This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Discriminative Deep Canonical Correlation Analysis for Multi-View Data

Debamita Kumar and Pradipta Maji, *Senior Member, IEEE*

***Abstract*— Over the past few years, multimodal data analysis has emerged as an inevitable method for identifying sample categories. In the multi-view data classification problem, it is expected that the joint representation should include the supervised information of sample categories so that the similarity in the latent space implies the similarity in the corresponding concepts. Since each view has different statistical properties, the joint representation should be able to encapsulate the underlying nonlinear data distribution of the given observations. Another important aspect is the coherent knowledge of the multiple views. It is required that the learning objective of the multi-view model efficiently captures the nonlinear correlated structures across different modalities. In this context, this article introduces a novel architecture, termed discriminative deep canonical correlation analysis (D2CCA), for classifying given observations into multiple categories. The learning objective of the proposed architecture includes the merits of generative models to identify the underlying probability distribution of the given observations. In order to improve the discriminative ability of the proposed architecture, the supervised information is incorporated into the learning objective of the proposed model. It also enables the architecture to serve as both a feature extractor as well as a classifier. The theory of CCA is integrated with the objective function so that the joint representation of the multi-view data is learned from maximally correlated subspaces. The proposed framework is consolidated with corresponding convergence analysis. The efficacy of the proposed architecture is studied on different domains of applications, namely, object recognition, document classification, multilingual categorization, face recognition, and cancer subtype identification with reference to several state-of-the-art methods.**

***Index Terms*— Boltzmann machine, canonical correlation analysis (CCA), deep learning, multimodal data.**

I. Introduction

ADVANCEMENT in information acquisition processes has entailed the development of predictive models for multimodal data analysis. The classification based on multi-view data has been exercised in numerous domains of applications, for example, object detection [1], tumor analysis [2], face recognition [3], 3-D saliency detection [4], and so on. Since each view has a fundamentally distinct representation of the underlying data distribution, it is primarily considered that information from different sources encompasses complementary as well as coherent knowledge, corresponding to the given observations. Thus, judicious integration of information from multiple views can potentially provide a more comprehensive and discriminative representation of the data, as compared to each of the unimodal representations.

A naive solution concatenates all the views to obtain a single data matrix which can then be applied to single view learning algorithms. However, the direct integration approaches suffer from the overfitting problem, which is predominant in the case of small training data sets. Also, it becomes difficult to reflect the individual statistical properties of each of the modalities in the unified representation for heterogeneous databases. In this regard, several approaches have been adopted in the existing literature to learn the joint representation of the data from the given multiple modalities. Various multi-view learning algorithms, based on correlation analysis [5], [6], [7], [8], discriminant analysis [9], [10], [11], and deep learning [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], have been developed that learn the necessary functions to model each of the views and then jointly optimize all the functions to enhance the generalization ability of the corresponding approaches.

A. Related Work

Considering the baseline framework of canonical correlation analysis (CCA) [22], several attempts have been made to characterize the inherent correlation across multiple views. While multiset CCA (MCCA) [6] is an extension of the classical two-set theory of CCA to several sets, regularized generalized CCA (RGCCA) [5] is a generalization of the regularized CCA that uses ridge regression optimization to prevent the problem of over-fitting. Chen et al. [7] have proposed graph-regularized MCCA (GMCCA) and graph-regularized kernel MCCA (GMKCCA) approaches that minimize the distance between the canonical variables and the common low-dimensional representation, based on the graph-induced knowledge of the common sources. Multi-view uncorrelated locality preserving projection (MULPP) [23] considers pairwise correlation and distance between the input views to obtain a low-dimensional projection of the given data. In [8], large-scale generalized CCA (LasCCA) has been proposed to handle large sparse views. A distributed algorithm for generalized CCA (DisCCA) has also been developed in [8] to reduce the run time of the algorithm. Since the CCA-based methods are, in general, unsupervised in nature, the projected subspaces lack discriminative information.

Typical discriminant analysis-based methods include multi-view discriminant analysis (MvDA) [9], which seeks a discriminant common space by jointly learning multiple view-specific linear transforms. Multi-view discriminant analysis with view-consistency (MvDA-VC) [11] maximizes the between-class variations and minimizes the within-class

Manuscript received 8 June 2021; revised 28 February 2022, 26 September 2022, and 6 February 2023; accepted 15 May 2023. *(Corresponding author: Pradipta Maji.)*

The authors are with the Biomedical Imaging and Bioinformatics Laboratory, Machine Intelligence Unit, Indian Statistical Institute, Kolkata 700108, India (e-mail: debamita_r@isical.ac.in; pmaji@isical.ac.in).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TNNLS.2023.3277633.

Digital Object Identifier 10.1109/TNNLS.2023.3277633

2162-237X © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

variations by enforcing the view-consistency constraint. You et al. [10] have proposed multi-view common component discriminant analysis (MvCCDA) to learn a discriminant common subspace by incorporating supervised information and local geometric information into the common component extraction process. Although the methods have the benefits of view discrepancy and discriminability, the common subspace is learned using linear transformations in most of the cases.

In recent years, there has been a surging interest in combining information from multiple modalities using deep learning based models. In deep multiset CCA (dMCCA) [12], feedforward networks have been used to map the given input modalities to a shared subspace. Couture et al. [13] have developed task optimal CCA (TOCCA) that focuses on both CCA and task-driven objectives using a deep architecture, whereas Srivastava et al. [14] have proposed a multimodal deep Boltzmann machine (MDBM) to learn a joint density model over the space of multimodal inputs. Fan et al. [15] have proposed a deep adversarial CCA (DACCA) model, which integrates adversarial learning techniques with the concept of CCA. The deep CCA with view generation (DCCA-VG) [16] focuses on learning multi-view representation for hyperspectral image classification by fusing spatial and spectral information. Dorfer and Widmer [17] have proposed deep and discriminative CCA (TDDCCA) that incorporates a discriminative regularizer into the objective of existing deep CCA to jointly avail the advantages of correlation and discriminability. Tensor CCA (TCCA) [18] discovers the higher order correlation by maximizing canonical correlation among all the views and seeks the optimal filter banks by analyzing a covariance tensor. Rastegar et al. [19] have proposed a multimodal deep learning framework, termed MDL-CW, which learns the cross-weights between representations of different modalities through a deep network. In multimodal graph neural network (MMGNN) [20], an image is represented as a graph, and then three aggregators are introduced to refine the nodes of the network. In [21], multi-view generative adversarial network (MVGAN) have been developed to automatically expand the labeled multi-view samples, and the expanded dataset is then used to train the multistream convolutional neural network.

Both TOCCA [13] and MDBM [14] approaches serve as feature extractors and need to employ an additional classifier for classification purposes. Though dMCCA [12], TOCCA, and TDDCCA [17] concentrate on learning the correlated subspace from the input multi-view data, they eventually disregard the underlying data distribution. Since the DACCA [15] model is susceptible to slight variation in the input characteristics, the performance of the model is highly dependent on the input signal-to-noise ratio. Moreover, MDBM [14], DCCA-VG [16], TCCA [18], and MDL-CW [19] models are unsupervised in nature, and hence, the corresponding joint representations fail to capture the discriminative information of the given observations. While MMGNN [20] and MVGAN [21] frameworks concentrate on view-specific information, they do not take into consideration the shared knowledge of different views.

B. Motivation

In the existing literature, several deep models have been considered to learn the joint representation of the given data from the input modalities. The dMCCA model [12] is developed based on the framework of a feedforward network, whereas stacked autoencoder has been considered for MDL-CW [19]. While graph neural network is used for the development of MMGNN [20], the generative adversarial network has been considered for the implementation of MVGAN [21]. However, the backpropagation learning algorithm of feed-forward networks considers only the training classification error to iteratively adjust the parameters of the model. Hence, the performance of the network largely depends on the training set of the given data. One of the difficulties of stacked autoencoders lies in the fact that if an error is present in the first layer of the network, it propagates through the final layer and causes the network to reconstruct the average of the training data. Since the majority of graph-based deep networks assume homogeneous graphs, it is difficult to directly apply the corresponding algorithm to heterogeneous graphs that contain different forms of node and edge inputs. The effectiveness of adversarial learning has a strong correlation with the distance between a test point and the manifold of training data embedded in the network. Consequently, adversarial networks are more likely to be vulnerable to blind-spot attacks.

On the other hand, DBM [24] is an effective paradigm of undirected generative models that efficiently captures the nonlinear dependencies between observed and latent variables by analyzing the energy landscape of the given observations. Unlike the feedforward counterparts, the learning objective of DBMs is to adjust the weights of the network such that the probability of observing the training data is maximized. One of the advantages of DBMs is the stochastic approximation procedure, which, apart from the usual bottom-up passes, includes top-down feedback to incorporate uncertainty associated with the given input data. Also, the problem of slow and intractable learning of contrastive divergence is alleviated by considering the variational learning approach, proposed by Salakhutdinov and Hinton [24]. Hence, the joint representation in the DBM based multi-view model is expected to encapsulate the underlying nonlinear data distribution of the given observations. However, the architecture of DBM is essentially unsupervised in nature. In multimodal learning, the joint representation should contain the discriminative information so that the similarity in the latent space implies the similarity in the corresponding concepts. It is also required that the learning objective of the multi-view model efficiently captures the correlated structures across different modalities.

C. Contribution of Current Study

In this regard, a novel architecture, termed as discriminative deep CCA (D2CCA), is proposed. In order to incorporate the cross-modal information, the theory of CCA is introduced to the learning objective of the proposed framework. The weights of the network are updated such that the individual latent spaces are transformed into maximally correlated subspaces. Hence, the joint representation, learned from the obtained subspaces, can efficiently capture the nonlinear correlated structures across different modalities. The class nodes are incorporated into the proposed deep architecture to include the supervised information at each layer of the network. Proper learning of the weights associated with the class nodes ensures

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

that the obtained representations will have better discriminative abilities as compared to the unsupervised counterparts. Also, considering the class nodes in the architecture allows the proposed model to predict the class labels of given observations without employing any additional classifier for classification purpose. Furthermore, the proposed framework is consolidated with corresponding convergence analysis. The proficiency of the D2CCA architecture is extensively studied and compared with several state-of-the-art methods on seven benchmark and five real-life cancer datasets considering both training–testing and tenfold cross-validation (CV).

II. Discriminative Deep CCA

In this section, a novel architecture, termed as D2CCA, is presented for multi-view data analysis. It judiciously integrates the theory of CCA and the merits of a new model, called multimodal discriminative DBM (MDDBM), to classify the given observations into different sample categories. Prior to explaining the proposed D2CCA model, the learning objective of DBM and the new architecture of MDDBM are described next.

A. Learning Objective of Deep Boltzmann Machine

Given the input view $\mathbf{v}$ and the class labels $\mathbf{y}$, learning of the DBM [24] corresponds to identifying the model parameter set $\boldsymbol{\theta}$ that maximizes the probability of observing the given observations. So, the objective function is given by the log-likelihood function as follows:

$$\begin{aligned}\ln \mathrm{L}(\boldsymbol{\theta}|\mathbf{v},\mathbf{y}) &= \ln P(\mathbf{v},\mathbf{y}|\boldsymbol{\theta}) \\ &= \ln \sum_{\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h},\mathbf{y})} - \ln \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} e^{-E(\mathbf{v},\mathbf{h},\mathbf{y})} \qquad (1)\end{aligned}$$

where $\mathbf{h}$ denotes the stack of hidden layers, $P(\mathbf{v},\mathbf{y}|\boldsymbol{\theta})$ represents the probability assigned to the observation $(\mathbf{v},\mathbf{y})$ by the model parameter set $\boldsymbol{\theta}$, and $E(\mathbf{v},\mathbf{h},\mathbf{y})$ signifies the energy of the joint configuration $\{\mathbf{v},\mathbf{h},\mathbf{y}\}$.

Since the parameter space of the model is quite large, obtaining the parameters that maximize (1) is computationally very intensive. The gradient ascent on the log-likelihood is most commonly used to determine the optimal parameters, which iteratively updates the parameters by an amount $\Delta\boldsymbol{\theta}^t$ based on the gradient of the log-likelihood. So, the update rule for the parameters is given by

$$\begin{aligned}\frac{\partial \ln \mathrm{L}(\boldsymbol{\theta}|\mathbf{v},\mathbf{y})}{\partial \boldsymbol{\theta}} = &-\sum_{\mathbf{h}} P(\mathbf{h}|\mathbf{v},\mathbf{y}) \frac{\partial E(\mathbf{v},\mathbf{h},\mathbf{y})}{\partial \boldsymbol{\theta}} \\ &+ \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} P(\mathbf{v},\mathbf{h},\mathbf{y}) \frac{\partial E(\mathbf{v},\mathbf{h},\mathbf{y})}{\partial \boldsymbol{\theta}}. \qquad (2)\end{aligned}$$

So, the gradient of the log-likelihood function reduces to the difference between the expectation of the gradient of energy function under model distribution, which is termed data-independent expectation, and under the conditional distribution of hidden representation given the input views and class label information, referred to as data-dependent expectation. Since exact maximum likelihood learning is intractable, variational learning is employed to estimate the data-dependent expectation, whereas data-independent expectation is approximated by the stochastic approximation procedure.

In variational inference [25], the posterior distribution $P(\mathbf{h}|\mathbf{v},\mathbf{y})$ is approximated with a tractable mean field distribution $Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \approx P(\mathbf{h}|\mathbf{v},\mathbf{y})$. Now

$$\ln P(\mathbf{v},\mathbf{y}) = \ln \sum_{\mathbf{h}} P(\mathbf{v},\mathbf{h},\mathbf{y}) = \ln \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \frac{P(\mathbf{v},\mathbf{h},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})}.$$

Since logarithmic is a concave function, applying Jensen's inequality [26], we get

$$\ln P(\mathbf{v},\mathbf{y}) \geq \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \ln \frac{P(\mathbf{v},\mathbf{h},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})} = \mathcal{L}_v. \qquad (3)$$

Thus, the mean field approximation provides a lower bound $\mathcal{L}_v$ on the log-likelihood function. The difference between the true posterior and the variational lower bound, obtained using mean field theory, is given by

$$\begin{aligned}\ln P(\mathbf{v},\mathbf{y}) - \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \left\{ \ln P(\mathbf{v},\mathbf{y}) + \ln \frac{P(\mathbf{h}|\mathbf{v},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})} \right\} \\ = \mathrm{KL}(Q(\mathbf{h}|\mathbf{v},\mathbf{y}) || P(\mathbf{h}|\mathbf{v},\mathbf{y})) \qquad (4)\end{aligned}$$

where $\mathrm{KL}(Q(\mathbf{h}|\mathbf{v},\mathbf{y})||P(\mathbf{h}|\mathbf{v},\mathbf{y}))$ is the Kullback–Leibler divergence between the two distributions P and Q. So, better approximation of $P(\mathbf{h}|\mathbf{v},\mathbf{y})$ implies tighter bound on $\ln P(\mathbf{v},\mathbf{y})$.

B. Multimodal Discriminative Deep Boltzmann Machine

This section introduces a new architecture, termed as MDDBM, for multimodal data classification. It incorporates the supervised information of class labels or sample categories into the architecture of DBMs to improve the discriminative ability of the architecture. In [14], the multimodal DBM (MDBM) is reported, which is an unsupervised model employed for feature extraction. So, the extracted features do not contain any class label information. However, it is expected that if the hidden layers of a network are guided by the supervised information of class labels, the obtained joint representation will have better discriminative ability. Also, incorporating class nodes in the architecture enables the model to predict the class label of given samples, without employing any additional classifier for classification.

Let us assume that the proposed model has M input views or modalities, where $\mathbf{v}^m = \{v_1^m, \ldots, v_i^m, \ldots\}$ represents the input view corresponding to the mth modality and $\mathbf{y} = \{y_1, \ldots, y_p, \ldots\}$ provides the class label information. Let us also assume that the model contains $L > 1$ hidden layers, out of which $L_0 > 0$ layers are modality-specific, while the rest of the $(L - L_0)$ layers are joint. The l-th layer of the m-th modality-specific hidden layer is represented by $\mathbf{h}^{lm} = \{h_1^{lm}, \ldots, h_j^{lm}, \ldots\}$, whereas the joint hidden representation, corresponding to the l-th layer, is referred to as $\mathbf{h}^l = \{h_1^l, \ldots, h_j^l, \ldots\}$. Here, the number of nodes in a representation is expressed by the corresponding capital letter. For example, the number of nodes in $\mathbf{v}^m$ is denoted by V^m. The energy E_s of the proposed model is defined in (5), while the corresponding architecture is depicted in Fig. 1.

Here, the bidirectional weight parameters w_{ij}^{1m}, $w_{jk}^{(l+1)m}$, $w_{jk}^{(L_0+1)m}$, and $w_{jk}^{(l+1)}$ connect the i-th visible node of the m-th modality to the j-th hidden node of first modality-specific hidden layer from the m-th modality, j-th hidden node of

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

l-th modality-specific hidden layer to k-th hidden node of $(l+1)$-th modality-specific hidden layer from m-th modality, j-th hidden node of modality-specific hidden layer L_0 from modality m to k-th hidden node of first joint hidden layer, and j-th hidden node of l-th joint hidden layer to k-th hidden node of $(l+1)$-th joint hidden layer, respectively. Similarly, the parameters u_{pj}^{lm} and u_{pj}^{l} connect p-th class node to j-th hidden node of l-th modality-specific hidden layer from m-th modality, and p-th class node to j-th hidden node of l-th joint hidden layer, respectively. The bias parameters a_i^m, b_j^{lm}, b_j^l, and d_p are associated with i-th visible node of m-th modality, j-th hidden node of l-th modality-specific hidden layer from m-th modality, j-th hidden node of l-th joint hidden layer, and p-th class node, respectively.

Considering the energy function E_s of (5), the parameter space of the model is defined by

$$\boldsymbol{\theta}_s = \{w^{1m}, \ldots, w^{(L_0+1)m}, w^{(L_0+2)}, \ldots, w^{L}, u^{1m}, \ldots u^{L_0m}, u^{(L_0+1)}, \ldots, u^{L}, a^m, b^{1m}, \ldots, b^{L_0m}, b^{(L_0+1)} \ldots, b^{L}, d\} \quad \forall m \in \{1, 2, \ldots, M\}.$$

Thus, through proper learning of the set of parameters $\{u^{1m}, \ldots, u^{L_0m}, u^{(L_0+1)}, \ldots, u^{L}, d\}\forall m$, the discriminatory information can be efficiently incorporated in the hidden representations of the model at each layer, which in turn, improves the proficiency of the model as feature extractor and allows the model to serve as classifier as well.

C. Integration of Multimodal Discriminative DBM and CCA

In MDDBM, the joint representation (say $\mathbf{h}^3$) is learned from the individual modality-specific representations (say $\mathbf{h}^{21}$ and $\mathbf{h}^{22}$) and class label information ($\mathbf{y}$). However, it may so happen that the individual views correspond to two very different sources. For example, one view may correspond to an image, whereas the other view refers to text modality. In such a scenario, the individual hidden representations correspond to completely different spaces. So, learning the joint representation from the two spaces may not be able to capture the cross-modal information. However, if the model is learned in such a way that $\mathbf{h}^{21}$ and $\mathbf{h}^{22}$ are highly correlated, then the inherent characteristics of the views can be efficiently modeled by the joint representation. So, given the input views, the objective of the proposed framework is to update the parameters of the model in such a way that the joint representation is learned from maximally correlated subspaces.

The CCA [22] is an effective statistical method for integrating information acquired from different views. It measures

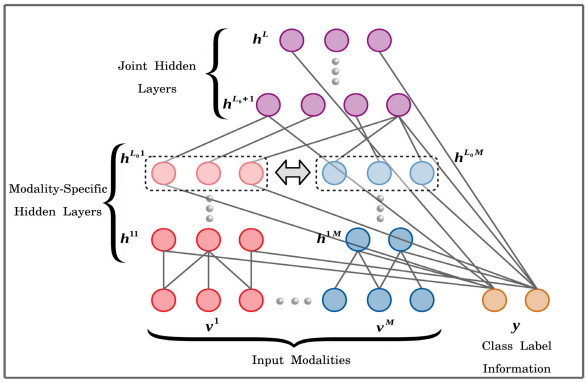

Fig. 1. Illustration of proposed multimodal discriminative deep framework.

the linear relationship between two multidimensional variables and finds the best linear transformation to achieve the maximum correlation between them. The objective of CCA is to extract latent features from two data sets $X_1 \in \Re^{p\times n}$ and $X_2 \in \Re^{q\times n}$, which are highly correlated. Each column of X_1 and X_2 corresponds to one of the n samples, and each row represents one variable. The CCA obtains two directional weight vectors, also termed as basis vectors, $\omega_1 \in \Re^p$ and $\omega_2 \in \Re^q$, corresponding to two mean-centered data matrices X_1 and X_2, respectively, such that the correlation between the respective projections onto these weight vectors, that is, between ${X_1}^T\omega_1$ and ${X_2}^T\omega_2$ is maximum. So,

$$(\omega_1, \omega_2) = \underset{\|{X_1}^T\omega_1\|_2=\|{X_2}^T\omega_2\|_2=1}{\arg\max} \left\{\left({X_1}^T\omega_1\right)^T\left({X_2}^T\omega_2\right)\right\}. \quad (6)$$

The objective of CCA is incorporated into the energy function E_s of the proposed MDDBM model, which turns out to be

$$E(\mathbf{v}, \mathbf{h}, \mathbf{y}) = E_s(\mathbf{v}, \mathbf{h}, \mathbf{y}) + E_c(\mathbf{h}); \quad (7)$$

$$\text{where} \quad E_c(\mathbf{h}) = -\sum_{m=1}^{M-1}\sum_{r=(m+1)}^{M}\sum_{j=1}^{H^{L_0m}} h_j^{L_0m} h_j^{L_0r} - \sum_{m=1}^{M}\lambda_m\left(1 - \sum_{j=1}^{H^{L_0m}}\left(h_j^{L_0m}\right)^2\right). \quad (8)$$

Here, λ_m is the Lagrange multiplier. It is assumed that H^{L_0m} is same for all $m \in \{1, 2, \ldots, M\}$. The criterion presented in (8) is termed the sum of correlations, which is used to integrate more than two sets of multidimensional variables.

The proposed multimodal discriminative deep model is illustrated in Fig. 1. The MDDBM architecture, described in

$$\begin{aligned} E_s(\mathbf{v}, \mathbf{h}, \mathbf{y}) = &-\sum_{m=1}^{M}\sum_{i=1}^{V^m}\sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m} h_j^{1m} - \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{p=1}^{Y}\sum_{j=1}^{H^{lm}} y_p u_{pj}^{lm} h_j^{lm} - \sum_{l=1}^{L_0-1}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}}\sum_{k=1}^{H^{(l+1)m}} h_j^{lm} w_{jk}^{(l+1)m} h_k^{(l+1)m} \\ &- \sum_{m=1}^{M}\sum_{j=1}^{H^{L_0m}}\sum_{k=1}^{H^{(L_0+1)}} h_j^{L_0m} w_{jk}^{(L_0+1)m} h_k^{(L_0+1)} - \sum_{l=(L_0+1)}^{L}\sum_{p=1}^{Y}\sum_{j=1}^{H^l} y_p u_{pj}^l h_j^l - \sum_{l=(L_0+1)}^{L-1}\sum_{j=1}^{H^l}\sum_{k=1}^{H^{(l+1)}} h_j^l w_{jk}^{(l+1)} h_k^{(l+1)} \\ &- \sum_{m=1}^{M}\sum_{i=1}^{V^m} a_i^m v_i^m - \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}} b_j^{lm} h_j^{lm} - \sum_{l=(L_0+1)}^{L}\sum_{j=1}^{H^l} b_j^l h_j^l - \sum_{p=1}^{Y} d_p y_p. \end{aligned} \quad (5)$$

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Section II-B, resembles Fig. 1 except for the correlation considered between different pairs of modalities. The parameter space of the model is defined by

$$\boldsymbol{\theta} = \boldsymbol{\theta}_s \cup \boldsymbol{\theta}_c; \quad \text{where} \quad \boldsymbol{\theta}_c = \{\lambda_m\} \quad \forall m \in \{1, 2, \ldots, M\}. \tag{9}$$

Thus, the concept of CCA is incorporated into the learning objective of the proposed D2CCA architecture by including only M number of λ_m parameters in the parameter space of the model. The energy function of the model decreases with the increase in the correlation among different views as the joint representation of the model is learned from maximally correlated subspaces. Now, from (2), it can be observed that in order to learn the parameters of the D2CCA model, the corresponding data-dependent and data-independent expectations are required to be estimated, which are described subsequently.

1) Estimation of Data-Dependent Expectations: Let us consider the following factorized distribution as the approximate posterior distribution of (3):

$$Q(\mathbf{h}|\mathbf{v}, \mathbf{y}) = \prod_{l=1}^{L_0} \prod_{m=1}^{M} \prod_{j=1}^{H^{lm}} q\left(h_j^{lm}|\mathbf{v}, \mathbf{y}\right) \prod_{l=(L_0+1)}^{L} \prod_{j=1}^{H^l} q\left(h_j^{l}|\mathbf{v}, \mathbf{y}\right); \tag{10}$$

where the hidden units $\{h_j\}$ are considered to be Bernoulli variables with $q(h_j|\mathbf{v}, \mathbf{y}) = \mu_j^{\{h_j=1\}}(1-\mu_j)^{\{h_j=0\}}$ and μ_j denotes the probability of being the state of h_j as 1. Thus, using mean field approximation, the stochastic binary values are replaced with real-valued probabilities.

The variational lower bound $\mathcal{L}_v$ on the log-likelihood function $\ln P(\mathbf{v}, \mathbf{h}, \mathbf{y})$ can be obtained by substituting (10) into right-hand side of (3), which is as follows:

$$\mathcal{L}_v = \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v}, \mathbf{y})\{\ln P(\mathbf{v}, \mathbf{h}, \mathbf{y}) - \ln Q(\mathbf{h}|\mathbf{v}, \mathbf{y})\}. \tag{11}$$

Considering the energy function (7) of the D2CCA model, the first term of the lower bound can be expressed as (12), while the second term can be expressed as (13). Detailed derivations of (12) and (13) are presented in Sections S1.1 and S1.2 of the supplementary material, respectively. Assembling (12) and (13), the variational bound $\mathcal{L}_v$ can be obtained using (11).

Based on the variational lower bound, the mean-field parameters of the proposed model can be computed for each hidden layer of the architecture. In order to obtain the mean field parameters ($\boldsymbol{\mu}$) for a particular layer, the variational bound, presented in (11), is maximized with respect to $\boldsymbol{\mu}$ for a fixed parameter set $\boldsymbol{\theta}$. Let us assume $H^{0m} = V^m$, $\boldsymbol{\mu}^{0m} = \mathbf{v}^m$, $H^0 = H^{L_0 m}$, $\mu_k^0 w_{kj}^1 = \sum_{m=1}^{M} \mu_k^{L_0 m} w_{kj}^{(L_0+1)m}$, and $\boldsymbol{\mu}^l = 0, \forall l > L$. So, $\dfrac{\partial \mathcal{L}_v}{\partial \mu_j^{lm}} = 0$ leads to

$$\mu_j^{lm} = \sigma\left(\sum_{k=1}^{H^{(l-1)m}} \mu_k^{(l-1)m} w_{kj}^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m} \mu_k^{(l+1)m} + \sum_{p=1}^{Y} y_p u_{pj}^{lm} + b_j^{lm}\right), \quad \text{for } 1 \le l < L_0 \text{ and } \forall m \tag{14}$$

where $\sigma(x) = 1/1+e^{-x}$ is the sigmoid function. Similarly;

$$\mu_j^{L_0 m} = \sigma\left(\sum_{k=1}^{H^{(L_0-1)m}} \mu_k^{(L_0-1)m} w_{kj}^{L_0 m} + \sum_{p=1}^{Y} y_p u_{pj}^{L_0 m} + b_j^{L_0 m} + \sum_{k=1}^{H^{(L_0+1)}} w_{jk}^{(L_0+1)m} \mu_k^{(L_0+1)} + \sum_{r\neq m=1}^{M} \mu_j^{L_0 r} - \lambda_m\right), \ \forall m. \tag{15}$$

The detailed derivations of (14) and (15) are presented in Section S1.3 of the supplementary material. Following similar steps, the update rule for $L_0 < l \le L$ can be obtained

$$\mu_j^{l} = \sigma\left(\sum_{k=1}^{H^{(l-1)}} \mu_k^{(l-1)} w_{kj}^{l} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^{(l+1)} \mu_k^{(l+1)} + \sum_{p=1}^{Y} y_p u_{pj}^{l} + b_j^{l}\right). \tag{16}$$

Thus, given training data along with the corresponding class label $\{\mathbf{v}^m, \mathbf{y}\}$, the equilibrium state of the model is

$$\begin{aligned}
&\sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v}, \mathbf{y}) \ln P(\mathbf{v}, \mathbf{h}, \mathbf{y}) \\
&= \sum_{m=1}^{M} \sum_{i=1}^{V^m} \sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m} \mu_j^{1m} + \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{p=1}^{Y} \sum_{j=1}^{H^{lm}} y_p u_{pj}^{lm} \mu_j^{lm} + \sum_{l=1}^{L_0-1} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} \sum_{k=1}^{H^{(l+1)m}} \mu_j^{lm} w_{jk}^{(l+1)m} \mu_k^{(l+1)m} \\
&+ \sum_{m=1}^{M} \sum_{j=1}^{H^{L_0 m}} \sum_{k=1}^{H^{(L_0+1)}} \mu_j^{L_0 m} w_{jk}^{(L_0+1)m} \mu_k^{(L_0+1)} + \sum_{l=(L_0+1)}^{L} \sum_{p=1}^{Y} \sum_{j=1}^{H^l} y_p u_{pj}^{l} \mu_j^{l} + \sum_{l=(L_0+1)}^{L-1} \sum_{j=1}^{H^l} \sum_{k=1}^{H^{(l+1)}} \mu_j^{l} w_{jk}^{(l+1)} \mu_k^{(l+1)} + \sum_{m=1}^{M} \sum_{i=1}^{V^m} a_i^m v_i^m \\
&+ \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} b_j^{lm} \mu_j^{lm} + \sum_{l=(L_0+1)}^{L} \sum_{j=1}^{H^l} b_j^{l} \mu_j^{l} + \sum_{p=1}^{Y} d_p y_p + \sum_{m=1}^{M-1} \sum_{r=(m+1)}^{M} \sum_{j=1}^{H^{L_0 m}} \mu_j^{L_0 m} \mu_j^{L_0 r} + \sum_{m=1}^{M} \lambda_m \left(1 - \sum_{j=1}^{H^{L_0 m}} \mu_j^{L_0 m}\right) - \ln Z
\end{aligned} \tag{12}$$

$$\begin{aligned}
&\sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v}, \mathbf{y}) \ln Q(\mathbf{h}|\mathbf{v}, \mathbf{y}) \\
&= \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} \left\{\mu_j^{lm} \ln \mu_j^{lm} + \left(1-\mu_j^{lm}\right) \ln\left(1-\mu_j^{lm}\right)\right\} + \sum_{l=(L_0+1)}^{L} \sum_{j=1}^{H^l} \left\{\mu_j^{l} \ln \mu_j^{l} + \left(1-\mu_j^{l}\right) \ln\left(1-\mu_j^{l}\right)\right\}
\end{aligned} \tag{13}$$

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

estimated using the concept of mean field theory. Now, based on the mean field parameters, obtained using (14)–(16), the parameter set $\boldsymbol{\theta}$ of the proposed architecture can be learned by maximizing the variational bound, which is given by

$$\frac{\partial \mathcal{L}_v}{\partial w_{ij}^{1m}} = v_i^m \mu_j^{1m}; \quad \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{lm}} = \mu_j^{(l-1)m} \mu_k^{lm}, \text{ for } 1 < l \leq L_0;$$
$$\frac{\partial \mathcal{L}_v}{\partial w_{jk}^{(L_0+1)m}} = \mu_j^{L_0 m} \mu_k^{(L_0+1)}; \quad \frac{\partial \mathcal{L}_v}{\partial u_{pj}^{lm}} = y_p \mu_j^{lm}, \text{ for } 1 \leq l \leq L_0;$$
$$\frac{\partial \mathcal{L}_v}{\partial w_{jk}^{l}} = \mu_j^{(l-1)} \mu_k^{l}, \text{ for } (L_0+1) < l \leq L; \quad \frac{\partial \mathcal{L}_v}{\partial a_i^m} = v_i^m;$$
$$\frac{\partial \mathcal{L}_v}{\partial u_{pj}^{l}} = y_p \mu_j^{l}, \text{ for } L_0 < l \leq L; \quad \frac{\partial \mathcal{L}_v}{\partial b_j^{lm}} = \mu_j^{lm}, \text{ for } 1 \leq l \leq L_0;$$
$$\frac{\partial \mathcal{L}_v}{\partial b_j^{l}} = \mu_j^{l}, \text{ for } L_0 < l \leq L; \quad \frac{\partial \mathcal{L}_v}{\partial d_p} = y_p; \quad \text{and}$$
$$\frac{\partial \mathcal{L}_v}{\partial \lambda_m} = \left(1 - \sum_{j=1}^{H^{L_0 m}} \mu_j^{L_0 m}\right), \quad \forall m \in \{1, 2, \ldots, M\}. \tag{17}$$

So, the data-dependent expectations are estimated by the gradient ascent on the lower bound of the proposed architecture.

2) Estimation of Data-Independent Expectations: Now, the second term of the log-likelihood function of (2), that is, energy gradient with respect to the model distribution, is estimated using the Markov Chain Monte Carlo-based stochastic approximation procedure [27]. The idea behind this approach is to sample a new state of the model from the current state based on the conditional distributions over visible and hidden nodes for a fixed parameter set $\boldsymbol{\theta}$. Considering $\mathbf{h}^l = 0, \forall l > L$, $h_k^0 w_{kj}^1 = \sum_{m=1}^{M} h_k^{L_0 m} w_{kj}^{(L_0+1)m}$, and $H^0 = H^{L_0 m}$, the conditional distributions for the D2CCA model $\forall m$, is given by

$$P\left(\mathbf{h}^{1m} | \mathbf{v}^m, \mathbf{y}, \mathbf{h}^{2m}\right) = \frac{P\left(\mathbf{h}^{1m}, \mathbf{v}^m, \mathbf{y}, \mathbf{h}^{2m}\right)}{\sum_{\mathbf{h}^{1m}} P\left(\mathbf{h}^{1m}, \mathbf{v}^m, \mathbf{y}, \mathbf{h}^{2m}\right)}$$
$$= \frac{\prod_{j=1}^{H^{1m}} e^{h_j^{1m}\left(\sum_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum_{p=1}^{Y} y_p u_{pj}^{1m} + \sum_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + b_j^{1m}\right)}}{\prod_{j=1}^{H^{1m}} \sum_{\mathbf{h}^{1m}} e^{h_j^{1m}\left(\sum_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum_{p=1}^{Y} y_p u_{pj}^{1m} + \sum_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + b_j^{1m}\right)}}. \tag{18}$$

The detailed derivation of (18) is presented in Section S1.4 of the supplementary material. So, for $m \in \{1, 2, \ldots, M\}$

$$P\left(h_j^{1m} | \mathbf{v}^m, \mathbf{y}, \mathbf{h}^{2m}\right) = \sigma\left(\sum_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum_{p=1}^{Y} y_p u_{pj}^{1m} + \sum_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + b_j^{1m}\right). \tag{19}$$

$$P\left(h_j^{lm} | \mathbf{h}^{(l-1)m}, \mathbf{h}^{(l+1)m}, \mathbf{y}\right) = \sigma\left(\sum_{k=1}^{H^{(l-1)m}} h_k^{(l-1)m} w_{kj}^{lm} + b_j^{lm} + \sum_{p=1}^{Y} y_p u_{pj}^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m} h_k^{(l+1)m}\right), \text{ for } 1 < l < L_0 \ \forall m \tag{20}$$

$$P\left(h_j^{L_0 m} | \mathbf{h}^{(L_0-1)m}, \mathbf{h}^{(L_0+1)}, \mathbf{h}^{L_0 r}, \mathbf{y}\right) = \sigma\left(\sum_{k=1}^{H^{(L_0-1)m}} h_k^{(L_0-1)m} w_{kj}^{L_0 m} + \sum_{p=1}^{Y} y_p u_{pj}^{L_0 m} + b_j^{L_0 m} + \sum_{k=1}^{H^{(L_0+1)}} w_{jk}^{(L_0+1)m} h_k^{(L_0+1)} + \sum_{r \neq m=1}^{M} h_j^{L_0 r} - \lambda_m\right), \quad \forall m. \tag{21}$$

The detailed derivations of (19)–(21) are presented in Sections S1.5–S1.7 of the supplementary material. Following similar steps, the subsequent conditional distributions can be obtained:

$$P\left(h_j^l | \mathbf{h}^{(l-1)}, \mathbf{h}^{(l+1)}, \mathbf{y}\right) = \sigma\left(\sum_{k=1}^{H^{(l-1)}} h_k^{(l-1)} w_{kj}^{l} + \sum_{p=1}^{Y} y_p u_{pj}^{l} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^{(l+1)} h_k^{(l+1)} + b_j^{l}\right) \text{ for } L_0 < l \leq L; \tag{22}$$

$$P\left(v_i^m | \mathbf{h}^{1m}\right) = \sigma\left(\sum_{j=1}^{H^{1m}} w_{ij}^{1m} h_j^{1m} + a_i^m\right), \quad \forall m; \tag{23}$$

$$P\left(y_p | \mathbf{h}^{11}, \ldots, \mathbf{h}^{L_0 M}, \mathbf{h}^{L_0+1}, \ldots, \mathbf{h}^{L}\right) = \frac{e^{X_p}}{\sum_{\tilde{p}=1}^{Y} e^{X_{\tilde{p}}}}; \tag{24}$$

$$X_p = \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} u_{pj}^{lm} h_j^{lm} + \sum_{l=(L_0+1)}^{L} \sum_{j=1}^{H^{l}} u_{pj}^{l} h_j^{l} + d_p. \tag{25}$$

Given that the convergence criteria are satisfied, which are to be discussed in Section III-A, if a Markov chain is run for a sufficient number of steps, then it can be ensured that the chain will converge to a unique stationary distribution such that the subsequent states of the chain will be accordingly distributed. The gradient of the energy function under model distribution is estimated by drawing samples from the obtained stationary distribution. So, many persistent chains are run in parallel, and states of the chains are sampled based on the conditional distributions, described in (19)–(24). Thus, the data-independent expectations with respect to the model parameters are approximated as follows, where the state variables, sampled from the model distribution, are denoted with a superscript tilde (e.g., $\tilde{v}$);

$$\frac{\partial E}{\partial w_{ij}^{1m}} = \tilde{v}_i^m \tilde{h}_j^{1m}; \quad \frac{\partial E}{\partial w_{jk}^{lm}} = \tilde{h}_j^{(l-1)m} \tilde{h}_k^{lm}, \text{ for } 1 < l \leq L_0;$$
$$\frac{\partial E}{\partial w_{jk}^{(L_0+1)m}} = \tilde{h}_j^{L_0 m} \tilde{h}_k^{(L_0+1)}; \quad \frac{\partial E}{\partial u_{pj}^{lm}} = \tilde{y}_p \tilde{h}_j^{lm}, \text{ for } 1 \leq l \leq L_0;$$
$$\frac{\partial E}{\partial w_{jk}^{l}} = \tilde{h}_j^{(l-1)} \tilde{h}_k^{l}, \text{ for } (L_0+1) < l \leq L; \quad \frac{\partial E}{\partial a_i^m} = \tilde{v}_i^m;$$
$$\frac{\partial E}{\partial u_{pj}^{l}} = \tilde{y}_p \tilde{h}_j^{l}, \text{ for } L_0 < l \leq L; \quad \frac{\partial E}{\partial b_j^{lm}} = \tilde{h}_j^{lm}, \text{ for } 1 \leq l \leq L_0;$$
$$\frac{\partial E}{\partial b_j^{l}} = \tilde{h}_j^{l}, \text{ for } L_0 < l \leq L; \quad \frac{\partial E}{\partial d_p} = \tilde{y}_p; \quad \text{and}$$
$$\frac{\partial E}{\partial \lambda_m} = \left(1 - \sum_{j=1}^{H^{L_0 m}} \left(\tilde{h}_j^{L_0 m}\right)^2\right), \quad \forall m \in \{1, 2, \ldots, M\}. \tag{26}$$

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Hence, the proposed model can be efficiently learned from data-dependent and data-independent estimates, obtained in (17) and (26), respectively.

Let N, S, t, and η be the number of training samples, number of persistent Markov chains, current epoch, and learning rate, respectively. Thus, the update rule for different parameters of the proposed D2CCA architecture, required to perform gradient ascent on the log-likelihood function corresponding to the energy function of (7), is as follows:

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^t + \Delta\boldsymbol{\theta}^t; \tag{27}$$

$$\text{where } \Delta\boldsymbol{\theta}^t = \eta\left\{\frac{1}{N}\sum_{n=1}^{N}\left(\frac{\partial \mathcal{L}_v}{\partial \boldsymbol{\theta}}\right)_n - \frac{1}{S}\sum_{s=1}^{S}\left(\frac{\partial E}{\partial \boldsymbol{\theta}}\right)_s\right\} - \rho\boldsymbol{\theta}^t + \zeta\,\Delta\boldsymbol{\theta}^{(t-1)}. \tag{28}$$

So, from (27) and (28), it can be observed that the update rules of the proposed D2CCA model follow the Hebbian rule, which is originally employed for the learning of the standard Boltzmann machine [28]. The energy function of the model is defined in such a way that it not only considers relation within a particular modality but also across different modalities. So, if a state of the model is stuck in a local minima of energy landscape, the learning will help the state to raise the energy of the state, so that the model can come out of the local minima [29]. The learning algorithm of the proposed D2CCA model is illustrated in Section S2 of the supplementary material.

III. Various Aspects of Proposed D2CCA Model

In this section, various aspects of the proposed framework are analyzed, which include convergence analysis and class evolution of the D2CCA model for dynamic streaming data.

A. Convergence Analysis

In the proposed model, variational learning is employed to estimate the data-dependent expectations, which provides a lower bound $\mathcal{L}_v : \Re^{|\boldsymbol{\theta}|} \mapsto \Re$ on the log-likelihood function (3) of the model. Given a particular state, the parameter set $\boldsymbol{\theta}$ of the model is updated by applying gradient ascent on $\mathcal{L}_v$. In this section, the convergence of the gradient ascent algorithm on $\mathcal{L}_v$ is discussed.

The gradient function of $\mathcal{L}_v$, corresponding to the energy function of (7), is given by

$$\nabla\mathcal{L}_v(\theta) = \left[\begin{array}{cccccc} \frac{\partial\mathcal{L}_v(\theta)}{\partial w_{ij}^{1m}} & \cdots & \frac{\partial\mathcal{L}_v(\theta)}{\partial w_{jk}^{(L_0+1)m}} & \frac{\partial\mathcal{L}_v(\theta)}{\partial w_{jk}^{L_0+2}} & \cdots & \frac{\partial\mathcal{L}_v(\theta)}{\partial w_{jk}^{L}} \\ \frac{\partial\mathcal{L}_v(\theta)}{\partial u_{pj}^{1m}} & \cdots & \frac{\partial\mathcal{L}_v(\theta)}{\partial u_{pj}^{L_0m}} & \frac{\partial\mathcal{L}_v(\theta)}{\partial u_{pj}^{L_0+1}} & \cdots & \frac{\partial\mathcal{L}_v(\theta)}{\partial u_{pj}^{L}} \\ \frac{\partial\mathcal{L}_v(\theta)}{\partial a_i^m} & \frac{\partial\mathcal{L}_v(\theta)}{\partial b_j^{1m}} & \cdots & \frac{\partial\mathcal{L}_v(\theta)}{\partial b_j^{L_0m}} & \frac{\partial\mathcal{L}_v(\theta)}{\partial b_j^{L_0+1}} & \cdots \\ \frac{\partial\mathcal{L}_v(\theta)}{\partial b_j^{L}} & \frac{\partial\mathcal{L}_v(\theta)}{\partial d_p} & \frac{\partial\mathcal{L}_v(\theta)}{\partial \lambda_m} & & & \end{array}\right]^T, \quad \forall i,j,k,p,m. \tag{29}$$

The gradient of $\mathcal{L}_v(\theta)$ with respect to each of the parameters can be obtained from (17), which makes it clear that the gradient function $\nabla\mathcal{L}_v(\theta)$ is independent of θ, that is, $\nabla\mathcal{L}_v(\theta_1) = \nabla\mathcal{L}_v(\theta_2), \forall\theta_1,\theta_2 \in \boldsymbol{\theta}$. So, it can be said that $\mathcal{L}_v$ is a differential function having β-Lipschitz continuous gradient for some $\beta \geq 0$, that is, $\|\nabla\mathcal{L}_v(\theta_1) - \nabla\mathcal{L}_v(\theta_2)\|_2 \leq \beta\|\theta_1 - \theta_2\|_2$. For a function having β-Lipschitz gradient, it is known that $\forall\theta_1,\theta_2 \in \boldsymbol{\theta}$,

$$\mathcal{L}_v(\theta_1) \leq \mathcal{L}_v(\theta_2) + \nabla\mathcal{L}_v(\theta_2)^T(\theta_1 - \theta_2) + \frac{1}{2}\beta\|\theta_1 - \theta_2\|_2^2. \tag{30}$$

Let, $\theta_1 = \theta^t$ and $\theta_2 = \theta^{t+1}$, where t denotes the epoch or iteration number. Now, rearranging the terms in (30), we get

$$\mathcal{L}_v(\theta^{t+1}) \geq \mathcal{L}_v(\theta^t) + \nabla\mathcal{L}_v(\theta^{t+1})^T(\theta^{t+1} - \theta^t) - \frac{1}{2}\beta\|\theta^{t+1} - \theta^t\|_2^2.$$

Now, using the fact that for the gradient ascent algorithm $\theta^{t+1} = \theta^t + \eta\nabla\mathcal{L}_v(\theta^t)$, where η denotes the learning rate, and $\nabla\mathcal{L}_v(\theta^{t+1}) = \nabla\mathcal{L}_v(\theta^t)$, we have

$$\mathcal{L}_v(\theta^{t+1}) \geq \mathcal{L}_v(\theta^t) + \eta\left(1 - \frac{1}{2}\beta\eta\right)\|\nabla\mathcal{L}_v(\theta^t)\|_2^2.$$

Assuming η to be small enough such that $\eta \leq 1/\beta$, we get $(1 - 1/2\beta\eta) \geq \frac{1}{2}$. Thus, we have

$$\mathcal{L}_v(\theta^{t+1}) \geq \mathcal{L}_v(\theta^t) + \frac{1}{2}\eta\|\nabla\mathcal{L}_v(\theta^t)\|_2^2. \tag{31}$$

Let θ^* be the optimal parameter set such that $\mathcal{L}_v(\theta^*) \geq \mathcal{L}_v(\theta), \forall\theta \in \boldsymbol{\theta}$. Since $\mathcal{L}_v(\theta)$ is a linear function of θ, we have

$$\mathcal{L}_v(\theta^t) = \mathcal{L}_v(\theta^*) + \nabla\mathcal{L}_v(\theta^t)^T(\theta^t - \theta^*). \tag{32}$$

Using (32) in (31), we get

$$\mathcal{L}_v(\theta^*) - \mathcal{L}_v(\theta^{t+1}) \leq -\nabla\mathcal{L}_v(\theta^t)^T(\theta^t - \theta^*) - \frac{1}{2}\eta\|\nabla\mathcal{L}_v(\theta^t)\|_2^2$$

$$\Rightarrow \mathcal{L}_v(\theta^*) - \mathcal{L}_v(\theta^{t+1}) \leq \frac{1}{2\eta}\left\{\|\theta^t - \theta^*\|_2^2 - \|\theta^{t+1} - \theta^*\|_2^2\right\}.$$

Taking summation over iteration till $t = \tau$, we get

$$\tau\mathcal{L}_v(\theta^*) - \sum_{t=0}^{\tau-1}\mathcal{L}_v(\theta^{t+1}) \leq \frac{1}{2\eta}\left\{\|\theta^0 - \theta^*\|_2^2 - \|\theta^\tau - \theta^*\|_2^2\right\}.$$

Since $\mathcal{L}_v(\theta)$ is an increasing function of θ, we can replace $\sum_{t=0}^{\tau-1}\mathcal{L}_v(\theta^{t+1})$ with $\tau\mathcal{L}_v(\theta^\tau)$ and the inequality will still hold. Thus,

$$\mathcal{L}_v(\theta^*) - \mathcal{L}_v(\theta^\tau) \leq \frac{1}{2\eta\tau}\left\{\|\theta^0 - \theta^*\|_2^2 - \|\theta^\tau - \theta^*\|_2^2\right\} \leq \frac{1}{2\eta\tau}\|\theta^0 - \theta^*\|_2^2. \tag{33}$$

From (33), it can be concluded that the algorithm converges with a rate $\mathcal{O}((1/\tau))$ after τ iterations if the learning rate is considered to be small enough, that is, $\eta \leq 1/\beta$.

Now, the stochastic approximation procedure is considered in the proposed model to approximate data-independent expectations. Convergence of the procedure to an asymptotically stable point is already established in [30]. One necessary condition requires the learning rate (η) to decrease with iteration t so that the algorithm eventually settles down to a fixed state. So, it is required that $\sum_{t=0}^{\infty}\eta_t = \infty$ and $\sum_{t=0}^{\infty}\eta_t^2 < \infty$. This condition can be trivially satisfied by setting $\eta_t = a/(b+t)$, for constants $a > 0$ and $b > 0$. Also, in practice, the sequence $|\theta^t|$ is bounded and the Markov chain is ergodic which, along with the condition on learning rate, establishes the convergence of the stochastic approximation procedure. Together with the condition on the variational learning (33), this ensures convergence of the proposed algorithm.

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

B. D2CCA for Class Evolution of Dynamic Streaming Data

The proposed deep architecture can deal with the problem of supervised learning, where the class labels of the given samples as well as the total number of classes are known a priori. However, there might be cases where the number of classes is not known beforehand, although the input samples are provided with the corresponding class labels. In dynamic data streams, the 'class evolution'occurs when a sample with a new class label comes into the data stream. In such a scenario, the modifications of the proposed D2CCA model that are needed to be adopted for accurate prediction of the class labels of the given samples are discussed next.

In D2CCA architecture, the number of class nodes is considered to be equal to the number of distinct class labels. Since the number of class labels is not known beforehand, let us assume a maximum possible class node, which is denoted by Y in the proposed D2CCA model. Now, out of the Y nodes, only one node will be activated for a particular input sample. Let, Y_0 be the total number of input classes. Then, it is evident from the learning of the proposed model that during the estimation of data-dependent expectations, described in Section II-C1, only Y_0 nodes will be activated, while the rest of the $(Y - Y_0)$ nodes will always remain at zero. Hence, the terms related to class nodes will contribute to the computation of variational lower bound in (11), mean field equations from (14) to (16), and gradient of lower bound in (17) only for $p \in \{1, \ldots, Y_0\}$ and will have a zero value for $p \in \{Y_0 + 1, \ldots, Y\}$. So, the terms related to class nodes contribute to the estimation of data-dependent expectations only for valid class nodes, while they remain quiescent for the rest of the nodes.

However, this is not the case for the estimation of data-independent expectations, presented in Section II-C2. For the computation of conditional distribution $P(y_p|\mathbf{h}^{11}, \ldots, \mathbf{h}^{L_0 M}, \mathbf{h}^{L_0+1}, \ldots, \mathbf{h}^{L})$ in (24), the value of X_p depends on the weights and biases associated with the node y_p, which may result into a prediction of class labels corresponding to the invalid class nodes. In order to overcome such a situation, the update rule for the weight and bias parameters related to the class nodes, that is, u_{pj}^{lm}, u_{pj}^{l}, and d_p, which is earlier demonstrated in (28), can be modified as follows:

$$\Delta u_{pj}^{lm^t} = \begin{cases} -u_{pj}^{lm^t}, & \text{if } \sum_{n=1}^{N} y_p \mu_j^{lm} = 0; \\ \eta\left\{\frac{1}{N}\sum_{n=1}^{N}\left(y_p \mu_j^{lm}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(\tilde{y}_p \tilde{h}_j^{lm}\right)\right\} - \rho u_{pj}^{lm^t} \\ +\zeta\, \Delta u_{pj}^{lm^{(t-1)}}, & \text{otherwise;} \end{cases} \quad \text{for } 1 \le l \le L_0. \tag{34}$$

$$\Delta u_{pj}^{lm^t} = \begin{cases} -u_{pj}^{l^t}, & \text{if } \sum_{n=1}^{N} y_p \mu_j^{l} = 0; \\ \eta\left\{\frac{1}{N}\sum_{n=1}^{N}\left(y_p \mu_j^{l}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(\tilde{y}_p \tilde{h}_j^{l}\right)\right\} - \rho u_{pj}^{l^t} \\ +\zeta\, \Delta u_{pj}^{l^{(t-1)}}, & \text{otherwise;} \end{cases} \quad \text{for } L_0 < l \le L. \tag{35}$$

$$\Delta d_p^t = \begin{cases} -d_p^t, & \text{if } \sum_{n=1}^{N} y_p = 0; \\ \eta\left\{\frac{1}{N}\sum_{n=1}^{N} y_p - \frac{1}{S}\sum_{s=1}^{S} \tilde{y}_p\right\} - \rho d_p^t + \zeta\, \Delta d_p^{(t-1)} \\ \text{otherwise.} \end{cases} \tag{36}$$

Thus, the update rules (34)–(36), considered with (27), ensure that weight and bias parameters associated with the class nodes, which remain quiescent during data-dependent estimation, remain at zero value throughout the learning of the D2CCA model. Hence, during the prediction of the class labels of given test samples, it can be ensured that only the valid class nodes will be activated.

IV. Experimental Results and Discussions

In this section, the classification performance of the proposed architecture is studied extensively and the corresponding results are reported. In order to demonstrate the efficacy of the proposed model, several existing algorithms are considered, which include RGCCA [5], MCCA [6], GMCCA [7], GMKCCA [7], LasCCA [8], DisCCA [8], MvDA [9], MvDA-VC [11], MDBM [14], dMCCA [12], TOCCA [13], DACCA [15], DCCA-VG [16], TDDCCA [17], TCCA [18], MDL-CW [19], MMGNN [20], and MVGAN [21]. Both training–testing and tenfold CV are performed to evaluate the performance of the proposed model as well as existing algorithms. In the case of training–testing, overall classification accuracy is considered, while for tenfold CV, mean, median, standard deviation, and p-values computed using paired-t (one-tailed) and Wilcoxon signed-rank (one-tailed) tests, with 95% confidence level, are employed. The source code of the proposed algorithm, written in C language, is available at www.isical.ac.in/∼bibl/results/d2cca/d2cca.html.

A. Description of Data Sets

In this study, seven benchmark databases, namely, Digits [31], Caltech [1], CiteSeer [32], Cora [32], NUS-WIDE-OBJECT (NW-OBJECT) [33], Reuters [34], and ORL [35], and five cancer datasets are considered to evaluate the performance of different algorithms. Digits, Caltech, NW-OBJECT, and ORL are image-based databases, CiteSeer and Cora consist of scientific publications with annotated labels, whereas Reuters is a multilingual categorization dataset.

Different subtypes are identified for five real-life cancer datasets, which include cervical carcinoma (CESC), colorectal carcinoma (CRC), kidney carcinoma (KIDNEY), lower grade glioma (LGG), and lung carcinoma (LUNG). These datasets are obtained from The Cancer Genome Atlas (TCGA) [36]. A brief description of the datasets, which includes the number of samples, number of classes, number of views, and number of features in each view, is presented in Table I. Each data set is randomly partitioned into two sets for training–testing and ten separate folds for tenfold CV. In both cases, the samples are equally distributed with respect to different classes. A detailed description of these databases is given at www.isical.ac.in/∼bibl/results/d2cca/d2cca.html.

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE I
DESCRIPTION OF DATASETS

	Data	Sample	Class	View	V^1	V^2	V^3	V^4	V^5	V^6
Benchmark	Digits	2000	10	6	240	76	216	47	64	6
	Caltech	2386	20	6	48	40	254	1984	512	928
	CiteSeer	3312	6	4	3703	3312	3312	3312	-	-
	Cora	2708	7	4	1433	2708	2708	2708	-	-
	NW-OBJECT	30000	31	5	64	225	144	73	128	-
	Reuters	18758	6	5	21531	24893	34279	15506	11547	-
	ORL	200	40	2	4096	4096	-	-	-	-
Omics	CESC	104	3	4	291368	192	174	12028	-	-
	CRC	261	2	4	293526	222	236	13465	-	-
	KIDNEY	305	2	5	300451	174	209	20502	9059	-
	LGG	374	3	5	293965	181	139	11973	6261	-
	LUNG	546	2	5	294668	180	216	20502	49230	-

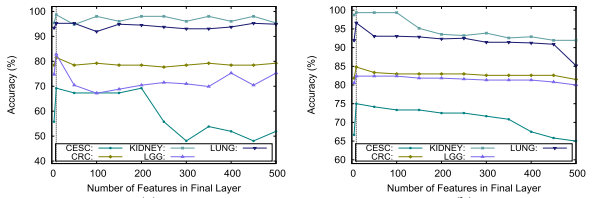

Fig. 2. Variation of classification accuracy with respect to number of features in final layer for omics datasets. (a) Training–testing. (b) Tenfold CV.

B. Model Architecture and Implementation Details

In case of the proposed D2CCA architecture, two modality-specific hidden layers and one joint hidden layer are considered for all the experiments. Each of the modality-specific hidden layers consists of 50 hidden nodes, whereas for the joint representation, an extensive experiment is carried out on five cancer data sets. The number of nodes or features at the final layer is varied from 5 to 500, and the variation of classification accuracy is studied for both training–testing and tenfold CV. The corresponding results are reported in Fig. 2. From the obtained results, it can be observed that the performance of the proposed model increases when the number of features in the final layer is increased from 5 to 10. If the number of features is further increased from 10, the performance of the model remains almost constant or even decreases in some cases. However, if the number of nodes in the joint layer is fixed at 10, considerable classification accuracy can be achieved irrespective of the experimental setup and data sets considered. Thus, the number of nodes in joint representation is considered to be 10 for the rest of the study.

The greedy layerwise pretraining [37] is performed to initialize the parameters of the architecture. The hidden nodes of the model are represented by the probability values and the parameters are updated based on the mini-batches of training samples. The number of epochs, the values of momentum, and weight decay are considered to be 100, 0.5, and 0.0005, respectively. The value of the learning rate is initialized at 0.01 and then gradually decreased with an increase in a number of epochs. For the estimation of data-independent expectations, 100 Gibbs steps and 20 separate Markov chains are considered.

C. Choice of Deep Model

In the existing literature, several deep models have been considered to learn the joint representation of the data from the given multiple modalities. The dMCCA model is developed based on the feed-forward network and stacked autoencoder has been considered for MDL-CW, while graph neural network is used for the development of MMGNN and the generative adversarial network has been considered for the implementation of MVGAN. However, the proposed D2CCA architecture has been developed based on the framework of the Boltzmann machine. In this section, the performance of the proposed architecture is compared with that of different existing deep frameworks, and the corresponding results are reported in Tables II and III for benchmark and omics databases, respectively.

TABLE II
EFFECTIVENESS OF PROPOSED ARCHITECTURE OVER SEVERAL DEEP MODELS ON BENCHMARK DATA

Data	dMCCA	MDL-CW	MMGNN	MVGAN	D2CCA
Digits	10.00	86.40	88.90	82.70	**91.00**
Caltech	33.71	54.25	74.27	77.31	**84.54**
CiteSeer	21.16	42.05	41.78	46.32	**73.12**
Cora	30.19	41.40	42.06	44.95	**80.47**
NW-OBJECT	17.80	18.20	37.87	27.61	**52.21**
Reuters	51.72	62.69	59.16	57.60	**84.60**
ORL	53.72	27.50	60.27	60.38	**78.50**

TABLE III
EFFECTIVENESS OF PROPOSED FRAMEWORK OVER DIFFERENT DEEP MODELS ON OMICS DATASETS

Data	Different Metrics		dMCCA	MDL-CW	MMGNN	MVGAN	D2CCA
CESC	Train-Test		51.92	65.38	55.77	57.69	**69.23**
	10-fold CV	Mean	40.83	69.17	69.17	61.67	**75.00**
		Median	41.67	66.67	66.67	58.33	**75.00**
		StdDev	12.08	14.72	13.64	12.55	**6.80**
		Paired-*t*:p	3.17E-05	1.66E-01	1.36E-01	3.16E-03	-
		Wilcoxon:p	2.45E-03	1.42E-01	1.05E-01	6.97E-03	-
CRC	Train-Test		73.85	76.15	74.62	79.23	**81.54**
	10-fold CV	Mean	74.07	77.04	80.00	81.85	**84.81**
		Median	74.07	77.78	81.48	81.48	**87.04**
		StdDev	0.00	3.83	4.35	**3.68**	6.86
		Paired-*t*:p	3.96E-04	1.22E-03	3.52E-02	9.84E-02	-
		Wilcoxon:p	5.56E-03	6.17E-03	4.52E-02	1.72E-01	-
KIDNEY	Train-Test		86.18	92.76	89.47	95.39	**98.68**
	10-fold CV	Mean	69.03	95.48	95.16	81.61	**99.35**
		Median	67.74	95.16	93.55	77.42	**100.00**
		StdDev	4.08	3.12	3.80	12.64	**1.36**
		Paired-*t*:p	1.10E-09	1.28E-03	4.74E-03	6.10E-04	-
		Wilcoxon:p	1.79E-03	5.56E-03	1.28E-02	5.66E-03	-
LGG	Train-Test		62.37	73.12	71.51	74.73	**82.80**
	10-fold CV	Mean	50.79	76.84	72.11	76.84	**82.37**
		Median	47.37	77.63	72.37	77.63	**82.89**
		StdDev	7.24	7.63	**3.96**	7.63	5.41
		Paired-*t*:p	2.19E-06	3.50E-03	1.45E-05	3.50E-03	-
		Wilcoxon:p	2.50E-03	8.09E-03	2.50E-03	8.09E-03	-
LUNG	Train-Test		59.34	93.77	90.84	92.67	**95.24**
	10-fold CV	Mean	60.71	93.93	82.14	94.11	**96.61**
		Median	58.93	95.54	93.75	94.64	**97.32**
		StdDev	5.05	4.31	19.32	3.95	**3.41**
		Paired-*t*:p	1.99E-10	2.33E-03	1.75E-02	1.47E-02	-
		Wilcoxon:p	2.52E-03	5.71E-03	2.17E-02	1.76E-02	-

From the results presented in Table II, it can be observed that dMCCA and MDL-CW fail to categorize the given observations from the benchmark databases correctly, but both MMGNN and MVGAN have achieved satisfactory results on the datasets. However, the highest classification accuracy is attained by the proposed model on all seven databases. In the case of omics data sets, corresponding to the results reported in Table III, significant improvement in performance can be noted for the proposed D2CCA model as compared with different deep architectures for both training–testing and tenfold CV. Statistical significance analysis demonstrates that the proposed

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE IV
EFFECTIVENESS OF PROPOSED ARCHITECTURE ON BENCHMARK DATA

Data	MDBM	MDDBM	D2CCA+SVM	D2CCA+Bayes	D2CCA
Digits	10.00	85.60	87.50	85.90	**91.00**
Caltech	2.53	74.14	83.90	71.74	**84.54**
CiteSeer	17.80	71.93	63.85	**94.10**	73.12
Cora	10.99	64.37	63.49	66.26	**80.47**
NW-OBJECT	26.07	43.75	33.56	34.16	**52.21**
Reuters	46.84	61.25	84.64	**85.47**	84.60
ORL	52.16	75.00	**86.25**	83.75	78.50

model achieves significantly better p-values for 34 cases and better but not significant p-values for the rest of the six cases.

D. Effectiveness of Proposed D2CCA Architecture

In this section, the effectiveness of different aspects of the proposed D2CCA architecture is demonstrated, which includes the importance of incorporating discriminative information, the significance of integrating CCA, and the effectiveness of the proposed method as a feature extractor. The corresponding results are reported in Fig. 3 and Tables IV and V. The scatter plots of Fig. 3 are depicted by considering the most relevant feature at the x-axis and the corresponding most significant feature at the y-axis, obtained using the concept of rough hypercuboid approach [38]. While the top row of Fig. 3 corresponds to the MDBM model, the plots of the middle and last rows are obtained from MDDBM and D2CCA architectures, respectively.

1) Importance of Incorporating Discriminative Information: The proposed MDDBM architecture is developed by integrating the merits of DBM and the supervised information of sample categories. While MDBM [39] is an unsupervised model, the proposed MDDBM approach incorporates class nodes into the architecture to improve the discriminative ability of the model. In this section, the importance of incorporating supervised information into the MDDBM architecture is established. The scatter plots of Fig. 3, obtained on benchmark databases, reveal that the samples from different classes overlap for MDBM, while the separation between the samples of various categories has improved in case of MDDBM. Similar results can be observed in Table IV, where MDDBM performs significantly better than MDBM for training–testing on all the benchmark databases. Considering the graphs of Fig. 3, corresponding to the omics datasets, it can be noticed that the samples from different cancer subtypes are better discriminated in the case of MDDBM as compared with MDBM. Results reported in Table V also confirm that MDDBM outperforms MDBM, irrespective of the experimental set-up and cancer datasets considered. The corresponding statistical significance tests are reported in Section S4 of the supplementary material, which establishes the importance of MDDBM over MDBM.

2) Significance of Integrating CCA: In the current study, two new architectures are proposed, namely, D2CCA and MDDBM, for multimodal data classification. The D2CCA architecture is developed by integrating the theory of CCA with MDDBM architecture so that the joint representation at the final layer is learned from the corresponding maximally correlated subspaces. In order to study the significance of incorporating CCA into the MDDBM architecture, the performance of D2CCA is compared with that of MDDBM on

TABLE V
EFFECTIVENESS OF PROPOSED ARCHITECTURE ON OMICS DATASETS

Data	Different Metrics		MDBM	MDDBM	D2CCA +SVM	D2CCA +Bayes	D2CCA
CESC	Train-Test		48.08	67.31	**75.00**	69.23	69.23
	10-fold CV	Mean	52.50	64.17	54.17	**80.00**	75.00
		Median	54.17	66.67	58.33	**79.17**	75.00
		StdDev	17.59	**5.62**	18.94	8.05	6.80
		Paired-t:p	1.05E-03	3.13E-03	8.68E-03	9.16E-01	-
		Wilcoxon:p	3.44E-03	7.31E-03	1.89E-02	8.58E-01	-
CRC	Train-Test		26.15	78.46	**86.15**	61.54	81.54
	10-fold CV	Mean	54.07	71.11	**90.00**	64.44	84.81
		Median	70.37	74.07	**90.74**	62.96	87.04
		StdDev	24.33	15.20	**4.29**	7.45	6.86
		Paired-t:p	3.66E-03	1.78E-02	9.92E-01	1.52E-04	-
		Wilcoxon:p	3.42E-03	5.81E-03	9.98E-01	3.44E-03	-
KIDNEY	Train-Test		69.08	98.03	69.08	92.76	**98.68**
	10-fold CV	Mean	70.97	90.32	75.16	93.87	**99.35**
		Median	67.74	91.94	67.74	93.55	**100.00**
		StdDev	10.20	8.60	12.17	4.92	**1.36**
		Paired-t:p	4.59E-06	4.77E-03	9.07E-05	2.89E-03	-
		Wilcoxon:p	2.48E-03	3.56E-03	3.33E-03	5.76E-03	-
LGG	Train-Test		65.05	79.57	67.74	**94.62**	82.80
	10-fold CV	Mean	27.63	63.95	18.42	69.74	**82.37**
		Median	18.42	63.16	18.42	69.74	**82.89**
		StdDev	14.90	6.08	0.00	6.71	**5.41**
		Paired-t:p	3.03E-07	2.42E-05	1.75E-11	1.62E-04	-
		Wilcoxon:p	2.52E-03	2.53E-03	2.49E-03	2.53E-03	-
LUNG	Train-Test		87.91	94.51	92.67	90.48	**95.24**
	10-fold CV	Mean	61.25	90.18	67.68	91.61	**96.61**
		Median	42.86	91.96	62.50	92.86	**97.32**
		StdDev	23.81	7.49	26.45	5.19	**3.41**
		Paired-t:p	4.42E-04	5.00E-03	2.35E-03	1.01E-04	-
		Wilcoxon:p	2.50E-03	3.98E-03	2.52E-03	2.52E-03	-

both benchmark and omics datasets. Considering the scatter plots of Fig. 3, corresponding to the benchmark databases, it can be observed that the separation between the samples of different classes is improved in the case of D2CCA as compared with MDDBM. This result is also reflected in Table IV, where the classification accuracy has improved for D2CCA in comparison with MDDBM on the benchmark data for training–testing.

The scatter plots of Fig. 3, corresponding to each of the omics datasets, reveal that all the given classes are well separated for D2CCA. In the case of MDDBM, although the samples from different classes can be distinguished for KIDNEY and LUNG data, they tend to overlap for CESC, CRC, and LGG datasets. This observation can also be validated from the results reported in Table V, where it can be observed that MDDBM performs better on KIDNEY and LUNG data, in comparison with rest of the omics datasets. However, the highest classification accuracy is achieved by D2CCA architecture on all the datasets, for both training–testing and tenfold CV. Statistical significance analysis demonstrates that the D2CCA model attains significantly better p-values for all the cases.

3) Effectiveness of D2CCA as Feature Extractor: Apart from multimodal data classification, the proposed D2CCA architecture can be considered as a feature extractor as well. In order to establish the discriminative ability of the features, extracted by the D2CCA architecture, the joint representation of the multi-view data is provided as input to various classifiers, namely, support vector machine (SVM) and Bayes classifier. From the results reported in Table IV, it can be

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

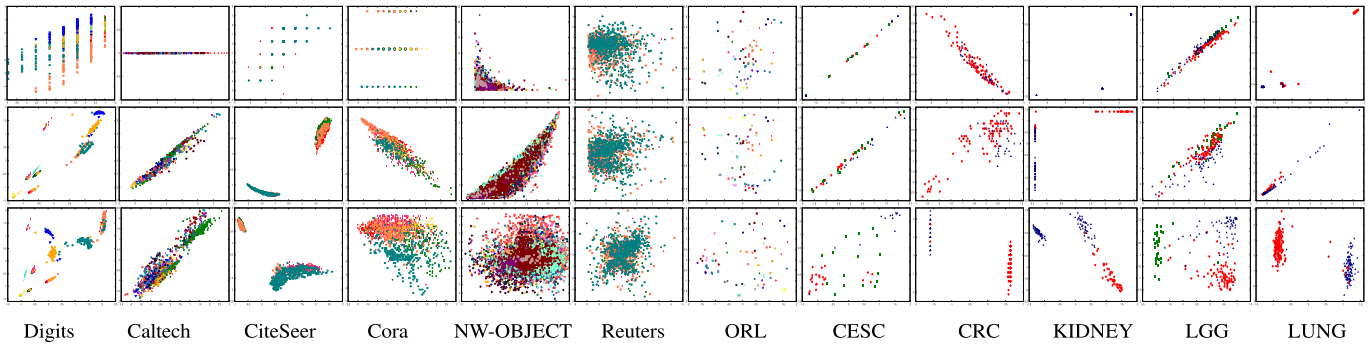

Fig. 3. Scatter plots of MDBM (top row), MDDBM (middle row), and D2CCA (bottom row) for benchmark and omics datasets.

TABLE VI

COMPARATIVE PERFORMANCE ANALYSIS OF EXISTING APPROACHES ON BENCHMARK DATASETS

Data	RGCCA	MCCA	GMCCA	GMKCCA	LasCCA	DisCCA	MvDA	MvDA-VC	dMCCA	TOCCA	DACCA	DCCA-VG	TCCA	TDDCCA	MDL-CW	MMGNN	MVGAN	D2CCA
Digits	90.30	87.00	11.20	6.60	10.20	5.60	92.40	93.50	10.00	96.50	84.60	89.00	**97.80**	85.90	86.40	88.90	82.70	91.00
Caltech	33.71	41.83	4.82	7.48	4.06	3.17	76.30	75.29	33.71	80.23	73.89	49.68	**87.58**	74.52	54.25	74.27	77.31	84.54
CiteSeer	27.61	58.13	23.43	24.98	22.25	20.71	37.69	43.51	21.16	39.60	55.22	37.33	56.40	40.42	42.05	41.78	46.32	**73.12**
Cora	52.16	32.85	30.97	30.19	31.63	30.19	53.94	55.72	30.19	52.39	50.06	44.62	56.94	53.05	41.40	42.06	44.95	**80.47**
NW-OBJECT	18.91	30.34	4.56	6.43	7.40	10.93	29.03	28.62	17.80	33.61	38.42	19.23	33.61	17.80	18.20	37.87	27.61	**52.21**
Reuters	55.26	57.50	24.78	28.69	28.67	23.27	56.01	55.15	51.72	57.38	56.38	64.38	75.97	57.27	62.69	59.16	57.60	**84.60**
ORL	18.75	18.75	51.25	5.00	73.75	72.50	3.75	28.75	53.72	54.72	57.16	32.50	66.67	56.83	27.50	60.27	60.38	**78.50**

TABLE VII

COMPARATIVE PERFORMANCE ANALYSIS OF EXISTING APPROACHES ON OMICS DATASETS

Data	Metrics		RGCCA	MCCA	GMCCA	GMKCCA	LasCCA	DisCCA	MvDA	MvDA-VC	dMCCA	TOCCA	DACCA	DCCA-VG	TCCA	TDDCCA	MDL-CW	MMGNN	MVGAN	D2CCA
CESC	Train-Test		61.54	38.46	42.31	44.23	42.31	36.54	42.31	40.38	51.92	36.54	47.12	65.38	61.54	53.98	65.38	55.77	57.69	**69.23**
	10-fold CV	Mean	75.00	45.83	49.17	38.33	35.00	39.17	46.67	50.00	40.83	43.33	39.81	67.50	60.19	**78.20**	69.17	69.17	61.67	75.00
		Median	**79.17**	50.00	50.00	41.67	33.33	33.33	41.67	50.00	41.67	50.00	39.42	70.83	61.54	78.20	66.67	66.67	58.33	75.00
		StdDev	13.03	13.75	14.41	11.92	15.61	10.43	15.32	14.16	12.08	14.05	5.37	16.87	5.74	**0.06**	14.72	13.64	12.55	6.80
		Paired-t:p	5.00E-01	1.91E-04	4.60E-04	5.04E-06	4.27E-06	3.21E-05	7.93E-04	3.54E-04	3.17E-05	1.42E-04	3.17E-07	9.67E-02	9.40E-06	9.15E-01	1.66E-01	1.36E-01	3.16E-03	-
		Wilcoxon:p	3.60E-01	2.46E-03	3.44E-03	2.50E-03	2.52E-03	2.38E-03	6.20E-03	2.50E-03	2.45E-03	2.49E-03	2.52E-03	7.63E-02	2.52E-03	8.34E-01	1.42E-01	1.05E-01	6.97E-03	-
CRC	Train-Test		83.85	73.85	83.85	50.77	78.46	73.08	80.77	83.08	73.85	73.85	85.77	77.69	**86.92**	72.50	76.15	74.62	79.23	81.54
	10-fold CV	Mean	81.85	60.74	78.52	54.07	78.52	62.59	83.70	**86.67**	74.07	74.07	81.69	76.30	82.92	48.80	77.04	80.00	81.85	84.81
		Median	83.33	62.96	81.48	55.56	77.78	68.52	85.19	**88.89**	74.07	74.07	81.54	75.93	83.85	48.76	77.78	81.48	81.48	87.04
		StdDev	5.64	10.36	7.57	8.59	4.20	13.23	5.00	6.34	0.00	0.00	3.12	4.68	3.55	**0.30**	3.83	4.35	3.68	6.86
		Paired-t:p	2.64E-02	1.94E-04	8.25E-03	3.32E-05	9.60E-03	1.28E-04	2.34E-01	7.39E-01	3.96E-04	3.96E-04	1.01E-01	4.73E-03	2.19E-01	2.97E-08	1.22E-03	3.52E-02	9.84E-02	-
		Wilcoxon:p	1.20E-01	2.50E-03	1.09E-02	2.52E-03	2.34E-02	2.52E-03	4.39E-01	9.16E-01	5.56E-03	5.56E-03	1.42E-01	1.25E-02	2.87E-01	2.53E-03	6.17E-03	4.52E-02	1.72E-01	-
KIDNEY	Train-Test		91.45	55.92	85.53	82.89	74.34	58.55	92.76	94.74	86.18	68.42	96.71	93.42	96.71	51.59	92.76	89.47	95.39	**98.68**
	10-fold CV	Mean	92.90	60.97	75.81	81.61	79.03	60.97	93.23	94.19	69.03	71.29	96.64	96.45	96.58	42.48	95.48	95.16	81.61	**99.35**
		Median	91.94	61.29	77.42	83.87	77.42	64.52	93.55	93.55	67.74	67.74	96.38	96.77	96.71	42.43	95.16	93.55	77.42	**100.00**
		StdDev	4.76	6.17	6.32	7.91	10.99	8.93	2.38	2.54	4.08	7.04	1.09	2.82	1.19	**0.22**	3.12	3.80	12.64	1.36
		Paired-t:p	7.75E-04	3.97E-09	1.26E-06	3.37E-05	1.21E-04	5.75E-08	3.69E-05	2.12E-06	1.10E-09	2.34E-07	1.10E-03	9.37E-03	5.49E-04	2.20E-16	1.28E-03	4.74E-03	6.10E-04	-
		Wilcoxon:p	5.71E-03	2.36E-03	2.46E-03	2.47E-03	2.39E-03	2.42E-03	2.39E-03	2.24E-03	1.79E-03	2.24E-03	4.56E-03	1.94E-02	2.45E-03	2.52E-03	5.56E-03	1.28E-02	5.66E-03	-
LGG	Train-Test		41.40	39.78	33.33	38.71	44.09	29.03	75.81	73.12	62.37	45.70	65.59	77.96	81.18	66.85	73.12	71.51	74.73	**82.80**
	10-fold CV	Mean	45.00	35.53	40.53	33.16	38.68	38.68	75.79	81.05	50.79	45.00	59.35	51.32	77.89	57.14	76.84	72.11	76.84	**82.37**
		Median	47.37	34.21	40.79	31.58	36.84	38.16	76.32	78.95	47.37	46.05	58.87	51.32	77.63	57.00	77.63	72.37	77.63	**82.89**
		StdDev	10.99	7.67	8.70	4.99	7.95	7.24	8.02	7.83	7.24	6.38	3.42	3.34	4.51	**0.39**	7.63	3.96	7.63	5.41
		Paired-t:p	2.77E-06	1.09E-07	7.85E-07	2.40E-09	4.46E-08	3.76E-08	1.13E-02	3.03E-01	2.19E-06	1.96E-07	6.98E-08	4.76E-08	3.34E-04	6.16E-08	3.50E-03	1.45E-05	3.50E-03	-
		Wilcoxon:p	2.53E-03	2.50E-03	2.53E-03	2.52E-03	2.53E-03	2.53E-03	1.83E-02	2.54E-01	2.50E-03	2.52E-03	2.53E-03	2.49E-03	5.76E-03	2.53E-03	8.09E-03	2.50E-03	8.09E-03	-
LUNG	Train-Test		87.91	46.52	68.86	86.08	82.42	47.62	92.31	91.58	59.34	57.14	95.05	89.74	93.41	67.42	93.77	90.84	92.67	**95.24**
	10-fold CV	Mean	87.68	51.43	68.39	86.07	85.18	50.71	94.82	95.54	60.71	57.14	95.71	94.11	95.42	67.72	93.93	82.14	94.11	**96.61**
		Median	86.61	50.89	69.64	87.50	85.71	48.21	96.43	95.54	58.93	57.14	95.60	95.54	95.24	67.65	95.54	93.75	94.64	**97.32**
		StdDev	4.16	3.13	7.48	8.32	6.68	8.84	4.16	3.29	5.05	0.00	1.17	4.69	1.00	**0.38**	4.31	19.32	3.95	3.41
		Paired-t:p	2.68E-05	3.13E-12	9.13E-08	1.63E-03	1.01E-04	2.76E-08	4.79E-02	9.67E-02	1.99E-10	2.13E-11	2.53E-01	1.48E-02	1.79E-01	5.71E-10	2.33E-03	1.75E-02	1.47E-02	-
		Wilcoxon:p	2.52E-03	2.53E-03	2.53E-03	2.53E-03	2.52E-03	2.53E-03	8.02E-02	9.61E-02	2.52E-03	2.43E-03	2.88E-01	1.76E-02	1.66E-01	2.53E-03	5.71E-03	2.17E-02	1.76E-02	-

observed that given the features extracted by the proposed architecture, reasonable accuracy is achieved using both SVM and Bayes classifier. In fact, significantly better performance can be noted from the Bayes classifier as compared with the D2CCA architecture for the CiteSeer dataset. A similar observation is noticed for the SVM on the ORL database. The results presented in Table V state that the samples from the CRC dataset can be efficiently recognized by the SVM, while the Bayes classifier can suitably identify different subtypes of LGG data for training–testing and CESC data for tenfold CV. However, considerable classification accuracy is achieved on all the omics datasets for both training–testing and tenfold CV using the proposed architecture itself. Statistical significance analysis reveals that out of a total of 20 cases, the proposed D2CCA framework attains significantly better p-values in 16 cases.

E. Comparative Performance Analysis

Finally, the classification performance of the proposed D2CCA architecture is compared with that of several existing methods on the benchmark as well as omics datasets, and the corresponding results are reported in Tables VI and VII. It is to be noted here that the proposed method predicts the class labels from the architecture itself based on the maximum class probability. Hence, no additional classifier is required in the proposed method. The existing algorithms include multiset CCA-based methods, multi-view discriminative analysis-based methods, and multi-view deep learning-based methods.

1) Performance of Multiset CCA Based Methods: In this section, the performance of the proposed method is analyzed with reference to several multiset CCA-based methods, namely, RGCCA [5], MCCA [6], GMCCA [7], GMKCCA [7], LasCCA [8], and DisCCA [8]. All the existing algorithms

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

extract 25 features from the given input views to represent the joint subspace, which are then applied to the input of SVM for classification purposes. From the results reported in Table VI, it can be observed that although RGCCA and MCCA achieve satisfactory results for the Digits dataset, and they have failed to obtain similar results for other benchmark databases. However, the proposed method attains the highest classification accuracy with respect to the existing CCA-based methods on all seven benchmark databases. In the case of omics datasets, represented in Table VII, the proposed method outperforms all the six multiset CCA-based methods on five cancer datasets for both training–testing and tenfold CV, except on the CRC dataset for training–testing, where RGCCA and GMCCA achieve highest classification accuracy. Statistical significance tests reveal that out of a total of 60 cases, the proposed architecture achieves significantly better p-values for 57 cases and better but not significant p-values in the remaining three cases.

2) Performance of Discriminative Analysis Based Methods: Various state-of-the-art multi-view discriminative analysis-based methods, namely, MvDA [9] and MvDA-VC [11], are considered for performance evaluation. For each of the methods, 25 features are extracted, and then given as input to the SVM for classification purpose. From the results reported in Table VI, it can be observed that both MvDA and MvDA-VC achieve considerable accuracy on all the benchmark databases. However, the highest classification accuracy is attained by the proposed method in all the cases, except for the Digits database. For omics datasets, corresponding results are reported in Table VII, which demonstrate that although MvDA-VC performs better than the proposed method on CRC dataset, the proposed architecture outperforms both the methods for rest of the cases. Statistical significance analysis demonstrates that out of a total of 20 cases, the proposed model attains significantly better p-values for 11 cases and better but not significant p-values for seven cases.

3) Performance of Deep Learning Based Methods: Finally, the performance of the proposed architecture is compared with that of several state-of-the-art multi-view deep learning based methods, namely, dMCCA [12], TOCCA [13], DACCA [15], DCCA-VG [16], TDDCCA [17], TCCA [18], MDL-CW [19], MMGNN [20], and MVGAN [21], and the corresponding results are reported in Tables VI and VII. For TOCCA, DACCA, DCCA-VG, TDDCCA, TCCA, and MDL-CW, 50, 80, 20, 50, $64|V|B$, and 600 features are extracted, respectively, from the given views, which are then applied to the input of the SVM for classification. Here, $|V|$ and B denote the number of input views and total number of batches, considered for a particular dataset. In the case of dMCCA, MMGNN, and MVGAN, 50 features are extracted at the final layer for each of the approaches. Since these methods are essentially feed-forward networks, they do not require any additional classifiers for class label prediction. The architecture for each of the models follows the same as suggested in the corresponding papers. From the results reported in Table VI, it can be noticed that except for the TCCA method on the Digits and Caltech databases, the proposed method performs considerably better than the existing multi-view deep learning based methods on the benchmark databases. For the omics datasets, results are presented in Table VII, which describes that the TCCA and TDDCCA approaches perform better than the proposed model on CRC data for training–testing and CESC data for tenfold CV, respectively. However, significant improvement in performance can be noted in the case of the D2CCA architecture as compared with the existing deep models, irrespective of the experimental setup and data sets considered. Statistical significance tests demonstrate that out of a total of 90 cases, the proposed model achieves significantly better p-values for 72 cases, and better but not significant p-values for 16 cases.

V. Conclusion

The major contribution of this article is fivefold, namely: 1) developing the MDDBM framework by incorporating supervised information into MDBM; 2) introducing D2CCA architecture based on the theory of CCA and MDDBM; 3) consolidating the theory of the proposed framework with convergence analysis; 4) demonstrating the effectiveness of the proposed architecture as feature extractor as well as classifier; and 5) illustrating the proficiency of the proposed method on different domains of applications, namely, object recognition, document classification, multilingual categorization, face recognition, and cancer subtype identification.

The learning objective of the proposed architecture includes the merits of deep Boltzmann machines. Incorporating supervised information into the objective function enhances the discriminative ability of the joint representation of the model, which in turn, entitles the architecture to serve as the feature extractor as well as the classifier. In order to capture the nonlinear correlated structures across multiple modalities, the theory of CCA is introduced to the learning objective of the proposed method. Theoretical analysis ensures the convergence of the proposed method to an asymptotically stable point, provided given sufficient conditions are met. The efficacy of the proposed architecture is established on several multi-view datasets. Significant improvement in performance is noticed in the case of the proposed model as compared with the existing approaches for both training–testing and tenfold CV.

The proposed framework is developed primarily based on the consensus principle. However, it has been shown in [23], that the complementary knowledge among different modalities may also contain useful information, which may essentially facilitate accurate classification of given observations into different categories. Hence, view discrepancy can be an important aspect that needs to be taken into consideration to develop a deep framework for multi-view data analysis. Another future direction can possibly be the estimation of the data-specific architecture of the proposed model instead of considering a uniform architecture for all the given databases. Although extensive experimentation is carried out on the datasets in order to obtain the uniform architecture of the D2CCA framework, a tradeoff has been made between the classification accuracy achieved on different databases. So, it is necessary to determine the optimal number of layers and the number of hidden nodes at a particular layer for each dataset in such a way that the challenges offered by different databases can be efficiently characterized.

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

REFERENCES

[1] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

[2] C. Feng, Y. Xu, J. Liu, Y. Gao, and C. Zheng, "Supervised discriminative sparse PCA for com-characteristic gene selection and tumor classification on multiview biological data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 10, pp. 2926–2937, Oct. 2019.

[3] C. Huang, H. Ai, Y. Li, and S. Lao, "High-performance rotation invariant multiview face detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 671–686, Apr. 2007.

[4] J. Han, H. Chen, N. Liu, C. Yan, and X. Li, "CNNs-based RGB-D saliency detection via cross-view transfer and multiview fusion," *IEEE Trans. Cybern.*, vol. 48, no. 11, pp. 3171–3183, Nov. 2018.

[5] A. Tenenhaus and M. Tenenhaus, "Regularized generalized canonical correlation analysis," *Psychometrika*, vol. 76, no. 2, pp. 257–284, Apr. 2011.

[6] J. R. Kettenring, "Canonical analysis of several sets of variables," *Biometrika*, vol. 58, no. 3, pp. 433–451, 1971.

[7] J. Chen, G. Wang, and G. B. Giannakis, "Graph multiview canonical correlation analysis," *IEEE Trans. Signal Process.*, vol. 67, no. 11, pp. 2826–2838, Jun. 2019.

[8] X. Fu et al., "Efficient and distributed algorithms for large-scale generalized canonical correlations analysis," in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, Dec. 2016, pp. 871–876.

[9] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," in *Proc. 12th Eur. Conf. Comput. Vis.*, 2012, pp. 808–821.

[10] X. You, J. Xu, W. Yuan, X.-Y. Jing, D. Tao, and T. Zhang, "Multi-view common component discriminant analysis for cross-view classification," *Pattern Recognit.*, vol. 92, pp. 37–51, Aug. 2019.

[11] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen, "Multi-view discriminant analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 188–194, Jan. 2016.

[12] K. Somandepalli, N. Kumar, R. Travadi, and S. Narayanan, "Multimodal representation learning using deep multiset canonical correlation," 2019, *arXiv:1904.01775*.

[13] H. D. Couture, R. Kwitt, J. S. Marron, M. Troester, C. M. Perou, and M. Niethammer, "Deep multi-view learning via task-optimal CCA," 2019, *arXiv:1907.07739*.

[14] N. Srivastava and R. R. Salakhutdinov, "Multimodal learning with deep Boltzmann machines," *J. Mach. Learn. Res.*, vol. 15, pp. 2949–2980, Oct. 2014.

[15] W. Fan et al., "Deep adversarial canonical correlation analysis," in *Proc. SIAM Int. Conf. Data Mining*, 2020, pp. 352–360.

[16] K. G. Toker and S. E. Yüksel, "Deep canonical correlation analysis for hyperspectral image classification," *Proc. SPIE*, vol. 11150, Oct. 2019, Art. no. 1115009.

[17] M. Dorfer and G. Widmer, "Towards deep and discriminative canonical correlation analysis," in *Proc. ICML Workshop Multi-View Represent. Learn.*, 2016, pp. 1–5.

[18] X. Yang, W. Liu, and W. Liu, "Tensor canonical correlation analysis networks for multi-view remote sensing scene recognition," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 6, pp. 2948–2961, Jun. 2022.

[19] S. Rastegar, M. S. Baghshah, H. R. Rabiee, and S. M. Shojaee, "MDL-CW: A multimodal deep learning framework with CrossWeights," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2601–2609.

[20] D. Gao, K. Li, R. Wang, S. Shan, and X. Chen, "Multi-modal graph neural network for joint reasoning on vision and scene text," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12746–12756.

[21] Q. Xuan, Z. Chen, Y. Liu, H. Huang, G. Bao, and D. Zhang, "Multiview generative adversarial network and its application in pearl classification," *IEEE Trans. Ind. Electron.*, vol. 66, no. 10, pp. 8244–8252, Oct. 2019.

[22] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, nos. 3–4, pp. 321–377, Dec. 1936.

[23] J. Yin and S. Sun, "Multiview uncorrelated locality preserving projection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3442–3455, Sep. 2020.

[24] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," in *Proc. Artif. Intell. Statist.*, 2009, pp. 448–455.

[25] R. M. Neal and G. E. Hinton, "A view of the EM algorithm that justifies incremental, sparse, and other variants," in *Learning in Graphical Models*. MIT Press, 1998, pp. 355–368.

[26] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, vol. 68. New York, NY, USA: Wiley, 1991, pp. 69–73.

[27] T. Tieleman, "Training restricted Boltzmann machines using approximations to the likelihood gradient," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 1064–1071.

[28] J. R. Movellan, "Contrastive Hebbian learning in the continuous Hopfield model," in *Connectionist Models*. Morgan Kaufmann, 1991, pp. 10–17.

[29] C. Peterson and J. R. Anderson, "A mean field theory learning algorithm for neural networks," *Complex Syst.*, vol. 1, no. 5, pp. 995–1019, 1987.

[30] L. Younes, "On the convergence of Markovian stochastic algorithms with rapidly decreasing ergodicity rates," *Stochastics, Int. J. Probab. Stochastic Processes*, vol. 65, nos. 3–4, pp. 177–228, Feb. 1999.

[31] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 4–37, Jan. 2000.

[32] R. Rossi and N. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proc. AAAI Conf. Artif. Intell.*, 2015, pp. 4292–4293.

[33] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng, "NUS-WIDE: A real-world web image database from National University of Singapore," in *Proc. ACM Int. Conf. Image Video Retr.*, 2009, pp. 1–9.

[34] M. R. Amini, N. Usunier, and C. Goutte, "Learning from multiple partially observed views—An application to multilingual text categorization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 22, 2009, pp. 28–36.

[35] J. Yang, D. Zhang, A. F. Frangi, and J.-Y. Yang, "Two-dimensional PCA: A new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, Jan. 2004.

[36] K. Tomczak, P. Czerwińska, and M. Wiznerowicz, "The cancer genome atlas (TCGA): An immeasurable source of knowledge," *Contemp. Oncol.*, vol. 19, no. 1A, p. A68, 2015.

[37] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," in *Proc. 12th Int. Conf. Artif. Intell. Statist.*, 2009, pp. 448–455.

[38] P. Maji, "A rough hypercuboid approach for feature selection in approximation spaces," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 1, pp. 16–29, Jan. 2014.

[39] N. Srivastava and R. Salakhutdinov, "Multimodal learning with deep Boltzmann machines," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2222–2230.

Debamita Kumar received the B.Tech. degree in electronics and communication engineering from the Maulana Abul Kalam Azad University of Technology, Kolkata, India, in 2011, and the M.Tech. degree in electronics and telecommunication engineering from the Indian Institute of Engineering Science and Technology, Shibpur, India, in 2016.

She is currently a Research Fellow with the Machine Intelligence Unit, Indian Statistical Institute, Kolkata. She has authored or coauthored a few articles in international journals and conferences. Her research interests include pattern recognition, machine learning, deep learning, and medical imaging.

Pradipta Maji (Senior Member, IEEE) received the Ph.D. degree from Jadavpur University, Kolkata, India, in 2005.

He is currently a Professor with the Machine Intelligence Unit, Indian Statistical Institute, Kolkata. He has authored or coauthored more than 150 articles in international journals and conferences. His research interests include machine learning, pattern recognition, computer vision, medical imaging, computational biology, and bioinformatics.

Dr. Maji is a fellow of the National Academy of Sciences, India. He received the 2008 Microsoft Young Faculty Award from the Microsoft Research Laboratory India Private, Ltd., the 2009 Young Scientist Award from the National Academy of Sciences, the 2011 Young Scientist Award from the Indian National Science Academy, India, and the 2015 Young Faculty Research Fellowship from the Ministry of Electronics and Information Technology, Government of India. He has been selected as the 2009 Young Associate of the Indian Academy of Sciences, India.

Authorized licensed use limited to: Indian Statistical Institute Kolkata. Downloaded on June 03,2023 at 07:37:36 UTC from IEEE Xplore. Restrictions apply.