arXiv:1907.01693v1 [cs.LG] 3 Jul 2019

Canonical Correlation Analysis (CCA) Based Multi-View Learning: An Overview

Chenfeng Guo and Dongrui Wu

Abstract—**Multi-view learning (MVL) is a strategy for fusing data from different sources or subsets. Canonical correlation analysis (CCA) is very important in MVL, whose main idea is to map data from different views onto a common space with the maximum correlation. The traditional CCA can only be used to calculate the linear correlation between two views. Moreover, it is unsupervised, and the label information is wasted in supervised learning tasks. Many nonlinear, supervised, or generalized extensions have been proposed to overcome these limitations. However, to our knowledge, there is no up-to-date overview of these approaches. This paper fills this gap, by providing a comprehensive overview of many classical and latest CCA approaches, and describing their typical applications in pattern recognition, multi-modal retrieval and classification, and multi-view embedding.**

Index Terms—**Canonical correlation analysis, multi-view learning, multi-modal retrieval, multi-view embedding**

I. Introduction

Many real-world datasets can be described from multiple "viewpoints", such as pictures taken from different angles of the same object, different language expressions of the same semantic, texts and images on the same web page, etc. The representations from different perspectives can be treated as different views. The essence of multi-view learning (MVL) is to exploit the consensual and complementary information between different views [1], [2] to achieve better learning performance.

MVL approaches can be divided into three major categories [3], [4]:

1) *Co-training* [5], [6], which exchanges discriminative information between two views by training the two models alternately.
2) *Multi-kernel learning* [7], [8], which maps data to different feature spaces with different kernels, and then combines those projected features from all spaces.
3) *Subspace learning* [9]–[11], which assumes all views are generated from a latent common space where shared information of all views can be exploited.

Canonical correlation analysis (CCA), first proposed by Hotelling [9] in 1936, is a typical subspace learning approach. Its main idea is to find pairs of projections for different views so that the correlations between them are maximized. One example is illustrated in Fig. 1. Since CCA takes the relationship between different feature sets into account, which is consistent with the idea of MVL, it is widely used in MVL [12]–[14], including multi-view dimensionality reduction [15], multi-view clustering [16], [17], multi-view regression [18], and so on.

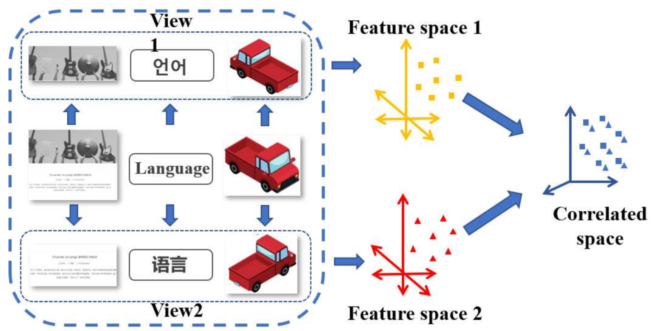


Fig. 1. Multi-view data and CCA-based subspace learning.

The traditional CCA has the following limitations:

1) It cannot handle more than two views.
2) It can only calculate the linear correlation between two views, whereas in many real-world applications the true relationship between the views may be nonlinear.
3) In supervised classification, labels are available; however, CCA, as an unsupervised algorithm, completely ignores the labels, and hence wastes information.

Many extensions have been proposed in the past few decades [19]–[27] to accommodate these limitations. Some representative ones are briefly introduced next.

In 1961, Horst [28] first proposed generalized canonical correlation analysis (GCCA) to estimate the pairwise correlations of multiple views. He provided two formulations: the sum of correlation (SUMCOR), and the maximum variance (MAXVAR). Carroll [29] in 1968 proposed to find a shared latent correlated space, which was shown to be identical to MAXVAR GCCA. In 1971, Kettenring [30] added three new formulations to GCCA, which maximizes the sum of the squared correlation (SSQCOR), minimizes the smallest eigenvalue (MINVAR), and minimizes the determinant of the correlation matrix (GENVAR), respectively. Two decades later, Nielsen [27] summarized four constraints for these five formulations, forming a total of 20 combinations. In 2007, Via [31] proposed least squares based CCA (LS-CCA), and showed that it is essentially identical to MAXVAR. Luo *et al.* [26] proposed in 2015 that the correlation of multiple views can be directly maximized by analyzing the high-order

C. Guo and D. Wu are with the Key Laboratory of the Ministry of Education for Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China. Email: cfguo@hust.edu.cn, drwu@hust.edu.cn.

Dongrui Wu is the corresponding author.

covariance tensor, namely tensor canonical correlation analysis (TCCA). In 2017, Benton *et al.* [24] proposed DNN-based deep generalized canonical correlation analysis (DGCCA), which was a nonlinear extension of MAXVAR GCCA.

Generally, there are three approaches to deal with complex nonlinear relationship between two views [20]. The most common one is to project data onto a higher dimensional space using the "kernel trick", e.g., kernel canonical correlation analysis (KCCA) [19]. However, global kernelization suffers from high computational complexity, and it is not easy to choose an optimal kernel function. The second approach tries to preserve the locality of data. Inspired by the graph model, Sun and Chen [20] proposed locality preserving canonical correlation analysis (LPCCA), which aimed to reduce the global nonlinear dimensionality while preserving the local linear structure of data. Since the distances between neighbors need to be calculated, it is time-consuming when the sample size is large. The third approach is based on deep neural networks (DNN), which can give a very complex mapping between data. Andrew *et al.* [23] first proposed deep canonical correlation analysis (DCCA) in 2013. Inspired by the autoencoder, Wang *et al.* [21] proposed deep canonically correlated autoencoders (DCCAE) in 2015. However, DNN models have poor interpretability, and require a large amount of data to fit.

The label information is critical for classification problems. In order to make full use of the discriminant information, Sun *et al.* [22] proposed discriminant canonical correlation analysis (DisCCA) in 2008, by taking the inter-class and intra-class similarities of different views into consideration. Similar to DisCCA, Sun *et al.* [32] further proposed multi-view linear discriminant analysis (MLDA), which combined CCA and linear discriminant analysis (LDA) [33]. Elmadany *et al.* [25] integrated neural networks into DisCCA to obtain a nonlinear supervised model. Benton *et al.* [24] made use of the discriminant information by treating the one-hot encoding matrix of the labels as an additional view.

Although CCA has been widely used in computer vision, information retrieval, natural language processing, brain-computer interfaces, etc. (see Section IV), there does not exist an up-to-date comprehensive overview on it. The most relevant overview we could find is [34], published in 2004. Many new and better-performing CCA approaches have been proposed in the last 15 years. So, it is desirable to have an updated overview of CCA. This paper fills this gap, by providing a comprehensive overview of many traditional and latest CCA approaches, and describing some representative applications. A summary of their main characteristics is shown in Table I.

The remainder of this paper is organized as follows: Sections II and III review CCA approaches for two views and more than two views, respectively. Section IV reviews some typical applications of CCA. Section V draws conclusions.

TABLE I
COMPARISON OF SOME TYPICAL CCA APPROACHES.

Approach	> 2 views	Supervised	Nonlinear	DNN
CCA				
sCCA				
KCCA			✓	
RCCA			✓	
LPCCA			✓	
DisDCCA		✓		
MLDA		✓		
MULDA		✓		
DCCA			✓	✓
DisDCCA		✓	✓	✓
DCCAE			✓	✓
DisDCCAE		✓	✓	✓
VCCA			✓	✓
GCCA	✓			
MCCA	✓			
LS-CCA	✓			
TCCA	✓			
DGCCA	✓	✓	✓	✓

TABLE II
NOTATIONS USED IN THIS PAPER.

Notation	Description
X, Y	Data matrices from two different views x and y
$(\mathbf{x}_i, \mathbf{y}_i)$	The i-th paired instances from Views x and y
N	Number of instances in each view
c	Number of classes
n_i	Number of instances in Class i
J	Number of views
K	Number of canonical vectors
d_x	Feature dimensionality of View x
$\mathbf{w}_x$	A canonical vector of View x
W_x	Canonical matrix of View x
Σ_{xy}	Covariance matrix of View x and View y
$\hat{\Sigma}_{xy}$	Regularized covariance matrix of View x and View y
r_x	Regularization coefficients of View x
m	Dimension of low rank approximation
$\cdot^T$	Transpose of a matrix or vector
I	The identity matrix
$\|\cdot\|_2$	2-norm of a vector
$\|\cdot\|_F$	The Frobenius-norm of a matrix

II. CCA FOR TWO VIEWS

This section first reviews the tradition CCA, and then introduces some representative nonlinear, sparse, and/or supervised extensions. Table II summarizes the notations used in this paper.

A. Canonical Correlation Analysis (CCA)

Let $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N] \in \mathbb{R}^{d_x \times N}$ and $Y = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N] \in \mathbb{R}^{d_y \times N}$ be two mean-zero data matrices with N instances and d_x and d_y features, respectively. CCA aims to find K pairs of linear projections $W_x = [\mathbf{w}_{x,1}, \mathbf{w}_{x,2}, \ldots, \mathbf{w}_{x,K}] \in \mathbb{R}^{d_x \times K}$ and $W_y = [\mathbf{w}_{y,1}, \mathbf{w}_{y,2}, \ldots, \mathbf{w}_{y,K}] \in \mathbb{R}^{d_x \times K}$, called canonical vectors, so that the correlations between $W_x^T X$ and $W_y^T Y$ are maximized.

Take a canonical vector $\mathbf{w}_x \in \mathbb{R}^{d_x \times 1}$ for X and a canonical vector $\mathbf{w}_y \in \mathbb{R}^{d_y \times 1}$ for Y for example. CCA maximizes the correlation coefficient ρ between $\mathbf{w}_x^T X$ and $\mathbf{w}_y^T Y$ [34], i.e.,

$$\rho\left(\mathbf{w}_x^T X, \mathbf{w}_y^T Y\right) = \frac{\mathbf{w}_x^T X Y^T \mathbf{w}_y}{\sqrt{\left(\mathbf{w}_x^T X X^T \mathbf{w}_x\right)\left(\mathbf{w}_y^T Y Y^T \mathbf{w}_y\right)}}. \tag{1}$$

Since (1) is invariant to the scaling of $\mathbf{w}_x$ and $\mathbf{w}_y$, it can be transformed into the following constrained form:

$$\begin{aligned} \max_{\mathbf{w}_x, \mathbf{w}_y} \; & \mathbf{w}_x^T X Y^T \mathbf{w}_y \\ s.t. \; & \mathbf{w}_x^T X X^T \mathbf{w}_x = 1, \mathbf{w}_y^T Y Y^T \mathbf{w}_y = 1. \end{aligned} \tag{2}$$

When the feature dimensionality is high, especially when $d_x > N$ (or $d_y > N$), the covariance matrix XX^T (or YY^T) is singular, and hence the optimization problem is underdetermined. Regularizations can be added to the covariance matrices to remedy this problem [35]–[37], by introducing:

$$\hat{\Sigma}_{xx} = \frac{1}{N} XX^T + r_x I, \tag{3}$$

$$\hat{\Sigma}_{yy} = \frac{1}{N} YY^T + r_y I, \tag{4}$$

where r_x and r_y are non-negative regularization coefficients.

There are two approaches for computing W_x and W_y directly. The first is to solve the following generalized eigenvalue decomposition problem [34]:

$$\begin{bmatrix} \mathbf{0} & \Sigma_{xy} \\ \Sigma_{yx} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w_x} \\ \mathbf{w_y} \end{bmatrix} = \lambda \begin{bmatrix} \hat{\Sigma}_{xx} & \mathbf{0} \\ \mathbf{0} & \hat{\Sigma}_{yy} \end{bmatrix} \begin{bmatrix} \mathbf{w_x} \\ \mathbf{w_y} \end{bmatrix}, \tag{5}$$

where

$$\Sigma_{xy} = \frac{1}{N} XY^T, \tag{6}$$

$$\Sigma_{yx} = \frac{1}{N} YX^T. \tag{7}$$

$\{[\mathbf{w}_{x,k}; \mathbf{w}_{y,k}]\}_{k=1}^K$ are then the K leading generalized eigenvectors. The correlation $\rho\left(\mathbf{w}_{x,k}^T X, \mathbf{w}_{y,k}^T Y\right)$ is equal to the k-th leading generalized eigenvalue.

The second solution [23] performs singular value decomposition (SVD) on matrix $T = \hat{\Sigma}_{xx}^{-1/2} \Sigma_{xy} \hat{\Sigma}_{yy}^{-1/2}$. Let $\tilde{W}_x$ and $\tilde{W}_y$ be the K leading left and right singular vectors of T. Then, the canonical matrices are $W_x = \hat{\Sigma}_{xx}^{-1/2} \tilde{W}_x$ and $W_y = \hat{\Sigma}_{yy}^{-1/2} \tilde{W}_y$. The correlation $\rho\left(\mathbf{w}_{x,k}^T X, \mathbf{w}_{y,k}^T Y\right)$ is equal to the k-th leading singular value of T.

Once W_x and W_y are obtained, the projected new features, called canonical variables, are computed by $Z_x = W_x^T X$ and $Z_y = W_y^T Y$.

B. Sparse CCA (sCCA)

Canonical matrices W_x and W_y calculated by CCA is dense. Based on penalized matrix decomposition (PMD) [38], Witten *et al.* [39] proposed penalized CCA, called sCCA in this paper, to obtain spare canonical vectors. sCCA also remedies the problem that canonical vectors are not unique when N is smaller than d_x and/or d_y.

PMD [40] performs rank-m approximation of an arbitrary matrix $M \in \mathbb{R}^{d \times N}$ by $\hat{M} = \sum_{i=1}^m d_i \mathbf{u}_i \mathbf{v}_i^T$, using SVD, where d_i is the i-th leading singular value, and $\mathbf{u}_i \in \mathbb{R}^{d \times 1}$ and $\mathbf{v}_i \in \mathbb{R}^{N \times 1}$ are the corresponding left and right singular vectors, respectively. The sparsity of $\hat{M}$ can be achieved by imposing LASSO-constraints on $\mathbf{u}_i$ and $\mathbf{v}_i$. When $m = 1$, minimizing $\|M - \hat{M}\|_F^2$ is equivalent to maximizing the following objective function:

$$\begin{aligned} &\max_{\mathbf{u},\mathbf{v}} \ \mathbf{u}^T M \mathbf{v} \\ &\quad s.t.\ \|\mathbf{u}\|_2^2 = 1,\ \|\mathbf{v}\|_2^2 = 1,\ p_1(\mathbf{u}) \leqslant c_1,\ p_2(\mathbf{v}) \leqslant c_2, \end{aligned} \tag{8}$$

where $p_1(\mathbf{u}) = \sum_{i=1}^d |u_i|$ and $p_2(\mathbf{v}) = \sum_{i=1}^N |v_i|$ represent the LASSO penalties, and parameters c_1 and c_2 control the degree of sparsity.

sCCA can be solved by replacing M in (8) with the cross covariance matrix XY^T, i.e.,

$$\begin{aligned} &\max_{\mathbf{w}_x,\mathbf{w}_y} \ \mathbf{w}_x^T XY^T \mathbf{w}_y \\ &\quad s.t.\ \|\mathbf{w}_x\|_2^2 = 1, \|\mathbf{w}_y\|_2^2 = 1, p_1(\mathbf{w}_x) \leqslant c_1, p_2(\mathbf{w}_y) \leqslant c_2. \end{aligned} \tag{9}$$

K pairs of canonical vectors, $\{(\mathbf{w}_{x,k}, \mathbf{w}_{x,k})\}_{k=1}^K$, can be obtained by solving a multi-factor PMD [38].

C. Kernel CCA (KCCA)

The traditional CCA cannot be applied when the correlation between different views is nonlinear. Kernel CCA (KCCA) uses (nonlinear) kernels to project data onto a higher dimensional space for correlation analysis.

Let the projections be ϕ_x and ϕ_y, and the projected views in the high-dimensional space be $\Phi_x = [\phi_x(\mathbf{x}_1), \phi_x(\mathbf{x}_2), \ldots, \phi_x(\mathbf{x}_N)]$ and $\Phi_\mathrm{y} = [\phi_y(\mathbf{y}_1), \phi_y(\mathbf{y}_2), \ldots, \phi_y(\mathbf{y}_N)]$, respectively. The objective function of KCCA is:

$$\begin{aligned} &\max_{\mathbf{w}_x,\mathbf{w}_y} \ \mathbf{w}_x^T \Phi_x \Phi_y^T \mathbf{w}_y \\ &\quad s.t.\ \mathbf{w}_x^T \Phi_x \Phi_x^T \mathbf{w}_x = 1, \mathbf{w}_y^T \Phi_y \Phi_y^T \mathbf{w}_y = 1. \end{aligned} \tag{10}$$

Expressing $\mathbf{w}_x$ and $\mathbf{w}_y$ as linear combinations of the columns of Φ_x and Φ_y, respectively [19]:

$$\mathbf{w}_x = \sum_{i=1}^N a^i \phi_x(\mathbf{x}_i) = \Phi_x \mathbf{a}, \tag{11}$$

$$\mathbf{w}_y = \sum_{i=1}^N b^i \phi_y(\mathbf{y}_i) = \Phi_y \mathbf{b}, \tag{12}$$

where $\mathbf{a} = \left[a^1, \cdots, a^N\right]^T$ and $\mathbf{b} = \left[b^1, \cdots, b^N\right]^T$ are linear coefficients. (10) can then be rewritten as:

$$\begin{aligned} &\max_{\mathbf{a},\mathbf{b}} \ \mathbf{a}^T \Phi_x^T \Phi_x \Phi_y^T \Phi_y \mathbf{b} \\ &\quad s.t.\ \mathbf{a}^T \Phi_x^T \Phi_x \Phi_x^T \Phi_x \mathbf{a} = 1,\ \mathbf{b}^T \Phi_y^T \Phi_y \Phi_y^T \Phi_y \mathbf{b} = 1. \end{aligned} \tag{13}$$

Let $K_x = \Phi_x^T \Phi_x$ and $K_y = \Phi_y^T \Phi_y$ be kernel matrices, i.e., $(K_x)_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$, where κ is a kernel function, such as a radial basis function (RBF). Then, (13) can be further simplified to:

$$\begin{aligned} &\max_{\mathbf{a},\mathbf{b}} \ \mathbf{a}^T K_x K_y \mathbf{b} \\ &\quad s.t.\ \mathbf{a}^T K_x K_x \mathbf{a} = 1, \mathbf{b}^T K_y K_y \mathbf{b} = 1. \end{aligned} \tag{14}$$

We can further add regularizations to the kernel matrices to make them numerically more stable [34], e.g., replace $K_x K_x$ with $K_x K_x + r_x K_x$, and $K_y K_y$ with $K_y K_y + r_y K_y$. The linear coefficient vectors $\mathbf{a}$ and $\mathbf{b}$ can then be solved by the following generalized eigen decomposition problem:

$$\begin{aligned} &\begin{bmatrix} \mathbf{0} & K_x K_y \\ K_y K_x & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} \\ &= \lambda \begin{bmatrix} K_x K_x + r_x K_x & \mathbf{0} \\ \mathbf{0} & K_y K_y + r_y K_y \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} \end{aligned} \tag{15}$$

D. Randomized Nonlinear CCA (RCCA)

KCCA has very high computational complexity. Lopez-Paz *et al.* [41] handled the nonlinear characteristics of data by performing random nonlinear projection, which greatly reduced the computational difficulty, with little scarification of performance:

$$\mathrm{RCCA}\,(X,Y) := \mathrm{CCA}(f_x(X), f_y(Y)) \approx \mathrm{KCCA}(X,Y), \tag{16}$$

where $f_x(\cdot)$ represents a mapping function, whose parameters $Q_x = \left(\boldsymbol{q}_{x,1}, \boldsymbol{q}_{x,2}, \ldots, \boldsymbol{q}_{x,D}\right) \in \mathbb{R}^{d_x \times D}$ are randomly sampled from a given data-independent distribution $p(\boldsymbol{q})$, e.g., a Gaussian distribution. The D-dimensional nonlinear random features Z_x can be obtained by:

$$\mathbf{z}_{x,d} := \left[\cos\left(\boldsymbol{q}_{x,d}^T \mathbf{x}_1 + b_d\right), \ldots, \cos\left(\boldsymbol{q}_{x,d}^T \mathbf{x}_N + b_d\right)\right]^T \in \mathbb{R}^{N\times 1}, \tag{17}$$
$$Z_x = \left[\mathbf{z}_{x,1}, \ldots, \mathbf{z}_{x,D}\right]^T \in \mathbb{R}^{D\times N},$$

where $\mathbf{b} = [b_1, b_2, \ldots, b_D]^T \in \mathbb{R}^{D\times 1}$ is a randomly generated bias term. The cosine mapping function in (17) can also be replaced by other nonlinear functions, e.g., sine. As in CCA, the computational complexity of RCCA also increases linearly with the sample size.

E. Locality Preserving CCA (LPCCA)

In addition to the computational complexity, the global kernelization strategy of KCCA ignores the local linear structure of complex data. Locality preserving CCA (LPCCA) [20] is another nonlinear CCA extension, which preserves the local linear structure of the data while performing global nonlinear dimensionality reduction.

LPCCA assumes that the corresponding instances of different views should be as close as possible in the common latent space, so it can be expressed in the following equivalent form [42]:

$$\min_{\mathbf{w}_x,\mathbf{w}_y} \sum_{i=1}^{N} \left\| \mathbf{w}_x^T (\mathbf{x}_i - \overline{\mathbf{x}}) - \mathbf{w}_y^T (\mathbf{y}_i - \overline{\mathbf{y}}) \right\|_2^2, \tag{18}$$
$$s.t. \sum_{i=1}^{N} \left\| \mathbf{w}_x^T (\mathbf{x}_i - \overline{\mathbf{x}}) \right\|^2 = 1, \sum_{i=1}^{n} \left\| \mathbf{w}_y^T (\mathbf{y}_i - \overline{\mathbf{y}}) \right\|^2 = 1$$

where $\overline{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i$ and $\overline{\mathbf{y}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{y}_i$ represent the mean vectors of X and Y, respectively. After some algebraic manipulations, (18) can be re-expressed as:

$$\max_{\mathbf{w}_x,\mathbf{w}_y} \mathbf{w}_x^T \cdot \sum_{i=1}^{N}\sum_{j=1}^{N} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{y}_i - \mathbf{y}_j)^T \cdot \mathbf{w}_y \tag{19}$$
$$s.t.\ \mathbf{w}_x^T \cdot \sum_{i=1}^{N}\sum_{j=1}^{N} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \cdot \mathbf{w}_x = 1,$$
$$\mathbf{w}_y^T \cdot \sum_{i=1}^{N}\sum_{j=1}^{N} (\mathbf{y}_i - \mathbf{y}_j)(\mathbf{y}_i - \mathbf{y}_j)^T \cdot \mathbf{w}_y = 1.$$

Let $\mathrm{Nei}\,(\mathbf{x}_i)$ be the neighbor set of $\mathbf{x}_i$. Define a similarity matrix S_x, whose ij-th element is:

$$S_{x,ij} = \begin{cases} \exp\left(-\left\|\mathbf{x}_i - \mathbf{x}_j\right\|_2^2 / t_x\right), & \mathbf{x}_j \in \mathrm{Nei}\,(\mathbf{x}_i) \\ 0, & \text{otherwise} \end{cases}, \tag{20}$$

where $t_x = \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{2\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{N(N-1)}$ represents the mean squared distance between all instances. The similarity matrix S_y of Y can be computed in a similar manner.

Substituting S_x and S_y into (19), the global correlation is decomposed into many local linear correlations between the neighboring instances. The objective of LPCCA is:

$$\max_{\mathbf{w}_x,\mathbf{w}_y} \mathbf{w}_x^T \cdot \sum_{i=1}^{N}\sum_{j=1}^{N} S_{x,ij}(\mathbf{x}_i - \mathbf{x}_j) S_{y,ij}(\mathbf{y}_i - \mathbf{y}_j)^T \cdot \mathbf{w}_y \tag{21}$$
$$s.t.\ \mathbf{w}_x^T \cdot \sum_{i=1}^{N}\sum_{j=1}^{N} S_{x,ij}(\mathbf{x}_i - \mathbf{x}_j) S_{x,ij}(\mathbf{x}_i - \mathbf{x}_j)^T \cdot \mathbf{w}_x = 1,$$
$$\mathbf{w}_y^T \cdot \sum_{i=1}^{N}\sum_{j=1}^{N} S_{y,ij}(\mathbf{y}_i - \mathbf{y}_j) S_{y,ij}(\mathbf{y}_i - \mathbf{y}_j)^T \cdot \mathbf{w}_y = 1,$$

which can be transformed into a generalized eigen decomposition problem:

$$\begin{bmatrix} \mathbf{0} & XS_{xy}Y^T \\ YS_{yx}X^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w}_x \\ \mathbf{w}_y \end{bmatrix} = \lambda \begin{bmatrix} XS_{xx}X^T & \mathbf{0} \\ \mathbf{0} & YS_{yy}Y^T \end{bmatrix} \begin{bmatrix} \mathbf{w}_x \\ \mathbf{w}_y \end{bmatrix}, \tag{22}$$

where

$$S_{xy} = D_{xy} - S_x \odot S_y, \tag{23}$$
$$S_{yx} = D_{yx} - S_y \odot S_x, \tag{24}$$
$$S_{xx} = D_{xx} - S_x \odot S_x, \tag{25}$$
$$S_{yy} = D_{yy} - S_y \odot S_y, \tag{26}$$

in which $\odot$ denotes the element-wise product operation, and D_{xy} is a diagonal matrix, with $(D_{xy})_{ii} = \sum_{j=1}^{N} (S_x \odot S_y)_{ij}$. D_{yx}, D_{xx} and D_{yy} are defined similarly.

F. Discriminative CCA (DisCCA)

Traditional CCA is unsupervised. In supervised classification, we have label information, which should be taken into consideration to help extract more discriminative features.

Discriminative CCA (DisCCA) is one such approach. It maximizes the within-class similarity and minimizes the between-class similarity. Rearrange X and Y according to the classes:

$$\hat{X} = \left[\mathbf{x}_1^{(1)}, \ldots, \mathbf{x}_{n_1}^{(1)}, \ldots\ldots, \mathbf{x}_1^{(c)}, \ldots, \mathbf{x}_{n_c}^{(c)}\right], \tag{27}$$
$$\hat{Y} = \left[\mathbf{y}_1^{(1)}, \ldots, \mathbf{y}_{n_1}^{(1)}, \ldots\ldots, \mathbf{y}_1^{(c)}, \ldots, \mathbf{y}_{n_c}^{(c)}\right], \tag{28}$$

where $\mathbf{x}_j^{(i)}$ and $\mathbf{y}_j^{(i)}$ are the j-th instance of Class i from the two views. The objective function of DisCCA is:

$$\max_{\mathbf{w}_x,\mathbf{w}_y} \mathbf{w}_x^T C_{\mathrm{w}} \mathrm{w}_y - \eta \cdot \mathbf{w}_x^T C_{\mathrm{b}} \mathbf{w}_y \tag{29}$$
$$s.t.\ \mathbf{w}_x^T \hat{X}\hat{X}^T \mathbf{w}_x = 1, \mathbf{w}_y^T \hat{Y}\hat{Y}^T \mathbf{w}_y = 1,$$

where η is a trade-off parameter, and the within-class similarity matrix C_{w} and between-class similarity matrix C_{b} are defined as:

$$C_{\mathrm{w}} = \sum_{i=1}^{c}\sum_{k=1}^{n_i}\sum_{l=1}^{n_j} \mathbf{x}_k^{(i)} \left(\mathbf{y}_l^{(j)}\right)^T = \hat{X} A \hat{Y}^T, \tag{30}$$

$$C_{\mathrm{b}} = \sum_{i=1}^{c}\sum_{\substack{j=1\\ j\neq i}}^{c}\sum_{k=1}^{n_i}\sum_{l=1}^{n_j} \mathbf{x}_k^{(i)} \left(\mathbf{y}_l^{(j)}\right)^T = -\hat{X} A \hat{Y}^T, \tag{31}$$

where A is a block-diagonal matrix:

$$A = \begin{bmatrix} \mathbf{1}_{n_1\times n_1} & & & & \\ & \ddots & & & \\ & & \mathbf{1}_{n_i\times n_i} & & \\ & & & \ddots & \\ & & & & \mathbf{1}_{n_c\times n_c} \end{bmatrix}. \tag{32}$$

Since $C_{\mathrm{w}} = -C_{\mathrm{b}}$, the optimization is independent of η, and hence (29) can be rewritten as:

$$\begin{aligned} \max_{\mathbf{w}_x,\mathbf{w}_y}\ & \mathbf{w}_x^T \hat{X} A \hat{Y}^T \mathbf{w}_y \\ s.t.\ & \mathbf{w}_x^T \hat{X}\hat{X}^T \mathbf{w}_x = 1, \mathbf{w}_y^T \hat{Y}\hat{Y}^T \mathbf{w}_y = 1, \end{aligned} \tag{33}$$

whose solution is similar to (1) and can also be expressed as a generalized eigen decomposition problem:

$$\begin{bmatrix} \mathbf{0} & \tilde{\Sigma}_{xy} \\ \tilde{\Sigma}_{yx} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{\mathrm{x}} \\ \mathbf{w}_{\mathrm{y}} \end{bmatrix} = \lambda \begin{bmatrix} \hat{\Sigma}_{xx} & \mathbf{0} \\ \mathbf{0} & \hat{\Sigma}_{yy} \end{bmatrix} \begin{bmatrix} \mathbf{w}_x \\ \mathbf{w}_y \end{bmatrix}, \tag{34}$$

where $\tilde{\Sigma}_{xy} = \frac{1}{N}\hat{X}A\hat{Y}^T$ and $\tilde{\Sigma}_{yx} = \frac{1}{N}\hat{Y}A\hat{X}^T$.

G. Multi-view Linear Discriminant Analysis (MLDA)

LDA [33] is a supervised algorithm for a single view that minimizes the within-class variance and maximizes the between-class variance. Multi-view linear discriminant analysis (MLDA) [32] combines LDA and CCA, which not only ensures the discriminative ability within a single view, but also maximizes the correlation between different views.

Define a between-class scatter matrix $S_b = \frac{1}{N}\hat{X}W\hat{X}^T$, where $\hat{X}$ is given in (27), $W = diag(W_1, W_2, \ldots, W_c)$, with all elements in $W_i \in \mathbb{R}^{n_i\times n_i}$ equal $\frac{1}{n_i}$. Then, for a single view, the objective function of LDA is:

$$\begin{aligned} \max_{\mathbf{w}}\ & \mathbf{w}^T S_{\mathrm{b},x} \mathbf{w} \\ s.t.\ & \mathbf{w}^T \hat{\Sigma}_{xx} \mathbf{w} = 1. \end{aligned} \tag{35}$$

Integrating with (1), the objective function of MLDA is:

$$\begin{aligned} \max_{\mathbf{w}_x,\mathbf{w}_y}\ & \mathbf{w}_x^T S_{\mathrm{b},x}\mathbf{w}_x + \mathbf{w}_y^T S_{\mathrm{b},y}\mathbf{w}_y + \eta \mathbf{w}_x^T \Sigma_{xy}\mathbf{w}_y \\ s.t.\ & \mathbf{w}_x^T \hat{\Sigma}_{xx}\mathbf{w}_x + \sigma \mathbf{w}_y^T \hat{\Sigma}_{yy}\mathbf{w}_y = 1, \end{aligned} \tag{36}$$

where η is a trade-off parameter, and $\sigma = \frac{tr(\hat{\Sigma}_{xx})}{tr(\hat{\Sigma}_{yy})}$. The constraint in (36) is in the form of summation to ensure a closed-form solution can be obtained. Using the Lagrangian multiplier, (36) can also be solved by generalized eigen decomposition:

$$\begin{bmatrix} S_{\mathrm{b},x} & \eta\Sigma_{xy} \\ \eta\Sigma_{yx} & S_{\mathrm{b},y} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{\mathrm{x}} \\ \mathbf{w}_{\mathrm{y}} \end{bmatrix} = \lambda \begin{bmatrix} \hat{\Sigma}_{xx} & \mathbf{0} \\ \mathbf{0} & \hat{\Sigma}_{yy} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{\mathbf{x}} \\ \mathbf{w}_{\mathbf{y}} \end{bmatrix}. \tag{37}$$

H. Multi-view Uncorrelated Linear Discriminant Analysis (MULDA)

Canonical vectors $\{(\mathbf{w}_{x,k}, \mathbf{w}_{y,k})\}_{k=1}^K$, calculated by MLDA, are correlated, which means the projected canonical variables contain redundant information. Multi-view uncorrelated linear discriminant analysis (MULDA) [32] aims to eliminate the information redundancy of the canonical variables using uncorrelated linear discriminant analysis (ULDA) [43].

ULDA imposes orthogonal constraints to LDA, and is solved recursively. Each time a projection vector $\mathbf{w}_k$ is calculated to be orthogonal to the $k-1$ projection vectors that have been obtained:

$$\begin{aligned} \max_{\mathbf{w}}\ & \mathbf{w}^T S_{\mathrm{b},x}\mathbf{w} \\ s.t.\ & \mathbf{w}_k^T \hat{\Sigma}_{xx}\mathbf{w}_i = 0, \quad i = 1, 2, \ldots, k-1. \end{aligned} \tag{38}$$

Integrating ULDA with MLDA, the objective function of MULDA can be written as:

$$\begin{aligned} \max_{\mathbf{w}_{x,k},\mathbf{w}_{y,k}}\ & \mathbf{w}_{x,k}^T S_{\mathrm{b},x}\mathbf{w}_{x,k} + \mathbf{w}_{y,k}^T S_{\mathrm{b},y}\mathbf{w}_{y,k} + \eta \mathbf{w}_{x,k}^T \Sigma_{xy}\mathbf{w}_{y,k} \\ s.t.\ & \mathbf{w}_{x,k}^T \hat{\Sigma}_{xx}\mathbf{w}_{x,k} + \sigma \mathbf{w}_{y,k}^T \hat{\Sigma}_{yy}\mathbf{w}_{y,k} = 1, \\ & \mathbf{w}_{x,k}^T \hat{\Sigma}_{xx}\mathbf{w}_{x,i} = 0, \mathbf{w}_{y,k}^T \hat{\Sigma}_{yy}\mathbf{w}_{y,i} = 0, \\ & i = 1, \ldots, k-1. \end{aligned} \tag{39}$$

The k-th pair of canonical vectors, $(\mathbf{w}_{x,k}, \mathbf{w}_{y,k})$, is the leading eigenvector of

$$\begin{bmatrix} P_x & \mathbf{0} \\ \mathbf{0} & P_y \end{bmatrix} \begin{bmatrix} S_{\mathrm{b},x} & \eta\Sigma_{xy} \\ \eta\Sigma_{yx} & S_{\mathrm{b},y} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{x,k} \\ \mathbf{w}_{y,k} \end{bmatrix} = \lambda \begin{bmatrix} \hat{\Sigma}_{xx} & \mathbf{0} \\ \mathbf{0} & \hat{\Sigma}_{yy} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{x,k} \\ \mathbf{w}_{y,k} \end{bmatrix}, \tag{40}$$

where

$$P_x = I - \hat{\Sigma}_{xx} D_x^T \left(D_x \hat{\Sigma}_{xx} D_x^T\right)^{-1} D_x, \tag{41}$$

$$D_x = \left[\mathbf{w}_{x,1}, \mathbf{w}_{x,2}, \cdots, \mathbf{w}_{x,k-1}\right]^T. \tag{42}$$

P_y and D_y are defined similarly.

Unlike DisCCA, MLDA and MULDA only focus on the within-view class scatters, without considering the between-view class scatter. Sun *et al.* [32] proposed that CCA in MLDA and MULDA can be replaced by DisCCA, to also consider the between-view class scatter. Moreover, MLDA and MULDA can also be extended using kernels to consider the nonlinear relationship between different views.

I. Deep CCA (DCCA) and Discriminative DCCA (DisDCCA)

Deep canonical correlation analysis (DCCA) [23], shown in Fig. 2, first extracts nonlinear features through deep neural networks (DNNs), and then uses linear CCA to calculate the canonical matrices.

Let $\mathbf{f}_x$ and $\mathbf{f}_y$ be two DNNs, and $H_x = \mathbf{f}_x(X)$ and $H_y = \mathbf{f}_y(Y)$ be their outputs, respectively. We already know that the total correlation of the K canonical variables equals the sum

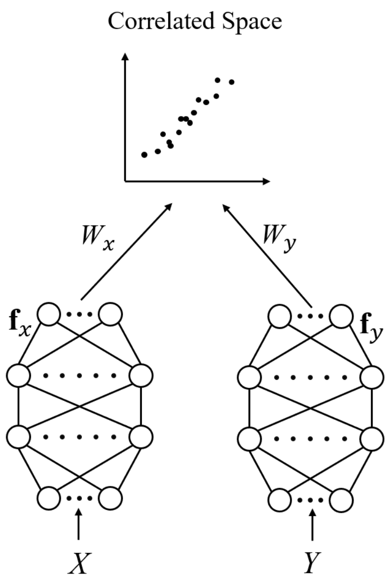


Fig. 2. Schematic diagram of DCCA. Two deep neural networks $\mathbf{f}_x$ and $\mathbf{f}_y$ first extract nonlinear features from X and Y, respectively, and then a linear CCA is applied for correlation analysis.

of the first K singular values of matrix $T = \hat{\Sigma}_{xx}^{-1/2}\Sigma_{xy}\hat{\Sigma}_{yy}^{-1/2}$ from II-A. Define

$$\hat{T} = \left(\frac{1}{N}H_x H_x^T + r_x I\right)^{-\frac{1}{2}} \left(\frac{1}{N}H_x H_y^T\right)\left(\frac{1}{N}H_y H_y^T + r_y I\right)^{-\frac{1}{2}}. \tag{43}$$

The objective function of DCCA is:

$$\max_{\mathbf{f}_x,\mathbf{f}_y,\mathbf{w}_x,\mathbf{w}_\mathbf{y}} \sum_{k=1}^{K} \sigma_k(\hat{T}) \tag{44}$$
$$s.t.\ \mathbf{w}_x\left(\frac{1}{N}H_x H_x^T + r_x I\right)\mathbf{w}_x^T = 1,$$
$$\mathbf{w}_y\left(\frac{1}{N}H_y H_y^T + r_y I\right)\mathbf{w}_y^T = 1,$$

where $\sigma_k(\hat{T})$ denotes the k-th largest singular value of $\hat{T}$.

Andrew *et al.* [23] employed a full-batch optimization algorithm (L-BFGS) to optimize (44), because the covariance matrix of the entire training set needs to be computed. If the dataset is divided into small batches, then the covariance matrix computed from each batch may not be accurate. However, full-batch optimization is both memory-hungry and time-consuming, especially when the dataset is large. Wang *et al.* [44] proposed that with a large batch size, the instances in each batch can be sufficient for estimating the covariance matrix, and the efficient mini-batch stochastic gradient descent approach can be used.

Similar to DCCA, discriminative deep canonical correlation analsis (DisDCCA) [25] is a DNN-based extension of discriminative CCA. There are two major differences in their implementations. First, when calculating the loss of each batch, the instances are first rearranged according to their classes, and then $\frac{1}{N}H_x H_y^T$ in (43) is replaced by $\frac{1}{N}H_x A H_y^T$, where A is defined in (32). Second, after the model is trained, DisCCA is used to obtain the canonical matrices W_x and W_y.

J. Deep Canonically Correlated Auto-Encoders (DCCAE) and Discriminative DCCAE (DisDCCAE)

Deep canonically correlated auto-encoders (DCCAE) [21] improves DCCA, by using auto-encoders to make sure the information captured by $\mathbf{f}_x$ and $\mathbf{f}_y$ can also accurately reconstruct the original X and Y. Fig. 3 shows the DCCAE model structure, where $\mathbf{g}_x$ and $\mathbf{g}_y$ represent the decoder networks for reconstructing X and Y, respectively. Therefore, two reconstruction errors are incorporated into (44):

$$\max_{\mathbf{f}_x,\mathbf{f}_y,\mathbf{g}_x,\mathbf{g}_y,\mathbf{w}_x,\mathbf{w}_\mathbf{y}} \sum_{k=1}^{K}\sigma_k\left(\hat{T}\right) - \frac{\lambda}{N}\left(\|X - \mathbf{g}_x\left(\mathbf{f}_x\left(X\right)\right)\|_F^2 + \|Y - \mathbf{g}_y\left(\mathbf{f}_y\left(Y\right)\right)\|_F^2\right) \tag{45}$$
$$s.t.\ \mathbf{w}_x(\frac{1}{N}H_x H_x^T + r_x I)\mathbf{w}_x^T = 1,$$
$$\mathbf{w}_y(\frac{1}{N}H_y H_y^T + r_y I)\mathbf{w}_y^T = 1.$$

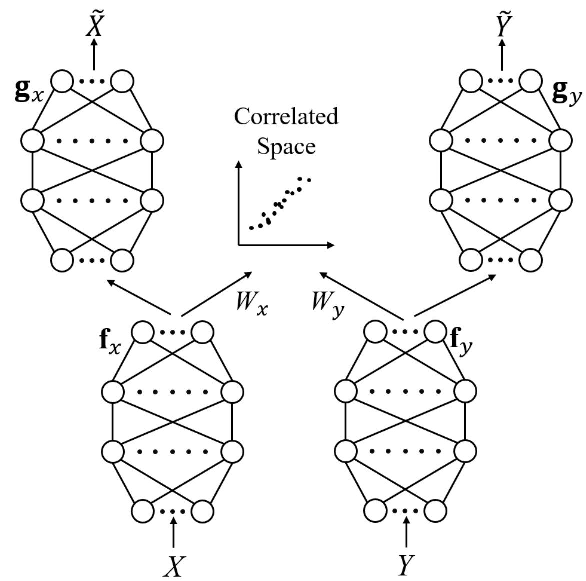


Fig. 3. Schematic diagram of DCCAE. Two deep neural networks $\mathbf{f}_x$ and $\mathbf{f}_y$ first extract nonlinear features from X and Y, respectively, and then a linear CCA is applied for correlation analysis. $\mathbf{g}_x$ and $\mathbf{g}_y$ are two decoders for reconstructing View X and View Y, respectively.

The test procedure is similar to DCCA, except that $W_x^T\mathbf{f}_x(\cdot)$ and $W_y^T\mathbf{f}_y(\cdot)$ are used as inputs to a classifier. The advantage of DCCAE is that the auto-encoders can alleviate the overfitting of the model.

K. Variational CCA (VCCA) and VCCA-private

Bach and Jordan [45] explained CCA from the perspective of a probabilistic latent variable model. It assumes that the instances $\mathbf{x}$ and $\mathbf{y}$ from two views are independently conditioned on the multivariate latent variable $\mathbf{z} \in \mathbb{R}^{d_z\times 1}$, and CCA aims to maximize the joint probability distribution of $\mathbf{x}$ and $\mathbf{y}$:

$$p(\mathbf{x},\mathbf{y},\mathbf{z}) = p(\mathbf{z})p(\mathbf{x}|\mathbf{z})p(\mathbf{y}|\mathbf{z}), \tag{46}$$

$$p(\mathbf{x},\mathbf{y}) = \int p(\mathbf{x},\mathbf{y},\mathbf{z})d\mathbf{z}. \tag{47}$$

Deep variational canonical correlation analysis (VCCA) [46], shown in Fig. 4(a), was extended from variational auto-encoders (VAE) [47]. First, the mean vector $\boldsymbol{\mu}_i$ and the diagonal covariance matrix $\boldsymbol{\Sigma}_i$ of $\mathbf{x}_i$ are calculated by the encoder $\mathbf{f}_x$. Then, L instances, $\{\mathbf{z}_i^{(l)}\}_{l=1}^{L}$, are randomly sampled from the distribution $\mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$. Finally, decoders $\mathbf{g}_x$ and $\mathbf{g}_y$ reconstruct instances $\mathbf{x}$ and $\mathbf{y}$, respectively.

The objective function of VCCA is:

$$\begin{aligned}\min_{\mathbf{f}_x,\mathbf{g}_x,\mathbf{g}_y} \frac{1}{N}\sum_{i=1}^{N} D_{KL}\left(q(\mathbf{z}_i|\mathbf{x}_i)||p\left(\mathbf{z}_i\right)\right) + \\ \frac{\lambda}{NL}\sum_{i=1}^{N}\sum_{l=1}^{L}\left(\log p\left(\mathbf{x}_i|\mathbf{z}_i^{(l)}\right) + \log p\left(\mathbf{y}_i|\mathbf{z}_i^{(l)}\right)\right),\end{aligned} \tag{48}$$

where the first term denotes the Kullback-Leibler (KL) divergence between the posterior distributions $q(\mathbf{z}_i|\mathbf{x}_i)$ and $p\left(\mathbf{z}_i\right)$, and the latter two terms denote the expectations of the log-likelihood under the approximate posterior distributions, which are equivalent to the reconstruction error. In testing, the mean vector $\boldsymbol{\mu}_i$ is used as the input features.

VCCA only takes the common latent variables $\mathbf{z}$ into consideration, which may not be sufficient to describe all information in X and Y. In order to more adequately capture the private information of the views, Wang *et al.* [46] further proposed VCCA-private, shown in Fig. 4(b). Its probabilistic model is defined as:

$$p(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{h}_x,\mathbf{h}_y) = p(\mathbf{z})p(\mathbf{h}_x)p(\mathbf{h}_y)p(\mathbf{x}|\mathbf{z},\mathbf{h}_x)p(\mathbf{y}|\mathbf{z},\mathbf{h}_y), \tag{49}$$

$$p(\mathbf{x},\mathbf{y}) = \iiint p(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{h}_x,\mathbf{h}_y)d\mathbf{z}d\mathbf{h}_x d\mathbf{h}_y. \tag{50}$$

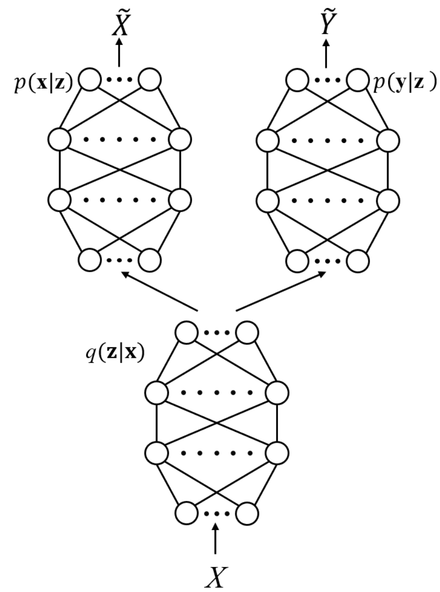
The corresponding objective function of VCCA-private is:

$$\begin{aligned}\min_{\mathbf{f}_{x,h},\mathbf{f}_{x,z},\mathbf{f}_{y,h},\mathbf{g}_x,\mathbf{g}_y} & \frac{1}{N}\sum_{i=1}^{N}(D_{KL}\left(q\left(\mathbf{z}_i|\mathbf{x}_i\right)||p\left(\mathbf{z}_i\right)\right) \\ & + D_{KL}\left(q\left(\mathbf{h}_{x,i}|\mathbf{x}_i\right)||p\left(\mathbf{h}_{x,i}\right)\right) \\ & + D_{KL}\left(q\left(\mathbf{h}_{y,i}|\mathbf{y}_i\right)||p\left(\mathbf{h}_{y,i}\right)\right)) \\ & + \frac{\lambda}{NL}\sum_{i=1}^{N}\sum_{l=1}^{L}\left(\log p\left(\mathbf{x}_i|\mathbf{z}_i^{(l)},\mathbf{h}_{x,i}^{(l)}\right)\right. \\ & \left. + \log p\left(\mathbf{y}_i|\mathbf{z}_i^{(l)},\mathbf{h}_{y,i}^{(l)}\right)\right).\end{aligned} \tag{51}$$

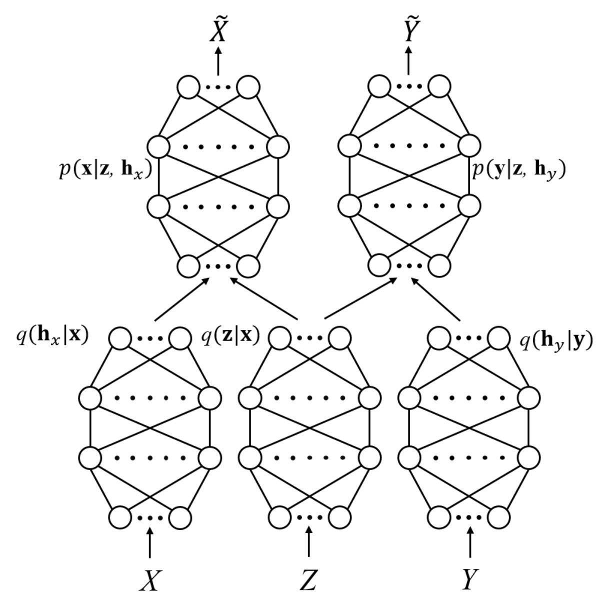
In testing, the mean vectors output by the three encoders are stacked as new features. According to [46], VCCA-private is more effective than VCCA.

L. Summary

This section has introduced the traditional CCA and its six nonlinear extensions, three supervised extensions, and one sparse extension. Three ideas were used in the nonlinear extensions, namely, kernel-based KCCA, locality preserving LPCCA, and DNN-based DCCA (DCCAE, DVCCA). The first two are more suitable for small datasets. The supervised extensions (DisCCA, MLDA and MULDA) are closely related to LDA, because they all take the between-class and within-class scatter matrices into consideration.



(a)



(b)

Fig. 4. Schematic diagrams of DVCCA and DVCCA-private. The encoders $q(\cdot|x)$ and $q(\cdot|y)$ are used to fit the mean and covariance of the input, and the $p(x|\cdot)$ and $p(y|\cdot)$ are decoders for reconstructing the input data.

III. CCA for More than Two Views

This section summarizes five representative CCA approaches for more than two views, which are SUMCOR-GCCA, MAXVAR-GCCA, LS-CCA, TCCA, and DGCCA.

A. SUMCOR-GCCA

CCA maximizes the correlation between two views. For more than two views, a natural extension is to maximize the sum of the pairwise correlations [28] [30], which is the idea of SUMCOR-GCCA.

Let $\left\{X_j \in \mathbb{R}^{d_j \times N}\right\}_{j=1}^{J}$ be a dataset containing J mean-zero views, where d_j is the feature dimensionality of View j. The objective function of SUMCOR-GCCA is:

$$\begin{aligned} \max_{\{\mathbf{w}_i\}_{i=1}^{J}} & \sum_{i=1}^{J}\sum_{j=1}^{J} \mathbf{w}_i^T X_i X_j^T \mathbf{w}_j \\ s.t.\ & \mathbf{w}_j^T X_j X_j^T \mathbf{w}_j = 1, \quad j = 1, \cdots, J. \end{aligned} \tag{52}$$

There is no closed-form solution for $\mathbf{w}_j$. Therefore, we followed [48] and solved it using the Manopt [49] package[1].

B. MAXVAR-GCCA

GCCA [29] assumes that each view can be generated from a set of multivariate latent variables $G = [\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_N]^T \in \mathbb{R}^{N \times K}$, in which all views are correlated. Its objective function is:

$$\min_{G, \{W_j\}_{j=1}^{J}} \sum_{j=1}^{J} \left\| G - X_j^T W_j \right\|_F^2 \quad s.t.\ G^T G = I, \tag{53}$$

where $W_j \in \mathbb{R}^{d_j \times K}$ is the canonical matrix of View j, and the constraint guarantees that G is a unit orthogonal matrix.

This objective is similar to a least squares problem. In order to avoid memory problems for large datasets, following [50], [51], an SVD is first used to obtain a rank-m approximation of each view:

$$X_j \approx U_j S_j V_j^T, \quad j = 1, \cdots, J, \tag{54}$$

where $S_j \in \mathbb{R}^{m \times m}$ is a diagonal matrix composed of the m largest singular values, and $U_j \in R^{d_j \times m}$ and $V_j \in \mathbb{R}^{N \times m}$ are the corresponding left and right singular matrices, respectively.

G then consists of the K leading eigenvectors of matrix $M = \tilde{M}\tilde{M}^T$, where

$$\tilde{M} = [V_1 T_1, \cdots, V_J T_J] \in \mathbb{R}^{N \times mJ}. \tag{55}$$

It has been shown [50] that the diagonal matrix T_j satisfies $T_j T_j^T = S_j \left(S_j^T S_j^T + r_j I\right)^{-1} S_j^T$. Hence, given S_j in (54), T_j can be easily computed.

Once G is obtained, W_j can be computed as:

$$W_j = \left(X_j^T X_j + r_j I\right)^{-1} X_j^T G. \tag{56}$$

Benton *et al.* [24] also proposed view-weighted GCCA to consider the varying importance of different views.

[1] www.manopt.org

C. Least Squares based Generalized CCA (LS-CCA)

LS-CCA [31] aims to minimize the distances among the canonical variables, so that different views are maximally overlapping with each other after mapping. Its objective function is:

$$\begin{aligned} \min_{\{\mathbf{w}_j\}_{j=1}^{J}} & \frac{1}{2J(J-1)} \sum_{i=1}^{J}\sum_{j=1}^{J} \left\| X_i^T \mathbf{w}_i - X_j^T \mathbf{w}_j \right\|_2^2 \\ s.t.\ & \frac{1}{J} \sum_{j=1}^{J} \mathbf{w}_j^T X X^T \mathbf{w}_j = 1. \end{aligned} \tag{57}$$

The k-th stacked canonical vector $\mathbf{w}^{(k)} = [\mathbf{w}_1^{(k)}; \cdots; \mathbf{w}_J^{(k)}]$ can be obtained by performing a generalized eigen decomposition:

$$\frac{1}{J-1}(R - D)\mathbf{w}^{(k)} = \lambda^{(k)} D \mathbf{w}^{(k)}, \tag{58}$$

where

$$R = \begin{bmatrix} \Sigma_{11} & \cdots & \Sigma_{1J} \\ \vdots & \ddots & \vdots \\ \Sigma_{J1} & \cdots & \Sigma_{JJ} \end{bmatrix}, \quad D = \begin{bmatrix} \hat{\Sigma}_{11} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \hat{\Sigma}_{JJ} \end{bmatrix}. \tag{59}$$

Since the computational cost of this solution is high, LS-CCA can also be solved in an iterative manner, e.g., using partial least squares (PLS) [31], so that the canonical vectors are obtained directly without using low rank approximation. Via *et al.* [31] showed that LS-CCA and MAXVAR-GCCA are equivalent. Moreover, when $J = 2$, LS-CCA degrades to CCA.

D. Tensor CCA (TCCA)

Traditional CCA optimizes the pairwise view correlation only, and ignores high-order statistics. TCCA [26] directly maximizes the correlation of all views, using a high-order covariance tensor. Its objective function is:

$$\begin{aligned} \max_{\{\mathbf{w}_j\}_{j=1}^{J}} & \rho\left(\mathbf{w}_1^T X_1, \mathbf{w}_2^T X_2, \ldots, \mathbf{w}_J^T X_J\right) \\ s.t.\ & \mathbf{w}_j^T X X^T \mathbf{w}_j^T = 1,\ j = 1, \ldots, J, \end{aligned} \tag{60}$$

where ρ denotes the correlation among the J views. Luo *et al.* [26] showed that

$$\rho\left(\mathbf{w}_1^T X_1, \ldots, \mathbf{w}_J^T X_J\right) = \mathcal{C}_{1,2,\ldots,J} \times_1 \mathbf{w}_1^T \ldots \times_J \mathbf{w}_J^T, \tag{61}$$

where $\times_j$ is the j-mode product, and $\mathcal{C}_{1,2,\ldots,J}$ denotes a $d_1 \times d_2 \times \cdots \times d_J$ covariance tensor. Let $\circ$ be the tensor product. Then,

$$\mathcal{C}_{1,2,\ldots,J} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_{1i} \circ \mathbf{x}_{2i} \circ \ldots \circ \mathbf{x}_{Ji}, \tag{62}$$

where $\mathbf{x}_{ji}$ $(j = 1, \ldots, J)$ denotes the i-th instance of View j. Let $\mathbf{u}_j = \hat{\Sigma}_{jj}^{1/2} \mathbf{w}_j$, and

$$\mathcal{M} = \mathcal{C}_{1,2,\ldots,J} \times_1 \hat{\Sigma}_{11}^{-1/2} \times_2 \hat{\Sigma}_{22}^{-1/2} \ldots \times_J \hat{\Sigma}_{JJ}^{-1/2}. \tag{63}$$

Then, (61) can be rewritten as:

$$\max_{\{\mathbf{u}_j\}_{j=1}^J} \mathcal{M} \times_1 \mathbf{u}_1^T \times_2 \mathbf{u}_2^T \ldots \times_J \mathbf{u}_J^T \quad (64)$$
$$s.t.\ \mathbf{u}_j^T \mathbf{u}_j = 1, \quad j = 1, \ldots, J.$$

This problem can be solved by decomposing $\mathcal{M}$ into K best rank-1 approximations $\left[\mathbf{u}_1^k, \mathbf{u}_2^k, \cdots, \mathbf{u}_J^k\right]_{k=1}^K$ with alternating least squares, and then stacking the corresponding vectors to obtain the canonical vectors $W_j = \left[\hat{\Sigma}_{jj}^{-1/2}\mathbf{u}_j^1, \cdots, \hat{\Sigma}_{jj}^{-1/2}\mathbf{u}_j^K\right]$ of View j.

E. Deep Generalized CCA (DGCCA)

DGCCA [52], shown in Fig. 5, is a DNN-based nonlinear extension of GCCA. Each view first passes through a multi-layer perceptron neural network to obtain nonlinear features $H_j = \mathbf{f}_j(X_j)$, $j = 1, \cdots, J$. The objective of DGCCA is defined as the sum of the reconstruction errors between the canonical variables $X_j^T W_j$ and the common representation G:

$$\min_{\{\mathbf{f}_j\}_{j=1}^J} \sum_{j=1}^J \left\| G - \mathbf{f}_j(X_j)^T W_j \right\|_F^2 \quad s.t.\ G^T G = I. \quad (65)$$

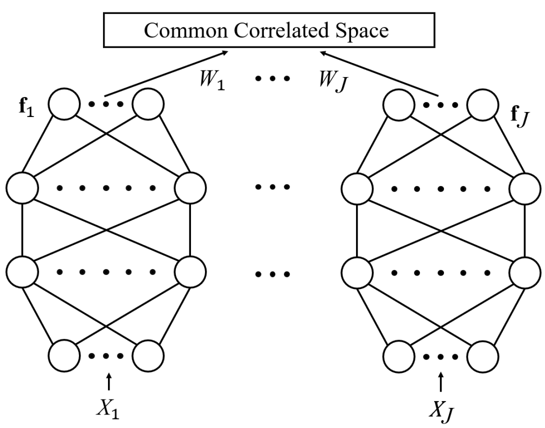


Fig. 5. A Schematic diagram of DGCCA for J views. Each view $X_j, j = 1, \cdots, J$, first passes through a multi-layer neural network $\mathbf{f}_j, j = 1, \cdots, J$, to obtain nonlinear features, and then GCCA is applied to project them onto a common correlated space.

In training, the network's outputs for a mini-batch of instances are used as the inputs to GCCA, and then the reconstruction error between each view and the common representation is calculated to update the corresponding network parameters. In testing, the common representation G or the stacked canonical variables $\left[X_1^T W_1, \cdots, X_J^T W_J\right]$ can be used as the input to a classifier.

DGCCA is unsupervised. To utilize the label information, Benton *et al.* [52] added the labels' one-hot coding matrix as an additional view.

F. Summary

This section has introduced five representative CCA approaches for more than two views. To our knowledge, TCCA is currently the only approach to directly calculate the correlation among more than two views. All other approaches perform the computation indirectly.

IV. Applications

As a well-known approach for analyzing correlations between two (or more) sets of variables, CCA and its extensions have been widely used in many applications. This section briefly reviews some representative ones.

A. Multi-View Pattern Recognition

CCA-based multi-view pattern recognition (classification and regression) can be supervised or unsupervised.

Generally, CCA is used to learn new feature representations (called canonical variables) for each view. There are two representative approaches for using these canonical variables. For example, in classification, one can feed the canonical variables of one view into a classifier [21], [22], [46], or concatenate the canonical variables from different views and feed them into a classifier [26], [53], [54]. The former is applicable in two scenarios: 1) both views are available in training, but only one view is available in testing; and, 2) the label is treated as the second view, and the classification of a test instance is computed from the nearest neighbors in the first projected view.

One popular application of CCA-based multi-view pattern recognition is computer vision. For example, the Daubechies wavelet transformed low-frequency image and the original image were treated as two different views for face recognition [22], [32], the facial image and a psychologist's semantic rating were used as two views in facial expression recognition [55], and gait and face modalities were used as two views in human gender recognition [54]. CCA has also been used to construct the connections between an image and the corresponding 3D pose parameters in pose recognition [20], and among different viewing perspectives in gait recognition [56].

Additionally, CCA has also found successful applications in phonetic recognition [44], [46], [57], handwritten digit recognition [22], [25], [32], [53], biometric structure prediction [26], text categorization [22], advertisement classification [26], brain-computer interfaces [58], [59], epileptic seizure detection [60], and so on.

B. Cross-Modal Retrieval and Classification

Due to the rapid growth of multimedia data on the Internet, cross-media retrieval has become a hot research topic in multimedia information retrieval and computer vision. Its main challenge is the "heterogeneity gap", i.e., data from different media types are inconsistent, which makes it difficult to measure the cross-media similarity of instances [61]. CCA can be a solution to this problem.

For example, a document may contain text and/or images, and our goal is to retrieve relevant texts according to a query image. One classical solution is to find a common representation for different views using CCA [62]–[64]. First, treating the text and images as two different views, k-dimensional canonical variables of each document are obtained using CCA. Then, canonical variables from different views are considered as coordinates in a common correlated space, as shown in Fig. 1. Given a query image, after mapping it onto the common correlated space with canonical vectors, the closest matches can be obtained by computing its nearest neighbors using the text canonical variables.

Cross-modal classification [65] has the same data distribution assumption as cross-modal retrieval, i.e., the paired canonical variables of two different views are close to each other in the common correlated space. Therefore, a classifier trained from the first view should also perform well in the second view.

C. Multi-View Embedding

Another important application of CCA is word embedding in natural language processing [66]–[68]. Faruqui and Dyer [68] proposed to learn word embeddings by considering the multilingual context using DCCA, and demonstrated that it gave better semantic representations in standard lexical semantic evaluation tasks. Benton *et al.* [24], [52] used GCCA to learn multi-view embeddings of social media users for friend recommendation.

Generally, multi-view embedding has two benefits. First, it can capture information from multiple views to obtain a representation, which usually has a better similarity measure. Second, it can learn a low-dimensional representation from high-dimensional multi-view data, which reduces the complexity of a subsequent pattern recognition algorithm.

V. Conclusions

CCA is widely used in MVL. However, the traditional CCA has some limitations: 1) it can only handle two views; 2) it can only optimize the linear correlation between different views; 2) it is unsupervised, and hence cannot make use of the label information in supervised learning. Many nonlinear, supervised, or generalized CCA extensions have been proposed for remedy. This paper reviews many representative CCA approaches, and describes their typical applications in patten recognition, cross-modal retrieval and classification, and multi-view embedding. To our knowledge, this is the most up-to-date comprehensive overview on CCA.

References

[1] J. Tang, Y. Tian, P. Zhang, and X. Liu, "Multiview privileged support vector machines," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3463–3477, 2018.

[2] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *CoRR*, vol. abs/1304.5634, 2013. [Online]. Available: http://arxiv.org/abs/1304.5634

[3] X. Xue, F. Nie, S. Wang, X. Chang, B. Stantic, and M. Yao, "Multi-view correlated feature learning by uncovering shared component," in *Proc. 31th AAAI Conf. on Artificial Intelligence*, San Francisco, CA, Feb. 2017, pp. 2810–2816.

[4] T. Baltrusaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 423–443, 2019.

[5] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. 12th Annual Conf. on Computational Learning Theory*, NY, Jul. 1998, pp. 92–100.

[6] S. Abney, "Bootstrapping," in *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA, Jul. 2002, pp. 360–367.

[7] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *Proc. 12th IEEE Int'l Conf. on Computer Vision*, Kyoto, Japan, Sep. 2009, pp. 221–228.

[8] M. Gonen and E. Alpaydn, "Multiple kernel learning algorithms," *Journal of Machine Learning Research*, vol. 12, pp. 2211–2268, 2011.

[9] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[10] T. Diethe, D. R. Hardoon, and J. Shawe-Taylor, "Multiview Fisher discriminant analysis," in *Proc. Neural Information Processing Systems Workshop on Learning from Multiple Sources*, Whistler, BC, Canada, Dec. 2008.

[11] T. Xia, D. Tao, T. Mei, and Y. Zhang, "Multiview spectral embedding," *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, vol. 40, no. 6, pp. 1438–1446, 2010.

[12] J. Zhao, X. Xie, X. Xu, and S. Sun, "Multi-view learning overview: Recent progress and new challenges," *Information Fusion*, vol. 38, pp. 43–54, 2017.

[13] S. Sun, "A survey of multi-view machine learning," *Neural Computing and Applications*, vol. 23, no. 7, pp. 2031–2038, 2013.

[14] T. Zhang, Z. Deng, D. Wu, and S. Wang, "Multi-view fuzzy logic system with the cooperation between visible and hidden views," *IEEE Trans. on Fuzzy Systems*, vol. 27, no. 6, pp. 1162–1173, 2019.

[15] D. P. Foster, R. Johnson, S. M. Kakade, and T. Zhang, "Multi-view dimensionality reduction via canonical correlation analysis," Toyota Technical Institute, Chicago, IL, Tech. Rep. TR-2008-4, 2008.

[16] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan, "Multi-view clustering via canonical correlation analysis," in *Proc. 26th Annual Int'l Conf. on Machine Learning*, Montreal, Quebec, Canada, Jun. 2009, pp. 129–136.

[17] M. B. Blaschko and C. H. Lampert, "Correlational spectral clustering," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Anchorage, Alaska, Jun. 2008.

[18] S. M. Kakade and D. P. Foster, "Multi-view regression via canonical correlation analysis," in *Proc. 20th Annual Conf. on Learning Theory*, San Diego, CA, Jun. 2007.

[19] S. Akaho, "A kernel method for canonical correlation analysis," *CoRR*, vol. abs/cs/0609071, 2006. [Online]. Available: http://arxiv.org/abs/cs/0609071

[20] T. Sun and S. Chen, "Locality preserving CCA with applications to data visualization and pose estimation," *Image and Vision Computing*, vol. 25, no. 5, pp. 531–543, 2007.

[21] W. Wang, R. Arora, K. Livescu, and J. Bilmes, "On deep multi-view representation learning," in *Proc. 32th Int'l Conf. on Machine Learning*, Lille, France, Jul. 2015, pp. 1083–1092.

[22] T. Sun, S. Chen, J. Yang, and P. Shi, "A novel method of combined feature extraction for recognition," in *Proc. 8th IEEE Int'l Conf. on Data Mining*, Washington, DC, Dec. 2008, pp. 1043–1048.

[23] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *Proc. 30th Int'l Conf. on Machine Learning*, Atlanta, GA, Jun. 2013, pp. 1247–1255.

[24] A. Benton, R. Arora, and M. Dredze, "Learning multiview embeddings of twitter users," in *Proc. 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, Germany, Aug. 2016, pp. 14–19.

[25] N. E. D. Elmadany, Y. He, and L. Guan, "Multiview learning via deep discriminative canonical correlation analysis," in *Proc. IEEE Int'l Conf. on Acoustics, Speech and Signal Processing*, Shanghai, China, Mar. 2016, pp. 2409–2413.

[26] Y. Luo, D. Tao, K. Ramamohanarao, C. Xu, and Y. Wen, "Tensor canonical correlation analysis for multi-view dimension reduction," *IEEE Trans. on Knowledge and Data Engineering*, vol. 27, no. 11, pp. 3111–3124, 2015.

[27] A. A. Nielsen, "Multiset canonical correlations analysis and multispectral, truly multitemporal remote sensing data," *IEEE Trans. on Image Processing*, vol. 11, no. 3, pp. 293–305, 2002.

[28] P. Horst, "Generalized canonical correlations and their applications to experimental data," *Journal of Clinical Psychology*, vol. 17, no. 4, pp. 331–347, 1961.

[29] J. D. Carroll, "Generalization of canonical correlation analysis to three or more sets of variables," in *Proc. 76th Annual Convention of the American Psychological Association*, Washington, DC, Sep. 1968, pp. 227–228.

[30] J. R. Kettenring, "Canonical analysis of several sets of variables," *Biometrika*, vol. 58, no. 3, p. 433451, 1971.

[31] J. Va, I. Santamara, and J. Prez, "A learning algorithm for adaptive canonical correlation analysis of several data sets," *Neural Networks*, vol. 20, no. 1, pp. 139–152, 2007.

[32] S. Sun, X. Xie, and M. Yang, "Multiview uncorrelated discriminant analysis," *IEEE Trans. on Cybernetics*, vol. 46, no. 12, pp. 3272–3284, 2016.

[33] R. A. Fisher, "The use of multiple measurements in taxonomicproblems," *Annals of Eugenics*, vol. 7, no. 2, p. 179188, 1936.

[34] 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, 2004.

[35] P. J. Bickel and E. Levina, “Regularized estimation of large covariance matrices,” *The Annals of Statistics*, vol. 36, no. 1, pp. 199–227, 2008.

[36] T. D. Bie and B. D. Moor, “On the regularization of canonical correlation analysis,” in *Proc. 4th Int'l Conf. on Independent Component Analysis and Blind Source Separation*, Nara, Japan, Apr. 2003, pp. 785–790.

[37] D. I. Warton, “Penalized normal likelihood and ridge regularization of correlation and covariance matrices,” *Journal of the American Statistical Association*, vol. 103, no. 481, pp. 340–349, 2008.

[38] D. M. Witten, R. Tibshirani, and T. Hastie, “A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis,” *Biostatistics*, vol. 10, no. 3, pp. 515–534, 2009.

[39] D. M. Witten, R. Tibshirani, and T. Hastie, “Extensions of sparse canonical correlation analysis with applications to genomic data,” *Statistical Applications in Genetics and Molecular Biology*, vol. 8, no. 1, pp. 1–27, 2009.

[40] C. Eckart and G. Young, “The approximation of one matrix by another of lower rank,” *Psychometrika*, vol. 1, no. 3, pp. 211–218, 1936.

[41] D. Lopez-Paz, S. Sra, A. Smola, Z. Ghahramani, and B. Schoelkopf, “Randomized nonlinear component analysis,” in *Proc. 31st Int'l Conf. on Machine Learning*, Bejing, China, Jun. 2014, pp. 1359–1367.

[42] L. Hoegaerts, J. A. Suykens, J. Vandewalle, and B. D. Moor, “Subset based least squares subspace regression in RKHS,” *Neurocomputing*, vol. 63, pp. 293–323, 2005.

[43] Z. Jin, J.-Y. Yang, Z.-S. Hu, and Z. Lou, “Face recognition based on the uncorrelated discriminant transformation,” *Attern Recognition*, vol. 34, no. 7, pp. 1405–1416, 2001.

[44] W. Wang, R. Arora, K. Livescu, and J. A. Bilmes, “Unsupervised learning of acoustic features via deep canonical correlation analysis,” in *Proc. 40th IEEE Int'l Conf. on Acoustics, Speech and Signal Processing*, Brisbane, Australia, Apr. 2015, pp. 4590–4594.

[45] F. R. Bach and M. I. Jordan, “A probabilistic interpretation of canonical correlation analysis,” Department of Statistics, University of California, Berkeley, CA, Tech. Rep. 688, 2005.

[46] W. Wang, H. Lee, and K. Livescu, “Deep variational canonical correlation analysis,” *CoRR*, vol. abs/1610.03454, 2016. [Online]. Available: http://arxiv.org/abs/1610.03454

[47] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in *Proc. 2nd Int'l Conf. on Learning Representations*, Banff, Canada, Apr. 2014.

[48] N. A. Asendorf, “Informative data fusion: Beyond canonical correlation analysis,” Ph.D. dissertation, The University of Michigan, Ann Arbor, MI, 2015.

[49] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, “Manopt, a matlab toolbox for optimization on manifolds,” *Journal of Machine Learning Research*, vol. 15, pp. 1455–1459, 2014.

[50] P. Rastogi, B. V. Durme, and R. Arora, “Multiview LSA: Representation learning via generalized CCA,” in *Proc. Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Denver, CO, May 2015, pp. 556–566.

[51] R. Arora and K. Livescu, “Kernel CCA for multi-view learning of acoustic features using articulatory measurements,” in *Proc. Symp. on Machine Learning in Speech and Language Processing*, Portland, Oregon, Sep. 2012.

[52] A. Benton, H. Khayrallah, B. Gujral, D. Reisinger, S. Zhang, and R. Arora, “Deep generalized canonical correlation analysis,” *CoRR*, vol. abs/1702.02519, 2017. [Online]. Available: http://arxiv.org/abs/1702.02519

[53] Y. J. Shin and C. H. Park, “Analysis of correlation based dimension reduction methods,” *Int'l Journal of Applied Mathematics and Computer Science*, vol. 21, no. 3, pp. 549–558, 2011.

[54] C. Shan, S. Gong, and P. McOwan, “Fusing gait and face cues for human gender recognition,” *Neurocomputing*, vol. 71, no. 10, pp. 1931–1938, 2008.

[55] W. Zheng, X. Zhou, C. Zou, and L. Zhao, “Facial expression recognition using kernel canonical correlation analysis (KCCA),” *IEEE Trans. on Neural Networks*, vol. 17, no. 1, pp. 233–238, 2006.

[56] X. Xing, K. Wang, T. Yan, and Z. Lv, “Complete canonical correlation analysis with application to multi-view gait recognition,” *Pattern Recognition*, vol. 50, pp. 107–117, 2016.

[57] R. Arora and K. Livescu, “Multi-view CCA-based acoustic features for phonetic recognition across speakers and domains,” in *IEEE Int'l Conf. on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, May 2013, pp. 7135–7139.

[58] Z. Lin, C. Zhang, W. Wu, and X. Gao, “Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs,” *IEEE Trans. on Biomedical Engineering*, vol. 53, no. 12, pp. 2610–2614, 2006.

[59] M. Nakanishi, Y. Wang, Y.-T. Wang, and T.-P. Jung, “A comparison study of canonical correlation analysis based methods for detecting steady-state visual evoked potentials,” *PLOS ONE*, vol. 10, no. 10, pp. 1–18, 2015.

[60] X. Tian, Z. Deng, K.-S. Choi, D. Wu, B. Qin, J. Wan, H. Shen, and S. Wang, “Deep multi-view feature learning for epileptic seizure detection,” *EEE Trans. on Neural Systems and Rehabilitation Engineering*, 2019, submitted.

[61] J. Chi, X. Huang, and Y. Peng, “Zero-shot cross-media retrieval with external knowledge,” in *Proc. 9th Int'l Conf. on Internet Multimedia Computing and Service*, Qingdao, China, Aug. 2017, pp. 200–211.

[62] N. Rasiwasia, J. C. Pereira, E. Coviello, G. Doyle, G. R. Lanckriet, R. Levy, and N. Vasconcelos, “A new approach to cross-modal multimedia retrieval,” in *Proc. 18th ACM Int'l Conf. on Multimedia*, Firenze, Italy, Oct. 2010, pp. 251–260.

[63] Y. Peng, X. Huang, and Y. Zhao, “An overview of cross-media retrieval: Concepts, methodologies, benchmarks, and challenges,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 28, no. 9, pp. 2372–2385, 2017.

[64] F. Wu, H. Zhang, and Y. Zhuang, “Learning semantic correlations for cross-media retrieval,” in *Proc. Int'l Conf. on Image Processing*, Atlanta, GA, Sep. 2006, pp. 1465–1468.

[65] S. Chandar, M. M. Khapra, H. Larochelle, and B. Ravindran, “Correlational neural networks,” *Neural Computation*, vol. 28, no. 2, pp. 257–285, 2016.

[66] P. Dhillon, D. Foster, and L. Ungar, “Multi-view learning of word embeddings via CCA,” in *Proc. in Neural Information Processing Systems*, Granada, Spain, Dec. 2011, pp. 199–207.

[67] A. Lu, W. Wang, M. Bansal, K. Gimpel, and K. Livescu, “Deep multilingual correlation for improved word embeddings,” in *Proc. Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Denver, CO, May 2015, pp. 250–256.

[68] M. Faruqui and C. Dyer, “Improving vector space word representations using multilingual correlation,” in *Proc. 14th Conf. of the European Chapter of the Association for Computational Linguistics*, Gothenburg, Sweden, Apr. 2014, pp. 462–471.