Multi-View Non-negative Matrix Factorization Discriminant Learning via Cross Entropy Loss

Jian-Wei Liu [1], Yuan-Fang Wang [1], Run-Kun Lu[1], Xiong-Lin Luo[1]

[1] Department of Automation, China University of Petroleum, Beijing Campus (CUP), Beijing, China
E-mail:liujw@cup.edu.cn
E-mail:714244712@qq.com
E-mail:983194327@qq.com
E-mail:luoxl@cup.edu.cn

Abstract: Multi-view learning accomplishes the task objectives of classification by leveraging the relationships between different views of the same object. Most existing methods usually focus on consistency and complementarity between multiple views. But not all of this information is useful for classification tasks. Instead, it is the specific discriminating information that plays an important role. Zhong Zhang et al explores the discriminative and non-discriminative information existing in common and view-specific parts among different views via joint non-negative matrix factorization. In this paper, we improve this algorithm on this basis by using the cross entropy loss function to constrain the objective function better. At last, we implement better classification effect than original on the same data sets and show its superiority over many state-of-the-art algorithms.

Key Words: Multi-View learning, Matrix factorization, Cross entropy loss, Discriminative learning, Classification

1. INTRODUCTION

In many scientific data analysis tasks, data are often collected through different measuring methods, such as various feature extractors or sensors, as usually the single particular measuring method cannot comprehensively describe all the information of the data. In this case, the features of each data example can be naturally divided into groups, each of which can be considered as a view. For instance, for images and videos, color features and texture features can be regarded as two views. It is important to make good use of the information from different views. Multi-view learning is a branch of machine learning which studies and utilizes the information and relationship between views.

There are also many entities represented with different views in the real world, such as web pages[1],[2], multi-lingual news[3]-[5], and neuroimaging[6]-[8]. And most of relevant literatures of multi-view learning regard the consistency and complementarity as two main underlying properties of the multi-view data[9], which like the bridges to link all views together[10]-[12]. Consistency assumptions indicate that all views share consistent information. Obviously, using only consistent information to take advantage of multi-view data is not enough, because each view also contains additional knowledge that other views do not have[1],[13],[14]. Therefore, the complementarity of research views is another important example of learning multi-view data.

However, it is a doubt that whether the consistent and complementary information really always support a better classification performance. Because some experience pre-experiments show that the predictive performance of multi-view data may even be worse than using single-view data in some real data sets. Zhang Zhong et al hold points that the multi-view data contains sections that are helpful and unhelpful for classification, which are called discriminate and non-discriminate information and the consistent or complementary information does not learn discriminative information directly[15]. The classifier constructed by multi-view data may give an even worse classification performance if the learned consistent and complementary information contains no clear discriminative information.

In this paper, we propose multi-view non-negative matrix factorization discriminant learning via cross entropy loss. we distinguish discriminative and non-discriminative information existing in the consistent and complementary parts, and only use discriminative information for classification. Specifically, multi-view data is factorized into common part shared across views and view-specific parts existing within each view as usual. Besides, for both common part and view-specific part, they are further factorized into discriminative part and non-discriminative part. In this situation, each view of data is factorized into four parts: the common discriminative part, the common non-discriminative part, the specific discriminative part and the specific non-discriminative part.

In previous research, mean square error is habitually used as a loss function for predictive labels, but the truth is not the case. As a result of gradient descent training, the mean squared error term for predicting the output label has a certain degree of defects. Its partial derivative value is very small when the output probability value is close to 0 or 1, which could cause the partial derivative value to almost disappear at the beginning of training. In order to make the

This work is supported by the Science Foundation of China University of Petroleum Beijing(2462018QZDX02)

training speed adjust in time with the error between predicted and actual values, we choose the cross-entropy loss function instead. Cross-entropy loss function is usually used to measure the performance of a classification model whose output is a probability value between 0 and 1. And cross-entropy loss increases as the predicted probability diverges from the actual label. Moreover, a supervised constraint is added to guide the joint NMF factorization to obtain better discriminative parts.

To find the optimal decomposition, we follow the block coordinate descent framework[16] to solve the objective function. And only the derived discriminative parts from common part and view-specific parts are used to construct a classifier. Finally, experimental results on seven real-world data sets verify the effectiveness of our proposed method.

In conclusion, our contributions are summarized as follows:

1) We propose a new multi-view non-negative matrix factorization discriminant learning algorithm, which utilizes the discriminative information of data to classify, and outperforms many state-of-the-art algorithms on seven real-world data sets.

2) We apply cross-entropy loss to the non-negative matrix factorization method for the first time, which is more reasonable for the objective function of predicting output label errors.

3) To find the optimal decomposition, we follow the block coordinate descent framework to solve the objective function. However, the cross-entropy loss function contains the nonlinear softmax function, which normalized the predicting output label probability. For better preform the block coordinate descent, we obtain the derivative of the cross-entropy loss function to each matrix vectors.

4) In order to visually compare the difference between cross entropy loss and mean squared error, we use a visualization tool, t-SNT, which is very suitable for high-dimensional data dimensionality reduction to 2D or 3D for visualization. According to the result, our method that use cross entropy loss function is able to find the feature matrix more accurately than the mean squared error function.

2. RELATED WORK

Two representative works of multi-view learning in the early days are canonical correlation analysis (CCA)[17] and co-training[1]. They represent two core ideas for dealing with multi-view problems through consistency and complementarity.

Studies in exploiting consistency generally looks for commonalities between multiple views, which have minimum disagreement. Canonical Correlation Analysis related algorithms[10],[18]-[21] project two or more views into latent subspaces by maximizing the correlations among projected views. Spectral methods[5],[22]-[25] use a weighted summation to merge graph Laplacian matrices from different views into one optimal map for further clustering or embedding. Matrix factorization based methods[4],[11],[26] jointly factorize multi-view data into a common centroid representation by minimizing the overall reconstruction loss of different views. In addition, multiple kernel learning (MKL)[27] can also be viewed as taking advantage of the consistency of different views, where each view is mapped to the new space using kernel tricks, and then all kernel matrices are combined to unify the kernel by minimizing predefined objective functions.

Another method is to explicitly preserve complementary information of different views. The co-training algorithms[27]-[30] treat each view as complementary. Generally, it iteratively trains two classifiers on two different views, and each classifier generates its complementary information to help other classifiers train in the next iteration.

In summary, most existing multi-view learning algorithms mainly focus on learning consistency and complementarity from multi-view data. Zhong Zhang et al[15] breaks the usual and explores the discriminative and non-discriminative information existing in common and view-specific parts among different views via joint non-negative matrix factorization, which provides novel ideas for multi-view learning.

3. THE PROPOSED METHOD

3.1. Non-negative Matrix Factorization

Given a non-negative matrix $X \in \mathbb{R}_+^{m\times n}$, which represents n datas with m fatures. NMF aims to find non-negative matrix factors $W \in \mathbb{R}_+^{m\times k}$ and $H \in \mathbb{R}_+^{n\times k}$ such that:

$$X \approx WH^T \tag{1}$$

Then the objective function can be written as flows:

$$\min_{W,H} \left\| X - WH^T \right\|_F^2 \tag{2}$$

$$\text{s.t. } W, H \geq 0 \tag{3}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Note that the original data matrix is a linear combination of all column vectors in W with weight of corresponding column vectors in H. Therefore, W and H are usually called the basis matrix and the coefficient matrix respectively.

For multi-view data, the representation of NMF-based approaches is as flows:

$$\min_{W,H} \sum_{v=1}^{n_v} \left\| X^{(v)} - WH^T \right\|_F^2 + \Phi(W,H) \tag{4}$$

$$\text{s.t. } W, H \geq 0 \tag{5}$$

where n_v denotes the number of views, and $X^{(v)}$ denotes the data matrix of v-th view. $\Phi(\cdot)$ is a regularization term for W and H.

3.2. Multi-view Learning via DICS

There are consistency and complementary among different multiple views. Therefore, DICS algorithm decomposes the multi-view data matrix into two parts: common part and view-specific parts. As other approaches[31]-[34], define W_C represents the common subspace shared by all views and $W_S^{(v)}$ represents the specific subspace for the v-st view. Therefore, the data matrix of each view can be written as $X^{(v)} = W_C H_C^T + W_S^{(v)} H_S^{(v)T}$. In order to learn the

discriminate information, not only the data matrix is divided into the common and view-specific parts, but also each part of data matrix is divided into the discriminative and non-discriminative part as follows:

$$W = [W_{CD}\, W_{CN}\, W_{SD}^{(v)}\, W_{SN}^{(v)}]$$
$$H = [H_{CD}\, H_{CN}\, H_{SD}^{(v)}\, H_{SN}^{(v)}] \quad (6)$$

Where W_{CD} and H_{CD} represent the common discriminate matrixes, while W_{CN} and H_{CN} represent the common non-discriminate matrixes. Similarly, $W_{SD}^{(v)}$ and $H_{SD}^{(v)}$ indicate the view-specific discriminate parts, while $W_{SN}^{(v)}$ and $H_{SN}^{(v)}$ indicate the view-specific non-discriminate parts.

Afterwards, the discriminate matrixes is used to predict the output label through a linear projection matrix $B = [B_{CD}\, B_{SD}^{(v)}]$, which is corresponding to H_{CD} and $H_{SD}^{(v)}$. Therefore, the objection function of DICS is further reformulated as follows:

$$\min_{W,H,B} \sum_{v=1}^{n_v} \left\| X^{(v)} - WH^T \right\|_F^2 + \Phi(W,H) + \gamma \left\| Y - \begin{bmatrix} B_{CD}\, B_{SD}^{(v)} \end{bmatrix} \begin{bmatrix} H_{CD}^T \\ H_{SD}^{(v)T} \end{bmatrix} \right\|_F^2 \quad (7)$$

$$\text{s.t. } W, H \geq 0 \quad (8)$$

where $Y \in \mathbb{R}^{c \times n}$ is the label matrix, c is the number of classes and n is the number of data instances. Moreover, $y_{i,j} = 1$ if the j-st instance belong to class i and otherwise $y_{i,j} = 0$. Word.

3.3. Improved Objective Function via Cross-Entropy Loss

Number section and subsection headings consecutively in Arabic numbers and type them in bold. Avoid using too many capital letters. If any further subdivision of a subsection is needed the titles should be 10 point and flushed left.

However, as a result of gradient descent training, the mean squared error term for predicting the output label has a certain degree of defects. Its partial derivative value is very small when the output probability value is close to 0 or 1, which could cause the partial derivative value to almost disappear at the beginning of training. In order to make the training speed adjust in time with the error between predicted and actual values, we choose the cross-entropy loss function instead.

Cross-entropy loss function is usually used to measure the performance of a classification model whose output is a probability value between 0 and 1. And cross-entropy loss increases as the predicted probability diverges from the actual label.

For multi-classification tasks, the form of the cross-entropy loss function is as follows:

$$J = -\sum_{i=1}^{c} y_i \log(p_i) \quad (9)$$

Where c indicates the number of classes, y_i is an indicator variable, if the instance belong to this class i, y_i is 1 otherwise 0. And p_i is the predicted probability that the observed sample belongs to class i.

Therefore, the improved objective function could be written as follows:

$$\min_{W,H,B} \sum_{v=1}^{n_v} \left\| X^{(v)} - WH^T \right\|_F^2 + \alpha \left\| W_D^T W_D \right\|_{1,1} + \beta \left\| H_D \right\|_{1,1} - \gamma \sum_{j=1}^{n} \sum_{i=1}^{c} y_{ij} \ln p_{ij} \quad (10)$$

Where α, β, γ are non-negative parameters to balance the regularization term and n_v denotes the number of views, n denotes the number of instances, c denotes the number of classes.

Where the $\|\cdot\|_{1,1}$ is a $l_{1,1}$ norm constraint on the discriminative matrix W_D, this term can be factorized into two parts: $\left\| W_D^T W_D \right\|_{1,1} = \sum_i w_{Di}^T w_{Di} + \sum_{i \neq j} w_{Di}^T w_{Dj}$. The first term is used to prevent overfitting, and the second term encourage basis vectors to be as orthogonal as possible, which reduces the redundancy of discriminative bases. What's more, using $l_{1,1}$ norm constraint on H_D makes the discriminate coefficients sparse. The reason is that data points of different classes should not possess identical basis vectors.

Where p_{ij} is a predictive label output normalized by softmax function. Specifically, its form is as follows:

$$p_{ij} = \frac{e^{\sum_{k=1}^{k_1} b_{CD_i,k} \cdot h_{CD_j,k} + \sum_{k=1}^{k_3} b_{SD_i,k}^{(v)} \cdot h_{SD_j,k}^{(v)}}}{\sum_{t=1}^{n} e^{\sum_{k=1}^{k_1} b_{CD_t,k} \cdot h_{CD_j,k} + \sum_{k=1}^{k_3} b_{SD_t,k}^{(v)} \cdot h_{SD_j,k}^{(v)}}} \quad (11)$$

Thus the objective function can be written as follows in the form of elements and vectors:

$$f(W,H,B)$$

$$\sum_{v=1}^{n_v} \left\| X^{(v)} - \sum_{i=1}^{k1} w_{CD_i} h_{CD_i}^T - \sum_{i=1}^{k2} w_{CN_i} h_{CN_i}^T - \sum_{i=1}^{k3} w_{SD_i}^{(v)} h_{SD_i}^{(v)T} - \sum_{i=1}^{k4} w_{SN_i}^{(v)} h_{SN_i}^{(v)T} \right\|_F^2$$

$$\alpha \sum_{i\ 1}^{k1\ k1} \sum_{j\ 1}^{k1\ k1} w_{CD_i}^T w_{CD_j} + \sum_{i\ 1}^{k3\ k3} \sum_{j\ 1}^{k3\ k3} w_{SD_i}^{(v)T} w_{SD_j}^{(v)} - 2 \sum_{i\ 1}^{k1\ k1} \sum_{j\ 1}^{k1\ k3} w_{CD_i}^T w_{SD_j}^{(v)}$$

$$\beta 1_{1\ n} \sum_{i} h_{CD_i}$$

matrix. And $1_{1\times n}$ is a row vector of length n with all elements 1.

By fixing all column vectors except the one we want to update, we can obtain the convex sub-problem respect to it, then solve it based on the block coordinate descent framework. In order to facilitate the derivation, the objective function is divided into two parts: the nonlinear part of cross-entropy loss function and the other linear parts.

The nonlinear part is written as follows:

$$\begin{aligned} L &= -\gamma \sum_{j=1}^{n}\sum_{i=1}^{c} y_{ij} \ln p_{ij} \\ p_{ij} &= \frac{e^{z_{ij}}}{\sum_{t=1}^{n} e^{z_{tj}}} \\ z_{ij} &= \sum_{k=1}^{k_1} b_{CD_{i,k}} \cdot h_{CD_{j,k}} + \sum_{k=1}^{k_3} b_{SD_{i,k}}^{(v)} \cdot h_{CD_{j,k}}^{(v)} \end{aligned} \tag{13}$$

Take the partial derivative of L to $h_{CD_{i,j}}$ as an example:

$$\frac{\partial L}{\partial h_{CD_{ij}}} = \frac{\partial L}{\partial p_{tj}} \cdot \frac{\partial p_{tj}}{\partial z_{ij}} \cdot \frac{\partial z_{ij}}{\partial h_{CD_{ij}}} \tag{14}$$

Since the softmax function contains the output of all layers, a partial derivative is obtained for all t layers.

The result of first term is as follows:

$$\frac{\partial L}{\partial p_{tj}} = -\sum_{j=1}^{n}\sum_{i,t=1}^{c} \frac{y_{ij}}{p_{tj}} \tag{15}$$

The second term of deriving the softmax function needs to consider two situations:

(1) Deriving the current node when $t = i$:

$$\begin{aligned} &\frac{\partial p_{tj}}{\partial z_{ij}} \\ &= \frac{\partial\left(\frac{e^{z_{ij}}}{\sum_{t=1}^{c} e^{z_{tj}}}\right)}{\partial z_{ij}} \\ &= \frac{e^{z_{ij}} \cdot \sum_{t=1}^{c} e^{z_{tj}} - e^{z_{ij}} \cdot e^{z_{ij}}}{(\sum_{t=1}^{c} e^{z_{tj}})^2} \\ &= p_{ij} \cdot (1 - p_{ij}) \end{aligned} \tag{16}$$

(2) Deriving the other nodes when $t \neq i$:

$$\frac{\partial p_{tj}}{\partial z_{ij}} = \frac{\partial\left(\frac{e^{z_{tj}}}{\sum_{t=1}^{c} e^{z_{tj}}}\right)}{\partial z_{ij}} = -\frac{e^{z_{tj}}}{(\sum_{t=1}^{c} e^{z_{tj}})^2} \cdot e^{z_{ij}} = -p_{tj} \cdot p_{ij} \tag{17}$$

And the third term is $b_{CD_{i,j}}$.

Then combine the results of the above three items:

$$\begin{aligned} \frac{\partial L}{\partial h_{CD_{ij}}} &= -\sum_{j=1}^{n}\sum_{i,t=1}^{c} y_{ij} \cdot \frac{1}{p_{tj}} \cdot \frac{\partial p_{tj}}{\partial z_{ij}} \cdot b_{CD_{i,j}} \\ &= \sum_{j=1}^{n}\sum_{i,t=1}^{c} [-\frac{y_{ij}}{p_{ij}} \cdot p_{ij} \cdot (1 - p_{ij}) + \sum_{t\neq i} \frac{y_{tj}}{p_{tj}} \cdot p_{tj} \cdot p_{ij}] \cdot b_{CD_{i,j}} \\ &= \sum_{j=1}^{n}\sum_{i,t=1}^{c} (-y_{ij} + p_{ij} \sum_{t} y_{tj}) \cdot b_{CD_{i,j}} \end{aligned} \tag{18}$$

Because Y is a label matrix with 0 or 1 elements, And for every sample $\sum_{t} y_{tj} = 1$.

So the partial derivative of L to $h_{CD_{i,j}}$ is written as follows:

$$\begin{aligned} \frac{\partial L}{\partial h_{CD_{ij}}} &= \sum_{j=1}^{n}\sum_{i=1}^{c} (p_{ij} - y_{ij}) \cdot b_{CD_{i,j}} \\ &= \sum_{j=1}^{n}\sum_{i=1}^{c} (\frac{e^{\sum_{k=1}^{k_1} b_{CD_{i,k}} \cdot h_{CD_{j,k}} + \sum_{k=1}^{k_3} b_{SD_{i,k}}^{(v)} \cdot h_{SD_{j,k}}^{(v)}}}{\sum_{t=1}^{c} e^{\sum_{k=1}^{k_1} b_{CD_{t,k}} \cdot h_{CD_{j,k}} + \sum_{k=1}^{k_3} b_{SD_{t,k}}^{(v)} \cdot h_{SD_{j,k}}^{(v)}}} \\ &\quad - y_{ij}) \cdot b_{CD_{i,j}} \end{aligned} \tag{19}$$

Then the partial derivative of the column vector h_{CD_i} is $\frac{\partial L}{\partial h_{CD_i}} = \sum_{i=1}^{n} (p_i - y_i)^T \cdot b_{CD_i}$, where p_i is calculated by the softmax function.

Make $[\,]_+$ to denote $max(0, \cdot)$, which projects the negative value to the boundary of feasible region of zero and guarantees the non-negative nature of the matrix. And we get the update process for each matrix with update rate η as follows:

$$\begin{aligned} w_{CD_i} &= w_{CD_i} \\ &+\eta\left[\sum_{v=1}^{n_v}\left(R^{(v)} h_{CD_i} - \alpha(W_{CD} 1_{k1\times 1} + W_{SD}^{(v)} 1_{k3\times 1})\right)\right]_+ \end{aligned} \tag{20}$$

$$w_{CN_i} = w_{CN_i} + \eta\left[\sum_{v=1}^{n_v} R^{(v)} h_{CN_i}\right]_+ \tag{21}$$

$$w_{SD_i}^{(v)} = w_{SD_i}^{(v)} + \eta\left[R^{(v)} h_{SD_i}^{(v)} - \alpha(W_{CD} 1_{k1\times 1} + W_{SD}^{(v)} 1_{k3\times 1})\right]_+ \tag{22}$$

$$w_{SN_i}^{(v)} = w_{SN_i}^{(v)} + \eta\left[R^{(v)} h_{SN_i}^{(v)}\right]_+ \tag{23}$$

$$h_{CD_i} = h_{CD_i} + \eta\left[\sum_{v=1}^{n_v}\begin{pmatrix} R^{(v)} w_{CD_i} - \frac{\beta}{2} 1_{n\times 1} \\ -\frac{\gamma}{2} Q^{(v)T} b_{CD_i} \end{pmatrix}\right]_+ \tag{24}$$

$$h_{CN_i} = h_{CN_i} + \eta\left[\sum_{v}^{n_v} R^{(v)T} w_{CN_i}\right]_+ \tag{25}$$

$$h_{SD_i}^{(v)} = h_{SD_i}^{(v)} + \eta\left[R^{(v)T} w_{SD_i}^{(v)} - \frac{\beta}{2} 1_{n\times 1} - \frac{\gamma}{2} Q^{(v)T} b_{SD}^{(v)}\right]_+ \tag{26}$$

$$h_{SN_i}^{(v)} = h_{SN_i}^{(v)} + \eta\left[R^{(v)T} w_{SN_i}^{(v)}\right]_+ \tag{27}$$

Where $R^{(v)}$ and $Q^{(v)}$ are as follows:

$$R^{(v)} = X^{(v)} - W_{CD}H_{CD}^{T} - W_{CD}H_{CD}^{T} - W_{SD}^{(v)}H_{SD}^{(v)T} - W_{SN}^{(v)}H_{SN}^{(v)T} \quad (28)$$

$$Q^{(v)} = \text{soft}\max(B_{CD}H_{CD}^{T} + B_{SD}^{(v)}H_{SD}^{(v)T}) - Y \quad (29)$$

Where η denotes the learning rate, and the second derivative of the objective function can be used in the actual calculation process, which is closest to the best step size.

Furthermore, the projection matrix B_{CD} and $B_{SD}^{(v)}$ could be calculated by other determined various. Let the cross-entropy loss function be zero, and the calculation method of projection matrix B_{CD} and $B_{SD}^{(v)}$ is as follows:

$$B_{CD} = \frac{1}{n_v}\sum_{v=1}^{n_v}\left(Y - B_{SD}^{(v)}H_{SD}^{(v)T}\right)H_{CD}\left(H_{CD}^{T}H_{CD} + \lambda I\right)^{-1} \quad (30)$$

$$B_{SD}^{(v)} = \left(Y - B_{CD}H_{CD}^{T}\right)H_{SD}^{(v)}\left(H_{SD}^{(v)T}H_{SD}^{(v)} + \lambda I\right)^{-1} \quad (31)$$

Where I is the identity matrix and λ is a minimum positive number.

4. EXPERIMENT

In this section, we experimentally evaluate the proposed improved algorithm in classification task on eight real world multi-view data sets, and analyse the convergence of our proposed algorithm.

4.1. Datasets

In this paper, we use eight real-world multi-view data sets to verify the performance of the proposed algorithm, including Reuters, YaleFace, BBC, Cornell, Texas, Washington, and Wisconsin datasets. And Cornell, Texas, Washington, and Wisconsin dataset are four subset of data sets selected from WebKB data sets. The properties of data sets are summarized in Table1. Comparison Algorithms

Table1. Characteristics of the datasets

Data Set	Characteristics			
	Instances	Views	Classes	Dimensions
Reuters	1200	5	6	2000 for all
YaleFace	256	2	8	2016 for all
BBC	685	4	5	4659/4633/ 4665/4684
Cornell	195	2	5	1703/585
Texas	187	2	5	1703/561
Washington	230	2	5	1703/690
Wisconsin	265	2	8	1703/795

We compare our algorithm with several single-view and multi-view algorithms to show its effectiveness, including KNN, NMF, SSNMF, GNMF, multiNMF, MVCC, MCL, DICS. And the parameters of all algorithms are selected within the range that the author suggested.

KNN(set $k = 1$) is regarded as the baseline algorithm and we apply KNN on all single views and report the best performance on the view. Also we apply the KNN algorithm on the concatenated feature vector(KNNcat).

NMF is applied on each of the single view data and the the concatenated feature vector (i.e. NMFcat), which is regarded as another baseline algorithm.

SSNMF is a supervised NMF variant[35], which incorporates a linear classifier to encode the supervised information. We select the regularization parameter λ within the range of [0.5:0.5:3].

GNMF is a manifold regularized version of NMF[3], which preserves the local similarity by imposing a graph Laplacian regularization. We use the normalized dot product (cosine similarity) to construct the affinity graph, and select the regularization parameter λ within the set of $\{10^0,10^1,10^2,10^3,10^4\}$.

MultiNMF is an NMF-based multi-view clustering algorithm[11], which can get compatible clustering results across multiple views. We select the regularization parameter λ within the set of $\{10^0,10^{-1},10^{-2},10^{-3}\}$.

MVCC is a novel multi-view clustering method based on concept factorization with local manifold regularization[26], which drives a common consensus representation for multiple views. We set parameter α to 100, and select β and γ within the set of {50,100,200,500,1000}.

MCL is a semi-supervised multi-view NMF variant with graph regularized constraint[4]. We select parameter α within the range of [100:50:250], β within the set of {0.01, 0.02, 0.03}, and set γ to 0.005 as author suggested.

DICS is the original model of our algorithm, which is an NMF-based multi-view learning algorithm, by exploring the discriminative and non-discriminative information existing in common and view-specific parts among different views via joint non-negative matrix factorization, and produce discriminative and non-discriminative feature from all subspaces. What's more, discriminative and non-discriminative features are further used to produce classification results. We select parameters α and β within a small range of $[0,1]$, and set parameter γ to 1.

4.2. Result

For all algorithms, we first perform a five-folds cross validation to select the parameters that has the best accuracy and generalization performance. Due to randomness, we run all algorithms 10 times on each dataset and report the mean values and standard deviation of accuracy.

All the classification results of eight multi-view datasets are summarized in Table2 and the best result on each dataset is highlighted in boldface. As we can see, the proposed algorithm achieves better accuracy on most of the datasets, and slightly worse than other algorithms on BBC and Wisconsin datasets, and It is worth mentioning that, the standard deviation of accuracy is much lower than DICS method.

Table2. Accuracy of Different Methods

Method	ACC(%)						
	Reuters	YaleFace	BBC	Cornell	Texas	Washington	Wisconsin
GNMF	40.8±1.2	50.0±2.5	38.0±1.5	41.0±1.8	57.9±1.8	69.6±2.2	52.8±1.4
MultiNMF	52.7±0.2	64.2±4.2	73.1±0.2	49.7±7.7	68.7±3.4	59.3±2.6	50.3±3.5
MVCC	54.4±1.9	33.3±6.9	95.8±2.6	60.8±5.0	64.7±5.5	62.8±3.8	64.3±2.7
DICS	70.3±4.0	89.1±3.2	90.2±2.4	72.8±6.1	81.6±4.0	77.4±6.0	85.1±4.5
Our method	70.5±1.3	90.2±1.5	91.7±1.2	75.3±2.2	83.8±2.1	83.5±2.2	83.6±1.9

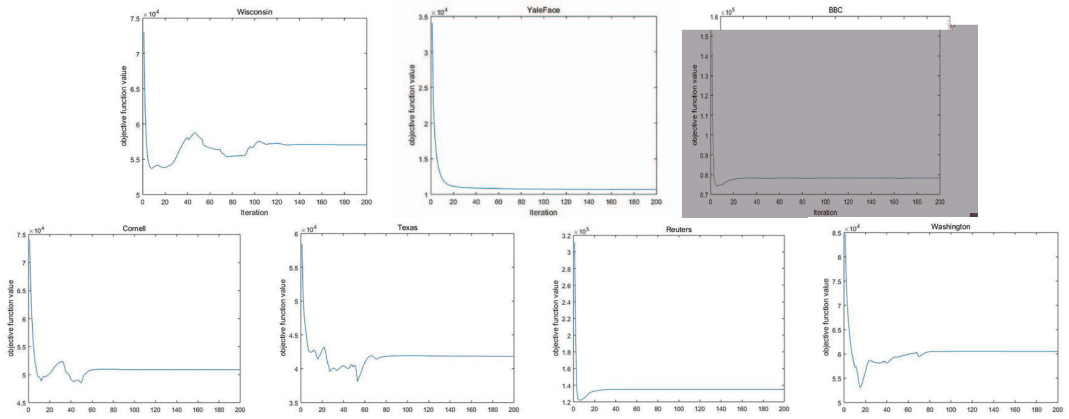

Fig 1. Iterative curves on all data sets.

4.3. Convergence Analysis

In order to empirically investigate the convergence property of our algorithm, we plot the iterative curves of objective function on five typical data sets in Fig 1, we can observe that the objective values drop sharply and then convergence curves begin to grow/decrease mildly, then it converges eventually. Usually, the algorithm will converge in no more than 100 iterations. Relatively, the DICS algorithm will converge in 50 iterations and the reason may be that the derivative of cross entropy loss function leads to higher algorithm complexity.

4.4. Discriminant Matrix Visualization

t-SNE (t-distributed stochastic neighbour embedding) is a machine learning algorithm used for dimensionality reduction[36]. It was proposed by Laurens van der Maaten and Geoffrey Hinton in 2008. In addition, t-SNE is a nonlinear dimensionality reduction algorithm, which is very suitable for high-dimensional data dimensionality reduction to 2D or 3D for visualization.

DICS and our method obtain the basis matrix W_D and the coefficient matrix H_D through iterative updating, which is used to generate the feature matrix of raw data matrix X. By using the t-SNE for dimension reduction processing of the feature matrix, it is possible to visually see the distinguishable feature matrix. Therefore, we compare DICS with our method by Two-dimensional visualized feature matrix to compare the help of two method feature matrices for classification. In the figure of t-SNE, each point represents an instance and different colors indicate different class labels. Therefore, most points with the same color are clustered into clusters in the figure. From Fig 2, we could see the t-SNE figure of our method having better clustering results, and there are fewer instances assigned to the wrong cluster. What's more, The clusters in our method are more concentrated. Thus, our method is able to find the feature matrix more accurately than DICS.

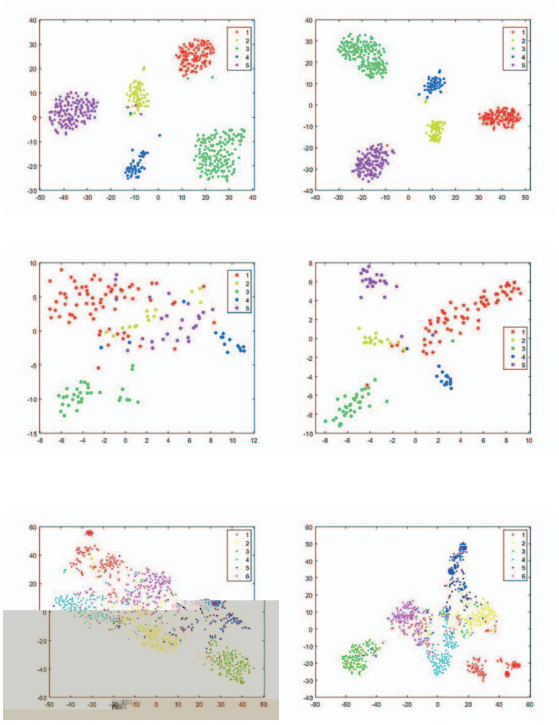

Fig 2. Compare t-SNE figures of the feature matrix on all data sets between DICS(left) and our method(right) and form top to bottom in order are the BBC, Cornell, Reuters, Texas, Washington, Wisconsin, Wisconsin and YaleFace datasets.

5. Conclusion and Future Work

In this paper, we propose a novel multi-view network. The proposed algorithm explores the discriminative and non-discriminative information existing in common and view-specific parts among different views via joint non-negative matrix factorization, and use the cross entropy loss to constrain the objective function, which shows better classification results than the mean square error. The experimental results on seven real-world data sets have demonstrated the effectiveness of our proposed algorithm.

For future studies, we plan to make the distance farther between the view-specific matrixes to improve the common and view-specific part, which is benefit to classify instances through features between different views.

REFERENCES

[1] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," *Proceedings of the eleventh annual conference on Computational learning theory*, Madison, Wisconsin, USA, 1998, pp. 92-100.

[2] D. Zhou and C. J. C. Burges, "Spectral clustering and transductive learning with multiple views," *Proceedings of the 24th international conference on Machine learning*, Corvallis, Oregon, USA, 2007, pp. 1159-1166.

[3] D. Cai, X. He, J. Han and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," in *nonnegativ7dirix ftation*,

[17] H. Hotelling, “Relations between two sets of variates,” *Breakthroughs in statistics*, Springer, New York, NY, 1992, pp. 162-190.
[18] J. D. R. Farquhar, D.R. Hardoon, H. Meng, J. Shawe-Taylor, and S. Szedmak, “Two view learning: SVM-2K, Theory an d Practice,” *Advances in Neural Information Processing Systems*, Vancouver, British Columbia, Canada, 2005, pp. 355–362.
[19] D. R. Hardoon, S. Szedmak and J. Shawe-Taylor, “Canonical correlation analysis: an overview with application to learning methods,” *Neural Computation*, vol. 16, no. 12, pp. 2639–2664, Dec. 2004.
[20] M. Kan, S. Shan, H. Zhang, S. Lao and X. Chen, “Multi-view discriminant analysis,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 188–194, Jan. 2016.
[21] A. Sharma, A. Kumar, H. Daume and D. W. Jacobs, "Generalized Multiview Analysis: A discriminative latent space," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, 2012, pp. 2160-2167.
[22] V.R. De Sa, “Spectral clustering with two views,” *ICML workshop on learning with multiple views*, 2005, pp. 20–27.
[23] F. Nie, J. Li and X. Li, “Parameter-free auto-weighted multiple graph learning: a framework for multiview clustering and semi-supervised classification,” *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, New York, NY, USA, 2016, pp. 1881-1887.
[24] T. Xia, D. Tao, T. Mei and Y. Zhang, “Multiview Spectral Embedding,” in *IEEE Transactions on Systems, Man, and Cybernetics*, Part B (Cybernetics), vol. 40, no. 6, pp. 1438-1446, Dec. 2010.
[25] D. Zhou and C. J. Burges, “Spectral clustering and transductive learning with multiple views,” *Machine Learning, Proceedings of the Twenty-Fourth International Conference*, Corvallis, Oregon, USA, 2007, pp. 1159–1166.
[26] H. Wang, Y. Yang and T. Li, “Multi-view clustering via concept factorization with local manifold regularization,” *IEEE 16th International Conference on Data Mining (ICDM)*, Barcelona, Spain, 2016, pp. 1245–1250.
[27] M. Gönen and E. Alpaydın, “Multiple kernel learning algorithms,” *Journal of machine learning research*, vol. 12, pp. 2211-2268, Jul. 2011.
[28] A. Kumar and H. Daume, “A co-training approach for multi-view spectral clustering,” *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, Washington, USA, 2011, pp. 393–400.
[29] W. Wang and Z. H. Zhou, “A new analysis of co-training,” *Proceedings of the 27th International Conference on Machine Learning*, Haifa, Israel, 2010, pp. 1135–1142.
[30] M. Zhang and Z. Zhou, “CoTrade: Confident Co-Training With Data Editing,” in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 6, pp. 1612-1626, Dec. 2011.
[31] S. K. Gupta, D. Phung, B. Adams, T. Tran and S. Venkatesh, “Nonnegative shared subspace learning and its application to social media retrieval,” *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, USA, 2010, pp. 1169–1178.
[32] S. K. Gupta, D. Phung, B. Adams and S. Venkatesh, “Regularized nonnegative shared subspace learning,” *Data Mining and Knowledge Discovery*, vol. 26, no. 1, pp. 57–97, Jan. 2013.
[33] H. Kim, J. Choo, J. Kim, C. K. Reddy and H. Park, “Simultaneous discovery of common and discriminative topics via joint nonnegative matrix factorization,” *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney, NSW, Australia, 2015, pp. 567–576.
[34] Liu, J., Jiang, Y., Li, Z., Zhou, Z.H., Lu, H.: Partially shared latent factor learning with multiview data. TNNLS 26(6), 1233–1246 (2015)
[35] H. Lee, J. Yoo and S. Choi, “Semi-Supervised Nonnegative Matrix Factorization,” in *IEEE Signal Processing Letters*, vol. 17, no. 1, pp. 4-7, Jan. 2010.
[36] L. Maaten and G. Hinton, “Visualizing data using t-SNE,” *Journal of machine learning research*, vol. 9, pp. 2579-2605, Nov. 2008.