textual clustering.

6.1.3 Video Datasets. The ultimate task of video clustering varies from action classification [155] to video anomaly detection [200]. Kinetics-400 and Kinetics-600[13] are two of the most famous video datasets. The others include UCF-101 dataset[14] and HMDB-51 dataset[15].

---

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

Table 3. The proposed taxonomy of existing methods and their properities

Taxonomy	Advantages	Disadvantages	Clustering Module	Type	Citation
Multi Stage	1. Easy to implement 2. Better interpretability	1. Sub-optimal representations 2. Limit on clustering performance	Multi Stage	Image	[183][88][229][239][157][181][39]
				Graph	[81][33][139][205]
				Textual	[26][186][25][172][87][117][105][2]
				Video	[155][200]
				Medical	[71][40]
Generative	1. Latent cluster structure discovery 2. Able to model complex data distribution	1. Unstable training 2. High computational complexity 3. Sensitive to prior distribution assumptions	VAE	Image	[45][215][93][159][89][119][197]
			GAN	Image	[138][10][218][144][57][59]
				Graph	[92]
Iterative	1. Clustering oriented representation learning 2. Discriminative space for clusteing	1. Error propagation 2. Limit on pseudo-label quality	Individual supervision	Image	[214][212][175][225][127][190][141]
				Video	[4]
				Medical	[99][136]
			Relational supervision	Image	[28][21][230][142]
Simultaneous	1. Clustering oriented representation learning 2.Discriminative space for clusteing 3. Unified optimization framework	1. Undesired prejudice of optimization focus 2. Sensitive to degenerate solutions	Auto-encoder based	Image	[206][68][60][69][77][112][156][164]
				Graph	[15]
				Medical	[184][78]
			Mutual Information based clustering	Image	[202][90][91][145]
			Contrastive	Image	[118][38][80][202][236][46][79][185][171][82][39]
				Textual	[222]
				Graph	[237]
			Hybrid methods	Image	[179][27][169]
				Textual	[222][133][182]

6.1.4 Graph Datasets. Commonly used graph datasets for node clustering can be referred from the following papers [33, 139, 174] and Stanford Network Analysis Project[16]. And there are also graph-level classification datasets like PROTINS[16] and MUTAG[41], which can be used to perform and evaluate graph-level clustering.

6.2 Evaluation Metrics

Evaluation metrics aim to evaluate 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).

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 follows:

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

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 [107].

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

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

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 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:

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

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:

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

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 pattern among instances, it has also benefit various downstream tasks. In this section, we discuss some typical applications of deep clustering.

---

[16]http://snap.stanford.edu/

7.1 Community Detection

Community Detection [55, 94, 121] aims at partitioning the graph network into several sub-graphs mainly according to connection density, which can be treated as node-level graph clustering task. Early works are mainly based on modularity measurement [14, 61], Maximum flows [54], graph cut [173] and its extension, spectral methods [193]. With the development of Graph Nerual Networks (GNNs)[128, 241], nodes are represented as individual instances in the low-dimensional space. As a result, the border between modern community detection [178] and graph clustering[115, 116, 242] is gradually getting blurry, and GNN based graph clustering [13, 187] have already been applied to many applications. However, different from early community detection that focus on the network topology, the graph clustering usually incorporates the node attributes and other side information. How to release the power of GNNs while reserve 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 [37, 51, 168] have specifically mentioned and addressed the problem of noise during clustering, which has enlighted a group of density based anomaly detection methods [24, 31]. The later anomaly detection methods [129, 130, 200] have utilized the clustering results and identify the anomaly instances as those far from the cluster centroids or border of each cluster. Currently, the deep clustering has shown great potential in forming a better clustering space for anomaly detection. Instead of performing anomaly detection after the deep clustering, the recent efforts have been put on conducting them in a unified framework: identify and remove the anomaly instances to reduce the impact on clustering [122], 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 which aims at dividing the pixels into disjoint regions. Generally speaking, image segmentation is doing pixel classification in a supervised manner and pixel clustering in an unsupervised manner [5, 36]. Currently, the deep clustering has been successfully applied in segmentation by utilizing the clustered regions to generate Scene Graph [213]. Yi et al. [216] 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 [29], 3D points were clustered to represent an object with geometric consistency. But such clustering based segmentation and object detection has no guarantee for small region and object, where the expected clustering result is highly unbalanced. And the global positional information of a pixel may be ignored when peoforming clustering.

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 requiring expert medical knowledge [99], which is hard to realize in the real-world scenarios. Recently, the deep clustering has been introduced to automatically categorize large-scale medical images [99]. Mit al et al. [136] introduce medical image clustering analysis for faster COVID-19 diagnostic. In the biological science field, single cell RNA sequencing (scRNA-seq) [49] gives an output cell-gene matrix for cell population and behaviour analysis and even new cell discovery. For this purpose, ScDeepCluster [184] and ItClust [78] develop their models based on

DEC [206] to cluster scRNA-seq data, and MARS [17] combining transfer learning and clustering to discover novel cell types. More application can be found in gene data clustering area[58].

7.5 Summary

In addition to the above successful deep clustering applications, the clustering also has great potential in many other domains, such as financial analysis [35, 64, 86], trajectory analysis [8, 147], and social media understanding [149, 219, 245]. Although most of the aforementioned methods has not incorporate deep learning techniques, with the increasing data volume and dimension, we believe that deep clustering will pave the way for these scenarios.

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 play an important role in training efficiency and stability [62]. This is more critical in deep clustering where both the representation learning module and clustering module are modeled by deep neural networks. Recently, the model pre-training [161] has been a popular network initialization technique which has also been introduced to the deep clustering [190]. However, the pre-training based initialization is appropriate for representation learning module but has not been well studied for the clustering module. Although there has been some initialization schemes on the shallow clustering [23], 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 paper largely focus on the partitioning clustering where each instance belong to only one cluster. Meanwhile, in the real-world scenario, each instance may belong to multiple clusters, e.g., users in a social network [207] may belong to several communities, and the video/audio in the social media may have several tags [203]. Among the deep clustering methods discussed in this paper, 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 the multi-label classification has been widely studied in the literature [123, 226], how to adapt to unsupervised clustering learning is still an open research question.

8.3 Degenerate Solution VS Unbalanced Data

The degenerate solution [21] has been one of the most noteworthy concerns in deep clustering, where all instances may be assigned to a single cluster. Many deep clustering methods have added extra constraints to overcome this problem [21, 80, 81, 91, 118, 153, 190], among which the entropy of cluster size distribution is the most widely used one. By maximizing the entropy, the instances are expected to be evenly assigned to each cluster and avoid degenerating to single cluster. It is worth noting that the success relies on the uniform distribution of ground truth labels which is coincidentally satisfied by the benchmark dataset such as CIFAR10 and CIFAR100. However, in real-world scenarios, such assumption is too strict and most datasets are unbalanced or long tailed [195]. The divergent target of uniform distribution and unbalanced dataset will seriously weaken the deep clustering. Recently, clustering unbalanced data [52, 120] has attracted increasing attention and this will boost the performance of clustering in real-world applications.

8.4   Boosting Representation with Deep Clustering

Throughout this paper, we can find that a good representation is essential for clustering. Although the clustering friendly representation learning has been studied in the literature, they are designed for the specific shallow clustering method. On the contra, the clustering structure denotes the high-order pattern of the dataset, which should be preserved in the comprehensive representation [114]. The deep clustering methods discussed in this paper focus on how to incorporate the representation learning to boost the clustering, meanwhile, how to in turn boost the representation learning by the clustering is still to be studied.

8.5   Deep Clustering Explanation

As an unsupervised task, the clustering process usually lacks human priors such as label semantics, number of clusters[137], which makes the clustering results hard to explain or understand. Some methods [223] have already combined the tags provided by the users to boost the explanation of the clustering results. However, it relies on the accurate human tagging which may not be realized in practical. With the development of causal inference in deep learning, the explanation of clustering among instances is hopefully to be improved. Both the research and industry community are expecting a generalized explanation framework for clustering, especially on the high-dimensional data. To conclude, how to utilize the casual inference techniques in the clustering is of great importance and deserve more attention.

8.6   Transfer Learning with Deep Clustering

Transfer learning [151] aims at bridging the gap between the training and test datasets with distribution shift. The general idea is to transfer the knowledge from the known data to the unknown test data. Recently, deep clustering is playing an increasing important role in unsupervised transfer learning [42, 180, 211, 240, 244], where the target domain is in the unsupervised manner. For example, ItClust [78] and MARS [17] has achieved success in scRNA-seq clustering (section 7.4), AD-Cluster [221] that use clustering has improved domain adaptive person re-identification. On the other way around, unsupervised transfer learning methods can also benefit deep clustering. Take UCDS [132] as an example, the unsupervised domain adaptation is used to perform clustering among variant domains. Distribution shift is one of the key factor affecting the performance of machine learning models, including deep clustering. Perform clustering analysis can give further understanding on unsupervised target domain, but how to transfer clustering result to knowledge and effectively minimize the distribution shift based on clustering result can be further explored.

8.7   Clustering with Anomalies

In subsection 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 restrict, 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 has been a few works on overcoming this problem [122], they are designed for shallow clustering methods. To this end, how to improve the deep clustering robustness to the anomaly instances and gradually improve the clustering performance by reducing the detected anomaly instances is still an open research question.

8.8   Efficient Training VS Global Modeling

To improve the training efficiency and scalability, most existing deep clustering methods have utilized the mini-batch training strategy, where the 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 the 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 utilized the cluster representations or prototypes [114] to store the global information, how to balance the training efficiency and model capability is still worth studying.

9 CONCLUSION

In this survey, we present a comprehensive and up-to-date overview of the deep clustering research field. We first summarize the cornerstones of deep clustering namely the representation learning module and the clustering module, followed by the representative design. Based on the ways of interaction between representation learning module and clustering module, we present the taxonomy of existing methods namely: multi-stage deep clustering, iterative deep clustering, generative deep clustering and simultaneous deep clustering. Then, we collect the benchmark datasets, metrics for evaluation and applications of deep clustering. Last but not least, we discuss the future directions in deep clustering that have potential opportunities.

REFERENCES

[1] Mohanad Abukmeil, Stefano Ferrari, Angelo Genovese, Vincenzo Piuri, and Fabio Scotti. 2021. A survey of unsupervised generative models for exploratory data analysis and representation learning. *Acm computing surveys (csur)* 54, 5 (2021), 1–40.

[2] 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 (2021), 114652.

[3] 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).

[4] Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. 2020. Self-supervised learning by cross-modal audio-video clustering. *Advances in Neural Information Processing Systems* 33 (2020), 9758–9770.

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

[6] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *International conference on machine learning*. PMLR, 214–223.

[7] Philip Bachman, R Devon Hjelm, and William Buchwalter. 2019. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems* 32 (2019).

[8] 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)*. IEEE, 1664–1670.

[9] 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.

[10] 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).

[11] 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.

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

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[19] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks* 106 (2018), 249–259.

[20] Feng Cao, Martin Ester, Weining Qian, and Aoying Zhou. 2006. Density-based clustering over an evolving data stream with noise. In *Proceedings of the 2006 SIAM international conference on data mining*. SIAM, 328–339.

[21] 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)*. 132–149.

[22] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems* 33 (2020), 9912–9924.

[23] 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.

[24] Mete Çelik, Filiz Dadaşer-Çelik, and Ahmet Şakir Dokuz. 2011. Anomaly detection in temperature data using DB-SCAN algorithm. In *2011 international symposium on innovations in intelligent systems and applications*. IEEE, 91–95.

[25] 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.

[26] 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.

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

[28] 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.

[29] Nenglun Chen, Lei Chu, Hao Pan, Yan Lu, and Wenping Wang. 2022. Self-Supervised Image Representation Learning with Geometric Set Consistency. *arXiv preprint arXiv:2203.15361* (2022).

[30] 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.

[31] Zhenguo Chen and Yong Fei Li. 2011. Anomaly detection based on enhanced DBScan algorithm. *Procedia Engineering* 15 (2011), 178–182.

[32] 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.

[33] 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.

[34] Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James Glass. 2019. An unsupervised autoregressive model for speech representation learning. *arXiv preprint arXiv:1904.03240* (2019).

[35] Liam Close, Rasha Kashef, et al. 2020. Combining artificial immune system and clustering analysis: A stock market anomaly detection model. *Journal of Intelligent Learning Systems and Applications* 12, 04 (2020), 83.

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

[37] 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.

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

[39] 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.

[40] 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.

[41] 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.

[42] 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.

[43] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[44] 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.

[45] Nat Dilokthanakul, Pedro AM Mediano, Marta Garnelo, Matthew CH Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. 2016. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648* (2016).

[46] 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.

[47] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).

[48] Robert Dzisevič and Dmitrij Šešok. 2019. Text classification using different feature extraction approaches. In *2019 Open Conference of Electrical, Electronic and Information Sciences (eStream)*. IEEE, 1–4.

[49] 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.

[50] Martin Ester. 2018. Density-based clustering. *Data Clustering* (2018), 111–127.

[51] 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.

[52] Xiang Fang, Yuchong Hu, Pan Zhou, and Dapeng Oliver Wu. 2021. Unbalanced Incomplete Multi-view Clustering via the Scheme of View Evolution: Weak Views are Meat; Strong Views do Eat. *IEEE Transactions on Emerging Topics in Computational Intelligence* (2021).

[53] 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.

[54] 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.

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

[56] Itziar Frades and Rune Matthiesen. 2010. Overview on techniques in cluster analysis. *Bioinformatics methods in clinical research* (2010), 81–107.

[57] Yanhai Gan, Xinghui Dong, Huiyu Zhou, Feng Gao, and Junyu Dong. 2021. Learning the Precise Feature for Cluster Assignment. *IEEE Transactions on Cybernetics* (2021).

[58] 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.

[59] 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.

[60] 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.

[61] 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.

[62] 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.

[63] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014).

[64] 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.

[65] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems* 33 (2020), 21271–21284.

[66] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. 1998. CURE: An efficient clustering algorithm for large databases. *ACM Sigmod record* 27, 2 (1998), 73–84.

[67] 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.

[68] Xifeng Guo, Long Gao, Xinwang Liu, and Jianping Yin. 2017. Improved Deep Embedded Clustering with Local Structure Preservation.. In *Ijcai*. 1753–1759.

[69] 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.

[70] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[71] Bryar A Hassan, Tarik A Rashid, and Hozan K 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 (2021), 104866.

[72] 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.

[73] 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.

[74] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* 2, 7 (2015).

[75] 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).

[76] Thomas Hofmann. 2013. Probabilistic latent semantic analysis. *arXiv preprint arXiv:1301.6705* (2013).

[77] 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.

[78] 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.

[79] Jiabo Huang and Shaogang Gong. 2021. Deep clustering by semantic contrastive learning. *arXiv preprint arXiv:2103.02662* (2021).

[80] 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.

[81] 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.

[82] Zhizhong Huang, Jie Chen, Junping Zhang, and Hongming Shan. 2021. Exploring Non-Contrastive Representation Learning for Deep Clustering. *arXiv preprint arXiv:2111.11821* (2021).

[83] 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.

[84] Anil K Jain, M Narasimha Murty, and Patrick J Flynn. 1999. Data clustering: a review. *ACM computing surveys (CSUR)* 31, 3 (1999), 264–323.

[85] 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.

[86] Deepak Jaiswal, Vikrant Kaushal, Pankaj Kumar Singh, and Abhijeet Biswas. 2020. Green market segmentation and consumer profiling: a cluster approach to an emerging consumer market. *Benchmarking: An International Journal* (2020).

[87] 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.

[88] Pan Ji, Tong Zhang, Hongdong Li, Mathieu Salzmann, and Ian Reid. 2017. Deep subspace clustering networks. *Advances in neural information processing systems* 30 (2017).

[89] Qiang Ji, Yanfeng Sun, Junbin Gao, Yongli Hu, and Baocai Yin. 2021. A Decoder-Free Variational Deep Embedding for Unsupervised Clustering. *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[90] 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)*. IEEE, 2196–2202.

[91] 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.

[92] 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.

[93] Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. 2016. Variational deep embedding: An unsupervised and generative approach to clustering. *arXiv preprint arXiv:1611.05148* (2016).

[94] Di Jin, Zhizhi Yu, Pengfei Jiao, Shirui Pan, Dongxiao He, Jia Wu, Philip Yu, and Weixiong Zhang. 2021. A survey of community detection approaches: From statistical modeling to deep learning. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[95] 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.

[96] Stephen C Johnson. 1967. Hierarchical clustering schemes. *Psychometrika* 32, 3 (1967), 241–254.

[97] Malvin H Kalos and Paula A Whitlock. 2009. *Monte carlo methods*. John Wiley & Sons.

[98] 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.

[99] 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.

[100] 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.

[101] 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.

[102] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[103] 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.

[104] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[105] 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.

[106] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. 2004. Estimating mutual information. *Physical review E* 69, 6 (2004), 066138.

[107] Harold W Kuhn. 1955. The Hungarian method for the assignment problem. *Naval research logistics quarterly* 2, 1-2 (1955), 83–97.

[108] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*. PMLR, 1188–1196.

[109] Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. 2020. Contrastive representation learning: A framework and review. *IEEE Access* 8 (2020), 193907–193934.

[110] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[111] Dong-Hyun Lee et al. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, Vol. 3. 896.

[112] Fengfu Li, Hong Qiao, and Bo Zhang. 2018. Discriminatively boosted image clustering with fully convolutional auto-encoders. *Pattern Recognition* 83 (2018), 161–173.

[113] Haoyang Li, Xin Wang, Ziwei Zhang, Zehuan Yuan, Hang Li, and Wenwu Zhu. 2021. Disentangled Contrastive Learning on Graphs. *Advances in Neural Information Processing Systems* 34 (2021).

[114] Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. 2020. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966* (2020).

[115] 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.

[116] Xiang Li, Yao Wu, Martin Ester, Ben Kao, Xin Wang, and Yudian Zheng. 2020. Schain-iram: An efficient and effective semi-supervised clustering algorithm for attributed heterogeneous information networks. *IEEE Transactions on*

knowledge and data engineering (2020).

[117] 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)*, Vol. 1. IEEE, 1426–1430.

[118] 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)*.

[119] 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)*. IEEE, 1–8.

[120] XW Liang, AP Jiang, T Li, YY Xue, and GT Wang. 2020. LR-SMOTE—An improved unbalanced data set oversampling based on K-means and SVM. *Knowledge-Based Systems* 196 (2020), 105845.

[121] 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 preprint arXiv:2005.08225* (2020).

[122] 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.

[123] Weiwei Liu, Haobo Wang, Xiaobo Shen, and Ivor Tsang. 2021. The emerging trends of multi-label learning. *IEEE transactions on pattern analysis and machine intelligence* (2021).

[124] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. 2021. Self-supervised learning: Generative or contrastive. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[125] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* 28, 2 (1982), 129–137.

[126] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. 2015. Learning transferable features with deep adaptation networks. In *International conference on machine learning*. PMLR, 97–105.

[127] Juncheng Lv, Zhao Kang, Xiao Lu, and Zenglin Xu. 2021. Pseudo-supervised deep subspace clustering. *IEEE Transactions on Image Processing* 30 (2021), 5252–5263.

[128] 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.

[129] Xiaoxiao Ma, Jia Wu, Shan Xue, Jian Yang, Chuan Zhou, Quan Z Sheng, Hui Xiong, and Leman Akoglu. 2021. A comprehensive survey on graph anomaly detection with deep learning. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[130] 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.

[131] 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.

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

[133] 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.

[134] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[135] 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.

[136] 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.

[137] 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.

[138] 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.

[139] 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.

[140] Andrew Ng, Michael Jordan, and Yair Weiss. 2001. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems* 14 (2001).

[141] Chuang Niu, Hongming Shan, and Ge Wang. 2021. Spice: Semantic pseudo-labeling for image clustering. *arXiv preprint arXiv:2103.09382* (2021).

[142] 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.

[143] 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).

[144] Foivos Ntelemis, Yaochu Jin, and Spencer A Thomas. 2021. Image clustering using an augmented generative adversarial network and information maximization. *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[145] Foivos Ntelemis, Yaochu Jin, and Spencer A Thomas. 2021. Information Maximization Clustering via Multi-View Self-Labelling. *arXiv preprint arXiv:2103.07368* (2021).

[146] 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.

[147] Xavier Olive and Jérôme Morio. 2019. Trajectory clustering of air traffic flows around airports. *Aerospace Science and Technology* 84 (2019), 776–781.

[148] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[149] 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.

[150] Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. 2018. Adversarially regularized graph autoencoder for graph embedding. *arXiv preprint arXiv:1802.04407* (2018).

[151] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2009), 1345–1359.

[152] Hae-Sang Park and Chi-Hyuck Jun. 2009. A simple and fast algorithm for K-medoids clustering. *Expert systems with applications* 36, 2 (2009), 3336–3341.

[153] 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.

[154] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[155] 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.

[156] Xi Peng, Jiashi Feng, Jiwen Lu, Wei-Yun Yau, and Zhang Yi. 2017. Cascade subspace clustering. In *Thirty-First AAAI conference on artificial intelligence*.

[157] Xi Peng, Shijie Xiao, Jiashi Feng, Wei-Yun Yau, and Zhang Yi. 2016. Deep subspace clustering with sparsity prior.. In *IJCAI*. 1925–1931.

[158] 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.

[159] 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)*. IEEE, 1–10.

[160] 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.

[161] 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.

[162] Carl Rasmussen. 1999. The infinite Gaussian mixture model. *Advances in neural information processing systems* 12 (1999).

[163] Douglas A Reynolds. 2009. Gaussian mixture models. *Encyclopedia of biometrics* 741, 659-663 (2009).

[164] Mina Rezaei, Emilio Dorigatti, David Ruegamer, and Bernd Bischl. 2021. Learning Statistical Representation with Joint Deep Embedded Clustering. *arXiv preprint arXiv:2109.05232* (2021).

[165] 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.

[166] 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.

[167] Jörg Sander, Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. 1998. Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data mining and knowledge discovery* 2, 2 (1998), 169–194.

[168] 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.

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

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

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

[172] 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.

[173] Jianbo Shi and Jitendra Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence* 22, 8 (2000), 888–905.

[174] Julian Shun, Farbod Roosta-Khorasani, Kimon Fountoulakis, and Michael W Mahoney. 2016. Parallel local graph clustering. *arXiv preprint arXiv:1604.07515* (2016).

[175] 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.

[176] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. 2015. Unsupervised learning of video representations using lstms. In *International conference on machine learning*. PMLR, 843–852.

[177] Michael Steinbach, George Karypis, and Vipin Kumar. 2000. A comparison of document clustering techniques. (2000).

[178] Xing Su, Shan Xue, Fanzhen Liu, Jia Wu, Jian Yang, Chuan Zhou, Wenbin Hu, Cecile Paris, Surya Nepal, Di Jin, et al. 2022. A comprehensive survey on community detection with deep learning. *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[179] 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).

[180] 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.

[181] Yaling Tao, Kentaro Takagi, and Kouta Nakata. 2021. Clustering-friendly representation learning via instance discrimination and feature decorrelation. *arXiv preprint arXiv:2106.00131* (2021).

[182] Karpagalingam Thirumoorthy and Karuppaiah Muneeswaran. 2021. A hybrid approach for text document clustering using Jaya optimization algorithm. *Expert Systems with Applications* 178 (2021), 115040.

[183] 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.

[184] 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.

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

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

[187] Anton Tsitsulin, John Palowitch, Bryan Perozzi, and Emmanuel Müller. 2020. Graph clustering with graph neural networks. *arXiv preprint arXiv:2006.16904* (2020).

[188] 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.

[189] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[190] 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.

[191] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[192] 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.

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

[194] 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.

[195] 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)*. IEEE, 4368–4374.

[196] Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*. PMLR, 9929–9939.

[197] Wenqing Wang, Junpeng Bao, and Siyao Guo. 2022. Neural generative model for clustering by separating particularity and commonality. *Information Sciences* 589 (2022), 813–826.

[198] 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.

[199] 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*.

[200] 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.

[201] S Wibisono, MT 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.

[202] 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.

[203] 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.

[204] 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.

[205] 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.

[206] Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*. PMLR, 478–487.

[207] 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.

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

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

[210] Xiaowei Xu, Martin Ester, H-P Kriegel, and Jörg Sander. 1998. A distribution-based clustering algorithm for mining in large spatial databases. In *Proceedings 14th International Conference on Data Engineering*. IEEE, 324–331.

[211] 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)*. IEEE, 4390–4395.

[212] 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.

[213] 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)*. 670–685.

[214] 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.

[215] 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.

[216] Faliu Yi and Inkyu Moon. 2012. Image segmentation: A survey of graph-cut methods. In *2012 international conference on systems and informatics (ICSAI2012)*. IEEE, 1936–1941.

[217] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems* 33 (2020), 5812–5823.

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

[219] 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.

[220] 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)*. IEEE, 415–419.

[221] 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.

[222] 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. *arXiv preprint arXiv:2103.12953* (2021).

[223] Hongjing Zhang and Ian Davidson. 2021. Deep Descriptive Clustering. *arXiv preprint arXiv:2105.11549* (2021).

[224] 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).

[225] 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.

[226] 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.

[227] Qin Zhang, Jia Wu, Hong Yang, Yingjie Tian, and Chengqi Zhang. 2016. Unsupervised Feature Learning from Time Series.. In *IJCAI*. New York, USA, 2322–2328.

[228] 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.

[229] 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.

[230] 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.

[231] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. 1996. BIRCH: an efficient data clustering method for very large databases. *ACM sigmod record* 25, 2 (1996), 103–114.

[232] 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.

[233] Zhen Zhang, Jiajun Bu, Martin Ester, Jianfeng Zhang, Chengwei Yao, Zhi Yu, and Can Wang. 2019. Hierarchical graph pooling with structure learning. *arXiv preprint arXiv:1911.05954* (2019).

[234] Ziwei Zhang, Peng Cui, and Wenwu Zhu. 2020. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering* (2020).

[235] 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.

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

[237] 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.

[238] 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.

[239] 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.

[240] Qiang Zhou, Shirui Wang, et al. 2021. Cluster adaptation networks for unsupervised domain adaptation. *Image and Vision Computing* 108 (2021), 104137.

[241] 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).

[242] 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), 105458.

[243] 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.

[244] 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)*. 10387–10396.

[245] 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.

[246] Wenwu Zhu, Xin Wang, and Peng Cui. 2020. Deep learning for learning graph representations. In *Deep learning: concepts and architectures*. Springer, 169–210.