Deep Semi-Supervised Learning via Tensor Label Propagation for High-Dimensional Low-Sample Size Data

Anonymous Authors

***Abstract*—Label propagation aims at assigning labels to abundant samples based on a few labeled samples and has been a popular technique in semi-supervised learning. However, traditional label propagation relies on a graph to propagate label information, which may fail to provide satisfactory performance especially for high-dimensional low-sample size samples. The reason for this is that a graph merely encodes pairwise similarities of samples and may fall short in representing the complex geometric structure of high-dimensional samples. To overcome such a limitation, we propose a deep semi-supervised learning method based on tensor similarity, which can capture high-order similarities and complex geometric structure of samples, thus facilitating the label propagation process. Specifically, we train the network on labeled samples to learn representation, subsequently, a uniform model is trained iteratively on all samples via high-order similarity and network optimization for label propagation. Extensive experiments on HDLSS datasets demonstrate the effectiveness of our proposed method in comparison to recent baseline methods.**

***Index Terms*—Semi-supervised learning, label propagation, high-order similarity, high-dimensional low-sample size**

I. Introduction

Semi-supervised learning is a crucial technique in the field of artificial intelligence. Semi-supervised learning methods combine limited labeled data with a large amount of unlabeled data for model training, which can enhance model performance while reducing labeling costs [1]–[3]. In recent years, various methods have emerged in semi-supervised learning. These methods can be roughly classified into four categories based on implementation principles: self-training [4], [5], co-training [6], generative models [7], and graph-based semi-supervised learning (GSSL) [8], [9]. Among them, graph-based semi-supervised learning methods have been widely researched and garnered attention due to their intuitive and efficient characteristics, giving rise to various variants. These methods are built on a crucial assumption that similar or proximate data points are likely to belong to the same class [1], [10], [11], enabling algorithms to learn additional information about data distribution from unlabeled data based on the graph structure, thereby enhancing overall learning effectiveness.

Graph-based semi-supervised learning (GSSL) methods typically involve two key steps: graph construction [12], [13] and label propagation [14], [15]. During the graph construction stage, labeled and unlabeled data are unified as nodes, with edges connecting nodes representing the similarity between samples. Once the graph constrction is completed, graph-based propagation algorithms can propagate label information from labeled nodes to unlabeled nodes. These methods are based on the assumption that nodes close to each other in the graph structure have the same label [16], [17]. However, traditional GSSL methods fail to obtain discriminative representations through nonlinear mappings. Furthermore, these methods are offline, able to analyze the original dataset only, and may not adapt well to subsequent new data.

In recent years, researchers have utilized the feature extraction capabilities of deep neural networks to address the aforementioned issue. These methods leverage the label propagation process in graph semi-supervised learning to train deep neural networks. Through the neural network, samples' low-dimensional representations are learned. Then, data similarity in the graph is calculated based on the low-dimensional representations after multiple layers of nonlinear mappings, followed by label propagation. For instance, Benato et al. [18] propose a semi-supervised deep network based on label propagation in 2D embedding space. Kamnitsas et al. [19] utilized dynamically constructed graphs and label propagation to achieve compact clustering in latent space. CL_PLP [20] proposed a deep semi-supervised learning algorithm that improves label propagation by combining contrastive self-supervised learning and partial label propagation strategy. Huang et al. [21] proposed the Correct and Smooth procedure, which uses a basic predictor to obtain preliminary predictions and then applies label propagation to correct and smooth the model predictions. While these methods equip online models with the capability to handle subsequent data, they have limitations when dealing with high-dimensional small-sample data. Firstly, the performance of graph-based deep methods largely depends on the quality of the constructed similarity graph, i.e., whether the graph structure can accurately reflect actual data relationships. High-dimensional data often exhibit the concentration effect when measuring based on pairwise similarities [22], [23], leading to similarity tending towards a constant. Additionally, the learning capacity of deep networks is limited when the sample size is insufficient, and inadequate labeled data may result in misleading label propagation.

To overcome these challenges, we present Tensor Label Propagation based Deep Semi-supervised Learning (TLPDSL). Our method consists of two main modules: a feature embedding network and a label propagation module. The feature embedding network is leveraged to extract deep features from input data. The pseudo-labels for the samples are derived by amalgamating multi-order similarity information

within the label propagation module, thereby yielding precise outcomes. The proposed TLPDSL is capable of effectively utilizing high-order similarity to enhance the accuracy and robustness of label propagation process. Our contributions are summarized as follow:

- High-order similarity is introduced to our label propagation process, which depicts intrinsic correlations among multiple samples thus compensate the pairwise similarity. The precision of label propagation is thereby significantly augmented.
- A novel deep semi-supervised learning algorithm is presented. The proposed method integrates both high-order and low-order similarities derived from deep features, allowing for effectively exploiting latent representations of data.
- Extensive experiments on different HDLSS datasets demonstrate the effectiveness of TLPDSL in comparison to baseline methods. The experimental results indicate that TLPDSL complements pairwise similarity with higher-order similarity to enhance classification performance.

II. Related work

A. Deep Semi-supervised Learning

The objective of Semi-Supervised Learning (SSL) is to leverage both labeled and unlabeled data with the aim of enhancing the effectiveness of supervised learning tasks [1]. SSL is particularly useful when labeled data is scarce or costly to obtain, which is a common situation in many real-world applications. Numerous approaches have been proposed to exploit data representations using Deep Neural Networks (DNNs). There are several ways to incorporate SSL into DNNs, such as deep generation methods, consistency regularization methods, graph-based methods, pseudo-labeling methods, and hybrid methods [24].

Raw data generally undergo two types of processing when implementing DSSL, namely consistency regularization and pseudo-labeling. Deep consistency regularization methods, ensure network output consistency under perturbations like noise, augmentation, or dropout. Pseudo-labeling methods, manage to assign pseudo-labels to unlabeled data based on network predictions. For example, Pseudo-Label [25] [26]uses network predictions as pseudo-labels and update them periodically. FixMatch [27] combines pseudo-labels with consistency regularization and only uses pseudo-labels with high confidence. Shi et al. [28] derives the deterministic weight of unlabeled samples based on the distance between the unlabeled sample and neighboring samples in the feature space. However, these methods mostly rely on the assumption that labeled data and unlabeled data come from the same distribution, which may not hold in some real-world scenarios. In addition, the absence of higher-order message may cause models to overlook the higher-order relationships and partial information present in the data, making them more susceptible to the influence of noise.

B. Label Propagation

Label propagation is a graph-based semi-supervised learning technique that assigns labels to unlabeled nodes by propagating from labeled nodes through graph edges. It assumes that data can be represented as a graph, with nodes representing data points and edges reflect their similarity.

The diversity among label propagation methods lies in how they construct the graph and define the linear system. Zhou et al. [10] proposed a label propagation method based on the normalized graph Laplacian operator, minimizing label score smoothness relative to the graph structure. [29] is proposed to mitigate the impact of outliers in graphs. Label propagation has advantages such as simplicity, scalability, and flexibility in choosing graph construction methods [14].

On the other hand, deep semi-supervised label propagation combines the advantages of deep neural networks and propagation algorithms. For instance, Iscen et al. [30] proposed utilizing a dataset's nearest neighbor graph to generate pseudo-labels for unlabeled data and trained a deep neural network with these predictions. Zhuang and Moulin [31] introduced a metric learning method that improves label propagation based semi-supervised deep learning by identifying and removing hard negative pairs in the similarity matrix, enhancing performance in a 2D embedding space. Lastly, A2LP [32] proposed an adaptive anchor-based label propagation algorithm that adjusts anchor positions and weights for efficient classification with limited labeled data. Although these methods can extract sample features, they struggle to capture local structures and complex relationships in the data, making it challenging to ensure data separability and robustness in high-dimensional and low-sample size (HDLSS) data.

III. Preliminaries

In this section, we discuss the problem setting and some of the fundamental knowledge of label propagation, then briefly introduce the high-order tensor similarity used in this paper. We begin with the definition of calculus used in this paper.

Definition 1: (Kronecker Product) The Kronecker product of two matrices $\boldsymbol{A} \in \mathbb{R}^{m_1 \times n_1}$ and $\boldsymbol{B} \in \mathbb{R}^{m_2 \times n_2}$ is defined as:

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{11}\boldsymbol{B} & \cdots & a_{1n_1}\boldsymbol{B} \\ \vdots & \ddots & \vdots \\ a_{m_1 1}\boldsymbol{B} & \cdots & a_{m_1 n_1}\boldsymbol{B} \end{bmatrix} \in \mathbb{R}^{m_1 m_2 \times n_1 n_2}. \quad (1)$$

Definition 2: (mode-k product) Let $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \cdots \times m_N}$ be a N-th order tensor and $\boldsymbol{U} \in \mathbb{R}^{q \times m_k}$ be a matrix. The mode-k product of $\mathcal{A}$ and $\boldsymbol{U}$ is a N-th order tensor denoted by $\mathcal{A} \times_k \boldsymbol{U} \in \mathbb{R}^{m_1 \times \cdots \times m_{k-1} \times q \times m_{k+1} \cdots \times m_N}$ such that

$$\begin{aligned} &(\mathcal{A} \times_k \boldsymbol{U})_{r_1, \cdots, r_{k-1}, t, r_{k+1}, \cdots, r_N} \\ &= \sum_{r_k=1}^{m_k} \mathcal{A}_{r_1, \cdots, r_{k-1}, r_k, r_{k+1}, \cdots, r_N} U_{t, r_k}. \end{aligned} \quad (2)$$

Problem formulation. Given sample set $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_l, \boldsymbol{x}_{l+1}, \cdots, \boldsymbol{x}_m]$ and label set $\mathcal{L} = \{1, \cdots, c\}$, the first l points $\boldsymbol{x}_i$ $(1 \leq i \leq l)$ are labeled as $y_i \in \mathcal{L}$, and the remaining samples $\boldsymbol{x}_u$ $(l+1 \leq u \leq m)$ are unlabeled.

The goal is to predict the labels of the unlabeled samples y_u $(l+1 \leq u \leq m)$.

Let $\boldsymbol{F} \in \mathbb{R}^{m \times c}$ denote the classification output of m samples, and $\boldsymbol{Y} \in \mathbb{R}^{m \times c}$ denotes the one-hot label matrix, where for labeled samples $\boldsymbol{Y}_{i,j} = 1$ if $\boldsymbol{x}_i$ is labeled as $y_i = j$, and for unlabeled samples $\boldsymbol{Y}_{p,q} = 0$ for $p \in \{l+1, \cdots, m\}$ and $q \in [c]$.

A. Label Propagation on Graph

Label Propagation on Graph [10] Let $\boldsymbol{A} \in \mathbb{R}^{m \times m}$ be the similarity matrix and $\boldsymbol{S} = \boldsymbol{D}^{-1/2}\boldsymbol{A}\boldsymbol{D}^{-1/2}$ be the normalized similarity matrix, where $\boldsymbol{D}_{ii} = \sum_j \boldsymbol{A}_{ij}$ is the degree matrix. Label propagation (LP) attempts to solve the following objective:

$$\min_{F} \alpha \sum_{i,j=1}^{m} \boldsymbol{A}_{ij} \left\| \frac{\boldsymbol{F}_{i,:}}{\boldsymbol{D}_{ii}} - \frac{\boldsymbol{F}_{j,:}}{\boldsymbol{D}_{jj}} \right\|_2^2 + (1-\alpha) \left\| \boldsymbol{Y} - \boldsymbol{F} \right\|_F^2, \quad (3)$$

where $\boldsymbol{F}_{i,:}$ represents the i row of $\boldsymbol{F}$ and α denotes a hyper-parameter for balance. The first term encourages smoothness such that nearby examples get the same predictions, while the second attempts to maintain predictions for the labeled examples. By setting the derivative of Eq. (3) to zero, one can derive the following solution:

$$\boldsymbol{F}^* = (\boldsymbol{I} - \alpha \boldsymbol{S})^{-1} \boldsymbol{Y}. \quad (4)$$

Finally, the class prediction for an unlabeled sample $\boldsymbol{x}_u$ is:

$$\hat{y}_u = \arg\max_{j<c} \boldsymbol{F}^*_{u,j}. \quad (5)$$

These processes can be further integrated into an iterative deep learning based framework [30], thereby enables the utilization of deep learning's distinctive feature extraction capabilities to facilitate more accurate label propagation.

B. Tensor Spectral Analysis

Traditional label propagation methods is demonstrated in Eq. (3). Where the first term is the smoothness constraint, which denotes that neighboring nodes tend to share the same label. This spirit is also applied in spectral clustering, its objective is defined as:

$$\min_{F} \sum_{i,j=1}^{n} w_{ij} ||\boldsymbol{F}_i - \boldsymbol{F}_j||^2, \quad (6)$$

where w_{ij} is an element in weight matrix W. The core principle of spectral clustering is to optimize the intra-cluster similarity in order to effectively maintain the volume of each subgraph after graph cut. Cai et al. [33] introduce a normalized similarity entropy metric that can evaluate the volume of similarity across different sample sizes with flexibility. It aims to seeks a concise low-dimensional representation by utilizing multi-order similarities. Let indicator matrix H $[h_1, h_2, \ldots, h_k] \in \mathbb{R}^{n \times k}$ be the sample assignment, such that $H_{ij} = |C_j|^{-1}$if $x_i \in C_j$, and zero otherwise. To obtain this optimal sample assignment $C_1, \ldots, C_k$, we maximize the normalized associativity by solving:

$$\max_{C_1,\ldots,C_k} \sum_{j=1}^{k} (\mathcal{P} \times_m h_j \times_{m-1} h_j \ldots \times_1 h_j), \quad (7)$$

where, h_j is the j-th column of matrix H, $\mathcal{P}$ is the normalized tensor calculated by order-k sample similarity.

Solving for maximum normalized associativity is NP-hard. Alternatively, a relaxation approach can be used by transforming the binary assignment matrix to an orthonormal matrix $Q \in \mathbb{R}^{n \times k}$, where $Q^T Q = I$. The simplified problem is:

$$\max_{Q^T Q = I} \sum_{j=1}^{k} (\mathcal{P} \times_1 q_j \times_2 q_j \ldots \times_m q_j), \quad (8)$$

where q_j is the j-th column of matrix Q.

Eq. (8) utilizes multi-order similarity to learn low-dimensional latent representations, strating by computing a pairwise similarity matrix $\boldsymbol{S} \in \mathbb{R}^{m \times m}$. Note that pairwise similarity is unable to tackle noise interference and the concentration effect under HDLSS setting. Higher-order similarity is necessary to overcome the aforementioned challenges, to construct such an similarity, it is natural to consider designing composite similarity based on pairwise similarity $\boldsymbol{S}$. Each entry of the composite similarity $\mathcal{T}$ can be defined as:

$$\mathcal{T}_{3ijk} = \boldsymbol{S}_{ij} \boldsymbol{S}_{kj}. \quad (9)$$

However, it is proved by Cai et al. [33], [34] that this tensor similarity has the same structure and properties as pairwise similarity, making it susceptible to noise and indistinguishable for samples with high feature dimensions [35]. To address these drawbacks, a indecomposable tensor similarity is proposed to provide complementary message missed by pairwise similarity. The approach to constructing indecomposable similarity will be detailed in Section IV-B1.

IV. Method

In this section, we present the method for deep semi-supervised learning based on tensor label propagation. Our method consists of two modules: a feature embedding network and label propagation. The former is used to extract deep features from the input data, while the latter integrates both pairwise and high-order information to infer pseudo-labels for unlabeled data. The overall framework of our method is shown in Figure 1.

A. Pretraining stage

Inspired by [30], we use a neural network $f_\theta : \mathcal{X} \to \mathbb{R}^c$ with parameter θ to map the input samples to the label space. This network consists of two parts: feature extraction $\phi : \mathcal{X} \to \mathbb{R}^d$ for extracting deep features $\boldsymbol{V}$, and a classification layer for outputting soft labels with deep features as input. We first randomly initialize the network parameters θ, and cross-entropy loss ℓ_s is introduced to train the network in a fully supervised manner on labeled samples set $\boldsymbol{X}_L$:

$$L_s = \min_{\theta} \sum_{i=1}^{l} \ell_s(f_\theta(\boldsymbol{x}_i), y_i), \quad (10)$$

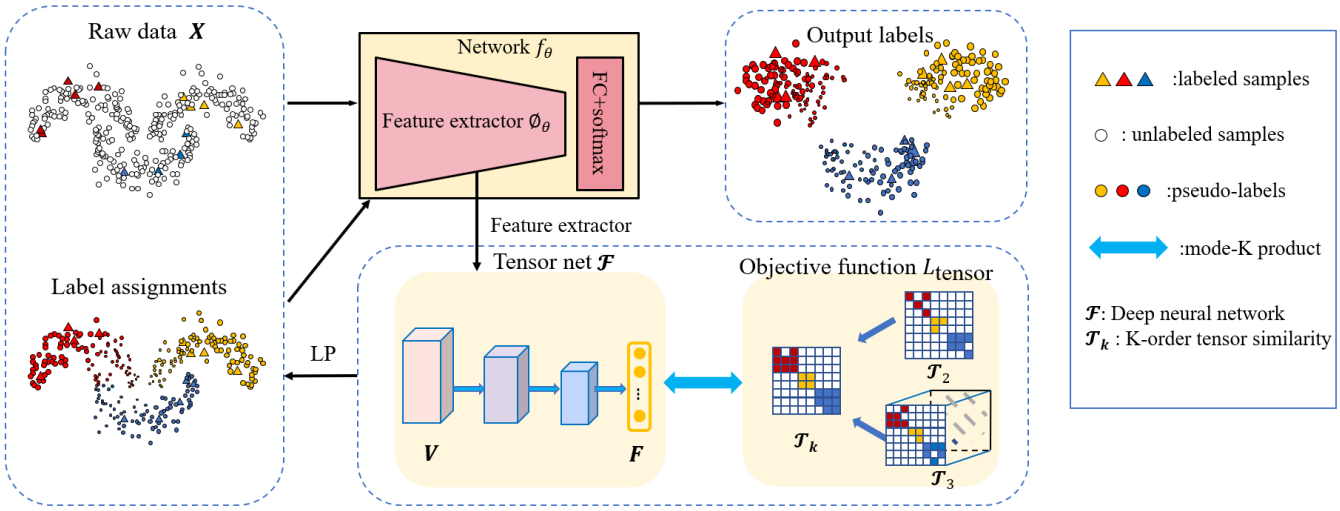


Fig. 1: Description of the TLPDSL method. Starting from a randomly initialized network f_θ, which is trained in a supervised manner on the labeled data. Subsequently, an iterative label propagation is conducted on all samples. The process involves constructing pairwise similarity $\mathcal{T}_2$ and third-order similaritie $\mathcal{T}_3$ from the features obtained during initialization stage, followed by optimizing L_{tensor} with a deep network to obtain pseudo-labels. Finally, f_θ is trained on the entrie dataset using both the labeled data and the unlabeled data with pseudo-labels.

which plays a role in the overall loss during the training of a network in a semi-supervised setup [28]. The function f_θ maps the raw data to confidence scores, and for the i-th sample in the network, the output is defined as $f_\theta(\boldsymbol{x}_i)$. Then, the predicted result is determined by selecting the highest score from the output vector, denoted as:

$$\hat{y}_i = \arg\max_j f_\theta(\boldsymbol{x}_i)_j, \tag{11}$$

with j represents the index of the vector dimension.

B. High-order similarity based Label Propagation

1) formulate tensor similarity: We use the trained feature extraction network ϕ_θ to obtain the feature descriptors $\boldsymbol{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_l, \mathbf{v}_{l+1}, \ldots, \mathbf{v}_n)$. Subsequently, we adopt a third-order indecomposable tensor similarity $\mathcal{T}_3 \in \mathbb{R}^{m\times m\times m}$ proposed by Cai et al. [33] to represent the high-order relationships between the feature descriptors $\boldsymbol{V}$ of m samples, defined by:

$$\mathcal{T}_{3_{ijk}} = 1 - \frac{\langle (\boldsymbol{v}_i - \boldsymbol{v}_j), (\boldsymbol{v}_k - \boldsymbol{v}_j) \rangle}{d_{ij} d_{jk}}, \tag{12}$$

for $i, j, k \in [1, m]$, where d_{ij} is the distance between the samples $\boldsymbol{x}_i$ and the $\boldsymbol{x}_j$, i.e., Euclidean distance. The entry $\mathcal{T}_{3ijk}$ denotes the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$ from the perspective of x_j. An intuitive idea is that, as long as x_k and x_i is close enough, regardless of the position of x_j, similarity will result in a larger value.

2) tensor-based label propagation: The spirit of tensor spectral clustering is used to involve tensor similarity into label propagation process. Specifically, the first term in Eq. (3) $\sum_{i,j=1}^{m} \boldsymbol{A}_{ij} \left\| \frac{\boldsymbol{F}_{i,:}}{\boldsymbol{D}_{ii}} - \frac{\boldsymbol{F}_{j,:}}{\boldsymbol{D}_{jj}} \right\|_2^2$ is replaced by $\frac{1}{\sum_{s=1}^{c} \mathcal{T}^{(3)} \times_1 \boldsymbol{F}_{:,s}^T \times_2 \boldsymbol{F}_{:,s}^T \times_3 \boldsymbol{F}_{:,s}^T}$, which is the inverse of tensor spectral clustering, the operator "$\times_k$" denotes mode-k product given by Eq. (2). Building upon this definition, we further derive the objective function:

$$\begin{aligned} &\min_F L_{tensor}(\boldsymbol{F}) \\ &= \min_F \alpha \frac{1}{\sum_{s=1}^{c} \mathcal{T}_3 \times_1 \boldsymbol{F}_{:,s}^T \times_2 \boldsymbol{F}_{:,s}^T \times_3 \boldsymbol{F}_{:,s}^T + \beta \mathcal{T}_2 \times_1 \boldsymbol{F}_{:,s}^T \times_2 \boldsymbol{F}_{:,s}^T} \\ &+ (1-\alpha) \|\boldsymbol{Y} - \boldsymbol{F}\|_F^2, \end{aligned} \tag{13}$$

where $\boldsymbol{F} \in \mathbb{R}^{m\times c}$ is the classification output matrix, $\boldsymbol{Y} \in \mathbb{R}^{m\times c}$ is the label matrix, α is the balance coefficient. $\mathcal{T}_2$ denotes conventional pairwise similarity that measure the relationship between sample pair $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$. $\mathcal{T}_3$ utilizes third-order similarity to assess the local interactions among multiple samples. The first term is the objective of tensor spectral clustering, which is predicated on a fundamental assumption that samples in close proximity are more likely to share the same label. The second term is to keep the prediction of labeled samples unchanged. By incorporating the third-order tensor $\mathcal{T}_3$ and the second-order tensor $\mathcal{T}_2$, we can achieve a more comprehensive understanding of the underlying structure of the data, allowing for more precise label propagation.

3) Optimization for L_{tensor}: Eq. (13) seems not to have a close-formed solution like Eq. (4) because one can check the derivative of $L_{tensor}(\boldsymbol{F})$ with respect to F as:

$$\frac{\partial L_{tensor}(\boldsymbol{F})}{\partial \boldsymbol{F}} = -\alpha \frac{\boldsymbol{M}}{\sum_{s=1}^{c} \mathcal{T}_3 \times_1 \boldsymbol{F}_{:,s}^T \times_2 \boldsymbol{F}_{:,s}^T \times_3 \boldsymbol{F}_{:,s}^T + \beta \mathcal{T}_2 \times_1 \boldsymbol{F}_{:,s}^T \times_2 \boldsymbol{F}_{:,s}^{T\,2}} + (2-2\alpha)(\boldsymbol{F}-\boldsymbol{Y}), \tag{14}$$

where $\boldsymbol{M}_{:,s} = \boldsymbol{I}_m \mathcal{T}_{3_{(1)}}(\boldsymbol{F}_{:,s} \otimes \boldsymbol{F}_{:,s}) + (\boldsymbol{I}_m \otimes \boldsymbol{F}_{:,s} + \boldsymbol{F}_{:,s} \otimes \boldsymbol{I}_m)^T (\mathcal{T}_{3_{(1)}})^T \boldsymbol{F}_{:,s} + \beta \mathcal{T}_2 \boldsymbol{F}$, $\boldsymbol{T}_{3_{(1)}}$ is the mode-1 unfolding of $\mathcal{T}_3$, and '$\otimes$' represents the Kronecker product.

To address this issue, we construct a label propagation neural network h_ω. The input of the network is the feature matrix $\boldsymbol{V}$, and the output is the pseudo-label matrix $\boldsymbol{F} \in \mathbb{R}^{n \times c}$, where each row of $\boldsymbol{F}$ represents the probability distribution of the corresponding sample on c classes. Let $\boldsymbol{F} = h_\omega(\boldsymbol{V})$, then we use Stochastic Gradient Descent (SGD) to optimize the objective function (13). For each batch, we use the backpropagation algorithm to update the weight parameter matrix ω of the feature embedding network, and then update the pseudo-label matrix $\boldsymbol{F}$. We iterate the optimization process until the maximum number of iterations is reached. For a possible iteration t:

$$\omega^{(t+1)} = \omega^{(t)} - \eta \frac{\partial L_{tensor}(\boldsymbol{F})}{\partial \omega^{(t)}}, \tag{15}$$

where η is the learning rate. By solving Eq. (13), we obtain $\boldsymbol{F}$, which further used to formulate pseudo-label given by Eq. (5).

C. Semi-surpervised learning with multi-order similarities

We input all samples into a feature extraction network pre-trained on labeled data to obtain embedding vectors $\boldsymbol{V}$. Then, we further learn a classification matrix from embedding $\boldsymbol{V}$ by multi-order similarities $\mathcal{T}_2$ and $\mathcal{T}_3$. Finally, label propagation is applied to generate pseudo-labels.

A deep neural network is trained on the labeled samples and unlabeled samples with pseudo-labels as:

$$L_{net} = \min_\theta \sum_{i=1}^{l} \delta_{y_i} \ell_s(f_\theta(\boldsymbol{x}_i), y_i) + \sum_{i=l+1}^{m} \gamma_i \delta_{\hat{y}_i} \ell_s(f_\theta(\boldsymbol{x}_i), \hat{y}_i), \tag{16}$$

where $\hat{y}_i$ is the pseudo-label of the unlabeled sample $\boldsymbol{x}_i$, determined by the index corresponding to the maximum value of the ith row of F. This equation represents the sum of the weighted versions of the supervised loss term Eq. (10) and the pseudo-label loss term. The latter shares similar structure with supervied loss.

Pseudo-label certainty for sample $\boldsymbol{x}_i$ is measured by γ_i as:

$$\gamma_i = 1 - \frac{H(\widetilde{\boldsymbol{F}}_{i,:})}{\log c}, \tag{17}$$

where $\widetilde{\boldsymbol{F}}$ is the row-wise normalize one of F as $\widetilde{\boldsymbol{F}}_{i,j} = \frac{\boldsymbol{F}_{i,j}}{\sum_k \boldsymbol{F}_{i,k}}$ and $H : \mathbb{R}^c \to \mathbb{R}$ is the entropy function $H(\widetilde{\boldsymbol{F}}_{i,:}) = -\sum_{j=1}^{c} \widetilde{\boldsymbol{F}}_{i,j} \log \widetilde{\boldsymbol{F}}_{i,j}$.

Class balance weight δ_{y_i} for class y_i is inversely proportional to class population as:

Algorithm 1 Tensor Label Propagation based Deep SSL

Inputs: Training examples X, labels Y_L
Parameter: θ, epochs T and T', learning rate η, iterations K
Output: Optimized θ

Initialize θ and ω randomly.
for epoch $\in [1, \ldots, T]$ **do**
  $\theta \leftarrow$ OPTIMIZE($L_s(X_L, Y_L; \theta)$)
end for
for epoch $\in [1, \ldots, T']$ **do**
  for $i \in \{1, \ldots, n\}$ **do**
    $v_i \leftarrow \phi_\theta(x_i)$
    $\mathcal{T}_3 \in \mathbb{R}^{n \times n \times n} \leftarrow$ third-order similarity (12)
    $\mathcal{T}_2 \in \mathbb{R}^{n \times n} \leftarrow$ pairwise similarity
    for k $\in [1, \ldots, K]$ **do**
      $F \leftarrow h_\omega(v_i)$ solve (13) with SGD
      $\omega^{(k+1)} \leftarrow$ using backpropagation (15)
    end for
  end for
  for $(i, j) \in U \times C$ **do**
    $\hat{F}_{ij} \leftarrow F_{ij} / \sum_k F_{ik}$
  end for
  for $i \in U$ **do**
    $\hat{y}_i \leftarrow \arg\max_j F_{ij}$
  end for
  for $i \in U$ **do**
    $\gamma_i \leftarrow$ certainty of $\hat{y}_i$ (17)
  end for
  for $j \in C$ **do**
    $\delta_j \leftarrow (|L_j| + |U_j|)^{-1}$
  end for
  $\theta \leftarrow$ OPTIMIZE($L_{net}(X, Y_L, \hat{Y}_U; \theta)$)
end for

$$\delta_{y_i} = \frac{1}{|L_{y_i}| + |U_{y_i}|}, \tag{18}$$

where $|L_{y_i}|$ and $|U_{y_i}|$ denote the number of class y_i in labeled samples and unlabeled samples, respectively.

The workflow of TLPDSL can be summarized as follows: Firstly, a randomly initialized network f_θ is trained on the labeled data to extract feature descriptors $\boldsymbol{V}$, which is leveraged to construct pairwise similarity $\mathcal{T}_2$ and third-order tensor similarity $\mathcal{T}_3$. Subsequently, we utilize a tensor-based label propagation network h_ω to optimize the loss function L_{tensor} and generate pseudo-labels for unlabeled samples (11). Finally, the deep network f_θ is trained on all data, including those labeled and pseudo-labeled, optimizing L_{net} Eq. (16). These steps are repeated for predefined number of iterations. The algorithmic process of our method TLPSDL is shown in Algorithm 1.

V. Experiments

In this section, the performance of the proposed Semi-supervised Label Propagation Deep Learning (TLPDSL)

method was evaluated on HDLSS datasets, comparing it with other state-of-the-art methods. The section begins with an introduction to the datasets and evaluation metrics for the experiment, followed by the implementation details and hyperparameter settings. Subsequently, the classification performance of TLPDSL on various datasets is demonstrated. Finally, the experimental results are discussed.

A. Datasets and Baselines

1) Datasets: Six different public datasets[1] were chosen to evaluate the effectiveness of the proposed method. Five of these datasets suffer from HDLSS problem, namely Colon, ALLAML, Leukemia, Prostate_GE and Lung. To demonstrate the ability of TLPDSL in addressing general problems, we also conducted tests on the image dataset USPS. Details about these datasets are summarized in Table II. To better display our findings, we utilize training, validation, and testing datasets as referenced in [36]. In this work, for each given dataset, the average results of 10 random splits is reported, where 80% of the data is used for training and 20% for testing. In addition, 20% of the samples are labeled.

State-of-the-art semi-supervised label propagation methods are selected as our baselines, these respective methods are:

- MLP is a basic feedforward neural network that uses backpropagation to optimize the model.
- A2LP [32] adapts the feature embeddings of the labeled data by minimizing a differentiable loss function, optimizing their positions in the manifold in the process.
- lapoleaf [37] develops a highly efficient non-iterative label propagation algorithm based on optimal leading forest (LaPOLeaF)
- ILP [38] presents a novel extension of the label propagation algorithm for applications where data samples are observed sequentially.
- GCN [39] learns node representations by aggregating information from nodes and their neighbors, and is widely used in semi-supervised tasks.
- GAT [40] enhances the expressive power of the GCN model by introducing a multi-head attention mechanism and adaptively adjusting weights between nodes.
- LPDSL [30] presents a transductive label propagation method based on the manifold assumption.

B. Experimental Setup

1) Evaluation Metrics: Three common evaluation metrics were used to quantify model performance, namely, accuracy, precision and specificity. In addition, the Receiver Operating Characteristic-Area Under Curve (ROC-AUC) was chosen as the supplementary metric for binary classification tasks, which depicts the model's ability to distinguish between positive and negative instances by plotting sensitivity (True Positive Rate) against False Positive Rate (1 - Specificity) at different classification thresholds.

[1]https://anonymous.4open.science/r/HDLSS_dataset-306E/

2) Implementation Details: The proposed framework consists of two modules: deep feature embedding network f_θ and tensor-based label propagation module. Starting with fully-supervised pre-training stage, where a 13-layer neural network [41] is employed, including a feature extractor ϕ_θ, followed by a fully-connected layer with softmax. An ℓ_2-normalization layer is futher implemented on the output of ϕ_θ to provide unit-norm descriptors for constructing high-order tensor similarity. Then, a label propagation fully-connected neural network h_ω is used to solve Eq. (13), yielding pseudo-labels for unlabeled samples. Lastly, the entire dataset is trained using f_θ.

3) Parameter settings: The network is trained for 180 epochs. Stochastic gradient descent (SGD) [42] with up to 20 iterations is used to solve Eq. (13). After each epoch of semi-supervised learning, we update the pseudo-labels, weights, and similarities. Grid search approach is implemented to explore the task-related hyperparameters α and β from the sets [0.1, 0.3, 0.5, 0.7, 0.9] and [0.01, 0.1, 1, 10, 100] respectively. Cosine annealing is utilized to decay the initial learning rate l_0 from 0.05. Meanwhile, the comparative method will remain unchanged as per the original work setting.

C. Comparison of Classification Performance

Table I presents a comparison of our proposed TLPDSL with state-of-the-art semi-supervised methods.[2] The best results are highlighted in bold. The result demonstrates the significant improvements of TLPDSL in all metrics across most datasets.

Specifically, TLPDSL surpasses LPDSL, which constructs a graph for label propagation based on pairwise similarity. The proposed TLPDSL, in particular, excels at handling HDLSS data, showing significant improvements over LPDSL across all HDLSS datasets while maintaining stable performance. This further implies TLPDSL is able to capture intrinsic structure among data.

A comparison of the proposed TLPDSL with recent label propagation methods is also listed in Table I. TLPDSL not only demonstrates the highest accuracy across all datasets but also maintains a significant margin over the runner-up methods in each case. Specifically, on Colon dataset, the ACC of TLPDSL is over 6% higher than the runner-up MLP. On Leukemia dataset, our proposed method shows a significant improvement of over 10% compared to the second-best method (GCN). With an accuracy of 93.333%, TLPDSL outperforms the second-best method which has an accuracy of 87.586% on ALLAML. On Prostate GE substantially higher than the second-best (LPDSL) at 87.619%, leading by about 7%. Furthermore, despite the well performance of some baseline methods on a few datasets, none of them can handle various types of HDLSS data, indicating their weak robustness under HDLSS setting. This highlights the effectiveness of the high-order information extracted by TLPDSL. Moreover, results on USPS dataset demonstrate that our method is equally capable as SOTA models in handling conventional tasks.

[2]Hardware used: 16-core Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz, NVIDIA(R) A100-40GB

TABLE I: Performance of TLPDSL on all Datasets

Dataset	Method	ACC (%)	Recall	F-Score	AUC	Specificity	Precision
Colon	MLP	75.400	0.723	0.726	0.723	0.722	0.744
	A2LP	62.540	0.625	0.607	0.625	0.779	0.658
	lapoleaf	63.103	0.707	0.684	0.702	0.707	0.720
	ILP	69.230	0.691	0.695	0.698	0.691	0.691
	GCN	70.000	0.755	0.750	0.750	0.750	0.771
	GAT	60.769	0.640	0.607	0.640	0.500	0.640
	LPDSL	67.692	0.670	0.665	0.770	0.670	0.665
	TLPDSL(Ours)	**81.538**	**0.798**	**0.801**	**0.720**	**0.798**	**0.808**
Leukemia	MLP	79.138	0.825	0.786	0.825	0.825	0.795
	A2LP	79.080	0.791	0.784	0.791	0.888	0.827
	lapoleaf	63.103	0.707	0.684	0.702	0.707	0.720
	ILP	70.600	0.706	0.706	0.500	0.500	0.706
	GCN	84.482	0.892	0.866	0.872	0.891	0.847
	GAT	85.335	0.820	0.816	0.820	0.820	0.814
	LPDSL	74.667	0.810	0.745	0.992	0.810	0.786
	TLPDSL(Ours)	**96.000**	**0.970**	**0.958**	**1.000**	**0.970**	**0.955**
ALLAML	MLP	78.793	0.810	0.781	0.810	0.810	0.782
	A2LP	82.130	0.821	0.817	0.821	0.905	0.848
	lapoleaf	69.321	0.764	0.745	0.761	0.764	0.773
	ILP	71.665	0.850	0.775	0.805	0.750	0.716
	GCN	77.586	0.928	0.926	0.900	0.900	0.935
	GAT	75.862	0.672	0.685	0.685	0.672	0.747
	LPDSL	86.667	0.900	0.862	1.000	0.900	0.862
	TLPDSL(Ours)	**93.333**	**0.950**	**0.930**	**1.000**	**0.950**	**0.950**
Prostate GE	MLP	57.805	0.585	0.464	0.585	0.585	0.412
	A2LP	66.444	0.664	0.658	0.664	0.799	0.676
	lapoleaf	64.707	0.717	0.708	0.716	0.717	0.738
	ILP	69.047	0.643	0.653	0.675	0.643	0.670
	GCN	76.543	0.619	0.591	0.632	0.632	0.687
	GAT	71.951	0.712	0.639	0.712	0.712	0.611
	LPDSL	87.619	0.923	0.918	0.923	0.923	0.923
	TLPDSL(Ours)	**94.286**	**0.965**	**0.963**	**0.997**	**0.965**	**0.965**
Lung	MLP	88.466	0.650	0.704	0.941	0.926	0.872
	A2LP	60.200	0.602	0.505	0.751	0.912	0.861
	lapoleaf	61.583	0.839	0.735	0.795	0.839	0.766
	ILP	90.240	0.902	0.903	0.900	0.963	0.902
	GCN	77.914	0.800	0.712	0.882	0.866	0.645
	GAT	89.571	0.802	0.847	0.981	0.946	0.907
	LPDSL	73.659	0.823	0.751	0.972	0.950	0.774
	TLPDSL(Ours)	**95.122**	**0.912**	**0.906**	**0.908**	**0.990**	**0.915**
USPS	MLP	88.774	0.890	0.888	0.987	0.987	0.892
	A2LP	83.94	0.839	0.839	0.911	0.981	0.854
	lapoleaf	72.430	0.779	0.788	0.877	0.779	0.854
	ILP	82.830	0.828	0.828	0.903	0.981	0.828
	GCN	87.750	0.879	0.877	0.989	0.986	0.881
	GAT	89.375	0.894	0.893	0.984	0.988	0.895
	LPDSL	**91.200**	0.905	**0.892**	0.991	0.989	**0.902**
	TLPDSL(Ours)	91.100	**0.906**	0.891	**0.994**	**0.989**	0.900

TABLE II: Statistics on tested datasets

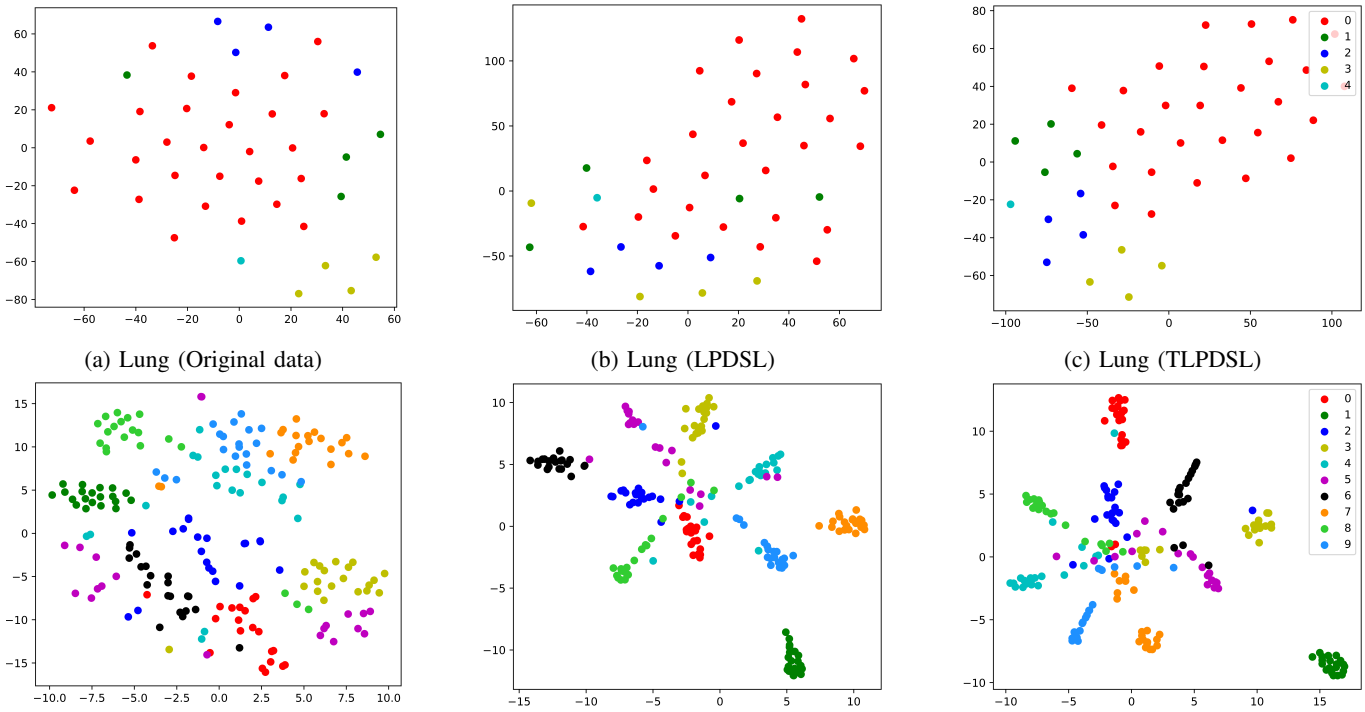
Index	Dataset	Instances	Features	Classes
1	Colon	62	2000	2
2	Leukemia	72	7070	2
3	ALLAML	72	7129	2
4	Prostate GE	102	5966	2
5	Lung	203	3312	5
6	USPS	9298	256	10

To further comprehensively evaluate the performance of TLPDSL, additional metrics such as F-Score is included in our experiments. The results demonstrate that TLPDSL significantly outperforms baseline methods on these supplementary metrics. Indicating that high-order similarity tensors can supplement pairwise similarity to obtain more accurate pseudo-label results, thereby allowing for robust performance under HDLSS setting.

In summary, the experiments show that high-order similarity compensates for the limitations of traditional second-order similarity information, especially in alleviating the curse of dimensionality associated with HDLSS data. Furthermore, by repeatedly interacting with training data, TLPDSL learns to recognize patterns and features in the input data related to recognition, thereby producing more discriminative boundaries.

D. Visualization

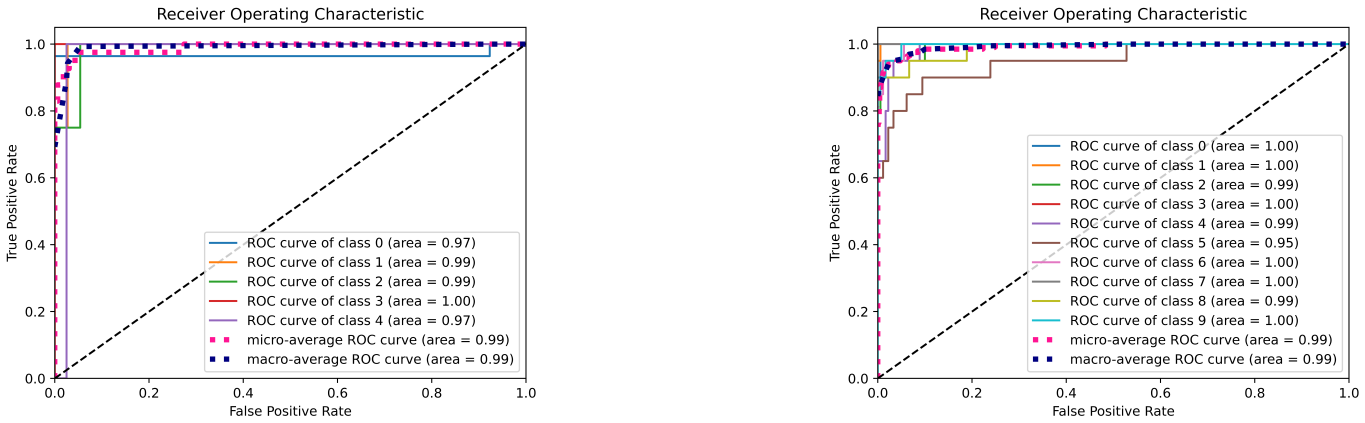
To more intuitively display the experimental results, we utilized t-SNE for visualizing the results of test set on the Lung and USPS datasets. Specifically, we use the last layer embedding of TLPDSL(or LPDSL) and depict the learned representation of test set. As shown in Figure 2, on Lung dataset,



(a) Lung (Original data) (b) Lung (LPDSL) (c) Lung (TLPDSL)

(d) USPS (Original data) (e) USPS (LPDSL) (f) USPS (TLPDSL)

Fig. 2: Visualization of the learned embeddings on Lung and USPS datasets.



(a) Lung (b) USPS

Fig. 3: The illustration of ROC curves of TLPDSL obtained on Lung and USPS for classification

the visualization of TLPDSL demonstrates that data points from different categories are more distinctly clustered, and the boundaries between classes are more pronounced compared to those in LPDSL. This further indicates that TLPDSL can learn meaningful representations of the data. Moreover, on the USPS dataset, the boundaries between classes are still clear, and the inter-cluster distances are relatively large, indicating that TLPDSL is also capable of effectively discriminating between samples in regular datasets. In conclusion, the t-SNE visualization provides compelling evidence of distinctive performance of TLPDSL.

As shown in Figure 3, the proposed TLPDSL model exhibits high discriminability, as evidenced by the ROC curves being closer to the top left corner of the plot. This indicates a higher true positive rate and a lower false positive rate, suggesting that our model has a strong ability to distinguish between different classes.

VI. CONCLUSION

In summary, this paper presents a novel deep semi-supervised learning approach tailored for high-dimensional low-sample size (HDLSS) datasets. The proposed TLPDSL

integrates multiple-order similarities to address the limitations of traditional label propagation in capturing intricate data correlations. Our contributions include the use of deep neural networks for feature extraction and a label propagation network incorporating both low-order and high-order representations. Extensive comparative experiments with advanced label propagation methods indicate that TLPDSL can effectively manage HDLSS data while offering excellent robustness, and it also performs exceptionally well with general data.

REFERENCES

[1] O. Chapelle, B. Sch"olkopf, and A. Zien, Eds., *Semi-supervised learning*, 2006.

[2] W. Dong-DongChen and Z. WeiGao, "Tri-net for semi-supervised deep learning," in *Proceedings of twenty-seventh international joint conference on artificial intelligence*, 2018, pp. 2014–2020.

[3] H. Wu and S. Prasad, "Semi-supervised deep learning using pseudo labels for hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1259–1270, 2017.

[4] D. Wei, Y. Yang, and H. Qiu, "Improving self-training with density peaks of data and cut edge weight statistic," *Soft Computing*, vol. 24, pp. 15 595–15 610, 2020.

[5] D. Wu, M. Shang, X. Luo, J. Xu, H. Yan, W. Deng, and G. Wang, "Self-training semi-supervised classification based on density peaks of data," *Neurocomputing*, vol. 275, pp. 180–191, 2018.

[6] Z.-H. Zhou and M. Li, "Semi-supervised learning by disagreement," *Knowledge and Information Systems*, vol. 24, pp. 415–439, 2010.

[7] D. P. Kingma, S. Mohamed, D. Jimenez Rezende, and M. Welling, "Semi-supervised learning with deep generative models," *Advances in neural information processing systems*, vol. 27, 2014.

[8] X. Zhu, *Semi-supervised learning with graphs*, 2005.

[9] C. Zhuang and Q. Ma, "Dual graph convolutional networks for graph-based semi-supervised classification," in *Proceedings of the 2018 world wide web conference*, 2018, pp. 499–508.

[10] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Seventeenth Annual Conference on Neural Information Processing Systems (NIPS 2003)*, 2004, pp. 321–328.

[11] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[12] F. Dornaika, R. Dahbi, A. Bosaghzadeh, and Y. Ruichek, "Efficient dynamic graph construction for inductive semi-supervised learning," *Neural Networks*, vol. 94, pp. 192–203, 2017.

[13] X. Fang, N. Han, W. K. Wong, S. Teng, J. Wu, S. Xie, and X. Li, "Flexible affinity matrix learning for unsupervised and semisupervised classification," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 4, pp. 1133–1149, 2018.

[14] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proceedings of the 20th International conference on Machine learning (ICML-03)*, 2003, pp. 912–919.

[15] Z. Zhang, F. Li, L. Jia, J. Qin, L. Zhang, and S. Yan, "Robust adaptive embedded label propagation with weight learning for inductive classification," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 8, pp. 3388–3403, 2017.

[16] N. Yang, Y. Sang, R. He, and X. Wang, "Label propagation algorithm based on non-negative sparse representation," in *International Conference on Intelligent Computing for Sustainable Energy and Environment*, 2010, pp. 348–357.

[17] Z. Hua and Y. Yang, "Robust and sparse label propagation for graph-based semi-supervised classification," *Applied Intelligence*, pp. 1–15, 2022.

[18] B. C. Benato, J. F. Gomes, A. C. Telea, and A. X. Falcão, "Semi-supervised deep learning based on label propagation in a 2d embedded space," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 25th Iberoamerican Congress, CIARP 2021, Porto, Portugal, May 10–13, 2021, Revised Selected Papers 25*, 2021, pp. 371–381.

[19] K. Kamnitsas, D. Castro, L. Le Folgoc, I. Walker, R. Tanno, D. Rueckert, B. Glocker, A. Criminisi, and A. Nori, "Semi-supervised learning via compact latent space clustering," in *International conference on machine learning*, 2018, pp. 2459–2468.

[20] Y. Gan, H. Zhu, W. Guo, G. Xu, and G. Zou, "Deep semi-supervised learning with contrastive learning and partial label propagation for image data," *Knowledge-Based Systems*, vol. 245, p. 108602, 2022.

[21] Q. Huang, H. He, A. Singh, S.-N. Lim, and A. R. Benson, "Combining label propagation and simple models out-performs graph neural networks," *arXiv preprint arXiv:2010.13993*, 2020.

[22] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *Database Theory—ICDT 2001: 8th International Conference London, UK, January 4–6, 2001 Proceedings 8*, 2001, pp. 420–434.

[23] D. François, V. Wertz, and M. Verleysen, "The concentration of fractional distances," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 7, pp. 873–886, 2007.

[24] X. Yang, Z. Song, I. King, and Z. Xu, "A survey on deep semi-supervised learning," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[25] D.-H. Lee *et al.*, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on challenges in representation learning, ICML*, vol. 3, no. 2, 2013, p. 896.

[26] Q. Li, L. Chen, S. Jing, and D. Wu, "Pseudo-labeling with graph active learning for few-shot node classification," in *2023 IEEE International Conference on Data Mining (ICDM)*, 2023, pp. 1115–1120.

[27] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, and C.-L. Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *Advances in neural information processing systems*, vol. 33, pp. 596–608, 2020.

[28] W. Shi, Y. Gong, C. Ding, Z. M. Tao, and N. Zheng, "Transductive semi-supervised deep learning using min-max features," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 299–315.

[29] C. Gong, T. Liu, D. Tao, K. Fu, E. Tu, and J. Yang, "Deformed graph laplacian for semisupervised learning," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 10, pp. 2261–2274, 2015.

[30] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Label propagation for deep semi-supervised learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5070–5079.

[31] F. Zhuang and P. Moulin, "Deep semi-supervised metric learning with mixed label propagation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 3429–3438.

[32] M. Lazarou, Y. Avrithis, G. Ren, and T. Stathaki, "Adaptive anchor label propagation for transductive few-shot learning," in *2023 IEEE International Conference on Image Processing (ICIP)*, 2023, pp. 331–335.

[33] H. Cai, Y. Wang, F. Qi, Z. Wang, and Y.-m. Cheung, "Multiview tensor spectral clustering via co-regularization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–14, 2024.

[34] H. Peng, Y. Hu, J. Chen, H. Wang, Y. Li, and H. Cai, "Integrating tensor similarity to enhance clustering performance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2582–2593, 2020.

[35] S. Sarkar and A. K. Ghosh, "On perfect clustering of high dimension, low sample size data," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 9, pp. 2257–2272, 2019.

[36] B. Liu, Y. Wei, Y. Zhang, and Q. Yang, "Deep neural networks for high dimension, low sample size data." in *IJCAI*, 2017, pp. 2287–2293.

[37] J. Xu, T. Li, Y. Wu, and G. Wang, "Lapoleaf: Label propagation in an optimal leading forest," *Information Sciences*, vol. 575, pp. 133–154, 2021.

[38] I. Chiotellis, F. Zimmermann, D. Cremers, and R. Triebel, "Incremental semi-supervised learning from streams for object classification," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 5743–5749.

[39] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[40] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[41] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," *arXiv preprint arXiv:1610.02242*, 2016.

[42] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.