Auto-partial Multi-view culture for Image Clustering and Semi-supervised classification

Mr.Gangaraju Raghavendra Datta Bhargava Sriram[1]

Mr.Mohammed Khaja Amir Kafeel[2]

Application Development Associate,Accenture Solutions Pvt. Ltd.
Hitech city, Madhapur,Hyderabad – 500081, Telangana, India.

Abstract**:** Due to the efficiency of learning relationships and complex structures hidden in data, graph-oriented methods have been widely investigated and achieve promising performance. Generally, in the field of multi-view learning, these algorithms construct informative graph for each view, on which the following clustering or classification procedure are based. However, in many real world dataset, original data always contain noise and outlying entries that result in unreliable and inaccurate graphs, which cannot be ameliorated in the previous methods. In this paper, we propose a novel multi-view learning model which performs clustering/semi-supervised classification and local structure learning simultaneously. The obtained optimal graph can be partitioned into specific clusters directly. Moreover, our model can allocate ideal weight for each view automatically without additional weight and penalty parameters. An efficient algorithm is proposed to optimize this model. Extensive experimental results on different real-world datasets show that the proposed model outperforms other state-of-the-art multi-view algorithms.

Index Terms**:** Auto-Weight Learning, Multi-view Clustering, Semi-supervised Classification.

I. INTRODUCTION

With the development of technology, the rate of gathering and accumulating information has reached an unprecedented level. Numerous data that contain heterogeneous features representing objects from different views have arisen in many scientific fields, such as computer vision, genetics, data mining, pattern recognition, etc. For example, in visual data, an image could be represented by different descriptors, such as SIFT [1], HOG [2], GIST [3], LBP [4], CENTRIST [5], Colour Moment [6]; in biological data, each human gene can be measured by gene expression, Array-comparative genomic hybridization (aCGH), Single-nucleotide polymorphism(SNP) and methylation; for a specific scientific paper, its keywords and citations can be regarded as two separate views. It might be satisfying for an individual view of data to accomplish some work, such as clustering, classification, regression, but methods that properly combine many views which contain different fractional information will improve the final performance. Numerous multi-view learning approaches have been proposed in the literature.

In semi-supervised learning domain [7], [8], Co-training [9] is a representative paradigm. It firstly trains two classifiers with labeled data, and classifies the unlabeled data separately. Next some predicted data that are of most confidence are added to the other classifier's training set, then the procedure repeats. [10] proposed an alignment-based semi-supervised learning model to classify gene expression data samples by seeking an optimal alignment between different samples' probe series.

In unsupervised learning domain, multi-view learning methods could be divided into three main categories: tensor-based methods, subspace-based methods, and graph-based methods. Tensor-based methods are powerful to analyze multi-view data's latent pattern. They model multi-view data as a tensor and discover latent pattern hidden in multi-view data, each view can be seen as a slice of the tensor. It has been successfully applied to many domains such as data mining, web search, image recognition and scientific computing [11], [12], [13], [14], [15]. Subspace-based approaches are based on the assumption that the views are generated from a single latent source, and the variation within the views is independent with such latent source. Graph-based methods have been widely investigated in many research and bring many state-of-the-art multi-view clustering methods.

II. RELATED WORK

A. semi-supervised classification

Under the manifold assumption, graph-based methods regard labeled and unlabeled examples as vertices of a graph and utilize edges to propagate information from labeled ones to unlabeled ones. [16] introduced an adaptive multi-modal semisupervised classification (AMMSS) algorithm which considers each type of feature as one modality, it learns a shared class indicator matrix and weights for different modalities. [17] use sparse weights to linearly combine different graphs to implement label propagation (SMGI). Multiple kernel learning methods are naturally combined with multi-view data. [18] learned a kernel matrix by solving the semidefine programming problem; [19] formulated the multiple kernel learning as an efficient semi-infinite linear program; [20] proposed a new kernel fusion scheme by optimizing the L_2-norm of multiple kernels in bioinformatics.The unknown-sample-oriented methods are really needed in real world applications. Based on cotraining method, [21] proposed a co-regularization approach to learn a multi-view classifier from partially labeled data based on the view consensus. A similar approach investigating a semi-supervised least squares regression algorithm has been proposed in [22]. This kind of methods first train multiple classifiers for different type features, and then maximize the consistency among all of the views by punishing the disagreement among unlabeled data.

B. clustering

In higher-order data sets, multi-linear structures can be captured by tensor decompositions for tensors are higherorder generalizations of matrices. Liu et al. [23] proposed a tensor-based multi-view clustering framework, in which two new formulations are developed: modeling the clustering work based on the integration of the Frobenius-norm objective function, or based on matrix integration in the Frobenius-norm objective function. Selee et al. [24] introduce a new tensor decomposition called Implicit Slice Canonical Decomposition (IMSCAND) in which each similarity is stored as a slice in a tensor. In [25], Cao et al. discovery a lower-rank approximation of the original tensor data though a `$_1$-norm optimization function and then compute high-order singular value decomposition of such approximate tensor to obtain the final clustering results. However, the tensor factorization methods emphasize the consistence eigenvectors across different views, following by *K*-means [26] step on eigenvectors, the final clustering labels consistence through different views cannot be ensured.

Subspace-based methods are often optimized by learning to discriminate each view with the shared variable independently and then updating the parameters for the shared space. [27] proposed a convex formulation of multi-view subspace learning method which can be solved efficiently. [28] applied largemargin principle to learn a latent space and show the better results. In order to address insufficiency in each individual view, [29] discovered a latent intact representation of the data and integrate the encoded complementary information. Cao et al. [30] proposed a multi-view clustering framework utilizing the Hilbert Schmidt Independence Criterion (HSIC) as a diversity term to ensure the complementarity of different views. Gao et al. proposed a multi-modal subspace clustering model that perform subspace clustering on different modality respectively and then unify them. Chaudhuri et al. [31] proposed multiview clustering method via Canonical Correlation Analysis (CCA), it computes two sets of variables and maximizes the correlation between them in the embedded space, but it only captures the pairwise correlations between different views, the high order correlations underlying the multiple views are ignored.

Graph-based methods are pretty conspicuous for efficiency and excellent clustering performance [32], [33], [34]. Kumar et al. [35] proposed a co-regularized approach for multi-view spectral clustering in which they co-regularize the clustering hypotheses to make different graphs agree with each other. Cai et al. [36] proposed multi-modal spectral clustering (MMSC) algorithm to integrate heterogeneous image feature, it learns a commonly shared Laplacian matrix by unifying different modals and add a non-negative relaxation to improve the robustness of image clustering. Li et al. [37] proposed a new large-scale multi-view spectral approach (MVSC) based on bipartite graph, it's computational complexity is nearly closed to linear to the number of data points. However,as previously mentioned, almost all of these methods have at least two problems, i.e. unreliable similarity matrix and improper neighbor assignment. These problems make the similarity matrix can't be fully relied, and eventually lead to suboptimal result.

Although graph-based multi-view learning methods achieve state-of-the-art performance, there still exist some limits. For one thing, such methods conduct the following procedure base on the constructed similarity matrix from original data but rarely modify it. Real world datasets always contain noise and outlying entries that result in the unreliable similarity matrix which will impair the finally performance. For another, those methods combining different views often have additional weight parameters to set, which is unsatisfactory especially in unsupervised clustering task. In this paper, we propose an auto-weighted graph-based multi-view learning approach, It is worthwhile to summarized the main contributions of this paper as follows:

1. There is no explicit weight parameter for each view, our approach can learn the weight factors automatically after finite iterations. It's nearly free of parameter so that be more practical to deal with real world application.
2. The proposed approach performs multi-view clustering/semi-supervised classification and local structure learning simultaneously. It adaptively learns local manifold structure, thus can update the graph to the ideal one for clustering.
3. A reasonable constraint is introduced to the approach. The similarity matrix obtained by local structure can be more accurate when we constrain similarity matrix to make it contain exact *c* connected components.
4. Comprehensive experiments on several real-world data sets show the effectiveness of the proposed approach, and demonstrate the advantage over other state-of-theart methods.

The rest of the paper is organized as follows. In Section III, we will propose the Multi-view learning with adaptive neighbors (MLAN) framework. In Section IV, we give an efficient algorithm to tackle the problem and some analysis of the algorithm. In Section V, we perform sound experiments on some state-of-the-art methods. At last, the conclusions and future work are presented in Section VI.

Notations are summarized here throughout the paper. All the matrices are written as uppercase. For a matrix $M \in \mathrm{R}^{n\times d}$, the *i*-th row and the (*ij*)-th element of *M* are denoted by m_i and m_{ij}, respectively. The transpose of matrix *M* is denoted by M^T. The trace of matrix *M* is denoted by $Tr(M)$. The `$_2$-norm of vector v is denoted by kvk$_2$. 1 denotes a column vector with all the elements as one, and the identity matrix is denoted by *I*. x and $\sigma(x)$ denote the average value and standard deviation of vector x, respectively.

III. METHODOLOGY

In this section, we will first introduce the assignment of adaptive neighbors; optimal similarity matrix can be directly partitioned into several cluster whose number is just equal to the number of data class, without *K*-means procedure that other spectral methods adopt. Then we address the issue of acquiring optimal linear combination of multiple graphs, the weight coefficient and corresponding penalty parameter can all be omitted.

A. Adaptive Local Structure Learning

One of the important factors of the graph-based methods' success is the preserving local manifold structure [38], [39], [40], high-dimensional data is considered to contain low dimensional manifold structure, so the obtained similarity matrix is crucial to the ultimate performance. Given a set of data points $\{x_1,x_2,\cdots,x_n\}$, denote data matrix $X \in \mathrm{R}^{n\times d}$, where n

is the number of data points and d is the dimension of features, we adopt the data preprocessing proposed in [41]. In details, $x_i \leftarrow (x_i - x))/\sigma(x)$. For each data point x_i, it belongs to one of the c classes, and can be connected by all the data points with the probability s_{ij} [42], and such probability can be seen as the similarity between them. Closer samples should have larger probability, thus s_{ij} has the negative correlation with the distance between x_i and x_j. The determination of probability s_{ij} can be seen as solving following problem:

$$\min_{s_i \in \mathbb{R}^{n\times 1}} \sum_{i,j}^{n} ||x_i - x_j||_2^2 s_{ij} + \alpha ||S||_F^2$$
$$s.t. \quad \forall i, s_i^T \mathbf{1} = 1, 0 \le s_{ij} \le 1 \qquad (1)$$

Where s_i is a vector with j-th element as s_{ij} in similarity matrix S. The second item is added for the consideration that there would be a trivial solution where only the nearest data point to the x_i is assigned probability 1 and all the other points' similarity would be 0 without such penalty item. In spectral analysis, $L_S = D_S - (S^T + S)/2$ is called Laplacian matrix, where the degree matrix D_S in $S \in \mathrm{R}^{n\times n}$ is the diagonal matrix whose i-th diagonal element is $^{\mathrm{P}}{}_j$ $(s_{ij} + s_{ji})/2$. Given the class indicator matrix $F = [f_1, \cdots, f_n]$, classical spectral clustering can be written as

$$\min_{F \in \mathbb{R}^{n\times c}, F^T F = I} Tr(F^T L_S F) \qquad (2)$$

The ideal neighbor assignment is that the data has exact c connected components for the clustering task which aims to partition the data into c clusters. Usually the neighbor assignment with Eq. (1) cannot reach the ideal case for any value of α. Upon most occasions, all the samples are connected as just one connected component. For the sake of achieving such goal, the probabilities s_{ij} in the Eq. (1) should be constrained so that the neighbor assignment becomes an adaptive process. It seems an impossible goal since such kind of structured constraint on the similarity matrix S is fundamental but also very difficult to handle. Nevertheless, we introduce a reasonable rank constraint to achieve this goal inspired by the important property of Laplacian matrix [43], [44]:

Theorem 1. *The multiplicity c of the eigenvalue 0 of the Laplacian matrix L_S (nonnegative) is equal to the number of connected components in the graph with the similarity matrix S.*

In view of the above consideration, we add a rank constrain to the L_S in problem 1 according to the Theorem 1:

$$\min_{s_i \in \mathbb{R}^{n\times 1}} \sum_{i,j}^{n} ||x_i - x_j||_2^2 s_{ij} + \alpha ||S||_F^2 \qquad (3)$$
$$s.t. \quad \forall i, s_i^T \mathbf{1} = 1, 0 \le s_{ij} \le 1, rank(L_S) = n - c$$

We assign adaptive neighbors to each of samples, which means that the similarity between data points will change, so similarity matrix S will be modified until it contains exact c connected component. Namely, not only the indicator matrix F can be learned, different from the traditional spectral clustering methods, our model can also learn similarity matrix S simultaneously. The learned S can be used for clustering directly according to Tarjan's strongly connected components algorithm [45].

B. Multi-view Data Fusion

For multi-view data, denote $\underset{v}{X_1, X_2, \cdots, X_v}$ be the data matrix of each view. $X_v \in \mathrm{R}^{n\times d}$, where n is the number of data and d^v is the feature dimension of the v-th view. As for graph-based methods, each view can construct similarity graph and maximize the performance quality on its own. In the context of multi-view clustering, there is an inherent problem that all methods have to deal with elaborately: when maximizing the within-view clustering quality, the clustering consistency across different views should be taken into consideration. The rough way that combining multiple views directly through similarity matrix addition or feature concatenation would not help improve the clustering performance, for fallible similarity matrix could lead to suboptimal result. A more reasonable manner is to integrate these views with suitable weights $w_v(v = 1, \cdots, V)$, and an extra regularization parameter γ is needed to keep weights distribution smooth:

$$\min_{S, w_v} \sum_v (w_v \sum_{i,j} ||x_i^v - x_j^v||_2^2 s_{ij}) + \gamma ||w_v||_2^2) + \alpha ||S||_F^2$$
$$s.t. \quad s_i^T \mathbf{1} = 1, 0 \le s_{ij} \le 1, w^T \mathbf{1} = 1, 0 \le w_v \le 1,$$
$$rank(L_S) = n - c \qquad (4)$$

(4) For unsupervised learning methods, the less parameter to be set, the strong robustness they possess. On the other hand, since parameters can be searched in a large range, methods with parameters like the above form often show better result than parameter-free methods. It's really elusive to pursue good performance while rely less on parameter searching. However, we will propose one to alleviate such challenging problem in the next section.

In our previous work [46], we have adopted the root function to integrate different graphs. One can ask what if adopting other functions. In this paper, we explore a series of exponential functions and propose a new multi-view learning with adaptive neighbor's method as the following form:

$$\min_S \sum_v (\sum_{i,j} ||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{p}{2}} + \alpha ||S||_F^2, \qquad (5)$$
$$s.t. \quad s_i^T \mathbf{1} = 1, 0 \le s_{ij} \le 1, rank(L_S) = n - c$$

Where each view shares the same similarity matrix, thus the goal of assigning each data point to the most suitable cluster in each view and ensuring clustering consistency across views is achieved. With the change of the value of $p(0 < p < 2)$, a series of exponential function could be utilized. There is no weight hyper parameter explicitly defined in our model. The Lagrange function of Eq. (5) can be written as

$$\sum_v (\sum_{i,j} ||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{p}{2}} + \alpha ||S||_F^2 + \mathcal{G}(\Lambda, S) \qquad (6)$$

where Λ is the Lagrange multiplier, G(Λ,S) is the formalized term derived from constraints. Taking the derivative of Eq. (6) w.r.t S and setting the derivative to zero, we have

$$\sum_v w_v \frac{\partial \sum_{i,j} ||x_i^v - x_j^v||_2^2 s_{ij}}{\partial S} + \frac{\alpha \partial ||S||_F^2}{\partial S} + \frac{\partial \mathcal{G}(\Lambda, S)}{\partial S} = 0 \qquad (7)$$

Where

$$w_v = \frac{p}{2(\sum_{i,j} ||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{2-p}{2}}} \qquad (8)$$

We can see that w_v is dependent on the target variable S, so that Eq. (7) cannot be directly solved. But if w_v is set to be stationary, Eq. (7) can be considered accounting for following problem

$$\begin{aligned} \min_S \quad & \sum_v w_v \sum_{i,j} ||x_i^v - x_j^v||_2^2 s_{ij} + \alpha ||S||_F^2, \\ s.t. \quad & s_i^T \mathbf{1} = 1, 0 \leq s_{ij} \leq 1, rank(L_S) = n - c \end{aligned} \tag{9}$$

Under the assumption that w_v is stationary, the Lagrange function of Eq. (5) also apply to Eq. (9), if we calculate S from Eq. (9), the value of w_v can be updated correspondingly, which inspires us to optimize Eq. (5) in an alternative way. After optimization, S tune to be S_b, according to Eq. (7), S_b is as least a local optimal solution to problem (5). Similarly, w_v tune to be $\widehat{w_v}$, and they are exactly the learned weights which linearly combining different graphs.

IV. OPTIMIZATION ALGORITHM

To solve the challenging problem (5), we should solve problem (9) iteratively. In the iterative procedure, parameters are updated one by one. The specific parameter updated in the last step could be seen as a constant during current step.

A. Clustering

Denote $\sigma_i(L_S)$ is the i-th smallest eigenvalue of L_S, because L_S is positive semi-definite, $\sigma_i(L_S) \geq 0$. So the constraint rank $(L_S) = n{-}c$ will be ensured if $\sum_{i=1}^{c} \sigma_i(L_S) = 0$. According to Ky Fan's Theorem [47],

We have

$$\sum_{i=1}^{c} \sigma_i(L_S) = \min_{F \in \mathbb{R}^{n \times c}, F^T F = I} Tr(F^T L_S F) \tag{10}$$

Then problem (9) is equivalent to the following problem

$$\begin{aligned} & \min_{S,F} \sum_v w_v \sum_{i,j} ||x_i^v - x_j^v||_2^2 s_{ij} + \alpha ||S||_F^2 + 2\lambda Tr(F^T L_S F) \\ & s.t. \quad s_i^T \mathbf{1} = 1, 0 \leq s_{ij} \leq 1, F^T F = I \end{aligned} \tag{11}$$

where λ is a very large number, the optimal solution to the problem (11) will make equation $\sum_{i=1}^{k} \sigma_i(L_S) = 0$ hold.

1) Fix S, update w_v and F: When S is fixed, we can easily calculate the value of w_v by Eq. (8). So the first and second item of problem (11) could be seen as constant, then it transforms into:

$$\min_{F \in \mathbb{R}^{n \times c}, F^T F = I} Tr(F^T L_S F) \tag{12}$$

The optimal solution F is formed by the c eigenvectors corresponding to the c smallest eigenvalues of L_S.

2) Fix w_v and F, update S: Since w_v is fixed, the first item of Eq. (9) can be replaced as $\sum_{i,j} \sum_v w_v ||x_i^v - x_j^v||_2^2 s_{ij}$. Denote $d_{ij}^x = \sum_v w_v ||x_i^v - x_j^v||_2^2$, which represents the weighted distance between data points x_i and x_j. Then the problem (9) becomes

$$\begin{aligned} \min_S \quad & \sum_{i,j} (d_{ij}^x s_{ij} + \alpha s_{ij}^2) + 2\lambda Tr(F^T L_S F) \\ s.t. \quad & s_i^T \mathbf{1} = 1, 0 \leq s_{ij} \leq 1 \end{aligned} \tag{13}$$

There is an elementary but very important equation in spectral analysis

$$\sum_{i,j} ||f_i - f_j||_2^2 s_{ij} = 2Tr(F^T L_S F) \tag{14}$$

Denote $d_{ij}^f = ||f_i - f_j||_2^2$, note that the problem (13) is independent between different i, we can deal with following problem individually for each i:

$$\begin{aligned} \min_{s_i} \quad & \sum_{j=1}^{n} (d_{ij}^x s_{ij} + \alpha s_{ij}^2 + \lambda d_{ij}^f s_{ij}) \\ s.t. \quad & s_i^T \mathbf{1} = 1, 0 \leq s_{ij} \leq 1 \end{aligned} \tag{15}$$

Denote $d_i \in \mathbb{R}^{n \times 1}$ is a vector with the j-th element as $d_{ij} = d_{ij}^x + \lambda d_{ij}^f$, then the above problem can be written as follow:

$$\min_{s_i} \quad ||s_i + \frac{1}{2\alpha} d_i||_2^2 \quad s.t. \quad s_i^T \mathbf{1} = 1, 0 \leq s_{ij} \leq 1 \tag{16}$$

The intermediate variable α can be determined using the number of adaptive neighbours, by saying adaptive, we mean that the k nearest neighbours to any data point x_i are not steady, they change in every iteration since the weighted distance d^x_{ij} between every pair of x_i and x_j is updated. The determination of the α value will be described in the subsection IV-C.

B. Semi-supervised Classification

Denote l and u are the number of labeled and unlabeled points. Denote $Y_l = [y_1, \cdots, y_l]^T$, where $y_i \in R^{c \times 1}$ is the known indicator vector for the i-th sample, y_i is one-hot and the element $y_{ij} = 1$ means that the i-th sample belongs to the j-th class. Without loss of generality, we rearrange all the points and let the front l points be labeled. We split L_S and F into blocks, so they could be expressed respectively as $L_S = \begin{bmatrix} L_{ll} & L_{lu} \\ L_{ul} & L_{uu} \end{bmatrix}$ and $F = [F_l; F_u]$, $F_l = Y_l$. The optimization procedure is just the same as clustering depicted above, the only difference is updating the class indicator matrix F. When λ is a very large number, problem (9) is equivalent to the following problem

$$\begin{aligned} & \min_{S,F} \sum_v w_v \sum_{i,j} ||x_i^v - x_j^v||_2^2 s_{ij} + \alpha ||S||_F^2 + 2\lambda Tr(F^T L_S F) \\ & s.t. \quad s_i^T \mathbf{1} = 1, 0 \leq s_{ij} \leq 1, F_l = Y_l \end{aligned} \tag{17}$$

It could be written as

$$\min_{F \in \mathbb{R}^{n \times c}, F_l = Y_l} Tr(F^T L_S F) \tag{18}$$

According to the [48], the optimal solution to problem (18) can be calculated as

$$F_u = -L_{uu}^{-1} L_{ul} Y_l \tag{19}$$

After iteration, the final single class label could be assigned to unlabeled data points by following decision function

$$y_i = \arg\max_j F_{ij}, \quad \forall i = l+1, l+2, ..., n. \forall j = 1, 2, ..., c \tag{20}$$

By iteratively solving problem (9), the final S and F in the objective function Eq. (5) can be obtained and could be used for clustering and classification respectively. The Algorithm is summarized in Alg. 1.

C. Determine α using Adaptive Neighbors

The value of regularization parameter α could be from zero to infinite, it's difficult to tune in experiment. Let us recall the original intention of introducing parameter α. In problem (1), it determines number of the neighbor to data pointx_i: neighbor number will be one if α equal to zero, $n - 1$ if α becomes infinite. We assign k nearest neighbors to each point, for any x_i, the Lagrangian Function of problem (16) is:

$$\mathcal{L}(s_i, \phi, \varphi_i) = \frac{1}{2}||s_i + \frac{1}{2\alpha_i}d_i||_2^2 - \phi(s_i^T\mathbf{1} - 1) - \varphi_i^T s_i \tag{21}$$

where $\phi, \varphi_i \geq 0$ are Lagrangian multipliers and $d_{ij} = \sum_v w_v||x_i^v - x_j^v||_2^2 + \lambda||f_i - f_j||_2^2$. According to KKT condition [49], the optimal solution of s_i is:

$$s_{ij} = (-\frac{d_{ij}}{2\alpha_i} + \phi)_+ \tag{22}$$

where $\phi = \frac{1}{k} + \frac{1}{2k\alpha_i}\sum_{j=1}^k d_{ij}$ [42]. That x_i have k neighbors can be translate into $s_{ij} > 0, \forall 1 \leqslant j \leqslant k$ and $s_{i,k+1} = 0$. According to Eq. 22 and substitution ϕ, we have

$$\frac{k}{2}d_{ik} - \frac{1}{2}\sum_{j=1}^k d_{ij} < \alpha_i \leq \frac{k}{2}d_{i,k+1} - \frac{1}{2}\sum_{j=1}^k d_{ij} \tag{23}$$

Where $d_{i1}, d_{i2}, \cdots, d_{in}$ are sorted in ascending order. Hence, to make most of s_i has exact k non-zeros elements, we let α_i

```
Algorithm 1 Multi-view Learning with Adaptive Neighbors (MLAN)
Input:
  $X = \{X_1, X_2, \cdots, X_v\}, X_v \in \mathbb{R}^{n \times d^v}$, number of classes
  $c$, parameter $\lambda$, label matrix $F_l$.
Output:
  Clustering: similarity matrix $S \in \mathbb{R}^{n \times n}$ with exact $c$ connected components
  Classification: the predicted label matrix $F \in \mathbb{R}^{n \times c}$ for all data points.
  Initial the weight for each view, $w_v = \frac{1}{v}$, then each row $s_i$ of $S$ can be initialized by solving the following problem:
  $\min_{s_i^T\mathbf{1}=1, 0 \leq s_{ij} \leq 1} \sum_{j=1}^n (w_v \sum_v ||x_i^v - x_j^v||_2^2 s_{ij} + \alpha||S||_F^2.$
  repeat
    Update $w_v$ by using Eq. (8)
    Clustering: update $F$ by solving the problem (12)/ Classification: update the unlabeled fraction of $F$ by Eq. (19)
    Update each row of $S$ by solving the problem (16)
  until converge
  Classification: Assign the single class label to unlabeled point by Eq. (20).
```

Equal to the right item and set the final α be the average of them:

$$\alpha = \frac{1}{n}\sum_{i=1}^n \alpha_i = \frac{1}{n}\sum_{i=1}^n (\frac{k}{2}d_{i,k+1} - \frac{1}{2}\sum_{j=1}^k d_{ij}) \tag{24}$$

So α can be determined using the number of adaptive neighbors, by saying adaptive, we mean that the k nearest neighbors to any data point x_i are not steady, they change in every iteration since the weighted distance d^x_{ij} between every pair of x_i and x_j is updated. By iteratively solving problem (9), the final S and F in the objective function Eq. (5) can be obtained and could be used for clustering and classification respectively. The Algorithm is summarized in Alg. 1.

D. Convergence Analysis

The proposed algorithm can find a local optimal solution, to prove its convergence, we need to utilize the lemma introduce by [50], [51]: Lemma 1 For any positive real number a and b, the following inequality holds:

$$a^{\frac{p}{2}} - \frac{p}{2}\frac{a}{b^{\frac{2-p}{2}}} \leq b^{\frac{p}{2}} - \frac{p}{2}\frac{b}{b^{\frac{2-p}{2}}} \tag{25}$$

Theorem 2. *In Alg.* 1*, updated S will decrease the objective value of problem (5) until converge ,*

Proof. Suppose the updated S is S_e in each iteration, it's easy to know that:

$$\begin{aligned}&\sum_v \frac{p}{2}\frac{\sum_{i,j}||x_i^v - x_j^v||_2^2\widetilde{s_{ij}}}{(\sum_{i,j}||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{2-p}{2}}} + \alpha||\widetilde{S}||_F^2 \\ \leq &\sum_v \frac{p}{2}\frac{\sum_{i,j}||x_i^v - x_j^v||_2^2 s_{ij}}{(\sum_{i,j}||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{2-p}{2}}} + \alpha||S||_F^2\end{aligned} \tag{26}$$

According to Lemma 1, we have

$$\begin{aligned}&\sum_v(\sum_{i,j}||x_i^v - x_j^v||_2^2\widetilde{s_{ij}})^{\frac{p}{2}} - \sum_v \frac{p}{2}\frac{\sum_{i,j}||x_i^v - x_j^v||_2^2\widetilde{s_{ij}}}{(\sum_{i,j}||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{2-p}{2}}} \\ \leq &\sum_v(\sum_{i,j}||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{p}{2}} - \sum_v \frac{p}{2}\frac{\sum_{i,j}||x_i^v - x_j^v||_2^2 s_{ij}}{(\sum_{i,j}||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{2-p}{2}}}\end{aligned}$$

(27) Sum over Eq. (26) and Eq. (27) in the two sides, we arrive at:

$$\begin{aligned}&\sum_v(\sum_{i,j}||x_i^v - x_j^v||_2^2\widetilde{s_{ij}})^{\frac{p}{2}} + \alpha||\widetilde{S}||_F^2 \\ \leq &\sum_v(\sum_{i,j}||x_i^v - x_j^v||_2^2 s_{ij})^{\frac{p}{2}} + \alpha||S||_F^2\end{aligned} \tag{28}$$

Which completes the prove.

E. Connected to Spectral Clustering

Given a graph with the similarity matrix S, spectral clustering is to solve the problem 2, usually, the obtained F cannot be directly used for clustering since the graph with S does not has exact c connected components. K-means or other discretization procedures should be performed on F to obtain the final clustering results [52]. In the convergence of Algorithm 1, we also obtain an optimal solution F to the problem 2, the difference is that the similarity S is also learned by the algorithm. Thanks to the constraint rank(L_S) = $n-c$, the graph with the learned S will has exact c connected components. The proposed algorithm learns the similarity matrix S and the indicator matrix F simultaneously, while traditional spectral clustering only learns the F. Although the computational burden maybe increase $O(n^2d + tcn^2)$, where t is the number of iteration steps, our new algorithm could achieve better performance since the improvement of the similarity matrix. The future work could be making this technique applied on very large-scale datasets.

V. EXPERIMENT

Since our MLAN is kind of graph-based learning model, we will perform the proposed methods on four benchmark data sets, compared with other related graph based state-of-theart multi-view clustering and semi-supervised classification methods.

A. Data Set Descriptions

MSRC-v1 data set contain 240 images in 8 class as a whole. Following [36], we select 7 classes composed of tree, building, airplane, cow, face, car, bicycle and each class has 30 images. We extract three visual features from each image: colour moment (CM) with dimension 24, GIST with 512 dimension, CENTRIST feature with 254 dimension, and local binary pattern (LBP) with 256 dimension.

Handwritten numerals (HW)data set is comprised of 2,000 digital images, 200 images for each class from 0 to 9. There are Six public features are available: 76 Fourier coefficients of the character shapes (FOU), 216 profile correlations (FAC), 64 Karhunen-love coefficients (KAR), 240 pixel averages in 2 × 3 windows (PIX), 47 Zernike moment (ZER) and 6 morphological (MOR) features.

Caltech101 is an object recognition data set containing 101 categories of images. We follow previous work [37] and select the widely used 7 classes, i.e. Dolla-Bill, Face, Garfield, Motorbikes, Snoopy, Stop-Sign and Windsor-Chair and get 1474 images. Six features are extracted from all the images: i.e. 48 dimension Gabor feature, 40 dimension wavelet moments, 254 dimension CENTRIST feature, 1984 dimension HOG feature, 512 dimension GIST feature, and 928 dimension LBP feature.

NUS-WIDE is a real-world web image dataset for object recognition problem. We select the front 25 from the all 31 categories in alphabetical order (bear, bird, ... ,tower), and choose the first 120 images for each class. Five low-level features are extracted to represent each image: 64 color histogram, 144 color correlogram, 73 edge direction histogram, 128 wavelet texture, and 225 block-wise color moment.

B. Evaluation Metric

For classification task, the proportion of correct-classified data points, namely accuracy (ACC) is adopted to measure each method's performance; for clustering task, there evaluation metric are adopted to measure the performance: accuracy, normalized mutual information (NMI), and purity. As to each dataset, supposing ground-truth labels μ with c classes and clustering result labels v with $c\hat{}$ classes.

Denote μ_i and v_i as the corresponding ground truth label and clustering result label of any data sample x_i, and $\delta(x,y) = 1$ if $x = y$; $\delta(x,y) = 0$ otherwise. Then ACC is defined as follows:

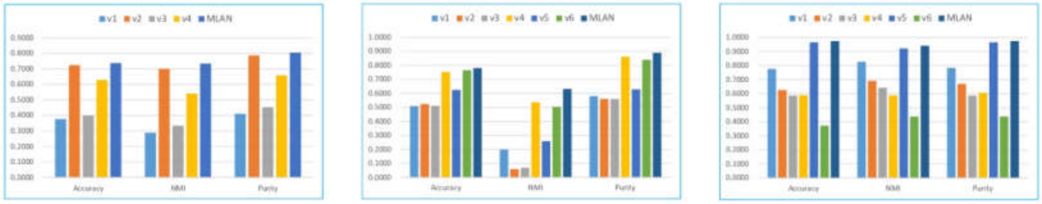
$$ACC(\mu,\nu) = \frac{\sum\limits_{i=1}^{n} \delta(\nu_i, map(\nu_i))}{n} \quad (29)$$

Where map(v_i) is the best mapping function which uses the Kuhn-Munkres algorithms to permute clustering labels to match the ground truth labels. A higher NMI indicates a better clustering performance. NMI provides sound indication of the shared mutual information between a pair of clustering [53]. It can be estimated by computing the confusion matrix.

$$NMI(\mu,\nu) = \frac{2\sum_{i=1}^{c}\sum_{j=1}^{\hat{c}} \frac{n_{ij}}{n} \log \frac{n_{ij}n}{\sum_{i=1}^{c} n_i \sum_{j}^{\hat{c}} n_j}}{-\sum_{i=1}^{c} \frac{n_i}{n} \log \frac{n_i}{n} - \sum_{j}^{\hat{c}} \frac{n_j}{n} \log \frac{n_j}{n}} \quad (30)$$

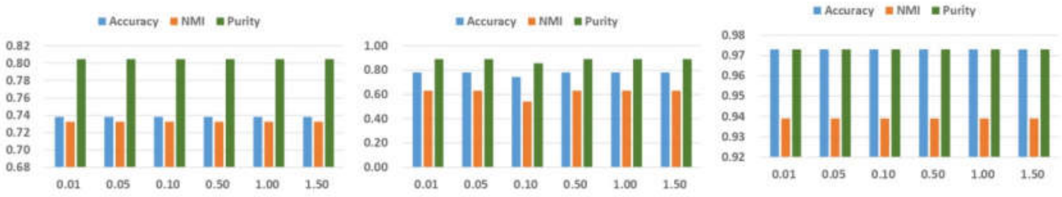
where n is the total number of data points, n_{ij} denotes the number of data in cluster i and class j, n_i and n_j denotes the data number belonging to the ground-truth (μ_i) and clustering result(v_j) respectively. A larger NMI indicates a better clustering performance.

Apart from accuracy and NMI, purity is another popularly used evaluation metric. For ground-truth set $\mu = \{\mu_1,\mu_2,\cdots,\mu_J\}$ and clustering result set $v = \{v_1,v_2,\cdots,v_K\}$, the purity is computed by first assigning each cluster to the class which is the most frequent in the cluster, and then counting the number of correctly assigned objects



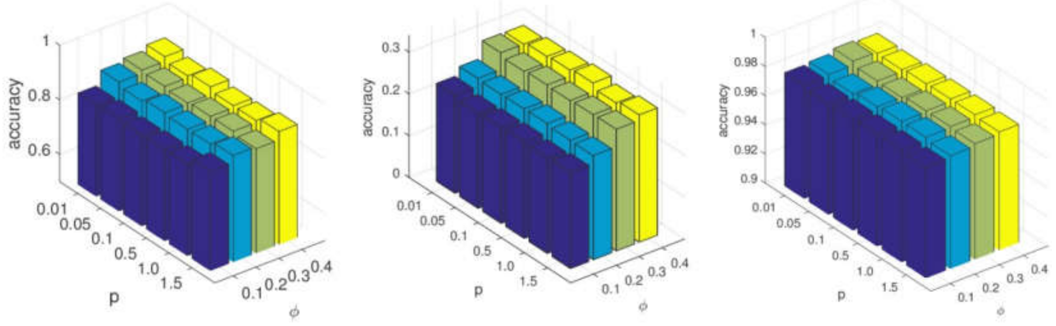
(a) MSRC (b) Caltech101 (c) Handwritten Numerals

Fig. 1: Comparison between MLAN and CAN method which only use one single feature.



(a) MSRC-v1 (b) Caltech101 (c) Handwritten Numerals

Fig. 2: The performance of MLAN in terms of clustering task with different value of parameter p.



(a) MSRC-v1 (b) NUS-WIDE (c) Handwritten Numerals

Fig. 3: The performance of MLAN in terms of semi-supervised classification task with different value of parameter p and ϕ.

Finally dividing by n:

$$Purity(\mu,\nu)=\frac{1}{n}\sum_{k}\max_{j}|\nu_k\cap\mu_j| \tag{31}$$

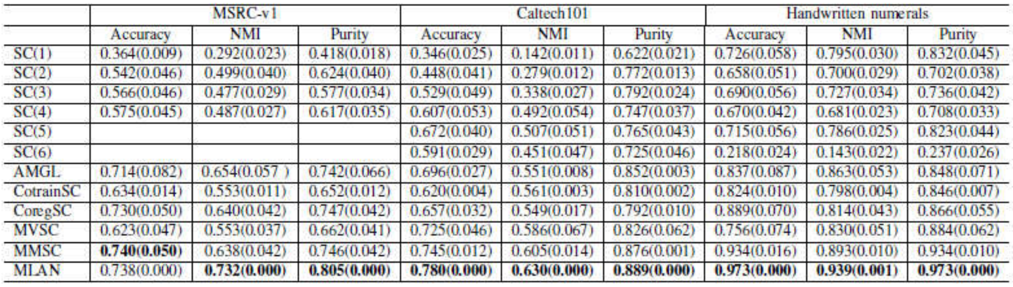
Like ACC and NMI evaluation metric, the higher the purity, the better clustering performance.

C. Comparison Scheme

We first compare the proposed AWMC approach with the single view CAN [42] method, which is exactly the model when view number in our approach is set to 1. Then we also compare with several state-of-the-art multi-view clustering algorithms . The brief introduction and parameter setting of these approaches are as follow:

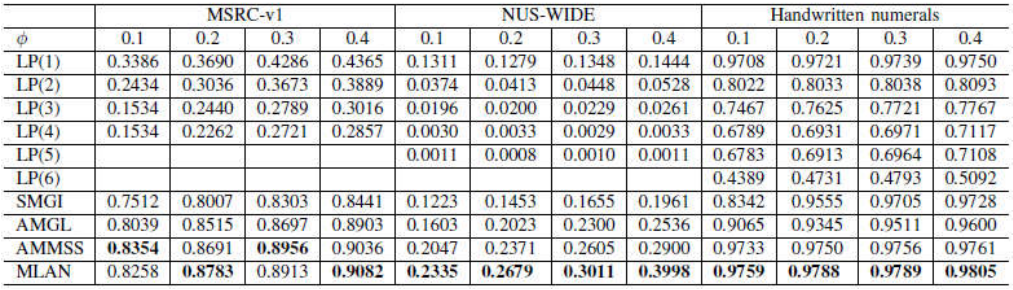
1. Single view Spectral Clustering (SC) [54]: Running spectral clustering on each single view as baseline.
2. Co-trained spectral clustering (CotrainSC) [55]: using co-training to learn eigenvectors which agree across different views, then apply K-means to generate final clustering results, it is a parameter-free method.
3. Co-regularized Spectral Clustering (CoregSC) [35]: it co-regularize the clustering hypotheses to make different
4. graphs agree with each other.
5. Multi-view Spectral Clustering (MVSC) [37]: one of state-of-the-art methods that is specially powerful in large-scale data clustering.
6. Multi-Modal Spectral Clustering (MMSC) [36]: it learns a commonly shared Laplacian matrix by unifying different modals and add a non-negative relaxation to improve the robustness of image clustering.
7. Auto-weighted multiple graph learning (AMGL) [39]: One parameter-free multi-view learning method base on the spectral clustering and could be applied to semisupervised clustering task.
8. Adaptive multi-modal semi-supervised classification (AMMSS) [16]: introduced an algorithm which considers each type of feature as one modality, it learns a shared class indicator matrix and weights for different modalities.
9. Sparse Multiple Graph Integration (SMGI) [17] use sparse weights to linearly combine different graphs to implement label propagation .
10. Multi-view Learning with Adaptive Neighbors (MLAN): proposed by this paper, it calculate the weight for each view automatically and can optimize the similarity graph during clustering step.

TABLE I: Clustering Methods' Performance ((mean and standard deviation))



	MSRC-v1			Caltech101			Handwritten numerals		
	Accuracy	NMI	Purity	Accuracy	NMI	Purity	Accuracy	NMI	Purity
SC(1)	0.364(0.009)	0.292(0.023)	0.418(0.018)	0.346(0.025)	0.142(0.011)	0.622(0.021)	0.726(0.058)	0.795(0.030)	0.832(0.045)
SC(2)	0.542(0.046)	0.499(0.040)	0.624(0.040)	0.448(0.041)	0.279(0.012)	0.772(0.013)	0.658(0.051)	0.700(0.029)	0.702(0.038)
SC(3)	0.566(0.046)	0.477(0.029)	0.577(0.034)	0.529(0.049)	0.338(0.027)	0.792(0.024)	0.690(0.056)	0.727(0.034)	0.736(0.042)
SC(4)	0.575(0.045)	0.487(0.027)	0.617(0.035)	0.607(0.053)	0.492(0.054)	0.747(0.037)	0.670(0.042)	0.681(0.023)	0.708(0.033)
SC(5)				0.672(0.040)	0.507(0.051)	0.765(0.043)	0.715(0.056)	0.786(0.025)	0.823(0.044)
SC(6)				0.591(0.029)	0.451(0.047)	0.725(0.046)	0.218(0.024)	0.143(0.022)	0.237(0.026)
AMGL	0.714(0.082)	0.654(0.057)	0.742(0.066)	0.696(0.027)	0.551(0.008)	0.852(0.003)	0.837(0.087)	0.863(0.053)	0.848(0.071)
CotrainSC	0.634(0.014)	0.553(0.011)	0.652(0.012)	0.620(0.004)	0.561(0.003)	0.810(0.002)	0.824(0.010)	0.798(0.004)	0.846(0.007)
CoregSC	0.730(0.050)	0.640(0.042)	0.747(0.042)	0.657(0.032)	0.549(0.017)	0.792(0.010)	0.889(0.070)	0.814(0.043)	0.866(0.055)
MVSC	0.623(0.047)	0.553(0.037)	0.662(0.041)	0.725(0.046)	0.586(0.067)	0.826(0.062)	0.756(0.074)	0.830(0.051)	0.884(0.062)
MMSC	**0.740(0.050)**	0.638(0.042)	0.746(0.042)	0.745(0.012)	0.605(0.014)	0.876(0.001)	0.934(0.016)	0.893(0.010)	0.934(0.010)
MLAN	0.738(0.000)	**0.732(0.000)**	**0.805(0.000)**	**0.780(0.000)**	**0.630(0.000)**	**0.889(0.000)**	**0.973(0.000)**	**0.939(0.001)**	**0.973(0.000)**

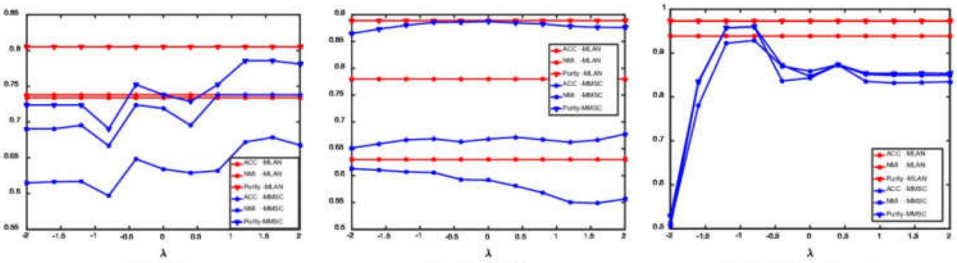
TABLE II: Semi-supervised Classification Methods' Performance



	MSRC-v1				NUS-WIDE				Handwritten numerals			
ϕ	0.1	0.2	0.3	0.4	0.1	0.2	0.3	0.4	0.1	0.2	0.3	0.4
LP(1)	0.3386	0.3690	0.4286	0.4365	0.1311	0.1279	0.1348	0.1444	0.9708	0.9721	0.9739	0.9750
LP(2)	0.2434	0.3036	0.3673	0.3889	0.0374	0.0413	0.0448	0.0528	0.8022	0.8033	0.8038	0.8093
LP(3)	0.1534	0.2440	0.2789	0.3016	0.0196	0.0200	0.0229	0.0261	0.7467	0.7625	0.7721	0.7767
LP(4)	0.1534	0.2262	0.2721	0.2857	0.0030	0.0033	0.0029	0.0033	0.6789	0.6931	0.6971	0.7117
LP(5)					0.0011	0.0008	0.0010	0.0011	0.6783	0.6913	0.6964	0.7108
LP(6)									0.4389	0.4731	0.4793	0.5092
SMGI	0.7512	0.8007	0.8303	0.8441	0.1223	0.1453	0.1655	0.1961	0.8342	0.9555	0.9705	0.9728
AMGL	0.8039	0.8515	0.8697	0.8903	0.1603	0.2023	0.2300	0.2536	0.9065	0.9345	0.9511	0.9600
AMMSS	**0.8354**	0.8691	**0.8956**	0.9036	0.2047	0.2371	0.2605	0.2900	0.9733	0.9750	0.9756	0.9761
MLAN	0.8258	**0.8783**	0.8913	**0.9082**	**0.2335**	**0.2679**	**0.3011**	**0.3998**	**0.9759**	**0.9788**	**0.9789**	**0.9805**

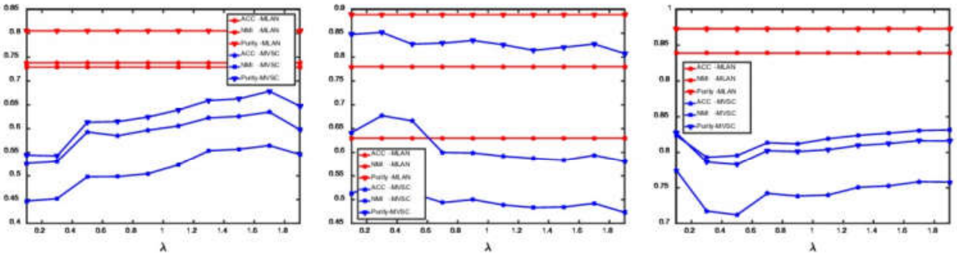
As to the compared methods, source codes are obtained from their authors' websites, since these state-of-the-art multiview clustering algorithms are graph-based, which need to calculate Laplacian matrix for each of the view. We utilize both normalized Laplacian matrix and non-normalized Laplacian matrix to perform the experiments. Such consideration is necessary because the experiment results display that in some situation, the normalized form show preferable performance than the non-normalized form; but in other situation, the superior to inferior changes in these two form. We report the better result in these two forms of the compared methods. Since most of the graph-based methods often need to utilize *k*-means method as the final clustering step, but we know that *k*-means method's result is heavily depend on the choose of the initial centroids, so we perform 50 times experiments for all methods on each of data sets. What's more, in order to make the experiments much fair enough, we set the parameters of each method just as their authors adopt and report the best results, and report the result of MLAN method with $p = 1$. To all dataset, each sample is assigned 9 nearest neighbors to construct graph. In terms of semi-supervised classification, some we denote φ (10%,20%, 30%, 40%) as the proportions of the labeled data.

D. Performance evaluation



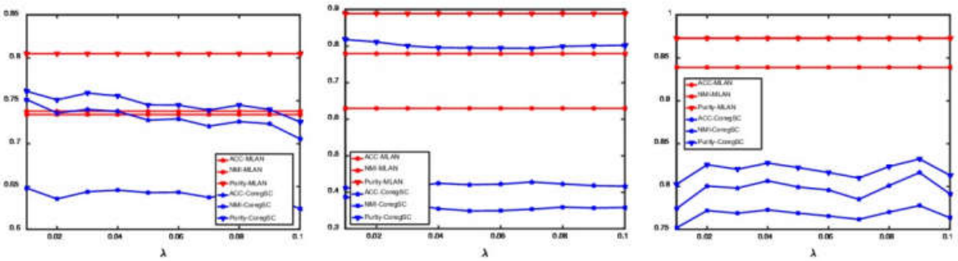
(a) MSRC (b) Caltech101 (c) Handwritten Numerals

Fig. 4: The comparison between MMSC and the proposed MLAN on different datasets.



(a) MSRC (b) Caltech101 (c) Handwritten Numerals

Fig. 5: The comparison between MVSC and the proposed MLAN on different datasets.



(a) MSRC (b) Caltech101 (c) Handwritten Numerals

Fig. 6: The comparison between CoregSC and the proposed MLAN on different datasets.

Multi-view clustering method achieves better performance than the best of single-view clustering method.

Particularly, the comparison between MLAN and such method only using one specific view's feature is conducted, seen in Figure 1. It validates the effectiveness of our MLAN method, proving that combing information from different views properly can improve the clustering result. Surprisingly, the clustering performance of the handwritten numerals dataset is better than some other semi-supervised classification methods. In addition, we seek the influence of exponential function with different parameter p, shown in the Figure 2 and Figure 3. We can see that there exist little difference, with the parameter p scanned in the whole range, the result keeps high-level performance. Besides, with the increasing proportion of labled data, the performance of semi supervised classification approaches raise. The much label information available, the more unlabeled data points will be more classified correctly. The performance of the proposed model MLAN exceeds other methods in both clustering and classification.

E. Parameter Sensitivity

There is only one parameter λ in our model brought by the Laplacian matrix rank constrains. For the sake of simpleness and accelerating the convergence procedure, we can initialize λ equal to the obtained value of α (or randomly chose from 1 to 30), and decrease it ($\lambda = \lambda/2$) if the connected components of S is greater than class number c or increase it ($\lambda = \lambda$ / 2) if smaller than c in each iteration.

For other compared methods, we set their parameters to the optimal value: CoregSC has the co-regularization parameter λ searched from 0 to 0.1 with step 0.01; MVSC has the weights' distribution control parameter r search in logarithm form ($log_{10}r$ from 0.1 to 2 with step size 0.2); MMSC has the penalty parameter $log_{10}\alpha$ searched from −2 to 2 with step 0.2; AMMSS has exactly the same parameter r as MVSC and regularization parameter λ in the range from 0 to 1 with step 0.1; SMGI has two regularization parameters λ_1 and λ_2 in the range from 0 to 1 with step 0.1.

From the table (I), we can see that the proposed MLAN method not only achieve excellent result but also be very robust to the parameter λ, it nearly can be seen parameterfree approach. By contrast, the performance of parameter-free method CotrainSC is unsatisfactory, and other state-of-the-art methods are sensitive to their hyperparameters, which can be seen from Figure 4, Figure 5, and Figure 6. However, the appropriate values of input parameters are often unknown for clustering and semi-supervised learning task, which highlight the superiority of our MLAN method.

VI. CONCLUSIONS

In this paper, we introduce a novel multi-view learning model named MLAN, which performs clustering/semi supervised classification and local structure learning simultaneously. With the reasonable rank constrain, the obtained optimal graph can be partitioned into specific clusters directly. Due to the robustness to the only parameter, MLAN nearly can be seen as parameter-free method, which is very commendable, especially for unsupervised clustering task. Extensive experimental results show that the proposed model achieves superior's performances.

REFERENCES

1) D. G. Lowe, "Distinctive image features from scaleinvariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
2) N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *The 9th European Conference on Computer Vision, Proceedings, Part II*, 2006, pp. 428–441.
3) A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.
4) T. Ojala, M. Pietikainen, and T. M¨ aenp¨ a¨a, "Multireso-¨ lution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
5) J. Wu and J. M. Rehg, "CENTRIST: A visual descriptor for scene categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1489–1501, 2011.
6) M. A. Stricker and M. Orengo, "Similarity of color images," in *Storage and Retrieval for Image and Video Databases III, San Diego/La Jolla, CA, USA, February 5-10, 1995*, 1995, pp. 381–392.
7) R. Zhang, F. Nie, and X. Li, "Semi-supervised classification via both label and side information," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*, 2017, pp. 2417–2421.
8) X. Fang, Y. Xu, X. Li, Z. Lai, and W. K. Wong,
9) "Robust semi-supervised subspace clustering via nonnegative low-rank representation," *IEEE Trans. Cybernetics*, vol. 46, no. 8, pp. 1828–1838, 2016.
10) A. Blum and T. M. Mitchell, "Combining labeled and unlabeled sata with co-training," in *Proceedings of the Eleventh Annual Conference on Computational Learning Theory, COLT 1998, Madison, Wisconsin, USA, July 2426, 1998.*, 1998, pp. 92–100.
11) Z. Tian and R. Kuang, "Integrative classification and analysis of multiple arraycgh datasets with probe alignment," *Bioinformatics*, vol. 26, no. 18, pp. 2313–2320, 2010.
12) H. Liu, J. Han, F. Nie, and X. Li, "Balanced clustering with least square regression," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, 2017, pp. 2231–2237.
13) D. M. Dunlavy, T. G. Kolda, and W. P. Kegelmeyer, "Multilinear algebra for analyzing data with multiple
14) linkages," in *Graph Algorithms in the Language of Linear Algebra*. SIAM, 2010.
15) E. Acar, T. G. Kolda, and D. M. Dunlavy, "An optimization approach for fitting canonical tensor decompositions," Tech. Rep. SAND2009-0857, 2009.
16) J. Sun, D. Tao, and C. Faloutsos, "Beyond streams and graphs: dynamic tensor analysis," in *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006, pp. 374–383.
17) X. Li, S. Lin, S. Yan, and D. Xu, "Discriminant locally linear embedding with high-order tensor data," *IEEE Trans. Systems, Man, and Cybernetics, Part B*, vol. 38, no. 2, pp. 342–352, 2008.
18) X. Cai, F. Nie, W. Cai, and H. Huang, "Heterogeneous image features integration via multi-modal semisupervised learning model," in *IEEE International Conference on Computer Vision*, 2013, pp. 1737–1744.
19) M. Karasuyama and H. Mamitsuka, "Multiple graph label propagation by sparse integration," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 12,
20) pp. 1999–2012, 2013.
21) G. R. G. Lanckriet, N. Cristianini, P. L. Bartlett, L. E. Ghaoui, and M. I. Jordan, "Learning the kernel matrix with semidefinite programming," *Journal of Machine Learning Research*, vol. 5, pp. 27–72, 2004.
22) S. Sonnenburg, G. Ratsch, C. Sch¨ afer, and B. Sch¨ olkopf,¨ "Large scale multiple kernel learning," *Journal of Machine Learning Research*, vol. 7, pp. 1531–1565, 2006.
23) S. Yu, T. Falck, A. Daemen, L. Tranchevent, J. A. K. Suykens, B. D. Moor, and Y. Moreau, "L2-norm multiple kernel learning and its application to biomedical data fusion," *BMC Bioinformatics*, vol. 11, p. 309, 2010.

AUTHOR's PROFILE

Mr. Gangaraju Raghavendra Datta Bhargava Sriram is working as Associate Software Developer in the technology SAP ABAP at Accenture. He completed his B.E in the stream of Electrical and Electronics Engineering from MVSR Engineering College in 2018.His areas of interest are Python and Machine Learning.

Mr. Mohammed Khaja Amir Kafeel is working as SAP ABAP Developer at Accenture. He completed his B.E in the stream of Electronics And Communication Engineering from MVSR Engineering College in 2018. His areas of interest are Data science with Python and Machine Learning.