# Discriminative Deep Joint Subspace Analysis for Multi-View Data:

## Correlation, Dependency to Spatial Proximity

A thesis submitted to Indian Statistical Institute
in partial fulfillment of the requirements for the degree of

**Doctor of Philosophy in Computer Science**

by

**Debamita Kumar**
Senior Research Fellow

Under the supervision of
**Dr. Pradipta Maji**, Professor


**Machine Intelligence Unit**
**Indian Statistical Institute, Kolkata**

**December 2023**

*You must believe.*
*-Oogway*

# Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisor Prof. Pradipta Maji for his support, advice, and guidance throughout my tenure. No word of thanks is enough to express my gratefulness. I acknowledge my gratitude to the Dean of Studies and the Director of the Indian Statistical Institute for providing me the research fellowship, grants, and infrastructure for research. I express my sense of indebtedness to all the faculty members of the Machine Intelligence Unit, Indian Statistical Institute, for their constant encouragement and motivation. I would also like to acknowledge the office personnel at the institute for all the prompt assistance that I have received throughout my tenure.

I express my gratitude to bardimuni, my beloved childhood teacher, for all the blessings you bestowed upon me and my sister. I sincerely hope and pray you get well soon and we all can have dinner together again. I would like to extend my profound thanks to all my Biomedical Imaging and Bioinformatics Lab members and alumni. I can not thank Suman and Ankita enough for the love and support that I have received from them over these years. I am deeply indebted to my didi, jiju, and chomchom for being in my life. I would also take the opportunity to thank Sayak and Sridatri Acharjya, my husband and daughter, for keeping my life so beautifully engaged. I am incomplete without my maa and baba. I am and will always be your daughter first.

Debamita Kumar 08.12.23

Debamita Kumar

# Abstract

The multi-view classification is an important machine learning paradigm, which explores the consistency and complementary properties of multiple views to discover patterns hidden in data. Although the integration of multiple views is expected to provide an intrinsically more powerful predictive model than its single-view counterpart, it poses its own set of challenges. The most important problems associated with multi-view classification are the handling heterogeneous nature of different views, selection of relevant and complementary views while generating discriminative joint subspaces for analysis, and capturing the lower dimensional non-linear geometry of each view. In a multi-view scenario, it is expected that the joint subspace should be learned in such a way that the similarity in the latent space implies the similarity in the corresponding concepts. The joint subspace should also reflect the intrinsic properties of each of the individual views and should efficiently capture the non-linear correlated structures across different views. Moreover, if some of the views correspond to images, the joint subspace should preserve the innate topological properties of the image views properly.

In this regard, the main contribution of the thesis is to develop some multi-view predictive models for the classification of observations into different concepts or classes. In order to evaluate the relevance of a particular view, a novel framework is introduced, by judiciously integrating the merits of rough sets and Bayes decision theory. While the former deals with the uncertainty due to inexactness, vagueness, and incompleteness in class definition, the latter addresses the uncertainty due to overlapping class boundaries. To learn a joint subspace that can efficiently encapsulate the underlying non-linear data distribution of the given observations, a multi-view deep predictive model, termed as multimodal discriminative deep Boltzmann machine (MDDBM), is introduced, based on the framework of deep Boltzmann machine. The supervised information of multi-view data is incorporated into the proposed deep architecture through the class nodes to make the joint subspace discriminative in nature. The theory of canonical correlation analysis is judiciously integrated with the learning objective of the proposed MDDBM to learn a joint subspace from the maximally correlated subspaces, while the concept of Hilbert-Schmidt independence criterion helps to encapsulate the cross-view dependency in terms of consensus and/or complementary knowledge from the input pairs of views. Based on the Bayes error analysis, an upper bound on the error probability of the proposed deep model is estimated, which facilitates determining the optimal architecture of the proposed model.

Finally, a deep predictive model is proposed to capture the spatial proximity within the image views, when both image and non-image views are available for the analysis. In order to recognize and represent the geometric structures of the image manifolds,

embedded in the ambient space, the theory of Laplacian eigenmap and the concept of simultaneous diagonalization of Laplacians are judiciously integrated with the learning objective of the MDDBM. In effect, the proposed deep predictive model can generate a joint discriminative subspace from the image and non-image views.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

In the era of digitalization, an exponential growth of data can be noted in numerous domains of applications. For example, there are about 1 trillion web pages; an average of one hour of video is uploaded to YouTube every second, amounting to 10 years of content every day; the genomes of thousands of people, each of which has a length of $3.8 \times 10^9$ base pairs, have been sequenced by various labs; Walmart handles more than 1M transactions per hour and has databases containing more than 2.5 petabytes ($2.5 \times 10^{15}$) of information [32]; and so on. This deluge of data calls for *data analysis* [66], which refers to the process of understanding and recognizing the interesting and non-trivial patterns properly from the given input data. A pattern refers to a portion of the data that repeats itself in a discernible way. It provides the information of data.

*Pattern recognition* is an automated process of exploring patterns and irregularities in the data [189]. The problem of searching for patterns in data is a fundamental one and has a long and successful history. For instance, the extensive astronomical observations of Tycho Brahe in the 16 th century allowed Johannes Kepler to discover the empirical laws of planetary motion, which in turn, provided a springboard for the development of classical mechanics [18]. The field of pattern recognition is concerned with the automatic discovery of regularities in data through the use of computer algorithms. This is where artificial intelligence comes into play. In particular, *artificial intelligence (AI)* can be defined as a systematic study and design of algorithms to automatically detect patterns in data, and then use the uncovered patterns to predict future data, or perform other kinds of decision making under uncertainty [138].

Nowadays, AI is a thriving field with many practical applications and active research topics. Many of the early successes of AI took place in relatively sterile and formal environments and did not require computers to have much knowledge about the world. It solved problems that are intellectually difficult for human beings but relatively straight-forward for computers, that is, problems that can be described by a list of formal, mathematical rules [57]. For example, IBM's Deep Blue chess-playing system defeated world champion Garry Kasparov in 1997. Chess is defined in relatively simple world, containing only sixty-four locations and thirty-two pieces that can move in only rigidly circumscribed ways. It can be completely described by a brief list of formal rules, easily provided ahead of time by the programmer. The true challenge of AI is to solve the tasks that are easy for people to perform but hard to describe formally, that is, problems that human beings solve intu-

itively, like recognizing spoken words or faces in images. A person's everyday life requires an immense amount of knowledge about the world. Much of this knowledge is subjective and intuitive, and therefore, difficult to articulate in a formal way. The key challenge in AI is how to get this informal knowledge into a computer.

One of the possible approach to solve the difficulties faced by the AI suggests that the systems need the ability to acquire their own knowledge, by extracting patterns from raw data. This capability is known as *machine learning*. It is a branch of AI that focuses on the use of data and algorithms to imitate the way humans learn; gradually improving its accuracy. The introduction of machine learning allowed computers to tackle problems involving knowledge of the real world and make decisions that appear subjective. Based on the learning strategy, pattern recognition and machine learning algorithms fall into three primary categories, which include supervised, unsupervised, and semi-supervised learning.

In the *predictive* or *supervised learning* approach, a two-stage methodology is adopted for identifying the patterns from the input data. The first stage includes the development of a model, which is termed as *training* phase, and the second stage involves the prediction of new or unseen data based on the developed model, which is referred to as *testing* phase. In the training phase, the goal is to learn a mapping from input $\mathbf{x}$ to output $y$, given a labelled set of input-output pairs $\mathcal{B} = \{\mathbf{x}_i, y_i\}_{i=1}^{N}$. Here, $\mathcal{B}$ denotes the training set and $N$ symbolizes the number of training examples. Each training input $\mathbf{x}_i$ is a $B$-dimensional vector of numbers, representing, say, the height and weight of a person. These are called *features*, *attributes* or *covariates*. The *output*, *target* or *response variable* is primarily a categorical or nominal variable from some finite set, $y_i \in \{1, \cdots, C\}$, or a real-valued scalar. When $y_i$ is categorical, the problem is known as *classification*, and when $y_i$ is real-valued, the problem is known as *regression*.

In the *descriptive* or *unsupervised learning* approach, only the input $\mathcal{B} = \{\mathbf{x}_i\}_{i=1}^{N}$ is given, and the goal is to find interesting non-trivial patterns in the data. This is a less well-defined problem, since it is not told what kinds of patterns to look for. The machine should learn to categorize the data based on the similarity in the patterns, present in the data. *Clustering* is an unsupervised technique where the goal is to find natural groups or clusters by interpreting the input data. The *semi-supervised learning* is less commonly used. Here, a combination of a small amount of labelled data and a large amount of unlabelled data is used to learn the model. An initial model is developed based on the limited labelled training data and then unlabelled data is used to refine the model.

Classification is a fundamental problem in pattern recognition and machine learning. The conventional supervised learning algorithms include Bayesian network [214], logistic regression [90], decision tree [98], random forest [13], support vector machine [197], nearest neighbors [38], and so on. The performance of these supervised learning algorithms depends heavily on how the input data is represented in the feature space. Much of the pattern recognition tasks can be solved by constructing the right set of features to extract for the task and then providing these features to the input of the machine learning algorithms. However, it is difficult to apprehend the features that need to be extracted from the input data for the accomplishment of a certain task. One possible solution is to use machine learning algorithms to discover not only the mapping from the feature representation to output but also the representation itself. This approach is known as *representation learning*. Learned representations often result in much better performance as compared to the hand-crafted counterparts, which allows the AI systems to rapidly adapt to the given tasks with

minimal human intervention.

The quintessential example of a representation learning algorithm is the *artificial neural network*, commonly referred to as *neural network*. It provides a powerful alternative to the conventional techniques of supervised machine learning which are often limited the assumptions of normality, linearity, variable independence, and so on. The inception of neural networks is motivated by the fact that the human brain computes in an entirely different way than the existing machine learning algorithms. The brain is a highly complex, non-linear information processing unit, which has the capability to organize its structural constituents, termed as *neurons*, to perform certain tasks. For example, human vision is an information processing task. The brain routinely accomplishes this task many times faster than the fastest digital computer in existence today. A neural network is designed to model the way in which the brain performs a particular task or function of interest. The formal definition of a neural network is presented in [4], which is as follows:

- *A neural network is a massively parallel distributed processor made up of simple processing units, which has a natural propensity of storing experiential knowledge and making it available for use. It resembles the brain in two respects:*

    1. *Knowledge is acquired by the network from its environment through a learning process.*
    2. *Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge.*

The procedure to perform the learning process is termed as the *learning algorithm* of the network. The objective of a learning algorithm is to modify the synaptic weights, or simply weights, of the network in an orderly fashion to achieve the desired objective. Since a neural network is comprised of non-linear neurons, it itself is non-linear. Non-linearity is an extremely important property, particularly if the underlying data distribution is inherently non-linear. Adaptability of the synaptic weights based on the environment is another essential property of neural networks.

The primary focus of neural networks is to disentangle the *factors of variation* that explain the observed data. In this context, the term *factors* refers to quantities that are not directly observed but they may exist as forces in the physical world that affect observable quantities. They provide useful simplifying explanations or inferred causes of the observed data. Factors can be regarded as concepts or abstractions that help to comprehend the rich variability in the data. For example, the factors of variation for analyzing a speech recording include the speaker's age, gender, accent, and the words that are been spoken. In case of complex problems, it becomes nearly impossible for the neural networks to identify such high-level, abstract factors of variation from the raw data.

One of the possible approach to overcome this situation is to define the surrounding world in terms of a hierarchy of concepts, with each concept defined in terms of its relation to simpler concepts. The hierarchy of concepts allows the network to learn complicated concepts by building them out of simpler ones. If a graph is drawn showing how these concepts are built on top of each other, then the graph will be deep, having many layers. Hence, this learning strategy is referred to as *deep learning*. Deep learning models resolve the difficulty of learning the desired complicated mapping by breaking it into a series of nested simple mappings, each described by a different layer of the model. The input is

presented at the *visible layer*, so named because it contains the observed variables. Then, a stack of *hidden layers* extracts increasingly abstract features from the given data. These layers are termed as hidden because the corresponding representations are not given in the data. The model needs to determine which concepts are useful for explaining the relationships in the observed data.

Deep learning has a long and rich history. Although the term *deep learning* appears to be relatively nascent, the field dates back to the 1950s. It has been abandoned many times for several years preceding its current popularity and has gone through many different names reflecting different philosophical viewpoints. In the present scenario, deep learning has evolved to be more effective as the amount of available training data has increased over time. Due to the development of acquisition equipments and sensors, a large amount of data has become more accessible nowadays. However, the unpredictably ambiguous nature of the data, along with the incompleteness in data representation, has restricted the performance of deep learning models [119]. In real world applications, data are collected from different sources of diverse domains. So, multiple representations of the same data are available, which are also referred to as different *modalities* or *views* of the data. It is primarily assumed that judicious integration of different views may provide comprehensive and descriptive description of the inherent characteristics of the input data.

## 1.1 Multi-View Data Analysis

*Multi-view learning* is an emerging machine learning paradigm that focuses on discovering patterns in data represented by multiple distinct views [182]. Data can naturally be described in terms of multiple views. For example, a webpage can be described by the words appearing on the page itself and the words underlying the links pointing to the webpage from other pages [19]. In natural language processing tasks, the same document can have multiple representations in different languages [7]. Although each of the individual views can characterize the input data, multiple views often provide information distinct to each other, which can alleviate the difficulty of describing the intrinsic properties of the given data. A naive solution concatenates all the views to obtain a single data matrix, which can then be applied to single view learning algorithms. However, it becomes difficult to reflect the individual statistical properties of each of the modalities in the unified representation. In recent years, several methods of integrating imperative information from multiple views have been developed in numerous domains of applications, such as integration of multi-omics data, biomedical imaging, multilingual categorization, multi-camera face and facial expression recognition, and so on [156, 225]. The enormous success of the multi-view learning can be explained through several arguments. The significant ones are discussed next.

1. **Complete Perspective:** Each view of the given multi-view data has a fundamentally distinct representation of the underlying data distribution. Hence, it can be primarily assumed that integration of different views may provide comprehensive and inclusive description of the inherent characteristics of the input data.

2. **Complementary Information:** Different views of the given data may provide some knowledge, which is distinct from the rest of the views. Therefore, the underlying

complementary information of different views can be suitably exploited to learn the diverse knowledge regarding the latent inherent structure of the data.

3. **Cross-Platform Analysis:** Due to the presence of multiple views of the given input data, it is possible to relate the observed variables from different views. For example, combining the information of functional magnetic resonance images ($f$MRI) with the single nucleotide polymorphism (SNP) data makes it easier to locate the changes in the brain regions that are caused by the associated SNP changes in the genes.

4. **Resilience to Noise:** In real-life applications, data is usually corrupted by noise. However, if information from multiple views is integrated, then it is most likely that a noisy data point in one view is indemnified by the corresponding data points of other views.

Despite these benefits, there are number of difficulties associated with the multi-view data analysis, some of which are discussed in the following section.

## 1.2 Challenges in Multi-View Data Analysis

The conventional supervised machine learning algorithms, such as nearest neighbors, support vector machine, decision trees, artificial neural networks, or even deep learning models, are primarily developed to process unimodal data. In order to adapt to multi-view environment, few modifications are required to be performed on these approaches. Hence, recognizing the inherent structures of data from the given multiple views offers its own set of challenges. The pivotal ones are discussed next.

1. **Joint Subspace:** For multi-view data analysis, multiple view-specific linear or non-linear mapping functions are required to be sought in order to transform the data from input spaces to a joint subspace. The resultant joint subspace should embody the inherent characteristics of each of the input views. In order to address the classification problem, supervised information of sample categories should be incorporated in the joint subspace. The consistency and diversity present among different views are also need to be reflected in the corresponding subspace. Since each view is characterized by different statistical properties, representing all the imperative information in the joint subspace is a difficult task.

2. **Latent Distribution:** In multi-view learning, it is essential that the underlying non-linear data distribution of the given observations is efficiently encapsulated in the joint subspace. Since the views are assumed to be generated from the latent distribution, the joint subspace is expected to characterize and classify the given data accurately. However, different views of the data provide different representations of the underlying data distribution.

3. **Heterogeneous Nature of Data:** A naive solution of multi-view data analysis concatenates all the views to obtain a single data matrix, which can then be applied to single view learning algorithms. However, the direct concatenation approach is not always meaningful since different views are generally captured at different scales.

The data in each view is represented with distinct unit, having unique variance. For example, in real-life cancer data sets, the $\beta$-values of DNA methylation data usually lie in $[0, 1]$, whereas the RNA sequence-based gene expression data is quantified in terms of RPM (reads per million), having real values in the order of $10^5$. The concatenation of such diverse views is essentially dominated by the view with higher variance value.

4. **Heterogeneity in Views:** In multi-view learning, it may so happen that the individual views correspond to completely different spaces. For example, in imaging genetics, one view may correspond to multidimensional fMRI, while the other view refers to the one-dimensional SNP array. Similarly, in image annotation and image retrieval tasks, the views are generally represented by natural images and their corresponding description or tags. The set of functions required to model the one-dimensional text modality will essentially be different from the set of functions required to model the image modality. In such a scenario, learning the joint subspace from the individual spaces may not be able to capture the cross-modal information.

5. **Irrelevant and Redundant Views**: In real-life applications, the observations in the given input views are usually prone to noise, generating from the measurement errors. In absence of proper pre-processing procedure, the noise propagates or even gets amplified during the data integration process. Based on the assumption that each view provides some information which is fundamentally distinct from the information provided by the rest of the views, all the input views are generally considered for multi-view data analysis. However, presence of redundant and irrelevant views may affect the overall performance of the corresponding approaches.

6. **Class Evolution:** The supervised machine learning algorithms usually address the multi-view classification problem, where the class labels of the given observations as well as the total number of input classes are known a priori. However, there might be cases where the number of classes is not known beforehand, although the input observations are provided with the corresponding class labels. In dynamic data streams, the *class evolution* occurs when an observation with a new class label comes in the data stream.

7. **Incomplete Views**: In multi-view data analysis, it is primarily assumed that all the views represent information corresponding to the same set of samples. However, in real-life scenario, the data on various views may get corrupted during the data collection or pre-processing phase. As a consequence, a particular set of samples may be present in one view, while the corresponding samples may not be observed in another view, which is referred to as *incomplete view*.

Certain challenges, such as, obtaining joint subspace, heterogeneity in views, and presence of irrelevant, redundant, and incomplete views, are inherent to multi-view data analysis only, whereas recognizing latent distribution and class evolution in dynamic data stream are pertinent to single view data analysis as well. From the discussion on the aforementioned challenges, it can be inferred that some advanced supervised machine learning algorithms are required to be developed in order to discover and analyze the intrinsic structures or patterns present in the multiple views of the given input data.

## 1.3 Challenges in Deep Learning

The goal of deep learning is to be able to comprehend high-dimensional data with rich structures, for example, natural images [226], audio waveforms representing speech [222], documents containing multiple words [89], and so on. Deep predictive models consider such high-dimensional data as input and summarize it with a categorical label, for example, what object is present in the photo, what word is spoken in the recording, or what topic the document is about. Although deep learning models have achieved a phenomenal success in almost every research field and application domains, there still exists certain aspects which limit the construction as well as the overall performance of deep learning models. Some of these challenges are discussed next.

1. **Determination of Architecture:** In deep learning, there does not exist any guiding principle or rule based on which the architecture of the model can be determined. The number of neurons at a particular layer or the number of layers in a deep model is either heuristically determined based on, for example, evolutionary algorithms, or existing networks, pre-trained on natural images, are considered whose parameters are fine-tuned based on the target database, except for the output layer of the model which needs to be trained from scratch. However, the architecture of the model, defined based on such aforementioned approaches, does not take into account the diversities present in the nature of the problem as well as the complexities associated with the given input data.

2. **Overfitting:** It is an undesirable behavior in which a deep learning model tries to fit the training data entirely and ends up memorizing the data patterns, noise, and random fluctuations. As a consequence, the model performs well on training data, but similar performance is not observed on unseen data. Thus, the model fails to generalize and tends to learn specific and low-level features that may not be relevant or useful for other domains or tasks. It prevents the model from leveraging the existing knowledge and skills to learn new or related ones. Overfitting occurs due to several reasons, such as:

   - the number of training samples is too small to reflect the data characteristics properly;
   - the training data contains irrelevant and redundant information;
   - the model is trained on same set of samples for a long period of time;
   - the model is highly complex and so, it learns the noise present in the input data;

   In effect, the overall performance of the deep learning model is degraded.

3. **Computationally Expensive:** Training and deploying deep learning models can be computationally very expensive and time-consuming as compared to the conventional machine learning algorithms, as they involve complex mathematical operations and multiple layers of neurons. Deep learning models necessitate the use of a large amount of computational resources, which include powerful central processing unit, high-performance graphics processing unit, storage, and so on. The amount of computational power, needed to efficiently train a deep learning model, depends on the depth and complexity of the model.

4. **Lack of Interpretability:** In deep learning, lack of interpretability is an important issue in which the outputs or decisions provided by the deep models are difficult to be explained. For this reason, deep learning models are often considered as black boxes. It is still difficult to analyze in deep models that how it is determined which feature to extract at which layer for proper modelling of the given input data. This can pose problem in applications that require transparency and accountability, such as medical image analysis.

5. **Security:** Deep learning models are susceptible to small input perturbations, in most cases imperceptible to the human eye. It is shown in [174] that small additive noise to an input image leads to misclassification by the model which was identified with 99.99% confidence earlier. Such perturbations, which can fool a trained model, are known as adversarial attacks. The study of adversarial attacks and robustness against them has become a great deal of research in deep learning.

Despite of having these challenges, deep learning models have vast inroads into many applications with outstanding performance. This is due to the fact that deep learning models achieve great power and flexibility by perceiving the environment as a nested hierarchy of concepts and representations, with each concept defined in relation to simpler concepts. It is possible for the deep models to perform tasks which is more expensive than the classification task. Density estimation [122], denoising [115], missing value imputation [87], and sampling [80] are just to name a few. Because of the powerful feature abstraction ability, a surging interest is noted in recent years for combining information from multiple views using deep learning models. In this regard, the problem of developing deep predictive models for the analysis of multi-view data is addressed in this thesis.

## 1.4 Scope and Organization of Thesis

In this regard, the thesis focuses on devising a set of novel models in order to address the classification problem of multi-view data. The significant challenges of multi-view classification include heterogeneity in data representation, incompleteness in class definition, and overlapping class boundaries, which substantially degrade the performance of the classification algorithms. In multi-view environment, it is expected that a joint subspace is defined in such a way that it can efficiently encapsulate the latent non-linear data distribution of the given observations. The joint subspace should also contain the discriminative information so that the similarity in the latent space implies the similarity in the corresponding concepts. The consistency and diversity present among different views are important aspects of multi-view data analysis, which are required to be properly explored for efficient representation and recognition of the given observations. Furthermore, if one or more input views correspond to images, then joint subspace should be learned in such a way that the topological properties of the image views are properly preserved along with the inherent characteristics of the rest of the views. As a consequence, the interpretability or explainability of the proposed models can be enhanced in addressing the multi-view classification problem. The primary contribution of the thesis is to develop few novel models, which can appropriately characterize the given classes and obtain a joint representation, incorporating all the imperative information of the input views such that the given observations are

classified into multiple categories accurately. Integration of information from various views can be observed as example of data fusion technique.

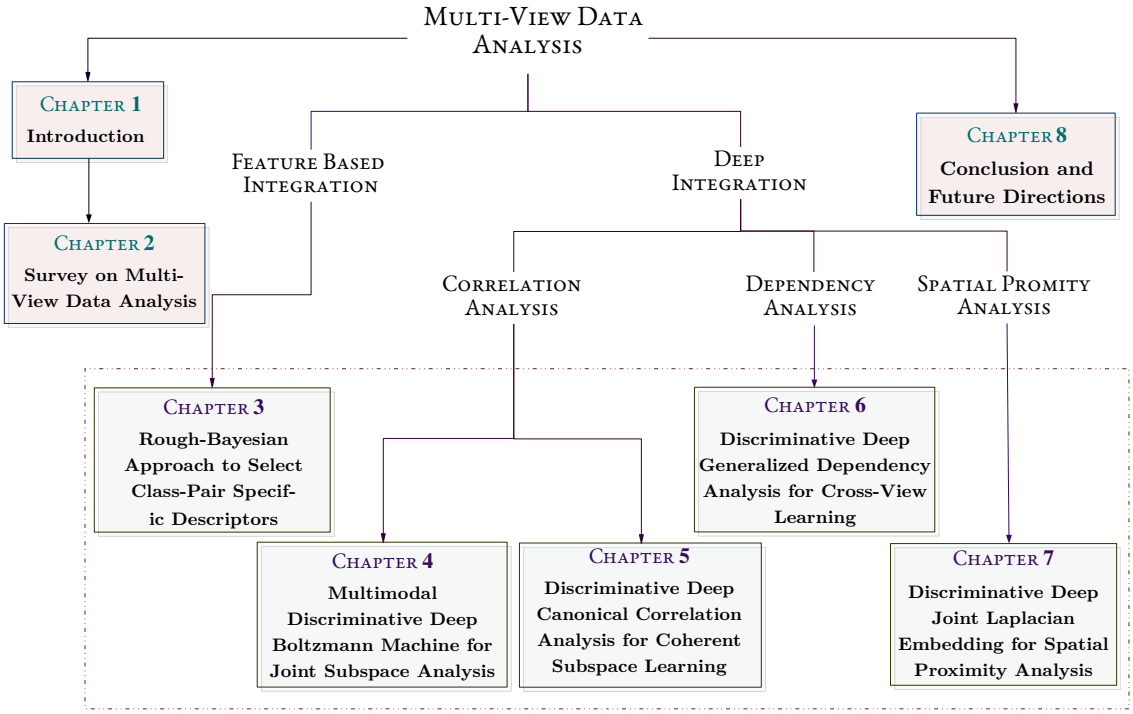



Figure 1.1: Outline of the thesis.

The outline of the thesis is depicted in Figure 1.1. It contains eight chapters. An introduction to multi-view data analysis and it's significance in the present scenario are discussed in Chapter 1. The major challenges associated with the multi-view learning are also described in this chapter. A concise study on state-of-the-art multi-view learning approaches is presented in Chapter 2.

The existing multi-view learning approaches consider a uniform set of descriptors for describing all the given input classes, which restricts the recognition rates of the corresponding approaches. In this context, Chapter 3 introduces a novel method, which is developed based on the hypothesis that a particular set of descriptors may be significant for differentiating observations belonging to a specific pair of classes, but may not be relevant for identifying observations from other pairs of classes. Also, it is assumed that all the features in a descriptor do not contribute uniformly in describing the inherent characteristics of the given observations. Therefore, a set of relevant descriptors is identified for representing the intrinsic properties of a particular pair of classes, and then the final feature set for multiple classes is formed from the relevant descriptors of all possible pairs of classes. Judiciously integrating the merits of rough sets and Bayes decision theory, a new framework, termed as Rough-Bayesian model, is introduced to evaluate the relevance of a descriptor in discriminating observations from a given pair of classes. While rough set theory deals

with the uncertainty due to the incompleteness in class definition, the probabilistic model addresses the uncertainty due to overlapping classes by measuring the belongingness of an observation to a specific class, based on some presumed intensity distribution of the class. During the computation of the relevance of each given descriptors, the proposed method takes care of the presence of both noisy features in a descriptors and noisy observations in an input class. Finally, support vector machine is used to predict the class labels of the given observations. The performance of the proposed method is studied with reference to several state-of-the-art approaches on a set of real-life human epithelial type 2 (HEp-2) cell image databases. An important finding is that the accuracy for classifying HEp-2 cell images is significantly increased if class-pair specific descriptors are considered, instead of selecting a uniform set of descriptors for multiple classes. Some of the results of this chapter are reported in [102, 103].

Although the classical approaches have achieved some promising results in numerous domains of applications, it is difficult for the hand-crafted features to effectively analyze the hidden attributes of the input data. On the other hand, features extracted by the deep models are data as well as task specific. Deep learning models can effectively learn complex, non-linear, and abstract representations of the given data by allowing multiple hierarchical layers. In this regard, Chapter 4 introduces a novel deep multi-view predictive model, termed as multimodal discriminative deep Boltzmann machine (MDDBM). The proposed deep model can extract discriminative and descriptive features from the given views, learn a joint subspace by integrating all the imperative information from the feature representations corresponding to each of the input views, and also classify the given observations into multiple categories. The proposed framework is developed based on the architecture of deep Boltzmann machine (DBM) in multi-view environment to encapsulate the underlying non-linear data distribution of the given observations. The class nodes are incorporated into the proposed deep architecture to include the supervised information at each layer of the network. Through proper learning of the weights associated with the corresponding class nodes, it can be ensured that the obtained representations at each layer of the network 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. The performance of the MDDBM model is extensively studied and compared with several state-of-the-art multi-view learning approaches on various benchmark and real-life cancer data sets. Some of the results of this chapter are reported in [104].

In multi-view environment, it may so happen that the individual views correspond to completely different sources. In such a scenario, the individual hidden representations correspond to essentially different spaces. So, learning the joint subspace from the individual spaces may not be able to capture the cross-modal information. However, if the proposed model is learned in such a way that the modality-specific subspaces are highly correlated, then the inherent characteristics of the views can be efficiently modeled by the joint subspace. In this regard, a novel architecture, termed as discriminative deep canonical correlation analysis (D2CCA), is proposed in Chapter 5, by judiciously integrating the merits of MDDBM, introduced in Chapter 4, and the theory of canonical correlation analysis (CCA). The weights of the network are updated such that the individual latent spaces are transformed to maximally correlated subspaces. Hence, the joint representation, learned

from the obtained subspaces, can efficiently capture the non-linear correlated structures across different modalities. Also, considering the class nodes in the architecture includes the supervised information of sample categories at each layer of the network, which in turn, allows the proposed D2CCA model to perform as feature extractor as well as classifier. Furthermore, the proposed framework is consolidated with corresponding convergence analysis. The proficiency of the D2CCA architecture is extensively studied and compared with numerous state-of-the-art methods on several benchmark and real-life cancer data sets. Some of the results of this chapter are reported in [104].

Along with the correlated structures, the complementary knowledge among different modalities may also contain useful information, which may essentially facilitate accurate classification of the given observations into different categories. In this regard, a novel deep learning model, termed as discriminative deep generalized dependency analysis (D2GDA), is proposed in Chapter 6 based on the MDDBM framework, introduced in Chapter 4. Thus, the proposed model can efficiently encapsulate the underlying non-linear data distribution of the given observations. Inclusion of supervised information at each layer of the network enhances the discriminative ability of the D2GDA model. Based on the concept of Hilbert-Schmidt independence criterion, a loss function is proposed to efficiently capture the cross-view dependency across several views. A view-pair specific constraint is incorporated in the loss function to extract the relevant cross-view information in terms of consensus and/or complementary knowledge from the given input pairs of views. Based on the Bayes error analysis, an upper bound on the error probability of the proposed deep model is estimated in terms of the model architecture. Hence, instead of heuristically determining the framework of the proposed model, an optimal architecture is estimated for each given database. While the number of layers is obtained from the total error probability of the model, the number of nodes at each layer is computed based on the Hilbert-Schmidt independence criterion. An analytic formulation demonstrates that the proposed model is the generalization of several state-of-the-art feature extraction techniques. The proposed approach is further consolidated with the convergence analysis. Finally, the proficiency of the proposed model is studied on numerous domain of applications, namely, object recognition, document classification, multilingual categorization, and cancer subtype identification. Some of the results of this chapter are reported in [105].

Combining information from multiple views is particularly challenging when the input views involve both image and non-image information. It is primarily due to the fact that as opposed to the non-image counterparts, the image modalities embody neighbourhood information which needs to be encapsulated properly for efficient representation of the given input data. In this regard, a deep predictive model, termed as discriminative deep joint Laplacian embedding (D2JLE), is developed in Chapter 7, which can process multiple image and non-image modalities simultaneously. For proper characterization of a particular image view, the corresponding manifold needs to be appropriately modelled from the given input images in such a way that moving into the latent space results into moving on the manifold. Hence, an objective function is developed based on the theory of Laplacian eigenmap, which can efficiently encapsulate the underlying low-dimensional embedding from the input high-dimensional pixel space. In case of multiple image views, the intrinsic geometric structures of the corresponding image manifolds need to be consolidate appropriately in the joint subspace. So, an objective function is formulated based on the concept of simultaneous diagonalization of Laplacians in which an approximate com-

mon eigenbasis is computed from the Laplacians simultaneously. In the proposed D2JLE model, the two objective functions are judiciously integrated with the learning objective of the MDDBM, introduced in Chapter 4, to capture the imperative information from both image and non-image views. The relevance of each view is evaluated based on the discrimination criterion of the corresponding view and the joint subspace is learned from the weighted combination of the individual subspaces. Based on the Bayes error analysis, an upper bound on the error probability of the proposed model is estimated in terms of the model architecture, which allows the model to determine the optimal architecture of the model for each database considered. The proposed D2JLE model is further consolidated with convergence analysis. The proficiency of the proposed model is demonstrated on several benchmark, HEp-2 cell image, and the real-life Virus databases with reference to several state-of-the-art multi-view learning approaches.

Finally, the concluding remarks are presented in Chapter 8 with discussions on future directions and improvements of the proposed research work.

# Chapter 2

# Survey on Multi-View Data Analysis

The basic notions of multi-view data analysis, along with a concise literature survey, are discussed in this chapter.

## 2.1 Multi-View Data Analysis

A multi-view data having $M$ views is represented by a set of $n$ samples, $\{x_1, x_2, \ldots, x_n\}$, where $M > 1$. Different views or modalities of the given input data can be characterized by various sets of descriptors. Hence, throughout the thesis, the terms *view*, *modalities*, and *descriptors* are interchangeably used, and multi-view data set is also referred to as a *multimodal data set*. The views are either represented by feature value based data or relational data. In case feature value based representation, a set of $M$ data matrices $X_1, X_2, \ldots, X_m, \ldots, X_M$ is considered to signify an $M$-view data set, where each data matrix corresponds to one of the $M$ input views. Hence, $X_m$ denotes a $(d_m \times n)$ matrix, where $d_m$ indicates the total number of input features for $m$-th view and $n$ refers to the number of samples, observed in a $d_m$-dimensional measurement space. The input data matrices are generally defined on Euclidean space, where $X_m$ is represented by numeric values in $\Re^{n \times d_m}$. However, other data types, such as, binary, categorical, and textual, are also observed. The measurement space, as well as the number of observed variables, $d_m$, need not be the same across different views. The matrices $X_1, \ldots, X_M$ may vary in terms of their scale, unit, variance, dimension (column-wise), and data distribution. In case of relational data, the $M$ views are typically represented by $M$ similarity (distance) matrices $W_1, W_2, \ldots, W_m, \ldots, W_M$. Each $W_m$ is a $(n \times n)$ matrix which is defined as $W_m = [w_m(i,j)]_{n \times n}$, where $w_m(i,j) \geqslant 0$ is the similarity (distance) between samples $x_i$ and $x_j$ in the $m$-th view.

Figure 2.1 shows an example of multimodal omics data set with feature vector based representation. The advent of whole genome sequencing technologies have led to the generation of different types of *omics* data from different levels of the genome. As shown in Figure 2.1, the DNA methylation, protein expression, gene expression, and copy number variation data can be observed from the epigenomic, proteomic levels of the genome, transcriptomic, and genomic, respectively. In a multimodal data set, these observations can be made for a common set of $n$ samples or patients whose genome is being sequenced. The

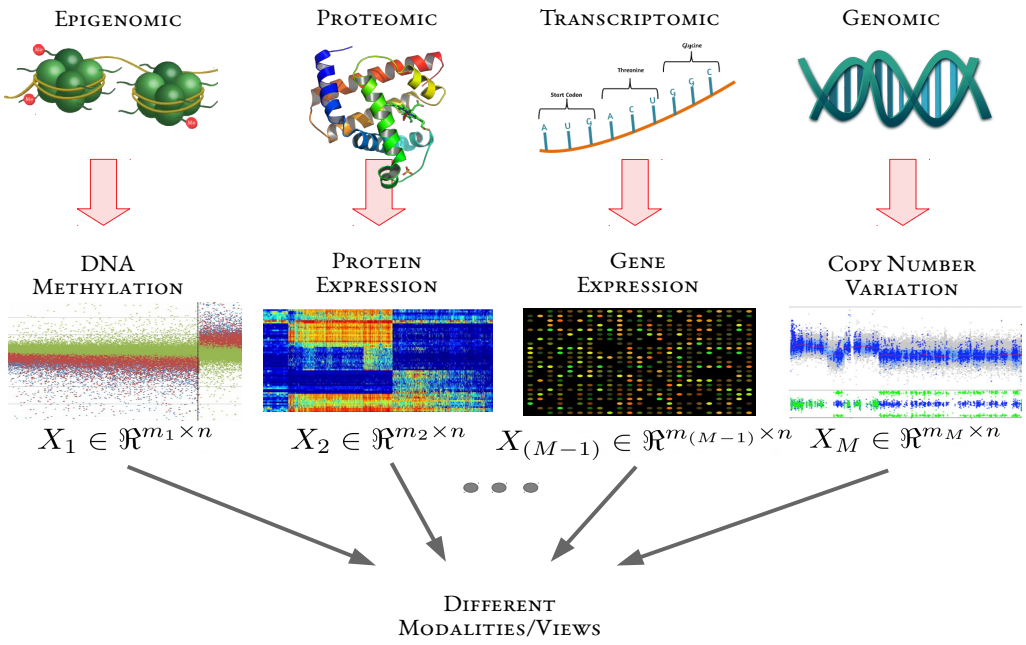



Figure 2.1: Different views of multi-omics data analysis.

resulting data set is a collection of $M$ views, denoted by $X_1, X_2, \ldots, X_m, \ldots, X_M$. Each $X_m$, in this case, is a $(n \times d_m)$ data matrix consisting of the expression levels of $d_m$ genes, or micro-RNAs, or proteins for those $n$ samples.

The area of multi-view learning is relatively nascent. However, due to the performance of multi-view learning in numerous domains of applications, a rich literature of machine learning research is developed over the past decade.

## 2.2 Multi-View Learning Approaches

The conventional machine learning algorithms, such as nearest neighbors, support vector machine, decision trees, artificial neural networks, or even deep learning models, are primarily developed to process unimodal data. 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 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 recent years, a surging interest is noted in combining information from multiple views. The existing algorithms can be approximately partitioned into five categories, namely, subspace learning, multiple kernel learning, co-training, embedding, and deep multi-view learning.

### 2.2.1 Subspace Learning

The main objective of subspace learning-based methods is to obtain a latent subspace shared by multiple modalities, where each input view can be generated from this latent subspace. The subspace learning effectively addresses the *curse of dimensionality* problem, as the latent subspace has lower dimensionality than that of any input view. Canonical correlation analysis (CCA) [78] finds linear relationships between two multidimensional views. It obtains two-directional weight vectors, also termed as basis vectors, and the empirical correlation between the respective projections onto these weight vectors is maximum. The CCA has been widely used in multi-view regression [91] and clustering [23] fields.

A generalization of Fisher's discriminant analysis has been proposed in [35] to explore the latent subspace spanned by a multimodal data set. This generalization is supervised although CCA does not incorporate the class information. Multi-view metric learning [155] has been developed to construct projections from multi-view data. The latent subspace is used to infer another view from the observation view. To establish the connections between the two views through latent subspaces, the Markov network [26], maximization of mutual information [131], and Gaussian process [173] have been used. In [165], a latent subspace is used to factorize private and shared information from different views. The main objective of factor analysis is to obtain latent factors, which summarize the input data. Partial least squares (PLS) [207] is another popular statistical technique that has been used to find fundamental relations between two views.

### 2.2.2 Multiple Kernel Learning

The main objective of multiple kernel learning (MKL) is to control the search space capacity of possible kernel matrices to achieve good generalization. The kernels in MKL correspond to different views, and the integration of different kernels may improve the learning performance. Thus, MKL is widely used to analyze multi-view data sets. Over the past few years, MKL has become one of the important techniques to analyze multi-view data sets. It achieves attention due to the utilization of various optimization techniques [107] as well as the recognization ability by exploring possible combinations of base kernels [227]. In [107], MKL has been formulated as a semi-definite programming problem. MKL is used to develop a dual formulation of the quadratically-constrained quadratic program as a second-order cone program problem in [11], where a sequential minimal optimization algorithm has been developed to efficiently obtain the optimal solution.

### 2.2.3 Co-training

Co-training [20] is one of the earliest models to analyze multimodal data. It learns alternately by maximizing the mutual correspondence between two unlabeled views. There are many modifications which have been done in the recent past. In [144], generalized expectation-maximization has been done, where adjustable probabilistic labels are assigned to unlabeled data. Some robust semi-supervised learning algorithms have been proposed in [139], where active learning is combined with co-training. In [219], Bayesian undirected graphical models are developed for co-training and a novel co-training kernel for Gaussian process classifiers. In [175], a co-regularization framework has been introduced where classi-

fiers are learned in each view through forms of multi-view regularization. Some co-training based multi-view clustering algorithms have been proposed in [100, 101].

### 2.2.4 Embedding

To overcome the representational differences, an alternative transformed representation has to be created for each view. An algorithm has been proposed in [110], where each view is projected into a homogeneous meta-space. The dimension of the projected space is the same for each view with the same scale. In [198], consensus embedding has been introduced, where according to the minimum predictive value, embeddings selected from different views have to be combined. The boosted embedding concatenation has been reported in [191], where supervised information is used in the fusion process. In [52], an algorithm of boosted embedding concatenation has been proposed based on the Adaboost classifier, which evaluates and provides weight on each embedding to integrate different views.

### 2.2.5 Deep Multi-View Learning

Due to the powerful feature extraction capability, deep learning methods have gained attention in recent years. By using multiple hierarchical layers, deep learning models can learn non-linear, subtle, complex, and abstract representations of the target data from multiple views. Several deep multi-view learning algorithms exist in the literature, such as multi-view convolutional neural network [109], multi-view auto-encoder [49], multi-view generative adversarial network [209], multi-view graph neural network [54], multi-view deep belief net [180], and multi-view recurrent neural network [129]. The conventional learning methods are also extended into the deep framework, such as, deep canonical correlation analysis [9], deep multi-view matrix factorization [224], deep multi-view spectral learning network [84], and deep multi-view information bottleneck [5].

- **Multi-View Convolutional Neural Network:** The convolutional neural network (CNN) [109] has demonstrated outstanding performance in numerous domains of application. As opposed to single-view CNN architectures, the multi-view CNN is defined as modeling from multiple feature sets with access to multi-view information of the target data. The multi-view CNN architecture aims to integrate imperative information from different views so as to obtain more discriminative common representations. The existing multi-view CNN architectures are usually partitioned into the following two types; one-view-one-net mechanism and multi-view-one-net mechanism. The multi-view CNN based on one-view-one-net mechanism adopts one convolutional neural network for each view and extracts feature representation corresponding to each view separately, then multiple representations are fused through subsequent part of the network. Multi-view-one-net mechanism feeds multi-view data into the same network to get the final representation. In essence, the difference between one-view-one-net mechanism and multi-view-one-net mechanism lies in the fusion methods of different views.

  CNN has been proven to be an effective way to extract high-level features from input data automatically. Many variants of the CNN model have been proposed, includ-

ing principal component analysis network (PCANet), canonical correlation analysis network (CCANet), multiple scale CCANet (MS-CCANet) and multiview CCANet (MCCANet). The PCANet is specialized for single view feature abstraction, while in many real-world practices, data are frequently observed from many more views. Although CCANet, MS-CCANet and MCCANet can be applied to two or more view data, they consider only the pair-wise correlation by calculating a series of two-order covariance matrices. However, the high-order consistence, which can only be explored by collectively and simultaneously examining all views, remains undiscovered. In this context, tensor canonical correlation analysis (TCCA) network is developed in [213]. Particularly, TCCA learns filter banks by simultaneously maximizing arbitrary number of views with high-order-correlation and solves the optimization problem by decomposing a covariance tensor. After the convolutional stage, binarization and block-wise histogram strategies are utilized to generate the final feature. In deep multiset CCA (dMCCA) [178], feed-forward networks have been used to map the given input modalities to a shared subspace such that the joint representation maximizes the ratio of between and within modality covariance of the given observations. Couture et al. [30] have developed task optimal CCA (TOCCA) that focuses on both CCA and task driven objectives using a deep architecture.

- **Multi-View Auto-Encoder:** Auto-encoder (AE) is a variation of neural network and has obtained promising results in various applications, such as data retrieval [49], human pose recovery [76], and disease analysis [223]. AEs are unsupervised feature learning method in the deep learning literature that consist of two objective functions; encoding function and decoding function. Specifically, the encoding function aims to map an input data to a compressed hidden representation. The decoding function aims to reconstruct the data from its compressed hidden representation. The hyper-parameters of AE architecture are obtained by minimizing the error of the reconstruction, which can be estimated by losses, for example, mean-squared loss. Some recent works are proposed to learn the common representation of multi-view data by using AEs. For example, Rastegar et al. [157] have proposed a multimodal deep learning framework, termed as MDL-CW, which exploits the cross-weights between modality-specific representations and gradually learns interactions between the given modalities through a deep network. In this way, the modality that contains higher level information can help to find a better representation for other modalities. It is also theoretically shown that intra-modality information can be captured by considering these inter-modality interactions.

  Inspired by MDL-CW, Feng et al. [49] present a correspondence auto-encoder (Corr-AE) to conduct cross-modal retrieval, which simultaneously learns the shared information of multiple modalities and the specific information in each individual modalities. The main idea of the Corr-AE is to minimize the correlation learning error between multiple modalities and the feature learning errors of each modality. The Corr-AE model is composed of the two subnetworks, each of which is a basic auto-encoder. The two subnetworks are combined by designing a code layer with a predefined similarity measurement. Feng et al. [49] also propose a full-modal version of Corr-AE, which can be regarded as an integration of standard auto-encoder and Corr-AE. The human pose recovery problem based on video has been addressed

in [76], where a multi-layered deep neural network has been used to construct low-rank hypergraph Laplacian. A discriminative margin-sensitive auto-encoder has been introduced in [223] to diagnose Alzheimer's disease and for recognition of protein folds accurately.

- **Multi-View Generative Adversarial Network:** As an unsupervised deep learning model, generative adversarial networks (GAN) [58] has been successfully applied into many domains and obtained promising results, such as image-to-image translation [85] and image in-painting [36]. Typically, the basic GAN includes a generative model $G$ and a discriminative model $D$. Thus, it has the most prominent character of adversarial training. The generative model $G$ characterizes the distribution of source data, while the discriminative model $D$ is designed to estimate the probability distribution from the training data. Generating images with multiple views from a single-view input is a fundamental research topic for broad applications in computer vision, robotics and graphics. The GAN-based multi-view generation methods first use encoder $E$ to map input images into latent space $Z$, then decoder $G$ is adopted to generate novel views.

  Based on the existing generative adversarial network (GAN) paradigm, multi-view GAN (MVGAN) is developed in [209] to handle the uncertainties associated with multiple views. At first, the MVGAN framework provides a principle to model the output distribution based on the input observations of two views by representing the hidden nodes with conditional probabilities. It also provide implicit mappings from input space to latent space and from latent space to output space. A natural way to extend the concept for multiple views is to define the mapping function from several views to single representation space. However, it is experimentally shown that the resulting model exhibits undesirable behaviours. Therefore, the MVGAN model proposes a constrain based on the idea that adding one more view to any subset of views must decrease the uncertainty on the output distribution, that is, the more views are provided, the less variance the output distribution has. This behaviour is encouraged by using a Kullback-Leibler divergence regularization.

- **Multi-View Graph Neural Network:** Graph neural networks (GNN) [167] reconciles the expressive power of graphs in modeling interactions with deep models in terms of learning representation and has gained increasing attention due to its capability of modeling graph structured data. They process variable-size permutation-invariant graphs and learn low-dimensional representations through an iterative process of transferring, transforming and aggregating the representations from topological neighbors. Recently, GNN has achieved outstanding performance in graph-structured data analysis, such as social network [82] and knowledge graphs [64].

  Recently, GNN also has achieved promising performance in the scenario of multi-view learning, such as multi-view graph conventional networks and multi-graph clustering. In multimodal graph neural network (MMGNN) [54], an input data is represented as a graph, consisting of three sub-graphs. Then, three aggregators are introduced which guide the message passing from one graph to another to refine the nodes of the network. The initial representations of the nodes in the three sub-graphs are obtained from priors, learned from the deep convolution neural networks. The up-

dated representation are considered to contain richer and more precise information of the given input data, facilitating the predictive model to provide accurate prediction of the class label information. In [43], a novel task-guided multi-view graph auto-encoder clustering framework has been reported, which can learn node embeddings by applying the content information. To analyze the global poverty problem, a graph structure based on the convolutional network has been proposed [97]. This model can be applied to predict whether a person is living below the poverty line, to predict the adoption of economic inclusion, or to predict the gender of mobile phone subscribers. A redesigned graph neural network, collaborated with a convolutional neural network, has been introduced in [210], to obtain a feature representation of multi-view images.

- **Multi-View Deep Belief Net:** The deep belief net (DBN) is proposed by Geoffrey Hinton et al. [71], which adopts the restricted Boltzmann machine (RBM) as its fundamental component. RBM is a simple model, which just contains two layers; a hidden layer and a visible layer. These two layers are connected by synaptic weights. The DBN architecture is composed of several RBMs by stacking each RBM on top of each other. The main aim of DBN is to capture the underlying data distribution of the observations. To learn the multi-view representations, Srivastava et al. [180] present a multi-view DBM based on generative model, which can model the joint distributions of various data sources, such as audio, image and text. Taking the image and text modality as an example, Srivastava et al. [180] first design different DBN models to extract the high-abstract representations of different modalities. Then, a top RBM with one-layer neural network is adopted to characterize the joint representation of multi-view data by passing the combined features of each individual modality. A hybrid model based on RBM has been reported in [6], and cross-modality as well as inter-modality features are extracted to detect the sequential event. A multi-view face recognition approach has been proposed in [3] based on DBN to capture the complementary representation of deep and local features. Another multimodal deep Boltzmann machine algorithm has been proposed in [184], where several patient phenotypes and gene expression data are processed simultaneously to identify the importance of different genes.

- **Multi-View Recurrent Neural Network:** For dealing with time series data, Sutskever et al. propose a recurrent neural network (RNN) [183], which has been successfully applied into various analysis tasks on time series data. The RNN model consists of three kinds of layers in different time frames, which are the input word layer, the recurrent layer, and the output layer. Mao et al. [129] propose a multi-view RNN architecture, which aims to generate captions for visual images. The multi-view RNN includes the three subnetworks; a vision network, a language network and a multi-view network. Specifically, the vision network usually adopts a deep CNN, such as Resnet, Alexnet and Inception, which aims to map the visual information of an image into its deep feature representation. The language network is to capture the task-specific representation and the temporal dependency. In the multi-view network, a hidden network is adopted to find the relationship between the vision representation and the learned language. Following [129], Karpathy et al. [95] present a multi-view alignment model based on RNN to bridge the inter-view relationship between visual

and textual data. In [1], a multi-view RNN model has been presented to address the indoor scene recognition problem. An algorithm based on multi-view RNN has been proposed in [166], which detects the wake and sleep state of a person by analyzing the data generated from his/her smartphones and wearable technologies.

## 2.3 Conclusion

A significant challenge of multi-view data analysis includes identification of most relevant views from the given multiple views which can efficiently characterize the inherent characteristics of the input pair of classes. In this context, a novel framework is developed in the next chapter by judiciously integrating the theory of rough sets and the Bayes decision theory to evaluate the efficacy of a view in discriminating observations from a given pair of classes.

# Chapter 3

# Rough-Bayesian Approach to Select Class-Pair Specific Descriptors

## 3.1 Introduction

In many real world applications, the data can naturally be described in terms of multiple views. For example, a webpage can be described by the words appearing on the page itself and the words underlying the links pointing to the webpage from other pages [19]. In natural language processing tasks, the same document can have multiple representations in different languages [7]. Although each of the individual views can characterize the input data, multiple views often provide information distinct to each other which can alleviate the difficulty of describing the intrinsic properties of the given data. In recent years, several methods of learning from multiple images have been proposed by considering the diversity of different views. These views may be obtained from multiple sources or different feature subsets. For example, a person can be identified by images of face, fingerprint, signature or iris with information obtained from multiple sources. Primarily, an image can be represented by its color or texture features, which can be considered as different feature subsets, also referred to as modalities, of the image. The classification of images based on multi-view data has been exercised in numerous domains of applications, namely, object detection [114], tumor analysis [48], face recognition [81], 3D saliency detection [65], and so on.

One of the emerging problems of image classification is recognition of staining patterns present in human epithelial type 2 (HEp-2) cells for automatic antinuclear antibody (ANA) analysis. The patterns observable in the cells provide the main information used for the analysis, since each pattern is related to the presence of specific ANAs. As the staining pattern is independent of each cell nucleus or cell including cytoplasm, each cell needs to be evaluated separately [177]. In common ANA analysis using indirect immunofluorescence (IIF), an expert carefully analyzes the staining patterns of HEp-2 cells through a fluorescence microscope. However, this procedure is not only time consuming but also subjected to inter-observer as well as intra-observer variability [133].

A significant amount of research has been undertaken in the last few years for automatic recognition of staining patterns present in HEp-2 cell images. Different global texture

descriptors based on gray level co-occurrence matrix are used in [181], while edge orientation histograms, rotation-invariant Gabor features, and modified Zernike moments are also found to provide important information of an HEp-2 cell image [22]. Faraki et al. [45] used covariance descriptor obtained from a bank of Gabor filters, while Kong et al. [99] applied maximum-response filter banks and histogram of gradients for HEp-2 cell texture analysis. The concept of bag of features model is used in [205], for the generation of cell signature. In [22], it has been shown that the local variations of intensity patterns are more effective than global information, obtained from the HEp-2 cell images, in differentiating various staining pattern classes. Several local texture descriptors, namely, local binary pattern (LBP) [148], rotation-invariant LBP ($LBP^{ri}$) [147], rotation-invariant uniform LBP ($LBP^{riu2}$) [147], co-occurrence of adjacent LBP (CoALBP) [146], completed LBP (CLBP) [61], rotation-invariant CoALBP (RICLBP) [145], median robust extended LBP [121], noise tolerant LBP [46], pairwise rotation-invariant co-occurrence LBP [153] and local ternary pattern [185], can be used to characterize a cell image. Some of them have already been found to be successful for HEp-2 cell pattern analysis [22]. In order to encode gradient and textural characteristics of the HEp-2 patterns, the concept of gradient-oriented co-occurrence of LBP has been introduced in [188], while the dominant LBP has been presented in [120] for texture analysis.

Recently, there has been a growing interest in the development of deep learning models for HEp-2 cell staining pattern recognition. In [118], a deep convolutional residual in residual network has been proposed for simultaneous segmentation and classification of HEp-2 cell images. The deep residual network has been applied in [112] for classifying HEp-2 cell images, while a deep feature extraction method, based on convolutional auto-encoders, has been introduced in [199] for staining pattern recognition. A deeper architecture with convolutional neural network (CNN) is developed in [88], and several preprocessing techniques have been applied in [160] to enhance the performance of deep CNN in identifying staining patterns present in HEp-2 cell images. However, the training of these deep learning models is computationally very expensive due to the large parameter space. To alleviate this problem, various deep architectures have recently been introduced in [55, 117], which enable building of deep architectures with comparatively smaller parameter space. In [55], the deep CNN has been used to classify HEp-2 cell staining patterns, whereas the features extracted by shallow and deep multi-scale convolutional component are fused in [117] for performance improvement. However, the availability of limited number of training images causes inappropriate learning of deep models which result into inaccurate classification of the HEp-2 cell images. Hence, data augmentation is necessary for these models to identify staining patterns properly.

In this background, it can be stated that the textural properties of the HEp-2 cell images are important for discrimination of the corresponding staining patterns as they contain important information about the structural arrangement of surfaces and their relationship to the surrounding environment. In literature, there exists several texture descriptors which translate the given images from the input space to the feature space in such a way that the intrinsic properties of the input images are properly described in the obtained feature space. Now, different texture descriptors encapsulate different characteristics of the images. For example, $LBP^{ri}$ [147] consider rotation invariant texture classification, whereas CoALBP [146] consider spatial relation among micropatterns. Also, an important aspect of texture is scale. The appearance of most textures is changed when viewed at

different scales. According to psychovisual studies, the human visual system processes images in multiple scales; thus capable of preserving both local and global information. In medical imaging perspective, the scale of the imagery may be different in many cases, and so it is important to understand how information changes over different scales of imagery. However, the existing approaches consider a uniform set of texture descriptors and/or scales for describing all the staining pattern classes, which restricts the recognition rates of HEp-2 cell classification.

The proposed method is developed based on the hypothesis that a fixed set of descriptors or scales may not be effective for classifying staining patterns present in HEp-2 cell images into multiple classes. A particular set of descriptors may be significant for classifying a pair of classes, but may not be relevant for other pairs of classes. Similarly, a uniform set of scales may be effective for classifying a pair of classes, but may not be significant for another pair of classes. Also, the proposed method assumes that all the features of a particular set do not contribute uniformly in describing the inherent characteristics of the staining patterns present in HEp-2 cell images. Furthermore, the intrinsic textures of various HEp-2 cell types are fairly different from each other. The HEp-2 cell stained images also have unpredictably ambiguous texture. This difficulty exists in both inter-class and intra-class examples. Moreover, due to the nature of human cell, the shared visual similarity of inter-class samples, like the texture of certain parts of the tissues, further increases the ambiguity. These difficulties limit the accuracy of HEp-2 cell classification. Also, there exists uncertainty due to inexactness, vagueness and incompleteness in HEp-2 class definition, as well as overlapping characteristics of HEp-2 cell classes. In this regard, the theory of rough sets can be used to handle uncertainties associated with HEp-2 cell pattern classes. It provides an effective means for the analysis of data by constructing upper and lower approximations of set concepts from the acquired data, based on information granules [149]. The probabilistic modelling, on the other hand, aims to measure the belongingness of an object to a specific class, based on a probability distribution. As rough sets and probabilistic models are complementary in nature, they can be integrated for modelling uncertainty inherent in classification of staining pattern classes of HEp-2 cells.

In this regard, the chapter introduces a novel method to identify a set of relevant descriptors under appropriate scales for the recognition of staining patterns present in HEp-2 cell images. The proposed method assumes that a uniform set of descriptors and/or scale is not effective for all HEp-2 cell classes. Therefore, a set of relevant local texture descriptors is first identified under appropriate scales for a pair of classes, and then the final feature set for multiple classes is formed from the relevant descriptors of all possible pairs of classes selected under appropriate scales. Judiciously integrating the merits of rough sets and Bayes decision theory, a new framework, termed as Rough-Bayesian model, is introduced to evaluate the relevance of a local texture descriptor and/or a scale for a pair of classes. While rough set theory deals with the uncertainty due to inexactness, vagueness, or incompleteness in HEp-2 class definition, the probabilistic model addresses the uncertainty due to overlapping classes by measuring the belongingness of an HEp-2 cell to a specific class, based on some presumed intensity distribution of the class. During the computation of the relevance of each descriptor and/or scale, the proposed method takes care of the presence of both noisy pixels in an HEp-2 cell image and noisy HEp-2 cell images in a staining pattern class. Finally, support vector machine (SVM) [197] is used to predict the staining patterns present in HEp-2 cell images. The performance of

the proposed method is studied with reference to several state-of-the-art approaches on a set of real-life HEp-2 cell image databases. Some of the results of this chapter are reported in [102, 103].

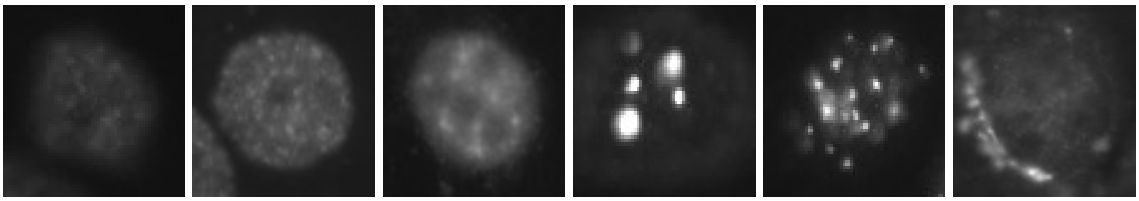
The rest of this chapter is organized as follows: Section 3.2 presents a new method to find a set of effective descriptors, along with the corresponding scales, for classification of staining patterns in HEp-2 cell images. Section 3.3 introduces a novel framework, termed as Rough-Bayesian model, to compute the relevance of a set of features, integrating judiciously the merits of rough sets and Bayes decision theory. Computational complexity of the proposed approach is presented in Section 3.4. The efficacy of the proposed method is studied with reference to several state-of-the-art approaches on a set of real-life HEp-2 cell image databases in Section 3.5. Concluding remarks are provided in Section 3.6.

## 3.2 Selection of Relevant Descriptors and Scales

This section presents a new method to find a set of effective descriptors, along with the corresponding scales, for classification of staining patterns in HEp-2 cell images.

### 3.2.1 Motivation

In general, a single scale of same local textural features is used for the recognition of HEp-2 patterns. Due to the drastic variation of different scales and noisy nature of the input HEp-2 cell images, the single scale and/or descriptor usually gives poor performance, which is reflected in the insufficient and inaccurate staining pattern representation of these images.



(a) Centromere (b) Homogeneous (c) Nucleolar (d) Speckled (e) Nuclear Membrane (f) Golgi

Figure 3.1: HEp-2 cell images from ICPR data set.

Let us assume that $\mathcal{M} = \{\mathcal{M}_1, \cdots, \mathcal{M}_p, \cdots, \mathcal{M}_t\}$ represents a set of $t$ modalities, where each modality $\mathcal{M}_p$ corresponds to a local descriptor considered under a particular scale. In the current example, four types of local descriptor, namely, LBP [148], $\text{LBP}^{\text{ri}}$ [147], $\text{LBP}^{\text{riu2}}$ [147], and CoALBP [146], are used, while four scales of LBP neighborhood such as 1 ($S_1$), 2 ($S_2$), 3 ($S_3$), and 4 ($S_4$) are considered. The 4-neighborhood is used for CoALBP, while for others 8-neighborhood is considered. Table 3.1 and Table 3.2 present several confusion matrices obtained for ICPR image database [75], which was used in HEp-2 cell classification contest organized by ICPR 2014. In particular, 68,429 cell images constitute the database, out of which 20% of data set, that is, 13,596 cell images are publicly available. For experimental purposes, the set of 13,596 images is partitioned into training and test sets, consisting of 6,797 and 6,799 cell images, respectively. In both training and test sets, the images are almost equally distributed with respect to six different pattern classes, namely, Centromere, Homogeneous, Nucleolar, Nuclear Membrane, Speckled, and

Table 3.1: Confusion Matrix for ICPR Test Data Set at Scale 4

| Actual/Predict | Centromere | Homogeneous | Nucleolar | Speckled | Nuclear Mem. | Golgi |
|---|---|---|---|---|---|---|
| Predicted Class Using **LBP** | | | | | | |
| Centromere | **79.94%(1096)** | 2.26%(31) | 3.72%(51) | 13.42%(184) | 0.29%(4) | 0.36%(5) |
| Homogeneous | 1.44%(18) | **69.77%(870)** | 3.69%(46) | 18.93%(236) | 5.37%(67) | 0.80%(10) |
| Nucleolar | 4.31%(56) | 6.39%(83) | **67.82%(881)** | 10.08%(131) | 4.08%(53) | 7.31%(95) |
| Speckled | 9.04%(128) | 20.76%(294) | 6.29%(89) | **60.38%(855)** | 1.62%(23) | 1.91%(27) |
| Nuclear Mem. | 0.18%(2) | 14.58%(161) | 4.26%(47) | 4.26%(47) | **71.74%(792)** | 4.98%(55) |
| Golgi | 0.83%(3) | 9.12%(33) | **38.40%(139)** | 9.67%(35) | 11.60%(42) | **30.39%(110)** |
| Predicted Class Using **LBP$^{ri}$** | | | | | | |
| Centromere | **83.08%(1139)** | 1.97%(27) | 4.74%(65) | 9.99%(137) | 0.07%(1) | 0.15%(2) |
| Homogeneous | 1.68%(21) | **81.96%(1022)** | 4.17%(52) | 7.22%(90) | 3.93%(49) | 1.04%(13) |
| Nucleolar | 3.00%(39) | 6.16%(80) | **81.45%(1058)** | 4.46%(58) | 3.00%(39) | 1.92%(25) |
| Speckled | 15.89%(225) | **38.56%(546)** | 12.43%(176) | **30.23%(428)** | 1.84%(26) | 1.06%(15) |
| Nuclear Mem. | 0.18%(2) | 11.14%(123) | 3.35%(37) | 1.36%(15) | **80.98%(894)** | 2.99%(33) |
| Golgi | 1.66%(6) | 10.50%(38) | **46.41%(168)** | 4.70%(17) | 8.84%(32) | **27.90%(101)** |
| Predicted Class Using **LBP$^{riu2}$** | | | | | | |
| Centromere | **85.34%(1170)** | 1.53%(21) | 3.06%(42) | 9.85%(135) | 0.22%(3) | 0.00%(0) |
| Homogeneous | 1.68%(21) | **69.53%(867)** | 3.69%(46) | 19.97%(249) | 5.13%(64) | 0.00%(0) |
| Nucleolar | 1.77%(23) | 3.93%(51) | **73.29%(952)** | 15.78%(205) | 5.23%(68) | 0.00%(0) |
| Speckled | 9.89%(140) | 20.06%(284) | 7.84%(111) | **59.25%(839)** | 2.90%(41) | 0.07%(1) |
| Nuclear Mem. | 0.82%(9) | 12.23%(135) | 6.43%(71) | 1.81%(20) | **78.71%(869)** | 0.00%(0) |
| Golgi | 0.55%(2) | 15.47%(56) | **59.94%(217)** | 8.84%(32) | 14.92%(54) | **0.28%(1)** |
| Predicted Class Using **CoALBP** | | | | | | |
| Centromere | **79.36%(1088)** | 4.38%(60) | 6.71%(92) | 8.90%(122) | 0.29%(4) | 0.36%(5) |
| Homogeneous | 0.88%(11) | **68.64%(856)** | 6.26%(78) | 19.09%(238) | 4.01%(50) | 1.12%(14) |
| Nucleolar | 7.85%(102) | 7.16%(93) | **73.98%(961)** | 7.62%(99) | 1.46%(19) | 1.92%(25) |
| Speckled | 7.06%(100) | 21.75%(308) | 4.66%(66) | **64.83%(918)** | 0.99%(14) | 0.71%(10) |
| Nuclear Mem. | 0.91%(10) | 5.71%(63) | 1.99%(22) | 3.89%(43) | **77.63%(857)** | 9.87%(109) |
| Golgi | 6.35%(23) | 5.25%(19) | 15.19%(55) | 6.35%(23) | 30.94%(112) | **35.91%(130)** |
| Predicted Class Using **Class-Pair Specific Descriptor** | | | | | | |
| Centromere | **83.81%(1149)** | 2.12%(29) | 5.18%(71) | 8.53%(117) | 0.07%(1) | 0.29%(4) |
| Homogeneous | 0.80%(10) | **75.70%(944)** | 3.29%(41) | 16.36%(204) | 3.37%(42) | 0.48%(6) |
| Nucleolar | 5.00%(65) | 5.16%(67) | **79.06%(1027)** | 7.85%(102) | 1.00%(13) | 1.92%(25) |
| Speckled | 6.50%(92) | 17.80%(252) | 4.59%(65) | **69.70%(987)** | 0.99%(14) | 0.42%(6) |
| Nuclear Mem. | 0.27%(3) | 6.88%(76) | 1.45%(16) | 2.54%(28) | **84.69%(935)** | 4.17%(46) |
| Golgi | 3.04%(11) | 2.49%(9) | 18.51%(67) | 5.25%(19) | 14.64%(53) | **56.08%(203)** |

Golgi. Representative images from the ICPR HEp-2 cell image database are presented in Figure 3.1.

#### 3.2.1.1 Performance at Fixed Scale

Each confusion matrix of Table 3.1 corresponds to each of the four local descriptors at scale $S_4$. The SVM with linear kernel, discussed in Appendix C, is used to evaluate the performance of different feature descriptors in the present work. While the first matrix corresponds to LBP, next three matrices, namely, second, third, and fourth, are obtained for LBP$^{ri}$, LBP$^{riu2}$, and CoALBP, respectively. The LBP, LBP$^{ri}$, LBP$^{riu2}$, and CoALBP, respectively, achieve 85.83%, 70.21%, 69.56%, and 100% classification accuracy on training cell images of ICPR database, while 67.72% (=(1096+870+881+855+792+110)/6799),

68.27%, 69.10%, and 70.75% accuracy on test images. However, from the confusion matrices, it can be seen that the class labels of 38.40%, 46.41%, and 59.94% of the cell images of Golgi class are predicted as Nucleolar class for LBP, $LBP^{ri}$, and $LBP^{riu2}$, respectively, while the predicted class of 30.94% of the cell images of Golgi class is Nuclear Membrane for CoALBP. Similarly, the class label of 38.56% cell images of Speckled class is predicted as Homogeneous class for $LBP^{ri}$. Moreover, for $LBP^{riu2}$, only 1 image, out of 362 cell images, of Golgi class is correctly identified, while the class label is correctly predicted for 85.34% of the cell images of Centromere class. For LBP, $LBP^{riu2}$, and CoALBP, nearly 20% of Homogeneous cells is predicted as Speckled cells, while almost same percentage of Speckled cells is classified as Homogeneous cells. So, a particular descriptor may be important for classifying a class-pair, but may not be effective for another pair of classes.

Table 3.2: Confusion Matrix for ICPR Test Data Set for $LBP^{ri}$

| Actual/Predict | Centromere | Homogeneous | Nucleolar | Speckled | Nuclear Mem. | Golgi |
|---|---|---|---|---|---|---|
| Predicted Class Using **Scale 1** | | | | | | |
| Centromere | **75.27%(1032)** | 0.22%(3) | 7.59%(104) | 16.92%(232) | 0.00%(0) | 0.00%(0) |
| Homogeneous | 0.08%(1) | **50.12%(625)** | 5.21%(65) | 30.07%(375) | 14.35%(179) | 0.16%(2) |
| Nucleolar | 4.85%(63) | 5.23%(68) | **60.20%(782)** | 18.78%(244) | 10.39%(135) | 0.54%(7) |
| Speckled | 5.44%(77) | 15.89%(225) | 12.50%(177) | **63.28%(896)** | 2.12%(30) | 0.78%(11) |
| Nuclear Mem. | 0.00%(0) | 13.86%(153) | 14.58%(161) | 8.24%(91) | **63.04%(696)** | 0.27%(3) |
| Golgi | 1.38%(5) | 18.51%(67) | **45.30%(164)** | 20.17%(73) | 13.54%(49) | **1.10%(4)** |
| Predicted Class Using **Scale 2** | | | | | | |
| Centromere | **84.32%(1156)** | 1.68%(23) | 5.91%(81) | 7.80%(107) | 0.29%(4) | 0.00%(0) |
| Homogeneous | 0.88%(11) | **72.49%(904)** | 2.73%(34) | 14.51%(181) | 9.22%(115) | 0.16%(2) |
| Nucleolar | 2.54%(33) | 4.39%(57) | **79.14%(1028)** | 7.85%(102) | 5.70%(74) | 0.38%(5) |
| Speckled | 15.61%(221) | 34.04%(482) | 8.90%(126) | **38.91%(551)** | 2.26%(32) | 0.28%(4) |
| Nuclear Mem. | 0.36%(4) | 11.87%(131) | 7.88%(87) | 2.26%(25) | **77.08%(851)** | 0.54%(6) |
| Golgi | 2.21%(8) | 13.81%(50) | **55.25%(200)** | 11.33%(41) | 11.60%(42) | **5.80%(21)** |
| Predicted Class Using **Scale 3** | | | | | | |
| Centromere | **86.14%(1181)** | 1.39%(19) | 3.36%(46) | 8.53%(117) | 0.22%(3) | 0.36%(5) |
| Homogeneous | 1.28%(16) | **65.12%(812)** | 6.50%(81) | 18.36%(229) | 7.94%(99) | 0.80%(10) |
| Nucleolar | 2.62%(34) | 3.23%(42) | **77.14%(1002)** | 11.93%(155) | 3.39%(44) | 1.69%(22) |
| Speckled | 10.17%(144) | 18.29%(259) | 5.72%(81) | **62.71%(888)** | 2.19%(31) | 0.92%(13) |
| Nuclear Mem. | 0.09%(1) | 13.77%(152) | 3.80%(42) | 2.72%(30) | **76.99%(850)** | 2.63%(29) |
| Golgi | 0.00%(0) | 8.29%(30) | **48.07%(174)** | 9.67%(35) | 10.22%(37) | **23.76%(86)** |
| Predicted Class Using **Scale 4** | | | | | | |
| Centromere | **83.08%(1139)** | 1.97%(27) | 4.74%(65) | 9.99%(137) | 0.07%(1) | 0.15%(2) |
| Homogeneous | 1.68%(21) | **81.96%(1022)** | 4.17%(52) | 7.22%(90) | 3.93%(49) | 1.04%(13) |
| Nucleolar | 3.00%(39) | 6.16%(80) | **81.45%(1058)** | 4.46%(58) | 3.00%(39) | 1.92%(25) |
| Speckled | 15.89%(225) | **38.56%(546)** | 12.43%(176) | **30.23%(428)** | 1.84%(26) | 1.06%(15) |
| Nuclear Mem. | 0.18%(2) | 11.14%(123) | 3.35%(37) | 1.36%(15) | **80.98%(894)** | 2.99%(33) |
| Golgi | 1.66%(6) | 10.50%(38) | **46.41%(168)** | 4.70%(17) | 8.84%(32) | **27.90%(101)** |
| Predicted Class Using **Class-Pair Specific Scale** | | | | | | |
| Centromere | **91.76%(1258)** | 1.17%(16) | 2.77%(38) | 4.16%(57) | 0.07%(1) | 0.07%(1) |
| Homogeneous | 0.16%(2) | **81.40%(1015)** | 2.65%(33) | 12.51%(156) | 2.89%(36) | 0.40%(5) |
| Nucleolar | 1.92%(25) | 4.39%(57) | **83.99%(1091)** | 6.62%(86) | 1.39%(18) | 1.69%(22) |
| Speckled | 4.10%(58) | 15.11%(214) | 3.67%(52) | **75.56%(1070)** | 1.48%(21) | 0.07%(1) |
| Nuclear Mem. | 0.09%(1) | 7.52%(83) | 1.36%(15) | 1.72%(19) | **85.24%(941)** | 4.08%(45) |
| Golgi | 1.66%(6) | 3.59%(13) | 21.82%(79) | 5.52%(20) | 6.91%(25) | **60.50%(219)** |

#### 3.2.1.2 Performance at Fixed Descriptor

Each confusion matrix of Table 3.2 corresponds to each scale of $LBP^{ri}$. While the first two matrices correspond to scales $S_1$ and $S_2$, next two matrices are obtained for scales $S_3$ and $S_4$. The $LBP^{ri}$ attains 60.67%, 68.56%, 73.30%, and 70.21% classification accuracy on training cell images of ICPR database for scales $S_1$, $S_2$, $S_3$, and $S_4$, respectively, while 59.35%, 66.35%, 70.88%, and 68.28% of accuracy on test images. From the first four confusion matrices, it is seen that the class labels of 45.30%, 55.25%, 48.07%, and 46.41% of the cell images of Golgi class are predicted as Nucleolar class for scales $S_1$, $S_2$, $S_3$, and $S_4$, respectively, while the class of 34.04% and 38.56% of the cell images of Speckled class is predicted as Homogeneous class for scales $S_2$ and $S_4$, respectively. On the other hand, the class labels of 30.07% of cell images of Homogeneous class are predicted as Speckled class for scale $S_1$. So, a specific scale may be significant in discriminating a class-pair, while it may not be important for another pair of classes.

Table 3.3: Confusion Matrix for ICPR Test Data Set Using Class-Pair Specific Descriptors and Scales

| Actual/Predict | Centromere | Homogeneous | Nucleolar | Speckled | Nuclear Mem. | Golgi |
|---|---|---|---|---|---|---|
| Centromere | **92.27%(1265)** | 1.02%(14) | 2.26%(31) | 4.30%(59) | 0.07%(1) | 0.07%(1) |
| Homogeneous | 0.08%(1) | **83.88%(1046)** | 1.04%(13) | 13.79%(172) | 0.96%(12) | 0.24%(3) |
| Nucleolar | 2.00%(26) | 2.77%(36) | **88.22%(1146)** | 4.31%(56) | 0.92%(12) | 1.77%(23) |
| Speckled | 3.04%(43) | 13.28%(188) | 3.11%(44) | **79.31%(1123)** | 1.06%(15) | 0.21%(3) |
| Nuclear Mem. | 0.00%(0) | 4.89%(54) | 1.45%(16) | 1.36%(15) | **89.04%(983)** | 3.26%(36) |
| Golgi | 1.66%(6) | 2.21%(8) | 13.81%(50) | 2.76%(10) | 9.67%(35) | **69.89%(253)** |

From all the confusion matrices reported in Tables Table 3.1 and Table 3.2, it is seen that a unique scale and/or a unique local descriptor may not be effective for all staining pattern classes. In other words, a scale or a descriptor may be significant for classifying a pair of classes, while may not be effective for another pair of classes. For a pair of classes, if the appropriate descriptors and corresponding scales can be identified properly, better classification accuracy can be achieved for classifying HEp-2 cells into multiple classes, as evident from last confusion matrices of Table 3.1 and Table 3.2. These two matrices correspond to the cases when descriptors and scales, respectively, are class-pair specific given fixed scale ($S_4$) and fixed descriptor ($LBP^{ri}$). In case of class-pair specific descriptor, as reported in Table 3.1, the classification accuracy on test images is 77.14%, while it is 82.28% if class-pair specific scale Table 3.2 is considered. When both descriptor and scale are considered to be class-pair specific, the classification accuracy on test images is increased to 85.54% and corresponding confusion matrix is presented in Table 3.3. More than 50% cells of Golgi class are correctly identified in these three cases, which is not possible if a single descriptor or a single scale is considered for the extraction of features of all classes.

### 3.2.2 Proposed Method

The proposed method selects important features from the relevant modalities for each pair of classes, and forms the resultant feature set for multiple staining pattern classes. The block diagram of the proposed method for HEp-2 cell staining pattern classification is

depicted in Figure 3.2. At first, the class specific feature set is formed for each of the HEp-2 cell staining pattern classes, considering the histograms of feature values for each of the input HEp-2 cell images, under a specific modality. Then, the class-pair specific feature sets are obtained for all possible class-pairs, from each of the class specific feature sets. The relevance of each class-pair specific feature set under a particular modality is evaluated based on rough sets and Bayes decision theory. After selecting a set of most relevant feature sets for each class-pair, the final feature set for multiple staining pattern classes is formed, which is used to train support vector machine. Each of these steps is elaborated next one by one.

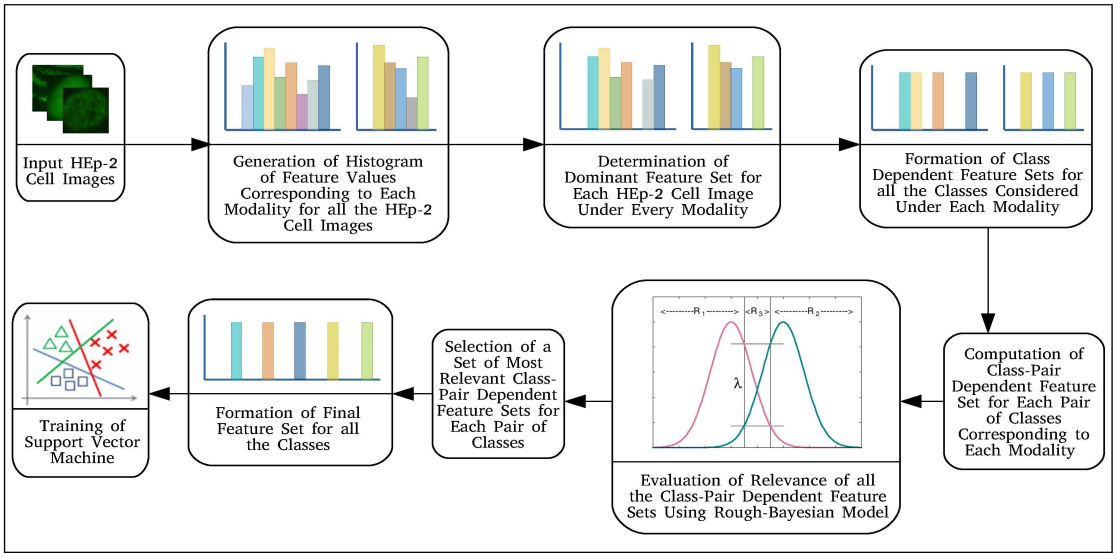


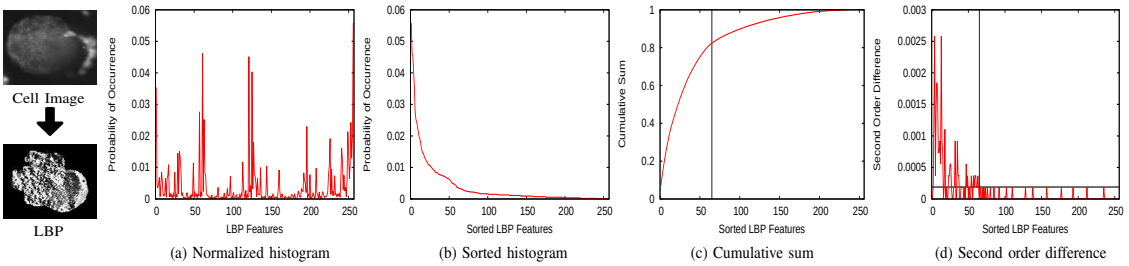

Figure 3.2: Block diagram of the proposed HEp-2 cell staining pattern recognition method.

Let us assume that $X = \{x_1, \cdots, x_j, \cdots, x_n\}$ be a set of $n$ number of HEp-2 cell images, where each image $x_j \in \Re^m$ is represented by a set $h_j = \{h_{j1}, \cdots, h_{jk}, \cdots, h_{jm}\}$, consisting of feature values of $m$ features $\{\mathcal{A}_1, \cdots, \mathcal{A}_k, \cdots, \mathcal{A}_m\}$. In the proposed work, four local descriptors, namely, LBP, LBP$^{\text{ri}}$, LBP$^{\text{riu2}}$, and CoALBP, are considered. So, $h_j$ is the normalized histogram of $x_j$ corresponding to either LBP, LBP$^{\text{ri}}$, LBP$^{\text{riu2}}$ or CoALBP. Let $\tilde{h}_j = \{\tilde{h}_{j1}, \cdots, \tilde{h}_{jk}, \cdots, \tilde{h}_{jm}\}$ be the sorted normalized histogram of $x_j$, in descending order, and $I_j = \{I_{j1}, \cdots, I_{jk}, \cdots, I_{jm}\}$ represents the feature order of $\tilde{h}_j$. Figure 3.3(a) presents the normalized LBP histogram at $S_2$ of an HEp-2 cell image belonging to Golgi class of ICPR data, while Figure 3.3(b) depicts the corresponding sorted histogram.

#### 3.2.2.1 Class Dependent Features

In order to find out the set of dominant features present in $\tilde{h}_j$, corresponding to each HEp-2 cell image $x_j$, the following function is defined:

$$\mathcal{S}(r) = \sum_{l=1}^{r} \tilde{h}_{jl}. \tag{3.1}$$




(a) Normalized histogram (b) Sorted histogram (c) Cumulative sum (d) Second order difference

Figure 3.3: Illustrative example to find out the dominant features of an HEp-2 cell image.

The function $\mathcal{S}(r)$ represents the cumulative sum of first $r$ normalized feature values of $\tilde{h}_j$. As each normalized feature value $\tilde{h}_{jk}$ is positive, the function $\mathcal{S}(r)$ is single-valued and monotonically increasing in the interval $1 \leqslant r \leqslant m$. Moreover, $0 \leqslant \mathcal{S}(r) \leqslant 1$ for $1 \leqslant r \leqslant m$. The second-order derivative of this function is the difference

$$\frac{\partial^2 \mathcal{S}(r)}{\partial r^2} = \mathcal{S}(r+1) + \mathcal{S}(r-1) - 2\mathcal{S}(r). \tag{3.2}$$

Figure 3.3(c) presents the variation of cumulative sum $\mathcal{S}(r)$ of first $r$ normalized feature values of $\tilde{h}_j$, corresponding to Figure 3.3(b). In this figure, the number of dominant features of $x_j$ is denoted by the point in $X$-axis, where the slope of the graph changes abruptly. This is due to the fact that further addition of features into the dominant feature set does not offer any significant increase in the cumulative sum of the normalized histogram of the object $x_j$. In order to identify this point, the difference operator is applied on $\mathcal{S}(r)$. Now, from the definition, it is known that the first-order difference yields a non-zero response along a ramp. With the change of the slope of the ramp, only the magnitude of the response of first-order difference operator changes. Hence, it is not possible to identify the point of interest with first-order difference response. On the other hand, in case of second-order difference, a non-zero response is only obtained at the beginning and end of the ramp, that is, where the slope changes, and a zero response is obtained along the ramp. Since the graph of $\mathcal{S}(r)$ can be considered as piece-wise linear after the slope changes, the second-order difference produces almost zero response in this region. Figure 3.3(d) depicts the second-order difference of $\mathcal{S}(r)$, corresponding to Figure 3.3(c). Thus, the number of dominant features for a particular object $x_j$ is defined as the feature index beyond which second-order difference yields near to zero response, indicating no further significant change in slope of the graph. As the second-order difference produces double sided response, modulus of the response is considered for computational advantage. The number of dominant features $d_j$, present in $\tilde{h}_j$, is thus defined as $d_j = r$, if $\left|\frac{\partial^2 \mathcal{S}(l)}{\partial l^2}\right| < \varepsilon_1$, $\forall l \in \{r+1, r+2, \cdots, m\}$. So, the average number of dominant features for the set $X$ is given by

$$\overline{d} = \frac{1}{n} \sum_{j=1}^{n} d_j. \tag{3.3}$$

The set of dominant features, denoted by $V_j$, corresponding to sample $x_j$, is defined as

$$V_j = \{\mathcal{A}_k \mid I_{jq} = k \text{ and } q \leqslant \overline{d}\}. \tag{3.4}$$

The features, which are dominant in sample $x_j$, belong to the set $V_j$, while rest of the features those are considered to be insignificant do not belong to $V_j$. Algorithm 3.1 presents the basic steps to determine the set of dominant features for each training cell image $x_j \in X$.

**Algorithm 3.1** Determination of Set of Dominant Features for Each Cell Image $x_j \in X$.

**Input:** Set of training cell images $X$.
**Output:** Set of dominant features for each cell image $x_j \in X$.

1: **for** each $x_j \in X$ **do**

(i) Compute the normalized histogram $h_j$ under a particular modality.

(ii) Sort the normalized histogram and compute cumulative sum of the sorted histogram.

(iii) Determine the second-order difference of the cumulative sum of sorted histogram.

(iv) Find out the number of dominant features $d_j$ of the sample $x_j$, upto which the second-order difference exhibits a non-zero response.

2: **end for**
3: Compute the average number of dominant features $\overline{d}$ over the entire set $X$.
4: **for** each $x_j \in X$ **do**

- Select the first $\overline{d}$ features from the sorted histogram $\tilde{h}_j$ as the set of dominant features for the sample $x_j$.

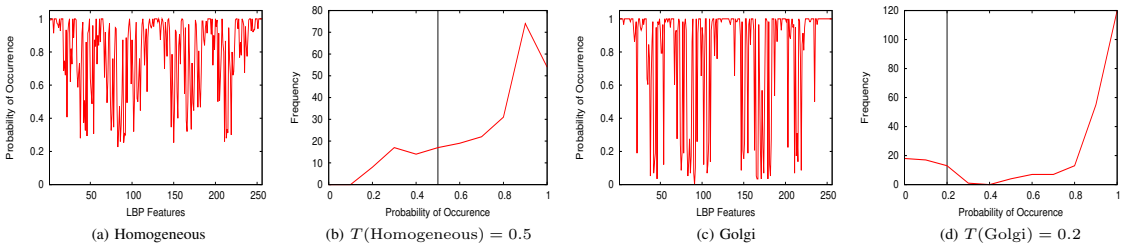
5: **end for**

Let us assume that each sample $x_j \in X$ belongs to one of the $c$ classes of the set $\{\omega_1, \cdots, \omega_i, \cdots, \omega_c\}$. It is likely that the samples of the same class will have similar dominant feature sets, while samples from different classes will have different sets of dominant features. Presence of a noisy sample in the set $X$ may introduce certain features in the dominant set which definitely do not bear any significant characteristics of the class, say $\omega_i$, to which the noisy sample belongs. Rather, presence of those features in the dominant set can cause erroneous decision regarding classification of the samples of class $\omega_i$. Hence, it is necessary to identify the features from the dominant set of a sample which correctly represent the characteristics of the class to which it belongs.

In order to address the aforementioned problem, the probability of occurrence of a feature $\mathcal{A}_k$ in the dominant sets of the samples of a particular class $\omega_i$ is defined as follows:

$$p(\mathcal{A}_k|\omega_i) = \frac{1}{|\omega_i|} \sum_{x_j \in \omega_i} v_{jk}; \tag{3.5}$$

$$\text{where } v_{jk} = \begin{cases} 1, & \text{if } \mathcal{A}_k \in V_j; \\ 0, & \text{otherwise.} \end{cases} \tag{3.6}$$




(a) Homogeneous (b) $T(\text{Homogeneous}) = 0.5$ (c) Golgi (d) $T(\text{Golgi}) = 0.2$

Figure 3.4: Illustration to find out dominant features of an HEp-2 cell class.

Here, $|\omega_i|$ represents the number of samples belonging to class $\omega_i$. Figure 3.4(a) and Figure 3.4(c) depict the probability of occurrence of LBPs in Homogeneous and Golgi classes, respectively. The features, which have occurred in the dominant sets of the samples belonging to class $\omega_i$ inadvertently, will have low probability of occurrence. Hence, the features having $p(\mathcal{A}_k|\omega_i)$ values below a certain threshold $T(\omega_i)$ can be considered to be insignificant for the class $\omega_i$ and can be discarded without loss of much information. Moreover, if the samples of a class $\omega_i$ show a large degree of variations among them, the sets of dominant features of those samples will also vary to a great extent. So, there will be a large number of features $\mathcal{A}_k$s having low $p(\mathcal{A}_k|\omega_i)$ values. Here, $T(\omega_i)$ is expected to have a low value. On the other hand, if the samples of a class, say $\omega_r$, exhibit similar properties, then most of the features $\mathcal{A}_k$s will have high $p(\mathcal{A}_k|\omega_r)$ values. In this case, $T(\omega_r)$ should have a high value. Hence, the values of probability of occurrence of features in the dominant sets of a class are not enough to determine the threshold of that class. The frequency of probability of occurrence values also needs to be considered. Hence, the histogram $H(\omega_i)$ of the probability of occurrence values is calculated as follows:

$$H_l(\omega_i) = H_l(\omega_i) + 1 \quad \text{if} \quad p(\mathcal{A}_k|\omega_i) \in \left(\frac{l-1}{L}, \frac{l}{L}\right], \quad \forall k; \tag{3.7}$$

where $1 \leqslant \sum_{l=1}^{L} H_l(\omega_i) \leqslant m$, $L$ is the number of bins in histogram and $l \in \{1, 2, \cdots, L\}$ denotes an index of the bins. The maximum value of $l$, for which

$$\sum_{q=1}^{l} H_q(\omega_i) < \varepsilon_2 \sum_{q=1}^{L} H_q(\omega_i), \tag{3.8}$$

determines the threshold for the class $\omega_i$ as follows:

$$T(\omega_i) = \frac{l}{L}. \tag{3.9}$$

The parameter $\varepsilon_2$, which is considered to be 0.1 in the current study, depicts the amount of acceptable loss. Figure 3.4(b) and Figure 3.4(d) represent the histograms corresponding to probability of occurrence values of Figure 3.4(a) and Figure 3.4(c), respectively. The threshold values for Homogeneous and Golgi classes of ICPR data, obtained using ((3.9)), are 0.5 and 0.2, respectively.

Based on the threshold $T(\omega_i)$, the class dependent feature set $\mathcal{N}(\omega_i)$ for the class $\omega_i$ is defined as

$$\mathcal{N}(\omega_i) = \{\mathcal{A}_k \mid p(\mathcal{A}_k|\omega_i) \geqslant T(\omega_i)\}. \tag{3.10}$$

So, the $k$-th feature $\mathcal{A}_k$ belongs to the feature set $\mathcal{N}(\omega_i)$ only if $\mathcal{A}_k$ is dominant for most of the samples belonging to class $\omega_i$ as well as represents the important characteristics of the class $\omega_i$. The main steps to generate class dependent feature set $\mathcal{N}(\omega_i)$, corresponding to the class $\omega_i$, are outlined in Algorithm 3.2.

**Algorithm 3.2** Determination of Class Dependent Feature Set

**Input:** Set of dominant features of each cell image $x_j \in X$.
**Output:** Class specific feature set $\mathcal{N}(\omega_i)$; $i = 1, \cdots, c$.

1: **for** each class $\omega_i$ **do**

(i) Calculate the probability of occurrence of features present in the dominant sets of samples belonging to the class $\omega_i$.

(ii) Compute histogram of the probability of occurrence values in the class $\omega_i$.

(iii) Determine class specific threshold $T(\omega_i)$ from the histogram of the class $\omega_i$.

(iv) Form a feature set $\mathcal{N}(\omega_i)$, corresponding to the class $\omega_i$, with the features having probability of occurrence values greater than $T(\omega_i)$.

2: **end for**

#### 3.2.2.2 Pairwise Class Dependent Features

Let $\mathcal{N}(\omega_i)$ and $\mathcal{N}(\omega_r)$ be two feature sets corresponding to two classes $\omega_i$ and $\omega_r$, respectively. These two sets represent the unique characteristics of two classes. The resultant feature set for a pair of classes $\omega_i$ and $\omega_r$, representing the characteristics of both the classes, can be defined as

$$\mathcal{N}(\{\omega_i, \omega_r\}) = \mathcal{N}(\omega_i) \cap \mathcal{N}(\omega_r). \tag{3.11}$$

Hence, the feature set $\mathcal{N}(\{\omega_i, \omega_r\})$ contains those features which are dominant or significant with respect to both the classes $\omega_i$ and $\omega_r$.

Given a set of modalities $\mathcal{M} = \{\mathcal{M}_1, \cdots, \mathcal{M}_p, \cdots, \mathcal{M}_t\}$, the proposed method evaluates the relevance of each feature set $\mathcal{N}_p(\{\omega_i, \omega_r\})$, corresponding to the $p$-th modality $\mathcal{M}_p$, for a pair of classes $\{\omega_i, \omega_r\}$. Let $\Gamma_p(\{\omega_i, \omega_r\})$ be the relevance of the feature set $\mathcal{N}_p(\{\omega_i, \omega_r\})$. The relevance $\Gamma \in [0, 1]$ of each feature set is computed based on Bayes decision theory and the concept of rough sets, introduced in Section 3.3. If the value of $\Gamma_p(\{\omega_i, \omega_r\})$ is 1, then all the samples belonging to two classes $\omega_i$ and $\omega_r$ can be classified correctly using the set of features selected under the $p$-th modality $\mathcal{M}_p$. If the value of $\Gamma_p(\{\omega_i, \omega_r\}) = 0$, then no sample of either classes can be correctly classified with the selected set of features. On the other hand, if $\Gamma_p(\{\omega_i, \omega_r\}) \in (0, 1)$, then some of the samples of two classes can be accurately classified, while others are misclassified. So, the value of $\Gamma_p(\{\omega_i, \omega_r\})$ refers to the relevance of the $p$-th modality $\mathcal{M}_p$ for differentiating a pair of classes $\{\omega_i, \omega_r\}$. After

selecting a set $\tilde{\mathcal{N}}_{ir} = \{\mathcal{N}_p(\{\omega_i, \omega_r\})\}$ of $\tilde{t}$ most relevant feature sets for a pair of classes $\{\omega_i, \omega_r\}$ from $t$ modalities, the final feature set $\mathcal{D}$ for all possible pairs of classes is obtained as follows:

$$\mathcal{D} = \bigcup \tilde{\mathcal{N}}_{ir}. \tag{3.12}$$

The basic steps to select a set of best $\tilde{t}$ feature sets, for a pair of classes $\{\omega_i, \omega_r\}$, are outlined in Algorithm 3.3; while Algorithm 3.4 presents the main steps of proposed method for HEp-2 cell image classification.

**Algorithm 3.3** Selection of Relevant Modality $\mathcal{M}_p \in \mathcal{M}$ for a Pair of Classes $\{\omega_i, \omega_r\}$

**Input:** Class specific feature sets $\mathcal{N}_p(\omega_i)$ and $\mathcal{N}_p(\omega_r)$ for each modality $\mathcal{M}_p \in \mathcal{M}$.
**Output:** A set of class-pair specific feature sets $\tilde{\mathcal{N}}_{ir} = \{\mathcal{N}_p(\{\omega_i, \omega_r\})\}$, along with corresponding modalities $\{\mathcal{M}_p\}$.

1: **for** each modality $\mathcal{M}_p \in \mathcal{M}$ **do**

(i) Compute class-pair specific feature set $\mathcal{N}_p(\{\omega_i, \omega_r\})$ using two class specific feature sets $\mathcal{N}_p(\omega_i)$ and $\mathcal{N}_p(\omega_r)$ of the classes $\omega_i$ and $\omega_r$, respectively.

(ii) Compute the relevance $\Gamma_p(\{\omega_i, \omega_r\})$ of the class-pair specific feature set $\mathcal{N}_p(\{\omega_i, \omega_r\})$ using **Algorithm 3.5**.

2: **end for**
3: Select the set $\tilde{\mathcal{N}}_{ir} = \{\mathcal{N}_p(\{\omega_i, \omega_r\})\}$ of best $\tilde{t}$ feature sets, based on the relevance values.

## 3.3 Rough-Bayesian Approach for Computation of Relevance

The problem of generation of final feature set using (3.12) for $c$ number of HEp-2 cell staining pattern classes boils down to the evaluation of each feature set, corresponding to a pair of classes considered under a particular modality. This section introduces a novel framework, termed as Rough-Bayesian model, to compute the relevance of a set of features, integrating judiciously the merits of rough sets and Bayes decision theory. The basics of the theory of rough sets is presented in Appendix B.

Let $\mathbb{U} = \{x_1, \cdots, x_j, \cdots, x_n\}$ be the set of $n$ objects, $\mathbb{C} = \{\mathcal{A}_1, \cdots, \mathcal{A}_k, \cdots, \mathcal{A}_m\}$ be the set of $m$ condition attributes or features, and $\mathbb{D}$ is the decision attribute set in $\mathbb{U}$. Let $\mathbb{U}/\mathbb{D} = \{\omega_1, \cdots, \omega_i, \cdots, \omega_c\}$ denote $c$ equivalence classes or information granules of $\mathbb{U}$ generated by the equivalence relation induced from the decision attribute set $\mathbb{D}$. The proposed method assumes that each class $\omega_i$ can be approximated by a pair of sets, namely, lower approximation $\underline{A}(\omega_i)$ and boundary region $B(\omega_i)$, based on the theory of rough sets. The upper approximation $\overline{A}(\omega_i) = [\underline{A}(\omega_i) \cup B(\omega_i)]$ [149]. If an object belongs to the lower approximation $\underline{A}(\omega_i)$ of a class $\omega_i$, it definitely belongs to that class. On the other hand, if an object belongs to the boundary region $B(\omega_i)$, it possibly belongs to that class and potentially belongs to another class. Both lower and upper approximations of a class $\omega_i$ can be defined by the equivalence classes of $\mathbb{U}$ generated by the equivalence relation induced from each condition attribute $\mathcal{A}_k \in \mathbb{C}$. The equivalence classes or information granules, corresponding to the condition attribute $\mathcal{A}_k$ or the set $\mathbb{C}$, are constructed based on Bayes

**Algorithm 3.4** Proposed Method for HEp-2 Cell Classification

**Input:** Set of training cell images $X$.
**Output:** Final feature set $\mathcal{D}$.

1: **for** each modality $\mathcal{M}_p \in \mathcal{M}$ **do**

(i) Obtain the set of dominant features for each of the training cell images $x_j \in X$ by **Algorithm 3.1**.

(ii) Determine class specific feature set $\mathcal{N}_p(\omega_i)$, where $i = 1, \cdots, c$, using **Algorithm 3.2**.

2: **end for**
3: **for** each class-pair $\{\omega_i, \omega_r\}$ **do**

- Select a set of class-pair specific feature sets $\tilde{\mathcal{N}}_{ir} = \{\mathcal{N}_p(\{\omega_i, \omega_r\})\}$, using **Algorithm 3.3**.

4: **end for**
5: Form the final feature set $\mathcal{D}$, from all the selected class-pair specific feature sets $\tilde{\mathcal{N}}_{ir}$'s, corresponding to all pairs of classes.

decision theory, considering two-class problem.

Let $\omega_1$ and $\omega_2$ be the two classes in which the objects belong. Let us assume that the a priori probabilities $P(\omega_1)$ and $P(\omega_2)$ of two classes are either known or they can be estimated from the available training objects. Further, the class-conditional probability density functions (pdf) $p(x|\omega_i), i = 1, 2$; representing the distribution of the feature vectors in each of the classes, are assumed to be known. The pdf is also referred to as likelihood function of $\omega_i$ with respect to $x$. Given the above inputs, the task is to compute the conditional probabilities $P(\omega_i|x), i = 1, 2$; each of them represents the probability that the object $x$ belongs to the respective class $\omega_i$. Integrating the theory of rough sets and Bayes decision theory, the classification rules for the proposed method are formulated as follows:

- **R1:** $x$ is classified to $\omega_1$ if $P(\omega_1|x) - P(\omega_2|x) > \lambda$;
- **R2:** $x$ is classified to $\omega_2$ if $P(\omega_1|x) - P(\omega_2|x) < -\lambda$;
- **R3:** $x$ is unclassified if $|P(\omega_1|x) - P(\omega_2|x)| \leqslant \lambda$.

The parameter $\lambda(\geqslant 0)$ controls the region of uncertainty due to inexactness in class definitions of $\omega_1$ and $\omega_2$, and overlapping boundaries between $\omega_1$ and $\omega_2$. According to Bayes rule,

$$P(\omega_i|x) = \frac{p(x|\omega_i)P(\omega_i)}{p(x)}; \quad i = 1, 2. \tag{3.13}$$

$$\text{Also,} \quad P(\omega_1|x) + P(\omega_2|x) = 1. \tag{3.14}$$

Combining rule **R1**, (3.13), and (3.14), we get

$$x \text{ is classified to } \omega_1 \text{ if } \frac{P(\omega_1)p(x|\omega_1)}{P(\omega_2)p(x|\omega_2)} > \frac{1+\lambda}{1-\lambda}. \tag{3.15}$$

Let us assume that the likelihood functions $p(x|\omega_i)$ of $\omega_i$ with respect to $x$ are multivariate normal distributions, that is,

$$p(x|\omega_i) = \frac{1}{\sqrt{(2\pi)^m|\Sigma_i|}} \exp\left\{-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)\right\};$$

where $\mu_i$ and $\Sigma_i$ are the mean of class $\omega_i$ and corresponding $m \times m$ covariance matrix, while $|\Sigma_i|$ denotes the determinant of $\Sigma_i$. So, the classification rule of **R1** reduces to:

$$x \text{ is classified to } \omega_1 \text{ if } d_{\mathrm{M}}^2(x,\mu_2,\Sigma_2) - d_{\mathrm{M}}^2(x,\mu_1,\Sigma_1) - \ln\frac{|\Sigma_1|}{|\Sigma_2|} > \Delta_{12}; \tag{3.16}$$

$$\text{where } \Delta_{12} = 2\left[\ln\left(\frac{1+\lambda}{1-\lambda}\right) - \ln\frac{P(\omega_1)}{P(\omega_2)}\right]; \tag{3.17}$$

$$\text{and } d_{\mathrm{M}}(x,\mu_i,\Sigma_i) = \sqrt{(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)} \tag{3.18}$$

is the Mahalanobis distance between $x$ and $\mu_i$.

Assuming the individual features, constituting the feature vector, are mutually uncorrelated, we get

$$\Sigma_i = \mathrm{diag}(\sigma_{i1}^2, \cdots, \sigma_{ik}^2, \cdots, \sigma_{im}^2); \tag{3.19}$$

where $\sigma_{ik}^2$ is the variance of the $k$-th feature. So,

$$x \text{ is classified to } \omega_1 \text{ if}$$
$$\sum_{k=1}^{m}\left[\frac{(x_k-\mu_{2k})^2}{\sigma_{2k}^2} - \frac{(x_k-\mu_{1k})^2}{\sigma_{1k}^2} - 2\ln\left(\frac{\sigma_{1k}}{\sigma_{2k}}\right)\right] > \Delta_{12}. \tag{3.20}$$

Similarly, from rules **R2** and **R3**, we get

$$x \text{ is classified to } \omega_2 \text{ if}$$
$$\sum_{k=1}^{m}\left[\frac{(x_k-\mu_{2k})^2}{\sigma_{2k}^2} - \frac{(x_k-\mu_{1k})^2}{\sigma_{1k}^2} - 2\ln\left(\frac{\sigma_{1k}}{\sigma_{2k}}\right)\right] < -\Delta_{12}; \tag{3.21}$$

$$\text{and } x \text{ is unclassified if}$$
$$\left|\sum_{k=1}^{m}\left[\frac{(x_k-\mu_{2k})^2}{\sigma_{2k}^2} - \frac{(x_k-\mu_{1k})^2}{\sigma_{1k}^2} - 2\ln\left(\frac{\sigma_{1k}}{\sigma_{2k}}\right)\right]\right| \leqslant \Delta_{12}. \tag{3.22}$$

Based on the concept of positive approximation accelerator, introduced in [126, 154], the computing performance of a condition attribute set $\mathbb{C}_{k+1} = \{\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_{k+1}\}$ can be improved using the following recursive expression principle:

$$POS^{\mathbb{U}_k}_{\mathbb{C}_{k+1}}(\mathbb{D}) = POS^{\mathbb{U}_k}_{\mathbb{C}_k}(\mathbb{D}) \cup POS^{\mathbb{U}_{k+1}}_{\mathcal{A}_{k+1}}(\mathbb{D}); \tag{3.23}$$

$$\text{where } \mathbb{U}_{k+1} = \mathbb{U}_k \backslash POS^{\mathbb{U}_k}_{\mathbb{C}_k}(\mathbb{D}); \quad \mathbb{U}_1 = \mathbb{U}; \quad \text{and } POS^{\mathbb{U}}_{\mathbb{C}}(\mathbb{D}) = \bigcup_{\omega_i \in \mathbb{U}/\mathbb{D}} \underline{A}(\omega_i). \tag{3.24}$$

Here, $POS^{\mathbb{U}}_{\mathbb{C}}(\mathbb{D})$ is termed as the positive region, which contains all objects of $\mathbb{U}$ that can be classified to classes of $\mathbb{U}/\mathbb{D}$ using the knowledge in attribute set $\mathbb{C}$. In effect, the decision attribute set $\mathbb{D}$ can be positively approximated using granulation orders $\mathbb{C}_k$ and $\mathbb{C}_{k+1}$ on the gradually reduced universe, respectively. So, instead of considering all $m$ features of $\mathbb{C}$ to generate information granules or equivalence classes, equivalence classes corresponding to each individual feature $\mathcal{A}_k \in \mathbb{C}$ can be generated separately, and then the decision attribute $\mathbb{D}$ can be positively approximated by the equivalence classes of individual features.

Let $\mathbb{U}/\mathcal{A}_k = \{\beta_1, \cdots, \beta_i, \cdots, \beta_{\tilde{c}}\}$ denote $\tilde{c}$ equivalence classes or information granules of $\mathbb{U}$ induced by the condition attribute $\mathcal{A}_k$. Based on Bayes decision theory, three information granules, namely, $\beta_1$, $\beta_2$ and $\beta_3$, are constructed for two classes corresponding to the condition attribute $\mathcal{A}_k$, as follows:

$$x_j \in \beta_1 \quad \text{if } \left[\frac{(x_{jk} - \mu_{2k})^2}{\sigma^2_{2k}} - \frac{(x_{jk} - \mu_{1k})^2}{\sigma^2_{1k}} - 2\ln\left(\frac{\sigma_{1k}}{\sigma_{2k}}\right)\right] > \tilde{\Delta}_{12}; \tag{3.25}$$

$$x_j \in \beta_2 \quad \text{if } \left[\frac{(x_{jk} - \mu_{2k})^2}{\sigma^2_{2k}} - \frac{(x_{jk} - \mu_{1k})^2}{\sigma^2_{1k}} - 2\ln\left(\frac{\sigma_{1k}}{\sigma_{2k}}\right)\right] < -\tilde{\Delta}_{12}; \tag{3.26}$$

$$\text{and } x_j \in \beta_3 \quad \text{if } \left|\left[\frac{(x_{jk} - \mu_{2k})^2}{\sigma^2_{2k}} - \frac{(x_{jk} - \mu_{1k})^2}{\sigma^2_{1k}} - 2\ln\left(\frac{\sigma_{1k}}{\sigma_{2k}}\right)\right]\right| \leqslant \tilde{\Delta}_{12}; \tag{3.27}$$

$$\text{where } \tilde{\Delta}_{12} = \frac{\Delta_{12}}{m} = 2\left[\ln\left(\frac{1+\delta}{1-\delta}\right) - \ln\frac{P(\omega_1)}{P(\omega_2)}\right]. \tag{3.28}$$

This can be viewed as a supervised granulation process, which utilizes class information. The parameter $\delta$ has the same interpretation as that of $\lambda$ in (3.17). Combining (3.17) and (3.28), the following relation holds between $\lambda$ and $\delta$:

$$\lambda = 1 - 2\left[1 + \left(\frac{1+\delta}{1-\delta}\right)^m \left(\frac{P(\omega_1)}{P(\omega_2)}\right)^{1-m}\right]^{-1}. \tag{3.29}$$

Let $\omega_i \subseteq \mathbb{U}$, and $E(\beta_p, \omega_i)$ be the relative degree of misclassification of the set $\beta_p$ with

respect to the set $\omega_i$, which is as follows [228]:

$$E(\beta_p, \omega_i) = \begin{cases} 1 - \frac{|\beta_p \cap \omega_i|}{|\beta_p|}, & \text{if } |\beta_p| > 0; \\ 0, & \text{otherwise.} \end{cases} \tag{3.30}$$

The quantity $E(\beta_p, \omega_i)$ is also referred to as the relative classification error. The number of misclassified objects is given by the product of $E(\beta_p, \omega_i)$ and $|\beta_p|$, which is referred to as an absolute classification error. According to the theory of variable precision rough sets [228], each class $\omega_i$ can be approximated using the measure $E(\beta_p, \omega_i)$ by constructing the $\alpha$-lower and $\alpha$-upper approximations of $\omega_i$, where $\alpha$ is the admissible classification error and $\alpha \in [0.0, 0.5)$ as per the majority requirement. The $\alpha$-lower approximation $\underline{A}_\alpha(\omega_i)$, $\alpha$-upper approximation $\overline{A}_\alpha(\omega_i)$, and $\alpha$-boundary region $B_\alpha(\omega_i)$ of the class $\omega_i$ can be defined as follows [228]:

$$\underline{A}_\alpha(\omega_i) = \bigcup \{\beta_p \mid E(\beta_p, \omega_i) \leqslant \alpha\}; \tag{3.31}$$

$$\overline{A}_\alpha(\omega_i) = \bigcup \{\beta_p \mid E(\beta_p, \omega_i) < 1 - \alpha\}; \tag{3.32}$$

$$B_\alpha(\omega_i) = \bigcup \{\beta_p \mid \alpha < E(\beta_p, \omega_i) < 1 - \alpha\}. \tag{3.33}$$

Hence, the lower approximation $\underline{A}_\alpha(\omega_i)$ is the collection of those elements of $\mathbb{U}$ that can be classified into $\omega_i$ with the classification error not greater than $\alpha$.

Using (3.24) and (3.31), the $\alpha$-dependency $\gamma_{\mathcal{A}_k}(\mathbb{D})$, also known as degree of dependency, of decision attribute $\mathbb{D}$ on the condition attribute $\mathcal{A}_k$ can be computed as follows [149,228]:

$$\gamma_{\mathcal{A}_k}(\mathbb{D}) = \frac{|POS^{\mathbb{U}}_{\mathcal{A}_k}(\mathbb{D})|}{|\mathbb{U}|} = \frac{1}{n} \left| \bigcup_{\omega_i \in \mathbb{U}/\mathbb{D}} \underline{A}_\alpha(\omega_i) \right|; \tag{3.34}$$

where $\gamma_{\mathcal{A}_k}(\mathbb{D}) \in [0, 1]$. However, as $\alpha$-lower approximation of a class contains both correctly classified and misclassified objects, the $\gamma$ measure of (3.34) fails to capture discriminative characteristics of a feature properly. In order to circumvent the above problem, a relevance measure is introduced next by discarding the adverse effect of misclassified objects.

The correctly classified objects belonging to the $\alpha$-lower approximations of $c$ classes can be arrayed as a $(c \times n)$ matrix $\mathbb{L}(\mathcal{A}_k) = [\mathrm{L}_{ij}(\mathcal{A}_k)]$, where $i$th row of the matrix

$$\mathrm{L}_i(\mathcal{A}_k) = \{\mathrm{L}_{i1}(\mathcal{A}_k)/x_1 + \mathrm{L}_{i2}(\mathcal{A}_k)/x_2 + \cdots + \mathrm{L}_{in}(\mathcal{A}_k)/x_n\}$$

corresponds to $\alpha$-lower approximation $\underline{A}_\alpha(\omega_i)$ of the $i$th class $\omega_i$, "+" means the operator of union in this case, and

$$\mathrm{L}_{ij}(\mathcal{A}_k) = \begin{cases} 1, & \text{if } x_j \in \underline{A}_\alpha(\omega_i) \text{ and } x_j \in \omega_i; \\ 0, & \text{otherwise.} \end{cases} \tag{3.35}$$

So, $\mathrm{L}_i(\mathcal{A}_k) \subseteq \underline{A}_\alpha(\omega_i)$. Since a correctly classified object belongs to the lower approximation of only one class, each column of $\mathbb{L}(\mathcal{A}_k)$ must contain at most one 1. Based on

the above definition, the relevance of a condition attribute $\mathcal{A}_k$, with respect to $c$ classes $\{\omega_1, \cdots, \omega_i, \cdots, \omega_c\}$, is defined as

$$\mathrm{Rel}_{\mathcal{A}_k}(\mathbb{D}) = \frac{1}{n} \sum_{i=1}^{c} \sum_{j=1}^{n} \mathrm{L}_{ij}(\mathcal{A}_k); \tag{3.36}$$

where $0 \leqslant \mathrm{Rel}_{\mathcal{A}_k}(\mathbb{D}) \leqslant 1$. Higher value of relevance measure indicates better attribute or feature for classification. In this regard, it should be noted that, for $\alpha = 0$, the definitions of set or class approximations, given in (3.31), (3.32) and (3.33) based on variable precision rough sets, reduce to that of Pawlak's rough sets [149]. So, for $\alpha = 0$, $\mathrm{L}_i(\mathcal{A}_k) = \underline{A}_0(\omega_i) = \underline{A}(\omega_i)$, and the relevance measure $\mathrm{Rel}_{\mathcal{A}_k}(\mathbb{D})$ defined in (3.36) is exactly same with the degree of dependency $\gamma_{\mathcal{A}_k}(\mathbb{D})$ reported in (3.34).

Given a set $\mathbb{C} = \{\mathcal{A}_1, \cdots, \mathcal{A}_k, \cdots, \mathcal{A}_m\}$ of $m$ condition attributes, in general, the equivalence classes corresponding to the set $\mathbb{C}$ can be formed from that of individual attributes. Based on the resultant equivalence classes of $\mathbb{C}$, the $\alpha$-lower approximations of $c$ classes can be found out using (3.31), and accordingly $\gamma_{\mathbb{C}}(\mathbb{D})$ or $\mathrm{Rel}_{\mathbb{C}}(\mathbb{D})$ can be computed using (3.34) or (3.36), respectively. However, as both $\gamma_{\mathbb{C}}(\mathbb{D})$ and $\mathrm{Rel}_{\mathbb{C}}(\mathbb{D})$, computed this way, consider only dependency in multidimensional feature space and do not take into account the relevance of individual attributes, they fail to identify relevant features [126, 127]. In order to identify relevant as well as complementary features, the union of $m$ individual $(c \times n)$ matrices $\{\mathbb{L}(\mathcal{A}_k)\}$, corresponding to $m$ condition attributes, is considered, which is as follows:

$$\tilde{\mathbb{L}}(\mathbb{C}) = \bigcup_{\mathcal{A}_k \in \mathbb{C}} \mathbb{L}(\mathcal{A}_k); \tag{3.37}$$

and corresponding relevance $\mathrm{Rel}_{\mathbb{C}}(\mathbb{D})$ is computed from $\tilde{\mathbb{L}}(\mathbb{C})$ using (3.36). Obviously,

$$\mathrm{Rel}_{\{\mathcal{A}_k, \mathcal{A}_l\}}(\mathbb{D}) \geqslant \max\{\mathrm{Rel}_{\mathcal{A}_k}(\mathbb{D}), \mathrm{Rel}_{\mathcal{A}_l}(\mathbb{D})\}. \tag{3.38}$$

**Algorithm 3.5** Computation of Relevance of Feature Set

**Input:** A set of objects $\mathbb{U} = \{x_1, \cdots, x_j, \cdots, x_n\}$, a set of condition attributes $\mathbb{C} = \{\mathcal{A}_1, \cdots, \mathcal{A}_k, \cdots, \mathcal{A}_m\}$, decision attribute set $\mathbb{D}$, and admissible classification error $\alpha$.
**Output:** $\mathrm{Rel}_{\mathbb{C}}(\mathbb{D})$, relevance of condition attribute set $\mathbb{C}$ with respect to decision attribute set $\mathbb{D}$.

1: **for** each condition attribute or feature $\mathcal{A}_k \in \mathbb{C}$ **do**
   - Form equivalence classes or information granules.
   - Construct $\alpha$-lower approximation for each class $\omega_i \in \mathbb{D}$ using (3.31).
   - Construct the matrix $\mathbb{L}(\mathcal{A}_k)$ using (3.35).

2: **end for**
3: Construct the matrix $\tilde{\mathbb{L}}(\mathbb{C})$, corresponding to the set $\mathbb{C}$, from $m$ number of $\mathbb{L}(\mathcal{A}_k)$ matrices using (3.37).
4: Calculate the relevance $\mathrm{Rel}_{\mathbb{C}}(\mathbb{D})$ of $\mathbb{C}$ with respect to $\mathbb{D}$ from $\tilde{\mathbb{L}}(\mathbb{C})$ using (3.36).

## 3.4 Computational Complexity

The basic steps of the proposed method to select the class-pair specific texture descriptors under appropriate scales for HEp-2 cell classification are outlined in Algorithm 3.4. In this section, the computational complexity of the proposed approach is presented.

Let us assume that $\mathcal{M} = \{\mathcal{M}_1, \cdots, \mathcal{M}_p, \cdots, \mathcal{M}_t\}$ represents a set of $t$ modalities, where each modality $\mathcal{M}_p$ corresponds to a local descriptor considered under a particular scale. Let $m$ denote the average cardinality of the feature set, corresponding to a particular modality, and $n$ is the number of HEp-2 cell images. At first, a set of dominant features under a particular modality $\mathcal{M}_p$ is obtained for each training HEp-2 cell image by Algorithm 3.1. The sorting operation of normalized histogram of $\mathcal{M}_p$ has a computational complexity of $\mathcal{O}(m \log m)$. The complexity of each of the steps, required for determination of cumulative sum, second-order difference and number of dominant features, is $\mathcal{O}(m)$. Now, the operations performed in Algorithm 3.1 are executed for $n$ images and $t$ modalities. Hence, the overall computational complexity for obtaining the set of dominant features is $(\mathcal{O}(tnm \log m) + \mathcal{O}(tnm)) \simeq \mathcal{O}(tnm \log m)$.

Let $\mathcal{N}_p(\omega_i)$ denote the class specific feature set for the class $\omega_i$ under the $p$-th modality $\mathcal{M}_p$ and $\tilde{n}$ is the average number of HEp-2 cell samples belonging to a particular class. The set $\mathcal{N}_p(\omega_i)$ is determined using Algorithm 3.2. The probability of occurrence of the features present in the dominant sets of samples belonging to the class $\omega_i$ is calculated with a complexity of $\mathcal{O}(\tilde{n})$. The computation of histogram $H(\omega_i)$ of probability of occurrence values in $\omega_i$ has complexity $\mathcal{O}(m)$. Considering $L$ number of bins in $H(\omega_i)$, the class specific threshold for the class $\omega_i$ is determined from the corresponding histogram with a complexity $\mathcal{O}(L)$. Finally, the set $\mathcal{N}_p(\omega_i)$ is formed with a complexity $\mathcal{O}(m)$. Now, it can be seen that all the steps of Algorithm 3.2 are executed for $c$ classes and $t$ modalities. So, the Algorithm 3.2 leads to an overall complexity of $\mathcal{O}(tc(\tilde{n} + m + L))$.

As indicated in Algorithm 3.3, the class-pair specific feature set $\mathcal{N}_p(\{\omega_i, \omega_r\})$ for each pair of classes $\{\omega_i, \omega_r\}$ under the modality $\mathcal{M}_p$ is computed from the class specific feature sets $\mathcal{N}_p(\omega_i)$ and $\mathcal{N}_p(\omega_r)$ of the classes $\omega_i$ and $\omega_r$, respectively. As this step does not include any information from other class-pair as well as other modalities, the set $\mathcal{N}_p(\{\omega_i, \omega_r\})$ is formed with a complexity of $\mathcal{O}(m)$. The relevance $\text{Rel}_{\mathbb{C}}(\mathbb{D})$ of the decision attribute set $\mathbb{D}$, containing the class label information of the samples belonging to $\{\omega_i, \omega_r\}$, on the condition attribute set $\mathbb{C} = \mathcal{N}_p(\{\omega_i, \omega_r\})$ is determined using Algorithm 3.3 For each feature $\mathcal{A}_k \in \mathcal{N}_p(\{\omega_i, \omega_r\})$, both means and standard deviations of two classes $\omega_i$ and $\omega_r$ are calculated with the complexity of $\mathcal{O}(\tilde{n})$. Similarly, $\tilde{c}$ number of equivalence partitions induced by the condition attribute $\mathcal{A}_k$, $\alpha$-lower approximations of two classes and the corresponding $\mathbb{L}(\mathcal{A}_k)$ matrix are formed with a complexity of $\mathcal{O}(\tilde{n})$. So, the computation of relevance of the feature set $\mathcal{N}_p(\{\omega_i, \omega_r\})$ using Algorithm 3.5 has $\mathcal{O}(m\tilde{n})$ complexity. Now, Algorithm 3.5 is executed for $t$ modalities and $\binom{c}{2}$ class-pairs. Hence, the total complexity is $\mathcal{O}(tc^2m\tilde{n})$. After computing the relevance values, $\tilde{t}$ most relevant feature sets are chosen from $t$ feature sets, corresponding to each pair of classes, with a complexity $\mathcal{O}(t)$ as $\tilde{t} << t$. So, for $\binom{c}{2}$ class-pairs, the total complexity is $\mathcal{O}(tc^2)$. Hence, the overall complexity of Algorithm 3.5 is $(\mathcal{O}(tc^2m\tilde{n}) + \mathcal{O}(tc^2)) \simeq \mathcal{O}(tc^2m\tilde{n})$.

Finally, in Algorithm 3.4, the resultant feature set $\mathcal{D}$ is formed as the union of $(\tilde{t} \cdot \binom{c}{2})$ feature sets. In order to reduce the computational complexity, $\mathcal{D}$ is represented as a binary string of zeros and ones with cardinality $(t \cdot m)$. The features, which are present

in the selected class-pair specific feature sets, are denoted with ones at the corresponding positions of the string, otherwise marked as zeros. Hence, $\mathcal{D}$ is formed with a computational complexity of $\mathcal{O}(tc^2m)$ as $\tilde{t} << t$. In effect, the computational complexity of the proposed method for selecting relevant and complementary features under various modalities for HEp-2 cell classification is $\mathcal{O}(tnm \log m)+\mathcal{O}(tcmn)+ \mathcal{O}(tcL)$ as $n = c \cdot \tilde{n}$.

## 3.5 Experimental Results and Discussions

This section presents the performance of the proposed descriptor selection method for HEp-2 cell image classification, along with a comparison with related approaches. The methods compared are several local texture descriptors, namely, LBP [148], $\text{LBP}^{\text{ri}}$ [147], $\text{LBP}^{\text{riu2}}$ [147], CoALBP [146], CLBP [61], and RICLBP [145], as well as several state-of-the-art methods for HEp-2 cell classification. The performance of different methods is also compared with respect to various scales of LBP neighborhood such as 1 ($S_1$), 2 ($S_2$), 3 ($S_3$), 4 ($S_4$), $S_{123}$: concatenation of ($S_1$, $S_2$, $S_3$), and $S_{124}$: concatenation of ($S_1$, $S_2$, $S_4$). The 4-neighborhood is used for CoALBP and RICLBP, while for others 8-neighborhood is considered. The SVM [197], discussed in Appendix C, is used to evaluate the performance of different local texture descriptors. Both training-testing and ten-fold cross-validation (CV) are performed to compute the classification accuracy. The comparative performance analysis of different algorithms is also studied using tables of means, medians, standard deviations, and p-values computed through paired-$t$ and Wilcoxon signed-rank tests, considering 95% confidence level.

### 3.5.1 Description of Data Sets

Three HEp-2 cell image databases, namely, MIVIA database (ICPR 2012 HEp-2 cell classification contest data set) [51], ICPR image database (ICPR 2014 HEp-2 cell classification contest data set) [75], and SNP HEp-2 database [206], are considered for the evaluation of the proposed method as well as related existing approaches. The cell images of the above three data sets were captured in different laboratory settings. For instance, SNP HEp-2 used objective lens magnitude 20×, while MIVIA HEp-2 used 40×. The MIVIA database contains 1455 cells obtained from 28 images: 721 training cell images and 734 test cell images. Six staining pattern classes, namely, Cytoplasmic, Fine Speckled, Nucleolar, Coarse Speckled, Homogeneous, and Centromere, are considered in this case. The ICPR database contains the set of 13,596 HEp-2 cell images, which is partitioned into 6797 training and 6799 test cell images. The cell images belong to six different patterns, namely, Centromere, Homogeneous, Nucleolar, Speckled, Nuclear Membrane, and Golgi. The SNP database contains 1806 labelled cell images obtained from forty slide images: 869 training images and 937 test images. The samples are grouped into five pattern classes, namely, Centromere, Homogeneous, Fine Speckled, Coarse Speckled and Nucleolar. Each of the three sets of 1455, 13596, and 1806 cell images of MIVIA, ICPR, and SNP databases, respectively, is also split into ten separate folds for ten-fold CV. In both training and test sets as well as in ten folds, the cell images are almost equally distributed with respect to different staining pattern classes. A detailed description of the data sets is reported in Appendix A.

### 3.5.2 Optimum Values of Different Parameters

The performance of the proposed modality selection method depends on the values of both $\delta$ and $\alpha$. While $\delta$ controls the region of uncertainty due to overlapping class boundaries between two pattern classes and inexactness in class definition, $\alpha$ imposes an explicit limitation on the admissible level of classification error. To find out the optimum value of $\delta$, extensive experiment is carried out on three HEp-2 image databases by varying the values of $\delta$ from 0.00 to 0.95 and corresponding results are presented in first graph of Figure 3.5. From the results reported in Figure 3.5, it is seen that the performance of the proposed method increases with the increase in value of $\delta$ upto 0.40, irrespective of the databases and experimental setup used. For $\delta > 0.40$, the performance remains almost constant in all the cases. So, the optimum value of $\delta$ is considered as 0.40. It ensures that an object definitely belongs to one of the staining pattern classes if its probability of belongingness in that class is at least 0.70.

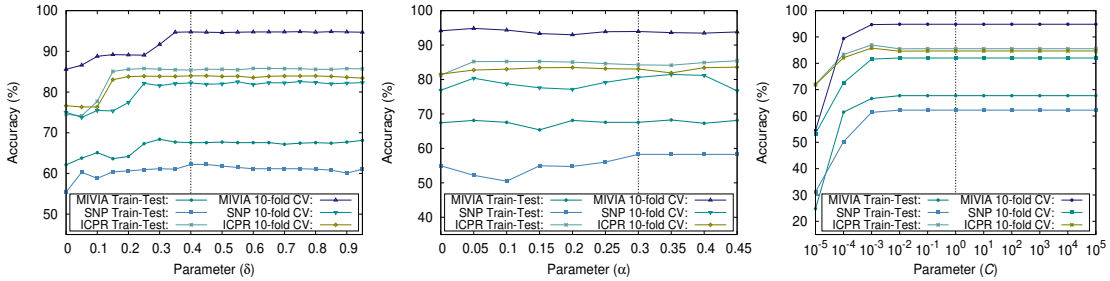



Figure 3.5: Variation of classification accuracy for different values of $\delta$, $\alpha$ and $C$.

Keeping the value of $\delta$ at 0.4, $\alpha$ is varied from 0.00 to 0.45 and the corresponding results are plotted in middle graph of Figure 3.5. All the results reported in Figure 3.5 reveal that the stable performance of the algorithm is obtained for medium values of $\alpha$ and considerable classification accuracy is achieved at $\alpha = 0.30$ for all the three databases, irrespective of experimental setup used. Hence, the value of $\alpha$ is considered to be 0.30 for the rest of the study. The optimum values of $\alpha = 0.3$ and $\delta = 0.4$ ensure that an object can be classified to one of the staining pattern classes with classification error not greater than 0.3, if its probability of belongingness in that class is at least 0.7.

The SVM [197] with linear kernel is used to analyze the performance of the proposed method. It is based on supervised statistical learning theory which tries to obtain an optimized hyperplane in the kernel space by maximizing the margin and minimizing the classification error between the given classes. The robust performance of the SVM is achieved by properly adjusting the parameter $C$, known as the penalty parameter in the kernel function. For large value of $C$, error minimization is predominant, that is, the optimization will choose a hyperplane with smaller margin so that all the training samples are correctly classified. Conversely, if the value of $C$ is small, then margin maximization is emphasized, even if more training samples are misclassified. So, in the proposed method, exhaustive search is conducted on a finite set of values of $C$, ranging from $10^{-5}$ to $10^{5}$, and the corresponding results are reported in last graph of Figure 3.5. From the results, it can be observed that for $C > 0.01$, the performance of the proposed method remains constant. So, the optimum value of $C$ is set at an intermediate value, which is 1 in this case.

### 3.5.3 Effectiveness of Difference Operator and Threshold

Prior to selecting relevant local texture descriptors under appropriate scales, the proposed method takes care of the presence of noisy pixels in an HEp-2 cell image as well as noisy images in a staining pattern class. The proposed approach considers the second-order derivative of $\mathcal{S}$ function and class specific threshold, introduced in (3.1) and (3.9), respectively, to make itself insensitive to noisy pixels and noisy images. In order to establish the importance of both $\mathcal{S}$ function and threshold, extensive experiment is carried out on three HEp-2 cell image databases. Figure 3.6 and Table 3.4 compare the performance of the proposed method with and without $\mathcal{S}$ function and class specific threshold.

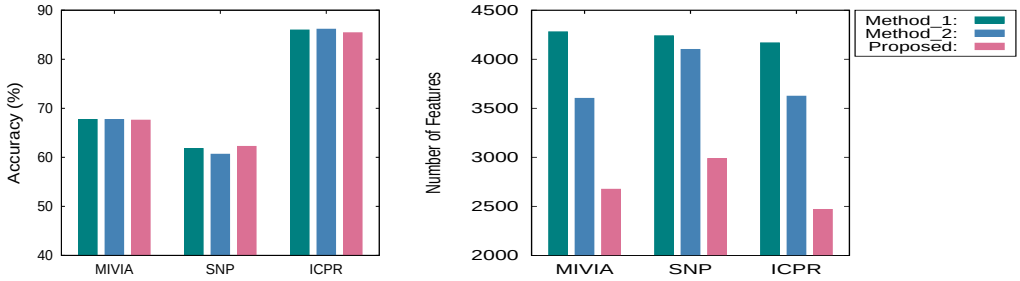



Figure 3.6: Performance of proposed method with and without $\mathcal{S}$ function and threshold for training-testing.

The Method_1 is the proposed method without using both second-order derivative of $\mathcal{S}$ function and threshold, while Method_2 is the proposed method without using only class specific threshold. All the results reported in Figure 3.6 and Table 3.4 establish the fact that the introduction of both $\mathcal{S}$ function and threshold helps to identify class specific relevant features. While the second-order derivative of $\mathcal{S}$ function helps to eliminate the effect of noisy pixels present in HEp-2 cell images, the class specific threshold takes care of the presence of noisy images in staining pattern classes. In effect, the proposed method obtains comparable classification accuracy, in both training-testing and 10-fold CV, with significantly lesser number of features compared to both Method_1 and Method_2.

### 3.5.4 Significance of Proposed Relevance Measure

In the current study, $\alpha$-lower and $\alpha$-upper approximations of a staining pattern class are obtained according to the theory of variable precision rough sets. The $\gamma$-measure takes into account the union of $\alpha$-lower approximations over all the classes, to compute the degree of dependency of decision attribute set on the condition attribute set. However, the $\alpha$-lower approximation of a class contains both the correctly classified and misclassified samples. So, the value of $\gamma$ fails to reflect the discriminative characteristics of the condition attribute set. On the other hand, the proposed relevance measure considers only the correctly classified samples of the $\alpha$-lower approximation of a class, where higher relevance value signifies better classification ability of the feature set.

In order to establish the effectiveness of proposed relevance measure with respect to $\gamma$-measure (henceforth, termed as Method_3), extensive experimentation is done on three HEp-2 cell image databases, and corresponding results are reported in Figure 3.7 for

Table 3.4: Performance of Proposed Method With and Without $\mathcal{S}$ Function and Threshold for 10-Fold CV

| Different Measures | Different Approaches | Mean | Median | StdDev | Paired-$t$:p | Wilcoxon:p |
|---|---|---|---|---|---|---|
| | | MIVIA | | | | |
| Accuracy | Method_1 | **94.90** | **95.58** | 1.67 | 6.21E-01 | 7.03E-01 |
| | Method_2 | **94.90** | 95.24 | **1.61** | 6.61E-01 | 6.60E-01 |
| | Proposed | 94.83 | 94.90 | 1.93 | - | - |
| Number of Features | Method_1 | 4188.00 | 4204.00 | 22.49 | 1.00E+00 | 9.98E-01 |
| | Method_2 | 3690.20 | 3691.00 | 20.69 | 1.00E+00 | 9.98E-01 |
| | Proposed | **2736.40** | **2737.50** | **15.77** | - | - |
| | | SNP | | | | |
| Accuracy | Method_1 | 81.91 | 81.15 | 6.79 | 4.24E-01 | 3.37E-01 |
| | Method_2 | **82.19** | 80.87 | 7.02 | 6.09E-01 | 5.24E-01 |
| | Proposed | 82.02 | **81.42** | **6.32** | - | - |
| Number of Features | Method_1 | 4044.80 | 4096.00 | 107.94 | 1.00E+00 | 9.98E-01 |
| | Method_2 | 4017.20 | 4043.50 | 80.41 | 1.00E+00 | 9.98E-01 |
| | Proposed | **2655.80** | **2674.50** | **68.10** | - | - |
| | | ICPR | | | | |
| Accuracy | Method_1 | **84.67** | 84.63 | 3.13 | 9.98E-01 | 9.97E-01 |
| | Method_2 | 84.56 | **84.67** | **2.90** | 9.93E-01 | 9.91E-01 |
| | Proposed | 84.06 | 83.75 | 3.24 | - | - |
| Number of Features | Method_1 | 4145.20 | 4173.00 | 82.52 | 1.00E+00 | 9.98E-01 |
| | Method_2 | 3682.80 | 3684.00 | 18.40 | 1.00E+00 | 9.98E-01 |
| | Proposed | **2549.80** | **2547.00** | **15.28** | - | - |
| Method_1: Without applying second-order derivative of $\mathcal{S}$ function and threshold; Method_2: Without applying threshold. | | | | | | |

training-testing and Table 3.5 for 10-fold CV. From the results reported in Figure 3.7 and Table 3.5, it can be observed that the proposed method with new relevance measure performs significantly better than Method_3, for all the three HEp-2 image data sets. Figure 3.7 and Table 3.5 also present the comparative performance analysis between proposed relevance measure with multidimensional approach (henceforth, termed as Method_4) and that with union of individual $\mathbb{L}$ matrices (proposed). All the results reported in Figure 3.7 and Table 3.5 indicate that the significantly increased classification accuracy can be achieved using the proposed method to compute the relevance measure, irrespective of data sets considered.

### 3.5.5 Relevance of Rough Sets and Bayes Decision Theory

For each pair of classes, the proposed method selects a set of features by evaluating its relevance. The set of features corresponds to a local texture descriptor selected under a particular scale. To compute the relevance of a feature set, the proposed method uses a new relevance measure, which is based on judicious integration of variable precision rough sets and Bayes decision theory. However, other feature evaluation measures such as

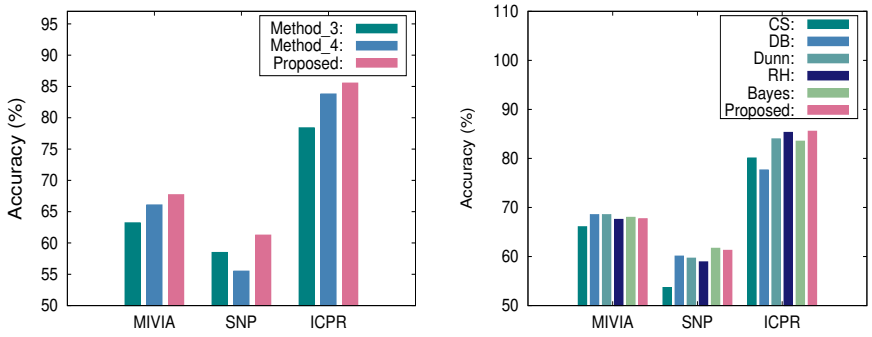



Figure 3.7: Performance analysis of proposed method with - left: $\gamma$-measure (Method_3) and multidimensional approach (Method_4); right: different feature evaluation indices for training-testing.

Table 3.5: Performance Analysis of Different Measures for Computing Relevance Using 10-Fold CV

| Different Measures | Mean | Median | StdDev | Paired-$t$:p | Wilcoxon:p |
|---|---|---|---|---|---|
| | MIVIA | | | | |
| Method_3 | 87.89 | 88.10 | 2.05 | 4.19E-05 | 2.53E-03 |
| Method_4 | 93.81 | 94.22 | 2.01 | 3.09E-02 | 4.24E-02 |
| Proposed | **94.83** | **94.90** | **1.93** | - | - |
| | SNP | | | | |
| Method_3 | 76.12 | 77.05 | 6.00 | 2.54E-03 | 6.26E-03 |
| Method_4 | 77.38 | 76.78 | **5.44** | 8.17E-04 | 3.44E-03 |
| Proposed | **82.02** | **81.42** | 6.32 | - | - |
| | ICPR | | | | |
| Method_3 | 77.16 | 75.61 | 5.05 | 3.14E-06 | 2.53E-03 |
| Method_4 | 82.72 | 82.36 | **3.24** | 5.11E-05 | 2.53E-03 |
| Proposed | **84.06** | **83.75** | **3.24** | - | - |
| Method_3: $\gamma$-measure; Method_4: Computing relevance in multidimensional feature space. | | | | | |

class separability (CS) index [39], Davies-Bouldin (DB) index [34], Dunn index [40], Bayes classifier [39], and rough hypercuboid (RH) [126] approach can also be used to compute the relevance of a feature set. Figure 3.7 and Table 3.6 depict the comparative performance analysis of different feature evaluation indices for HEp-2 cell image classification. From the results reported in Figure 3.7, it can be seen that the proposed relevance measure performs better than all other measures for both SNP and ICPR databases. For MIVIA database, it attains better classification accuracy than both CS and RH, but lower accuracy than Bayes classifier, DB and Dunn indices.

On the other hand, the results reported in Table 3.6 using 10-fold cross-validation confirm that the proposed feature evaluation index, based on rough sets and Bayes decision theory, attains highest mean values for three databases and highest median values for two databases. Out of total 30 cases, the proposed index performs significantly better

Table 3.6: Performance Analysis of Various Indices at Multiple Scales for 10-Fold CV

| Different Measures | Mean | Median | StdDev | Paired-$t$:p | Wilcoxon:p |
|---|---|---|---|---|---|
| | MIVIA | | | | |
| CS | 93.20 | 93.20 | 2.05 | 7.73E-03 | 6.35E-03 |
| DB | 94.35 | **94.90** | 1.93 | 8.64E-02 | 1.30E-01 |
| Dunn | 94.56 | **94.90** | 2.08 | 2.47E-01 | 2.41E-01 |
| Bayes | 94.69 | 94.56 | 1.81 | 3.10E-01 | 3.88E-01 |
| RH | 94.56 | **94.90** | **1.64** | 1.11E-01 | 1.95E-01 |
| Proposed | **94.83** | **94.90** | 1.93 | - | - |
| | SNP | | | | |
| CS | 77.76 | 76.78 | **5.04** | 2.21E-03 | 4.65E-03 |
| DB | 74.04 | 72.68 | 5.50 | 2.01E-04 | 3.46E-03 |
| Dunn | 74.04 | 72.40 | 6.27 | 2.41E-04 | 3.46E-03 |
| Bayes | **82.13** | **81.69** | 6.12 | 5.91E-01 | 4.76E-01 |
| RH | 81.69 | 80.60 | 6.45 | 2.34E-01 | 3.82E-01 |
| Proposed | 82.02 | 81.42 | 6.32 | - | - |
| | ICPR | | | | |
| CS | 80.54 | 79.46 | 4.87 | 7.95E-04 | 3.46E-03 |
| DB | 77.10 | 75.83 | 4.90 | 3.29E-06 | 2.53E-03 |
| Dunn | 80.80 | 81.25 | 4.24 | 3.46E-03 | 2.53E-03 |
| Bayes | 82.08 | 81.55 | **2.41** | 1.09E-04 | 2.53E-03 |
| RH | 83.32 | 82.58 | 2.71 | 3.08E-02 | 5.71E-02 |
| Proposed | **84.06** | **83.75** | 3.24 | - | - |

than other measures in 18 cases, and better but not significant in 11 cases. The better performance of the proposed index is achieved due to the fact that the theory of rough sets deals with the uncertainty arising from inexactness, vagueness, or incompleteness in HEp-2 pattern class definition, while Bayes decision theory addresses the uncertainty due to randomness and overlapping class boundary.

### 3.5.6 Importance of Class-Pair Specific Modalities

In general, the existing approaches, based on local texture descriptors, consider a fixed set of modalities for all the HEp-2 cell classes, where each modality corresponds to a specific local texture descriptor considered under a particular scale. However, the proposed method assumes that a fixed set of modalities may not be useful for all the classes. Rather, class-pair specific modalities should be considered while analyzing HEp-2 cell images. To establish the effectiveness of class-pair specific modalities over uniform modalities for all the classes, extensive experiment is carried out on three HEp-2 cell image databases, considering different sets of modalities incorporating local, global as well as deep features extracted from the input images.

### 3.5.6.1 Local Features

The performance of various local texture descriptors, which include LBP, $LBP^{ri}$, $LBP^{riu2}$, and CoALBP, applied at different scales, namely, $S_1$, $S_2$, $S_3$, and $S_4$, is compared to that of the proposed method given the set of descriptors as input and the obtained results are presented Figure 3.8 and Table 3.7 for both single and multiple modalities. From the results reported in top row of Figure 3.8 for single modality, it can be seen that the proposed method attains highest classification accuracy of training-testing in most of the cases. Similarly, all the results reported in bottom row of Figure 3.8, corresponding to three modalities, confirm that the proposed method achieves highest classification accuracy in all the cases. The results reported in Table 3.7, corresponding to 10-fold CV, show that the proposed method obtains highest mean and median values, irrespective of the databases and number of modalities used. Also, it attains significantly better results in all 132 cases, irrespective of statistical significance tests.

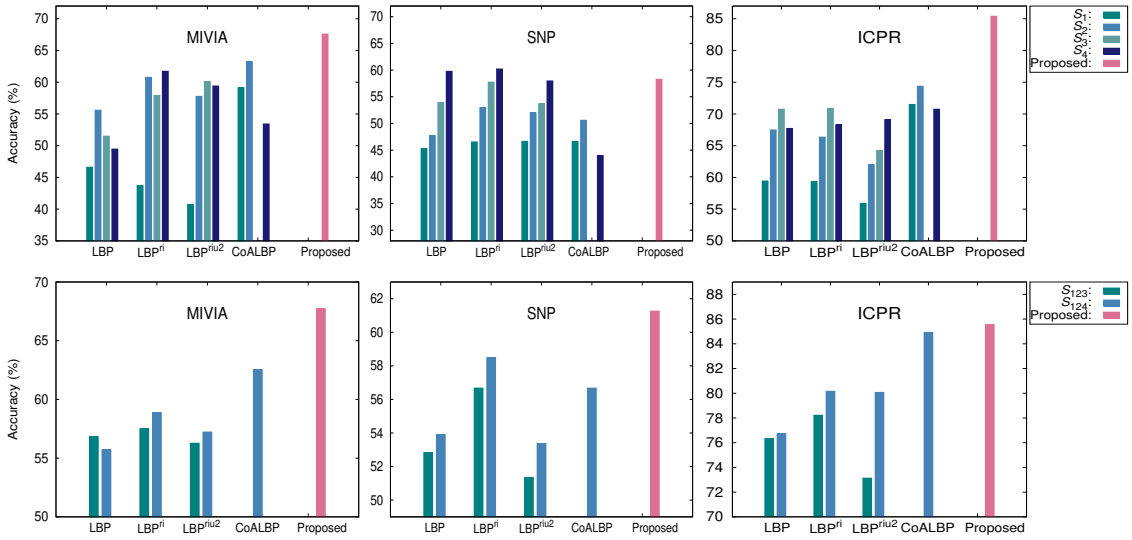



Figure 3.8: Performance of several local descriptors under different scales and proposed method at single (top row) and multiple (bottom row) modalities for training-testing.

In order to establish the effectiveness of the proposed approach under other sets of local features, Figure 3.9 and Table 3.8 compares the performance of the proposed method with that of individual modalities, where the proposed method signifies selection of class-pair relevant modality from the following given sets of modalities: (a) a set of eight modalities - sign (S) and magnitude (M) components of $LBP^{ri}$ under four scales; (b) a set of four modalities - LPQ under four window sizes; (c) a set of four modalities - BSIF under four window sizes; and (d) a set of seven modalities - CLBP under four scales and RICLBP at three scales;

From the results reported in Figure 3.9(a), it can be observed that the proposed method performs better than the individual modalities for SNP and ICPR databases, while it attains comparable results in case of MIVIA data set. However, the results reported in Table 3.8 demonstrates that the proposed method outperforms the individual modalities in all the databases for 10-fold CV. Statistical significance analysis demonstrates that the

Table 3.7: Comparative Performance Analysis at Single and Multiple Modalities for 10-Fold CV

| Different Descriptors | Different Scales | Mean | Median | StDv | Paired-$t$:p | Wilcoxon:p | Mean | Median | StDv | Paired-$t$:p | Wilcoxon:p | Mean | Median | StDv | Paired-$t$:p | Wilcoxon:p |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MIVIA | | | | | SNP | | | | | ICPR | | | | |
| LBP | $S_1$ | 74.01 | 74.49 | 2.44 | 1.10E-09 | 2.53E-03 | 60.71 | 60.93 | 7.82 | 1.25E-07 | 2.47E-03 | 57.44 | 56.31 | 8.26 | 8.50E-08 | 2.53E-03 |
| | $S_2$ | 83.33 | 83.33 | 3.52 | 6.04E-06 | 2.53E-03 | 64.48 | 65.85 | 8.32 | 5.23E-07 | 2.50E-03 | 67.33 | 67.35 | 6.79 | 7.38E-07 | 2.53E-03 |
| | $S_3$ | 82.45 | 81.63 | 2.46 | 7.20E-07 | 2.52E-03 | 66.50 | 67.49 | 7.54 | 2.39E-04 | 4.67E-03 | 72.12 | 72.52 | 5.02 | 4.08E-07 | 2.53E-03 |
| | $S_4$ | 81.43 | 81.29 | 2.81 | 1.60E-07 | 2.47E-03 | 68.47 | 66.39 | 5.55 | 1.33E-04 | 3.37E-03 | 70.01 | 68.31 | 5.25 | 4.59E-07 | 2.53E-03 |
| $LBP^{ri}$ | $S_1$ | 66.80 | 66.67 | 3.06 | 2.93E-12 | 2.50E-03 | 62.90 | 64.75 | 10.16 | 3.36E-06 | 2.52E-03 | 54.35 | 55.72 | 6.70 | 1.73E-08 | 2.53E-03 |
| | $S_2$ | 79.73 | 80.61 | 4.18 | 2.87E-07 | 2.52E-03 | 72.57 | 74.86 | 7.09 | 1.17E-03 | 4.67E-03 | 64.82 | 66.03 | 7.07 | 5.43E-07 | 2.53E-03 |
| | $S_3$ | 83.61 | 82.99 | 2.45 | 1.26E-08 | 2.45E-03 | 73.88 | 73.50 | 6.23 | 3.81E-03 | 8.30E-03 | 67.34 | 66.69 | 7.22 | 3.12E-06 | 2.53E-03 |
| | $S_4$ | 81.63 | 81.63 | 2.36 | 1.10E-07 | 2.50E-03 | 73.39 | 71.59 | 6.01 | 8.12E-04 | 4.67E-03 | 69.84 | 68.97 | 6.48 | 2.90E-06 | 2.53E-03 |
| $LBP^{riu2}$ | $S_1$ | 68.84 | 68.03 | 3.64 | 6.21E-09 | 2.52E-03 | 63.11 | 63.66 | 10.38 | 1.77E-07 | 2.53E-03 | 53.40 | 54.29 | 7.36 | 2.37E-08 | 2.53E-03 |
| | $S_2$ | 78.50 | 78.91 | 3.71 | 4.34E-08 | 2.52E-03 | 70.00 | 70.22 | 8.51 | 1.83E-05 | 2.52E-03 | 60.62 | 60.60 | 5.20 | 1.39E-09 | 2.52E-03 |
| | $S_3$ | 82.52 | 83.67 | 4.95 | 2.96E-05 | 2.52E-03 | 69.89 | 69.67 | 7.90 | 1.66E-05 | 2.53E-03 | 63.72 | 64.64 | 5.53 | 6.15E-08 | 2.53E-03 |
| | $S_4$ | 83.67 | 85.03 | 4.16 | 7.15E-05 | 2.52E-03 | 69.73 | 69.95 | 10.13 | 1.27E-05 | 2.50E-03 | 62.52 | 63.50 | 7.38 | 3.86E-06 | 2.53E-03 |
| CoALBP | $S_1$ | 88.23 | 88.44 | 2.20 | 2.15E-06 | 2.46E-03 | 69.29 | 71.31 | 7.66 | 1.12E-05 | 2.53E-03 | 68.88 | 67.20 | 6.86 | 1.88E-06 | 2.53E-03 |
| | $S_2$ | 89.66 | 89.12 | 2.26 | 1.05E-04 | 2.50E-03 | 69.34 | 69.40 | 7.15 | 2.09E-06 | 2.53E-03 | 73.55 | 73.04 | 4.97 | 9.97E-07 | 2.53E-03 |
| | $S_4$ | 83.06 | 83.33 | 3.18 | 1.84E-06 | 2.52E-03 | 57.16 | 57.10 | **4.94** | 1.67E-06 | 2.53E-03 | 70.62 | 69.85 | 3.79 | 2.61E-07 | 2.53E-03 |
| Proposed (Single) | | **93.95** | **93.20** | **1.55** | - | - | **80.60** | **81.69** | 8.48 | - | - | **83.04** | **81.95** | **3.05** | - | - |
| LBP | $S_{123}$ | 88.23 | 88.44 | 3.29 | 2.43E-04 | 2.53E-03 | 74.15 | 74.04 | 5.28 | 1.44E-04 | 2.52E-03 | 74.46 | 72.74 | 5.21 | 1.08E-06 | 2.53E-03 |
| | $S_{124}$ | 88.78 | 89.12 | 2.43 | 9.19E-05 | 2.47E-03 | 76.61 | 77.60 | 5.69 | 4.87E-04 | 3.98E-03 | 74.83 | 73.40 | 5.54 | 5.30E-06 | 2.53E-03 |
| $LBP^{ri}$ | $S_{123}$ | 86.46 | 86.73 | 3.21 | 1.18E-05 | 2.52E-03 | 78.14 | 78.14 | 5.83 | 1.21E-02 | 1.60E-02 | 77.19 | 78.28 | 3.93 | 3.50E-05 | 2.53E-03 |
| | $S_{124}$ | 85.37 | 85.71 | **1.85** | 8.83E-07 | 2.52E-03 | 78.91 | 77.32 | **5.19** | 9.92E-03 | 1.42E-02 | 74.03 | 75.46 | 4.64 | 5.71E-06 | 2.53E-03 |
| $LBP^{riu2}$ | $S_{123}$ | 85.92 | 86.05 | 2.66 | 1.28E-07 | 2.52E-03 | 76.89 | 77.87 | 7.56 | 2.04E-03 | 6.26E-03 | 70.13 | 69.99 | 4.38 | 1.81E-08 | 2.53E-03 |
| | $S_{124}$ | 86.67 | 87.07 | 3.32 | 1.49E-06 | 2.53E-03 | 78.63 | 78.14 | 7.33 | 5.82E-03 | 1.09E-02 | 71.60 | 71.06 | 4.38 | 1.03E-05 | 2.53E-03 |
| CoALBP | $S_{124}$ | 92.04 | 92.18 | 2.57 | 2.49E-03 | 4.56E-03 | 80.22 | 80.87 | 6.36 | 2.13E-02 | 1.82E-02 | 83.35 | 83.05 | 3.70 | 3.30E-02 | 4.63E-02 |
| Proposed (Three) | | **94.83** | **94.90** | 1.93 | - | - | **82.02** | **81.42** | 6.32 | - | - | **84.06** | **83.75** | **3.24** | - | - |

proposed approach achieves significantly better p-values for all the 48 cases. In case of input set corresponding to Figure 3.9(b), the proposed method performs better than the input modalities on all the databases for training-testing. From the results reported in Table 3.8, it can be noted that the proposed method outperforms the individual modalities on all the data sets for 10-fold CV. Statistical significance tests demonstrate that out of total 24 cases, the proposed model achieves significantly better p-values for 18 cases, and better but not significant p-values for 5 cases. Given the input set corresponding to Figure 3.9(c), the proposed method outperforms the individual modalities on all the data sets for training-testing. Table 3.8 describes that the proposed method performs better than the individual modalities on all the databases 10-fold CV. Statistical significance analysis demonstrates that the proposed approach achieves significantly better p-values for all the 24 cases. Lastly, given the input set corresponding to Figure 3.9(d), the proposed method performs better than the individual modalities on all the databases for training-testing. The results presented in Table 3.8 demonstrates that except for SNP database, the proposed method attains highest mean and median values for both MIVIA and ICPR data sets for 10-fold CV. Statistical significance tests reveal that out of total 42 cases, the proposed model achieves significantly better p-values for 32 cases, and better but not significant p-values for 6 cases.

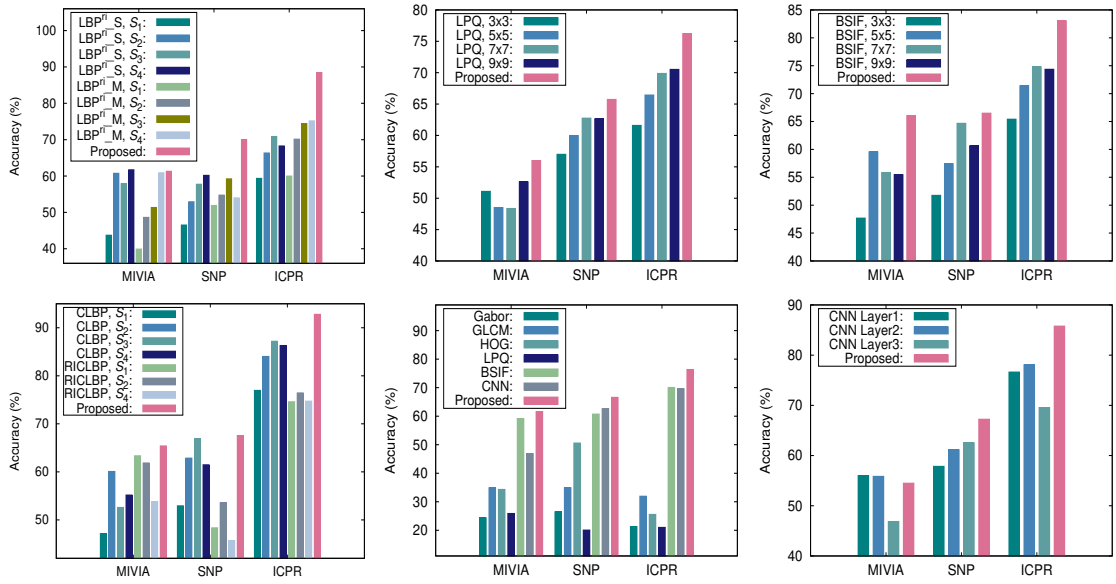



Figure 3.9: Performance of the proposed method at single modality considering different sets of descriptors.

### 3.5.6.2 Deep Features

Deep features are extracted at various layers of CNN [21] and provided as input modalities to the proposed approach which then identifies relevant modality based on the Rough-Bayesian model, for discriminating each pair of staining pattern classes accurately. The performance of the proposed method is compared with that of the individual modalities

Table 3.8: Comparative Performance Analysis of Different Descriptors at Single Modality for 10-Fold CV

| Different Descriptors | Different Scales | Mean | Median | StDv | Paired-$t$:p | Wilcoxon:p | Mean | Median | StDv | Paired-$t$:p | Wilcoxon:p | Mean | Median | StDv | Paired-$t$:p | Wilcoxon:p |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MIVIA | | | | | SNP | | | | | ICPR | | | | |
| $LBP^{ri}$_S | $S_1$ | 66.80 | 66.67 | 3.06 | 1.24E-09 | 2.52E-03 | 62.90 | 64.75 | 10.16 | 4.38E-05 | 2.53E-03 | 59.61 | 59.26 | 2.63 | 3.67E-08 | 2.53E-03 |
| $LBP^{ri}$_S | $S_2$ | 79.73 | 80.61 | 4.18 | 2.47E-05 | 2.50E-03 | 72.57 | 74.86 | 7.09 | 1.61E-04 | 2.52E-03 | 67.55 | 68.19 | **1.82** | 5.64E-07 | 2.53E-03 |
| $LBP^{ri}$_S | $S_3$ | 83.61 | 82.99 | 2.45 | 1.80E-05 | 2.52E-03 | 73.88 | 73.50 | 6.23 | 5.47E-05 | 2.53E-03 | 69.23 | 69.15 | 1.92 | 9.89E-07 | 2.53E-03 |
| $LBP^{ri}$_S | $S_4$ | 81.63 | 81.63 | 2.36 | 5.34E-07 | 2.50E-03 | 73.39 | 71.58 | 6.01 | 2.72E-05 | 2.52E-03 | 72.04 | 71.76 | 2.07 | 2.98E-06 | 2.53E-03 |
| $LBP^{ri}$_M | $S_1$ | 67.62 | 68.37 | 5.01 | 5.87E-08 | 2.53E-03 | 63.17 | 65.57 | 11.77 | 1.49E-04 | 2.53E-03 | 54.58 | 55.58 | 5.69 | 7.82E-09 | 2.53E-03 |
| $LBP^{ri}$_M | $S_2$ | 80.41 | 79.25 | 3.22 | 1.09E-06 | 2.50E-03 | 71.86 | 71.04 | 7.59 | 1.29E-04 | 2.52E-03 | 65.79 | 65.22 | 9.04 | 7.61E-07 | 2.53E-03 |
| $LBP^{ri}$_M | $S_3$ | 79.05 | 77.89 | 4.10 | 4.57E-06 | 2.52E-03 | 75.25 | 72.40 | 8.34 | 1.08E-03 | 3.46E-03 | 68.04 | 67.46 | 8.81 | 2.90E-06 | 2.53E-03 |
| $LBP^{ri}$_M | $S_4$ | 75.24 | 74.83 | 4.75 | 2.26E-06 | 2.52E-03 | 76.94 | 76.23 | 4.29 | 7.89E-05 | 2.53E-03 | 69.82 | 67.98 | 7.11 | 4.61E-07 | 2.53E-03 |
| Proposed | | **90.00** | **90.14** | **1.70** | - | - | **84.59** | **84.43** | **3.65** | - | - | **87.21** | **86.13** | 4.10 | - | - |
| LPQ | 3×3 | 75.92 | 76.19 | 2.59 | 1.63E-08 | 2.46E-03 | 65.25 | 63.66 | 6.80 | 9.22E-03 | 1.09E-02 | 63.97 | 65.04 | 10.38 | 1.20E-05 | 2.53E-03 |
| LPQ | 5×5 | 78.44 | 77.55 | 4.11 | 1.78E-05 | 2.53E-03 | 66.78 | 66.94 | 7.54 | 9.90E-02 | 1.93E-01 | 69.05 | 70.95 | 7.88 | 2.72E-06 | 2.53E-03 |
| LPQ | 7×7 | 79.59 | 80.27 | 2.62 | 8.29E-08 | 2.52E-03 | 71.86 | 72.68 | 8.30 | 4.78E-01 | 5.61E-01 | 71.33 | 70.80 | 7.27 | 5.44E-06 | 2.53E-03 |
| LPQ | 9×9 | 78.98 | 78.57 | 3.52 | 6.21E-08 | 2.52E-03 | 69.51 | 67.76 | 7.17 | 3.00E-01 | 3.23E-01 | 71.99 | 71.42 | 8.13 | 3.58E-03 | 6.26E-03 |
| Proposed | | **89.18** | **89.12** | **2.23** | - | - | **74.64** | **73.77** | **6.06** | - | - | **75.63** | **75.31** | **6.59** | - | - |
| BSIF | 3×3 | 79.66 | 80.95 | 3.21 | 9.31E-08 | 2.52E-03 | 66.99 | 68.58 | 9.63 | 7.36E-06 | 2.52E-03 | 66.02 | 66.84 | 8.39 | 1.36E-06 | 2.53E-03 |
| BSIF | 5×5 | 85.78 | 85.37 | 2.09 | 5.81E-08 | 2.47E-03 | 70.00 | 68.03 | **5.73** | 6.60E-06 | 2.52E-03 | 72.76 | 72.63 | 5.63 | 1.81E-07 | 2.53E-03 |
| BSIF | 7×7 | 85.92 | 86.73 | 3.19 | 1.94E-05 | 2.52E-03 | 74.32 | 72.13 | 5.94 | 6.20E-04 | 2.50E-03 | 75.22 | 73.70 | 5.46 | 8.23E-06 | 2.53E-03 |
| BSIF | 9×9 | 83.33 | 82.99 | 2.89 | 5.93E-07 | 2.52E-03 | 70.66 | 67.76 | 7.23 | 1.26E-05 | 2.46E-03 | 75.55 | 74.14 | 6.00 | 3.14E-04 | 2.53E-03 |
| Proposed | | **93.95** | **93.54** | **1.85** | - | - | **80.00** | **79.78** | 6.68 | - | - | **80.70** | **79.82** | **4.63** | - | - |
| CLBP | $S_1$ | 88.57 | 88.44 | 3.17 | 8.09E-05 | 2.53E-03 | 78.58 | 77.87 | 7.28 | 5.51E-01 | 8.44E-02 | 74.87 | 76.96 | 7.37 | 4.29E-06 | 2.53E-03 |
| CLBP | $S_2$ | 90.95 | 91.16 | 3.04 | 6.11E-04 | 3.46E-03 | **82.02** | 81.69 | 5.22 | 7.38E-01 | 1.20E-01 | 81.88 | 81.33 | 4.10 | 1.70E-07 | 2.53E-03 |
| CLBP | $S_3$ | 90.61 | 90.14 | **1.50** | 2.99E-07 | 2.49E-03 | 81.80 | 80.60 | 7.09 | 7.30E-01 | 2.22E-01 | 85.48 | 84.19 | 4.11 | 1.60E-05 | 2.53E-03 |
| CLBP | $S_4$ | 90.34 | 90.48 | 1.86 | 2.63E-06 | 2.50E-03 | 80.66 | 77.87 | 6.21 | 6.76E-01 | 1.31E-01 | 85.52 | 84.67 | 3.32 | 1.42E-05 | 2.53E-03 |
| RICLBP | $S_1$ | 87.62 | 87.76 | 2.58 | 1.64E-06 | 2.49E-03 | 72.62 | 72.40 | 8.00 | 2.60E-01 | 3.72E-02 | 70.59 | 69.88 | 7.30 | 4.54E-07 | 2.53E-03 |
| RICLBP | $S_2$ | 88.03 | 88.10 | 2.36 | 7.04E-07 | 2.49E-03 | 70.38 | 68.58 | 8.06 | 1.49E-01 | 3.72E-02 | 74.95 | 73.55 | 4.11 | 8.35E-10 | 2.53E-03 |
| RICLBP | $S_4$ | 84.22 | 84.35 | 2.95 | 6.43E-07 | 2.53E-03 | 57.60 | 57.10 | **3.96** | 5.95E-03 | 3.72E-02 | 69.99 | 69.04 | 4.99 | 2.01E-08 | 2.53E-03 |
| Proposed | | **96.60** | **96.60** | 1.73 | - | - | 77.65 | **84.15** | 21.15 | - | - | **90.11** | **89.10** | **2.50** | - | - |
| Gabor | | 32.04 | 32.65 | 3.31 | 7.43E-12 | 2.53E-03 | 31.15 | 30.60 | 5.49 | 1.25E-10 | 2.53E-03 | 36.48 | 43.18 | 11.23 | 4.30E-09 | 2.53E-03 |
| GLCM | | 53.40 | 53.74 | 4.08 | 1.04E-09 | 2.53E-03 | 53.17 | 53.28 | 12.06 | 9.49E-06 | 2.53E-03 | 41.30 | 45.38 | 11.83 | 3.73E-08 | 2.53E-03 |
| HOG | | 42.72 | 43.20 | 4.34 | 2.37E-09 | 2.52E-03 | 54.10 | 56.01 | 10.20 | 4.74E-08 | 2.52E-03 | 36.82 | 38.66 | **5.43** | 7.29E-12 | 2.53E-03 |
| LPQ | | 75.92 | 76.19 | **2.59** | 4.99E-06 | 2.53E-03 | 65.25 | 63.66 | 6.80 | 1.50E-05 | 2.53E-03 | 63.97 | 65.04 | 10.38 | 5.26E-06 | 2.53E-03 |
| BSIF | | 79.66 | 80.95 | 3.21 | 2.35E-05 | 2.53E-03 | 66.99 | 68.58 | 9.63 | 1.01E-04 | 2.53E-03 | 66.02 | 66.84 | 8.39 | 5.48E-07 | 2.53E-03 |
| CNN | | 76.53 | 77.89 | 4.39 | 1.75E-06 | 2.47E-03 | 68.58 | 64.75 | 9.27 | 8.63E-06 | 2.53E-03 | 70.60 | 70.98 | 7.64 | 2.42E-06 | 2.53E-03 |
| Proposed | | **89.12** | **89.80** | 3.53 | - | - | **81.64** | **81.15** | **6.42** | - | - | **76.52** | **77.06** | 6.45 | - | - |
| CNN | Layer1 | 82.79 | 81.97 | 3.19 | 4.85E-07 | 2.52E-03 | 75.30 | 75.41 | 6.80 | 1.22E-05 | 2.53E-03 | 73.83 | 72.96 | 4.33 | 1.17E-07 | 2.53E-03 |
| CNN | Layer2 | 87.21 | 87.41 | 2.72 | 2.82E-05 | 2.53E-03 | 78.20 | 77.05 | 6.29 | 2.18E-06 | 2.50E-03 | 74.64 | 74.17 | 4.43 | 8.28E-07 | 2.53E-03 |
| CNN | Layer3 | 76.53 | 77.89 | 4.39 | 2.12E-07 | 2.52E-03 | 68.58 | 64.75 | 9.27 | 1.20E-05 | 2.53E-03 | 70.60 | 70.98 | 7.64 | 5.45E-06 | 2.52E-03 |
| Proposed | | **94.42** | **94.56** | **1.46** | - | - | **87.05** | **87.16** | **4.25** | - | - | **84.72** | **84.30** | **2.91** | - | - |

and the obtained results are reported in Figure 3.9 for training-testing and Table 3.8 for 10-fold CV. From the reported results, it can be noted that the proposed method performs better than the each of the input modalities on all the databases for both training-testing and 10-fold CV. Statistical significance analysis demonstrates that the proposed approach achieves significantly better p-values for all the 18 cases.

#### 3.5.6.3 Combination of Features

Considering a set of six modalities, which include global features, namely, Gabor features, gray level co-occurrence matrix (GLCM), histogram of oriented gradients (HOG), local features, namely, LPQ and BSIF, and deep features extracted using CNN, the proposed method selects relevant modality to describe each pair of staining pattern classes. The performance of the proposed method is compared with that of the individual modalities and the obtained results are reported in Figure 3.9 for training-testing and Table 3.8 for 10-fold CV. From the reported results, it can be noted that the proposed method performs better than the each of the input modalities on all the databases for both training-testing and 10-fold CV. Statistical significance analysis demonstrates that the proposed approach achieves significantly better p-values for all the 36 cases.

### 3.5.7 Comparative Performance Analysis

Finally, the performance of the proposed method is extensively compared with that of several state-of-the art methods for texture classification and HEp-2 cell staining pattern recognition. For the proposed method, single modality corresponding to each class-pair is considered from the set of seven modalities, namely, CLBP under four scales and RICLBP at three scales. Results are reported with respect to overall classification accuracy (OCA), mean class accuracy (MCA) and execution time (in second).

#### 3.5.7.1 Texture Classification Methods

This section compares the performance of the proposed descriptor selection method with that of several state-of-the art texture classification methods, namely, dominant LBP (DLBP) [120], discriminative features for texture description (DFTD) [60], restricted Boltzmann machine (RBM) [108], discriminative deep Belief networks (disDBN) [123], CNN [21], deep encoding pooling (DEP) [211], and CNN_mRMR - a deep architecture combining CNN and minimum redundancy-maximum relevance (mRMR) framework [192]. Results are reported in Table 3.9 on three HEp-2 cell databases for training-testing. The performance of both DLBP and DFTD is evaluated for $S_{123}$ and $S_{124}$. While DLBP considers dominant features of LBP histogram [120], DFTD takes into account $\text{LBP}^{\text{ri}}$ features to form the discriminative set [60]. The SVM with linear kernel is used for staining pattern classification in all the cases. All the results reported in Table 3.9 establish the fact that the proposed method attains highest values of both OCA and MCA, irrespective of the databases used. Also, the execution time required by the proposed method is significantly lower than that of state-of-the-art deep learning based texture classification methods.

Table 3.9: Performance Analysis of Proposed Approach and Different Texture Classification Methods

| Different Methods | MIVIA | | | SNP | | | ICPR | | |
|---|---|---|---|---|---|---|---|---|---|
| | OCA | MCA | Time | OCA | MCA | Time | OCA | MCA | Time |
| DLBP, $S_{123}$ | 50.00 | 51.41 | 2.33 | 44.18 | 44.27 | 2.76 | 57.42 | 53.80 | 67.30 |
| DLBP, $S_{124}$ | 50.95 | 52.08 | 2.34 | 41.30 | 41.34 | 2.54 | 56.45 | 53.67 | 67.30 |
| DFTD, $S_{123}$ | 55.45 | 55.61 | **1.13** | 49.20 | 49.19 | **1.46** | 71.72 | 70.21 | **58.96** |
| DFTD, $S_{124}$ | 57.49 | 57.95 | 1.14 | 52.19 | 52.05 | 1.48 | 73.03 | 71.56 | **58.96** |
| RBM | 33.51 | 29.08 | 2826.39 | 47.49 | 47.38 | 3209.72 | 67.58 | 67.47 | 4558.93 |
| disDBN | 60.33 | 57.56 | 3114.46 | 54.66 | 52.69 | 7195.57 | 74.75 | 74.86 | 8889.01 |
| CNN | 46.87 | 39.39 | 2538.84 | 62.54 | 62.58 | 2888.41 | 69.57 | 69.33 | 18748.66 |
| DEP | 62.40 | 63.71 | 18353.64 | 57.20 | 56.38 | 22795.76 | 80.06 | 79.41 | 84428.37 |
| CNN_mRMR | 57.36 | 59.42 | 24631.54 | 50.48 | 49.46 | 27573.80 | 81.72 | 81.24 | 96925.61 |
| Proposed | **65.40** | **66.73** | 58.59 | **67.56** | **67.49** | 66.38 | **92.79** | **92.18** | 762.75 |

#### 3.5.7.2 HEp-2 Cell Classification Methods

This section compares the performance of the proposed HEp-2 cell staining pattern recognition method with that of several state-of-the art staining pattern classification methods, namely, RICLBP [145], Fisher tensor (FT) [45], automated pattern recognition system (APRS) [128], deep CNN model of Gao et al. (deepCNN) [55], dual convolutional auto-encoder (DCAE) [199], CNN using several preprocessing techniques (CNNPT) [160], deep CNN model of Jia et al. (DCNN) [88], and deep cross residual network (DCRNet_v2) [172]. Corresponding results are reported in Table 3.10 with respect to OCA, MCA and execution time (in second). While RICLBP, FT, and APRS methods rely on handcrafted features, other approaches consider features extracted using deep architectures for HEp-2 cell classification. From the results presented in Table 3.10, it can be seen that the proposed method for HEp-2 cell staining pattern recognition outperforms several state-of-the-art methods on all the three data sets, irrespective of the quantitative indices used. However, deepCNN [55] performs better than the proposed method only on SNP database, while DCRNet_v2 [172] attains slightly higher accuracy on MIVIA database.

Table 3.10: Performance Analysis of Proposed Approach and Different HEp-2 Cell Classification Methods

| Different Methods | MIVIA | | | SNP | | | ICPR | | |
|---|---|---|---|---|---|---|---|---|---|
| | OCA | MCA | Time | OCA | MCA | Time | OCA | MCA | Time |
| RICLBP | 63.49 | 64.67 | **17.98** | 63.82 | 63.82 | **14.56** | 83.91 | 81.70 | **179.85** |
| FT | 46.87 | 51.31 | 6374.39 | 48.03 | 48.01 | 3808.18 | 79.03 | 75.67 | 12820.44 |
| APRS | 54.90 | 52.61 | - | 53.79 | 53.60 | - | * | * | - |
| deepCNN | 64.03 | 64.17 | 2527.28 | **70.44** | **70.55** | 2875.60 | 84.95 | 84.63 | 18604.08 |
| DCAE | 42.64 | 46.52 | 8532.00 | 41.62 | 41.58 | 9120.96 | 62.49 | 62.91 | 57565.31 |
| CNNPT | 50.32 | 48.77 | 29999.30 | 47.28 | 46.41 | 35537.22 | 84.80 | 84.72 | 97760.40 |
| DCNN | 46.19 | 49.53 | 5278.31 | 46.21 | 45.13 | 6221.17 | 68.50 | 68.93 | 48444.35 |
| DCRNet_v2 | **66.49** | **66.99** | 12539.98 | 52.29 | 51.32 | 13947.27 | 87.20 | 85.60 | 871304.30 |
| Proposed | 65.40 | 66.73 | 58.59 | 67.56 | 67.49 | 66.38 | **92.79** | **92.18** | 762.75 |

It is worth mentioning here that APRS [128] considers the set of all 13,596 images

of ICPR data set to train the model, which includes the entire set of 6799 test images considered in the current study. So, the performance of APRS on ICPR data cannot be evaluated, and hence, corresponding entries are denoted by "*" in Table 3.10. To compare the performance of the proposed method with that of APRS on ICPR data, the HEp-2 cell images of Task 2 set of ICPR 2014 contest are used as test set. For Task 2, 1008 specimen images, belonging to seven HEp-2 cell classes, are made available. In [128], few cells from the specimen images have been extracted. Among them, 5032 cell images, along with the corresponding annotations, are shared by the authors of [128] for comparative performance evaluation. The APRS method achieves 82.05% OCA and 83.04% MCA on Task 2 data set, while the proposed method attains 72.48% OCA and 73.04% MCA. As the training model of APRS is provided by the authors of [128], the corresponding execution time, denoted by "-", is ignored in Table 3.10.

The better performance of the proposed method is achieved due to the fact that it considers class-pair specific modalities for analyzing HEp-2 cell images, rather than considering a fixed set of modalities for all the staining pattern classes. A set of dominant features is extracted under each modality, based on the probability of texture pattern occurrence, which makes the proposed algorithm insensitive to noisy HEp-2 cell images. Moreover, the proposed approach employs rough sets and Bayes decision theory to evaluate the relevance of each modality. While former deals with the uncertainty due to inexactness, vagueness, and incompleteness in HEp-2 class definition, latter addresses the uncertainty due to overlapping class boundaries. In effect, the proposed method provides significantly better performance as compared to existing methods.

## 3.6 Conclusion

The major contribution of this chapter is three-fold, namely, (i) development of a method to select a set of class-pair specific relevant texture descriptors under appropriate scales for HEp-2 cell pattern classification; (ii) introducing a new relevance measure, based on judicious integration of variable precision rough sets and Bayes decision theory; and (iii) demonstrating the effectiveness of the proposed method on several benchmark HEp-2 cell image databases.

The proposed class-pair specific modality selection method first selects local texture descriptors under appropriate scales for a particular pair of classes, and then forms the final feature set for multiple classes from the set of descriptors of all possible pairs of classes. To evaluate the quality of a modality in discriminating samples from a given class-pair, the theory of rough sets is judiciously integrated with Bayes decision theory. To make the proposed descriptor selection method insensitive to noisy pixels present in an HEp-2 cell image and noisy HEp-2 cell images in a staining pattern class, the concept of significant descriptors of an HEp-2 cell image and a staining pattern class has been introduced. Finally, the SVM is used to predict the class of HEp-2 cell images. The performance of the proposed approach is exhibited on three benchmark HEp-2 cell image databases, along with a comparative study with state-of-the-art methods. For three HEp-2 cell data sets, significantly better results are found for the proposed method compared to several existing methods. Along with the local, global, and deep features, textural features based on markov random fields can also be considered as the input set of descriptors for

the proposed class-pair specific descriptor selection approach.

The results obtained on different HEp-2 cell image databases demonstrate that the classification accuracy achieved by the proposed approach varies with the given set of modalities to a certain extent. This is due to the fact that the performance of the proposed method depends on the descriptive and discriminative ability of the features extracted from the input images. Also, the proposed approach attains highest classification accuracy on different data sets for different input sets of modalities. However, in case of deep features, the proposed method performs consistently well on all the databases. A possible explanation for the observation would be that the deep architectures takes into account the nature and complexity of the given classification problem while extracting features at various layers. Hence, in the next chapter, a deep architecture is developed based on the theory of Boltzmann machine, which can efficiently captures the non-linear dependencies between observed and latent variables by analyzing the energy landscape of the given samples or observations. It learns a joint subspace over the space of multimodal inputs. In order to improve the discriminative ability of the model, the class nodes are incorporated into the proposed architecture which enables the architecture to serve as feature extractor as well as classifier.

# Chapter 4

# Multimodal Discriminative Deep Boltzmann Machine for Joint Subspace Analysis

## 4.1 Introduction

With the development of acquisition equipments and sensors, a large amount of data has become more accessible nowadays. However, the unpredictably ambiguous nature of the data, along with the incompleteness in data representation, has restricted the performance of unimodal based data recognition systems [119]. In real world applications, data are collected from different sources of diverse domains or obtained from various feature extraction methods. So, multiple representations of the same data are available, which are also referred to as different modalities or views of the data. Since each view has a fundamentally distinct representation of the underlying data distribution, it is primarily assumed that integration of different views may provide comprehensive and discriminatory description of the inherent characteristics of the given input data. In recent years, a surging interest is noted for developing multi-view learning algorithms where the diversity of different views are exploited, along with the patterns observable in the individual views [212]. By establishing the connections between the attributes of different views of the given data, it is possible for the multi-view learning algorithms to obtain a more descriptive representation of the data than the single view learning approaches. Hence, development of multi-view learning algorithms has become a promising research topic with wide applicability.

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 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 existing literature to learn the joint representation of the data from the given multiple modalities. Various multi-view learning algorithms, based on correlation analysis [25, 53, 96, 124], discriminant analysis [92, 93, 217], and deep learning

[9, 30, 37, 44, 178, 180, 193, 202], 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.

Considering the baseline framework of canonical correlation analysis (CCA) [78] and its kernel extensions [10, 67, 124], a variety of theories and approaches are introduced to investigate the inherent correlation across several views. The multiset CCA (MCCA) [96] is introduced as an extension of the classical two-set theory of CCA to several sets. Chen et al. [25] 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 representations, based on the graph-induced knowledge of the common sources. In case of randomized CCA (rCCA) [124], linear CCA is performed on a pair of randomized non-linear mappings corresponding to the given input views. In [53], large-scale generalized CCA (LasCCA) has been proposed whose memory and computational costs vary linearly with the problem dimension and the number of non-zero data elements, respectively. As a result, it can easily handle very large sparse views. A distributed algorithm for generalized CCA (DisCCA) has also been developed in [53], which computes the canonical variables of different views in parallel to reduce the run time of the algorithm significantly.

Apart from the CCA based methods, several classical approaches have been developed for multi-view data analysis. Multi-view discriminant analysis (MvDA) [92] learns single unified discriminant common space from the given multiple views by jointly optimizing the view-specific transforms corresponding to each of the input views. Furthermore, the between-class and within-class variations are combined to form a generalized Rayleigh quotient, which can be solved analytically by using the generalized eigenvalue decomposition. Inspired by the observation that different views share similar structures, MvDA with view-consistency (MvDA-VC) [93] enforces consistency across different views to achieve a more robust common space. Since the CCA based methods are, in general, unsupervised in nature, the projected subspaces lack discriminative information. Also, the canonical variables, corresponding to each of the views, are learned either by linear projections or non-linear projections which are limited by a fixed kernel. Furthermore, an additional classifier is required for the classification purpose. Although the MvDA and MvDA-VC methods have the benefits of view discrepancy and discriminability, the common subspace is learned using linear transformations.

Although the classical approaches have achieved some promising results in numerous domains of applications, use of hand-crafted features and linear embedding functions have restricted these methods to capture non-linear nature of complex multi-view data. Non-linearity is described by a situation where there is no direct relationship between a dependent and an independent variable, that is, the dependent variables do not change in direct proportion to changes in any of the independent variables. As discussed in Chapter 3, the hand-crafted features are constructed for capturing certain specific aspects of the given data. They are restricted by the limited human knowledge and hence, it is difficult for the hand-crafted features to effectively analyze the hidden attributes of the input data. On the other hand, features extracted by the deep models, are data as well as task specific. Deep learning models can effectively learn complex, non-linear, and abstract representations of the given data by allowing multiple hierarchical layers. This is also validated in Chapter 3, where it can be observed that deep features provide outstanding performance

in comparison to the hand-crafted features for all data sets considered. Along with the success of various deep models, deep multi-view learning models have vast inroads into numerous applications with outstanding performance.

Recently, several deep learning models have been developed to integrate information from the given multiple views. The deep canonical correlation analysis (DCCA) [9] learns complex non-linear transformations of the given two views in such a way that the resulting representations are highly correlated, while deep canonically correlated autoencoders (DCCAE) [202] incorporates the objective of autoencoders into DCCA model. Since the DCCA and DCCAE models are unsupervised in nature, the joint representations fail to capture the discriminative information of the given observations. Also, both the models are applicable to two views only. In deep multiset CCA (dMCCA) [178], feed-forward networks have been used to map the given input modalities to a shared subspace such that the joint representation maximizes the ratio of between and within modality covariance of the given observations. Couture et al. [30] have developed task optimal CCA (TOCCA) that focuses on both CCA and task driven objectives using a deep architecture. Though dMCCA and TOCCA concentrate on learning the correlated subspace from the input multi-view data, they eventually disregard the underlying data distribution. The deep CCA with view generation (DCCA-VG) [193] focuses on learning multi-view representation for hyperspectral image classification by fusing spatial and spectral information. Dorfer et al. [37] have proposed towards deep and discriminative CCA (TDDCCA) that incorporates a discriminative regularizer into the objective of existing deep canonical correlation analysis to jointly avail the advantages of correlation and discriminability. Both the DCCA-VG and TDDCCA approaches serve as feature extractors and need to employ an additional classifier for classification purpose. Fan et al. [44] have proposed deep adversarial CCA (DACCA) model, which integrates adversarial learning techniques with the concept of CCA to simultaneously learn multi-view data representation and generate realistic multi-view samples. Since the DACCA model is susceptible to slight variation in the input characteristics of the given data, the performance of the model is highly dependent on the input signal-to-noise ratio.

From the state-of-the-art deep multi-view learning models, it can be stated that these models have significant contribution towards extracting relevant features from the given multiple views of the input data. However, they do not take into consideration the supervised information of sample categories in the corresponding learning objectives and hence, the extracted features lack the discriminative information. Also, the models require an additional classifier for categorizing the given observations into different classes. Hence, it is required to develop a generalized framework which can provide better classification performance through end-to-end learning. Now, deep Boltzmann machine (DBM) [163] is an effective paradigm of undirected generative models that efficiently captures the non-linear dependencies between observed and latent variables by analyzing the energy landscape of the given observations. Hence, the multi-view model based on DBM is expected to encapsulate the latent data distribution over the space of multimodal inputs. However, the architecture of DBM is essentially unsupervised in nature. In case of classification problem, it is expected that the similarity in the latent space implies the similarity in the corresponding concepts.

This chapter introduces a novel deep multi-view predictive model, termed as multimodal discriminative deep Boltzmann machine (MDDBM). It is developed by judiciously integrating the merits of DBM and the supervised information of sample categories. The

proposed deep model can extract discriminative and descriptive features from the given views, learn a joint subspace by integrating all the imperative information from the feature representations corresponding to each of the input views, and also classify the given observations into multiple categories. The proposed framework is developed based on the architecture of DBM in multi-view environment to encapsulate the underlying non-linear data distribution of the given observations. The class nodes are incorporated into the proposed deep architecture to include the supervised information at each layer of the network. Through proper learning of the weights associated with the corresponding class nodes, it can be ensured that the obtained representations at each layer of the network 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. The performance of the MDDBM model is extensively studied and compared with several state-of-the-art multi-view learning approaches on various benchmark and real-life cancer data sets considering both training-testing and ten-fold cross-validation. Some of the results of this chapter are reported in [104].

The rest of this chapter is organized as follows: Section 4.2 discusses the basics of Boltzmann machine, deep Boltzmann machine, and multimodal deep Boltzmann machine. Section 4.3 describes the architecture and learning of the proposed MDDBM model. The efficacy of the MDDBM framework is studied with reference to several state-of-the-art approaches on various benchmark and real-life cancer data sets in Section 4.4. Concluding remarks are provided in Section 4.5.

## 4.2 Basics of Boltzmann Machine

The basic concepts of Boltzmann machine (BM), deep Boltzmann machine (DBM), and multimodal deep Boltzmann machine (MDBM) are briefly discussed in this section.

### 4.2.1 Boltzmann Machine

Boltzmann machine is an effective paradigm of undirected graphical models, also known as Markov random fields, which provide a powerful mechanism for representing dependency structure between random variables. It has originally been developed in [2, 70] as a bidirectionally coupled neural networks, containing stochastic processing units which can be partitioned into two different types, namely, visible units and hidden units. While the observed data vectors are clamped on the visible units, the hidden units act as latent variables or features that allow the BM to model the probability distributions over visible state vectors that cannot be modelled by direct pairwise interactions between the visible units.

Let us consider a BM with $V$ visible units that represent the observed data vector and $H$ hidden units which account for the dependencies between observed variables. The state of the model at any instant is defined as $(\mathbf{v}, \mathbf{h}) \in \{0,1\}^{V+H}$. So, any distribution on $\{0,1\}^V$ can be modelled by the BM where the dimension of the target distribution is denoted by the number of hidden units $H$. In order to update the state of the $j$-th hidden unit, $h_j$, the total input to the node is to be computed which is the sum of the weights on connections

coming from rest of the active units and the corresponding bias term $b_j$:

$$\phi_{h_j} = \sum_{i=1}^{V} v_i w_{ij} + \sum_{k=1,k\neq j}^{H} h_k u_{kj} + b_j, \tag{4.1}$$

where $w_{ij}$ and $u_{kj}$ represent the values of the weights on the bidirectional connections between $i$-th visible unit and $j$-th hidden unit and $k$-th hidden unit and $j$-th hidden unit, respectively. Based on the conditional distributions of the model, it can then be shown that the $j$-th hidden unit turns on with a probability given by the following logistic function:

$$prob(h_j = 1) = \frac{\exp(\{h_j = 1\}\sum_{i=1}^{V} v_i w_{ij} + \sum_{k=1,k\neq j}^{H} h_k u_{kj} + b_j)}{\sum_{h_j\in\{0,1\}} \exp\{h_j(\sum_{i=1}^{V} v_i w_{ij} + \sum_{k=1,k\neq j}^{H} h_k u_{kj} + b_j)\}} = \frac{1}{1+e^{-\phi_{h_j}}}. \tag{4.2}$$

Thus, the hidden representation of the BM is obtained by applying non-linear transformation on the stochastic units. In a similar way, the updation rule for the state of a visible unit can be obtained as

$$prob(v_i = 1) = \frac{1}{1+e^{-\phi_{v_i}}}, \quad \text{where} \quad \phi_{v_i} = \sum_{j=1}^{H} w_{ij} h_j + \sum_{p=1,p\neq i}^{V} y_{ip} v_p + a_i. \tag{4.3}$$

Here $y_{ip}$ denotes the bidirectional real valued weight that connects the $i$-th visible unit with $p$-th visible unit and $a_i$ represents the bias term corresponding to the $i$-th visible unit.

If the units are updated sequentially in any order that does not depend on their total inputs, the network will eventually reach a Boltzmann distribution, also referred to as equilibrium or stationary distribution, in which the probability of a state vector $(\mathbf{v}, \mathbf{h})$ is determined solely by the energy of the vector relative to the energies of all possible binary state vectors:

$$P(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v},\mathbf{h})}, \quad \text{where} \quad Z = \sum_{\mathbf{v},\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h})}. \tag{4.4}$$

Hence, the joint probability distribution of the variables under the model is governed by the Gibbs distribution. Here, $Z$ is termed as the partition function which acts as a normalizing term. As in Hopfield networks [150], the energy of the state vector $(\mathbf{v}, \mathbf{h})$ is defined as

$$E_{bm}(\mathbf{v}, \mathbf{h}) = -\sum_{i=1}^{V}\sum_{j=1}^{H} v_i w_{ij} h_j - \sum_{j=1}^{H}\sum_{k=1,k\neq j}^{H} h_j u_{jk} h_k - \sum_{i=1}^{V}\sum_{p=1,p\neq i}^{V} v_i y_{ip} v_p - \sum_{i=1}^{V} a_i v_i - \sum_{j=1}^{H} b_j h_j. \tag{4.5}$$

If the weights on the connections are so chosen that the energies of state vectors represent the inferior quality of the vectors as solutions to an optimization problem, then the stochastic dynamics of the BM can be viewed as a way of escaping from local optima while searching for good solutions. The total input to a hidden unit then represents the difference in the energy depending on whether that unit is off or on, and the fact that unit occasion-

ally turns on even the corresponding input to the unit is negative means that the energy can occasionally increase during the search, thus allowing the search to jump over energy barriers. The crucial computational step in a BM is determining how the model, with its current generative parameters, might have used its hidden variables to generate an observed data vector, which is referred to as the learning of the BM. Stochastic generative models, like Boltzmann machines, generally have many different ways of generating any particular data vector. The best possible way would be to infer the probability distribution over the various possible settings of the hidden variables. The learning algorithm for BM [73] required randomly initialized Markov chains to approach the equilibrium distribution.

The major limitation for learning of BM models is the necessity to compute the partition function, whose purpose is to normalize the joint probability distribution over the set of random variables. Additionally, the derivative of the partition function is required to be estimated for learning of the model parameters. However, in most of the cases, the exact computation of the partition function or the corresponding derivative is intractable because it requires enumeration over an exponential number of terms. There has been extensive research on obtaining deterministic approximations [215, 216] or deterministic upper bounds [56, 200, 201] on the log-partition function of a BM. These methods take a variational perspective of estimating the log-partition function and rely critically on approximating the entropy of the undirected graphical model. Variational methods have become very popular, since they typically scale well to large applications. There have also been many developments in the use of Monte Carlo methods for estimating the partition function, including annealed importance sampling [141], bridged sampling [132], nested sampling [164], sequential Monte Carlo [135], Markov Chain Monte Carlo method [140], and many others.

Setting both $u_{jk} = 0$ and $y_{ip} = 0,\ \forall i, j, p, k$ recovers the well-known restricted Boltzmann machine (RBM) [176]. In contrast to the BM, inference in RBM is exact. Although exact maximum likelihood learning in RBM is still intractable, learning can be carried out efficiently using contrastive divergence (CD) [69]. It was further observed [204] that for contrastive divergence to perform well, it is important to obtain exact samples from the conditional distribution, which is in turn intractable while learning of the original BM.

### 4.2.2 Deep Boltzmann Machine

Generative models with only one hidden layer are much too simple for modelling the high-dimensional and richly structured sensory data that arrive at the visual cortex. It is believed that our visual systems contain multilayer generative models in which top-down connections can be used to generate low-level features of images from high-level representations, and bottom-up connections can be used to infer the high-level representations that would have generated an observed set of low-level features. Single cell recordings [111] and the reciprocal connectivity between cortical areas [47] both suggest a hierarchy of progressively more complex features in which each layer can influence the layer below. Vivid visual imagery, dreaming, and the disambiguating effect of context on the interpretation of local image regions [137] also suggest that the visual system can perform top-down generation.

Apparently, it was too difficult to perform inference in the complicated multilayer non-linear models. Until recently, several attempts have been formulated to develop multilayer non-linear models [94, 168]. Multiple hidden layers can be learned by treating the hidden

activities of one RBM as the data for training a higher-level RBM [72]. However, if multiple layers are learned in this greedy, layer-by-layer manner, the resulting composite model is not a multilayer or deep Boltzmann machine [71]. It is a hybrid generative model called a deep belief net that has undirected connections between its top two layers and downward directed connections between all its lower layers. Each layer of a deep Boltzmann machine (DBM) [163] captures complicated, higher-order correlations between the activities of hidden features in the layer below. DBMs are interesting for several reasons. As in case of deep belief networks, DBMs have the potential of learning internal representations that become increasingly complex, which is considered to be a promising way of solving several interesting real-life problems. Furthermore, unlike deep belief networks, the approximate inference procedure, in addition to an initial bottom-up pass, can incorporate top-down feedback, allowing DBMs to propagate uncertainty efficiently, which in turn, facilitate the model to deal with ambiguous inputs robustly.

Let us consider, $L$ denotes the number of layers in a DBM. The energy function of the DBM is given by

$$E_{dbm}(\mathbf{v}, \mathbf{h}^1, \cdots, \mathbf{h}^l, \cdots, \mathbf{h}^L) = -\sum_{i=1}^{V}\sum_{j=1}^{H^1} v_i w_{ij}^1 h_j^1 - \sum_{l=l}^{L}\sum_{j=1}^{H^l}\sum_{k=1}^{H^{(l+1)}} h_j^l w_{jk}^{(l+1)} h_k^{(l+1)} - \sum_{i=1}^{V} a_i v_i - \sum_{l=l}^{L}\sum_{j=1}^{H^l} b_j^l h_j^l; \tag{4.6}$$

where $H^l$ represents the number of hidden units in layer $l$ of the model, $b_j^l$ signifies the bias term corresponding to the $j$-th hidden unit at layer $l$, and $w_{ij}^1$ and $w_{jk}^{(l+1)}$ denote the values of the weights on the bidirectional connection between $i$-th visible unit and $j$-th hidden unit at first hidden layer of the model and $j$-th hidden unit at layer $l$ and $k$-th hidden unit at layer $(l+1)$, respectively.

It is to be noted here that if the posterior distribution over the hidden variables can be inferred for each data vector, then learning a DBM is relatively straightforward. Alternatively, if unbiased samples can be obtained from the posterior distribution, then also the learning of DBM becomes straightforward. In the later case, the parameters of the model need to be adjusted so as to increase the probability that the sampled states of the hidden variables in each layer would generate the sampled states of the hidden or visible variables in the layer below. If training vectors are selected with equal probability from the training set and the hidden states are sampled from the corresponding posterior distribution given the training vector, then learning of the model has a positive effect on the probability that the model would produce exactly the $N$ training vectors if it was run $N$ times.

### 4.2.3 Multimodal Deep Boltzmann Machine

In a multimodal setting, data consists of multiple modes, each modality having a different kind of representation and correlational structure. For example, text is usually represented as discrete sparse word count vectors, whereas an image is represented using pixel intensities or outputs of feature extractors which are real-valued and dense. Having very different statistical properties makes it much harder to discover relationships across modalities than

relationships among features in the same modality. There is a lot of structure in the data but it is difficult to discover the highly non-linear relationships that exist between low-level features across different modalities. A good multimodal learning model must satisfy certain properties. The joint representation must be such that similarity in the representation space implies similarity of the corresponding concepts so that the representation is useful for classification and retrieval. It is also desirable that the joint representation be easy to obtain even in the absence of some modalities. It should also be possible to fill-in missing modalities given the observed ones. In this context, Srivastava and Salakhutdinov have developed a multimodal deep Boltzmann machine (MDBM) model in [180] for learning multimodal data representations.

The energy function of the MDBM model is given by

$$E_{mdbm}(\mathbf{v}^m, \mathbf{h}^{1m}, \cdots, \mathbf{h}^{lm}, \cdots, \mathbf{h}^{Lm}, \mathbf{h}^{(L+1)}) = -\sum_{m=1}^{M}\sum_{i=1}^{V^m}\sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m} h_j^{1m}$$
$$-\sum_{l=l}^{L}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}}\sum_{k=1}^{H^{(l+1)m}} h_j^{lm} w_{jk}^{(l+1)m} h_k^{(l+1)m} - \sum_{m=1}^{M}\sum_{i=1}^{V^m} a_i^m v_i^m - \sum_{l=l}^{L}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}} b_j^{lm} h_j^{lm}; \quad (4.7)$$

where $M$ denotes total number of input views or modalities, $(L+1)$ signifies the total number of hidden layers of the model, out of which $L$ number of layers are modality-specific and the top most layer represents the joint representation. $\mathbf{v}^m$ represents the input view corresponding to the $m$-th modality, the $l$-th modality-specific hidden layer corresponding to the $m$-th modality is represented by $\mathbf{h}^{lm}$, and the joint representation is denoted by $\mathbf{h}^{(L+1)}$. The key idea of the model is to effectively utilize large amount of unlabelled data. Pathways for each modality are pretrained independently and plugged in together for performing joint learning. The model fuses multiple data modalities into a unified representation, which captures features that are useful for classification and retrieval. It also performs well when some modalities are absent and improves upon models trained on only the observed modalities.

## 4.3 Multimodal Discriminative Deep Boltzmann Machine

In this section, a novel architecture, termed as multimodal discriminative DBM (MDDBM), is discussed for multi-view data analysis. It judiciously integrates the merits of DBM and the supervised information of sample categories to accurately categorize the given observations into different class labels to efficiently encapsulate the underlying non-linear data distribution over the space of multimodal inputs. The inclusion of supervised information of class labels or sample categories into the architecture of DBM improves the discriminative ability of the model. MDBM [180] 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 label, the obtained joint subspace will have better discriminative ability. Also, incorporating class nodes in the architecture enables the model to predict the class label of given observations, without employing any additional classifier for classification.

### 4.3.1 Objective Function of Proposed MDDBM Model

Let us assume that the proposed model has $M$ input views or modalities, where $\mathbf{v}^m = \{v_1^m, \cdots, v_i^m, \cdots\}$ represents the input view corresponding to the $m$-th modality and $\mathbf{y} = \{y_1, \cdots, y_c, \cdots\}$ 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 hidden layers. The $l$-th modality-specific hidden representation corresponding to the $m$-th modality is denoted by $\mathbf{h}^{lm} = \{h_1^{lm}, \cdots, h_j^{lm}, \cdots\}$, whereas the joint hidden representation, corresponding to the $l$-th layer, is referred to as $\mathbf{h}^l = \{h_1^l, \cdots, h_j^l, \cdots\}$. 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 $E_{mddbm}(\mathbf{v}, \mathbf{h}, \mathbf{y})$ of the proposed model is defined in (4.8), while the corresponding architecture is depicted in Figure 4.1.

$$
\begin{aligned}
E_{mddbm}(\mathbf{v}, \mathbf{h}, \mathbf{y}) = &- \sum_{m=1}^{M} \sum_{i=1}^{V^m} \sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m} h_j^{1m} - \sum_{l=1}^{L_0-1} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} \sum_{k=1}^{H^{(l+1)m}} h_j^{lm} w_{jk}^{(l+1)m} h_k^{(l+1)m} \\
&- \sum_{m=1}^{M} \sum_{j=1}^{H^{L_0 m}} \sum_{k=1}^{H^{(L_0+1)}} h_j^{L_0 m} w_{jk}^{(L_0+1)m} h_k^{(L_0+1)} - \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{c=1}^{Y} \sum_{j=1}^{H^{lm}} y_c u_{cj}^{lm} h_j^{lm} - \sum_{m=1}^{M} \sum_{i=1}^{V^m} a_i^m v_i^m \\
&- \sum_{l=(L_0+1)}^{L} \sum_{c=1}^{Y} \sum_{j=1}^{H^l} y_c u_{cj}^l h_j^l - \sum_{l=(L_0+1)}^{L-1} \sum_{j=1}^{H^l} \sum_{k=1}^{H^{(l+1)}} h_j^l w_{jk}^{(l+1)} h_k^{(l+1)} - \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} b_j^{lm} h_j^{lm} \\
&- \sum_{l=(L_0+1)}^{L} \sum_{j=1}^{H^l} b_j^l h_j^l - \sum_{c=1}^{Y} d_c y_c. \qquad (4.8)
\end{aligned}
$$

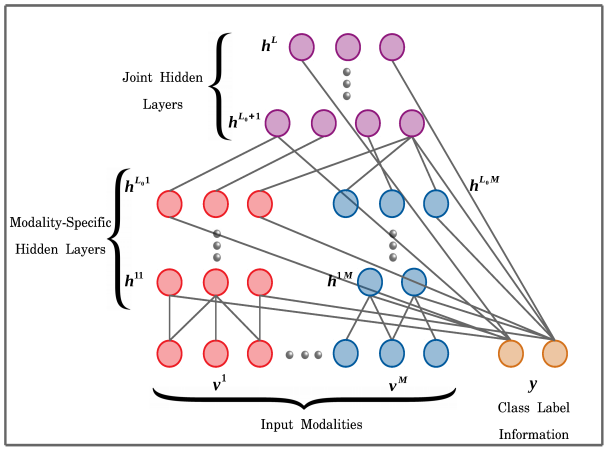



Figure 4.1: Illustration of proposed multimodal discriminative deep framework.

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 $l$-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 $u_{cj}^{lm}$ and $u_{cj}^{l}$ connect $c$-th class node to $j$-th hidden node of $l$-th modality-specific hidden layer from $m$-th modality, and $c$-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 $d_c$ 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 $c$-th class node, respectively.

Considering the energy function $E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})$ of (4.8), the parameter space $\boldsymbol{\theta}_{mddbm}$ of the model is defined by

$$\begin{aligned}\boldsymbol{\theta}_{mddbm} = \{&w^{1m},\cdots,w^{(L_0+1)m},w^{(L_0+2)},\cdots,w^{L},u^{1m},\cdots,u^{L_0m},u^{(L_0+1)},\cdots,u^{L},a^m,b^{1m},\\ &\cdots,b^{L_0m},b^{(L_0+1)},\cdots,b^{L},d\},\quad \forall m\in\{1,2,\cdots,M\}.\end{aligned}$$

Thus, through proper learning of the set of parameters $\{u^{1m},\cdots,u^{L_0m},u^{(L_0+1)},\cdots,u^{L},d\}$ $\forall m$, 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. Introducing the class nodes in the architecture allows the proposed model to serve as the classifier as well. Given the input view $\mathbf{v}$ and the class label information $\mathbf{y}$, learning of the MDDBM model corresponds to identifying the model parameter set $\boldsymbol{\theta}_{mddbm}$ that maximizes the probability of observing the given observations. So, the objective function is given by the log-likelihood function as follows:

$$\ln \mathrm{L}(\boldsymbol{\theta}_{mddbm}|\mathbf{v},\mathbf{y}) = \ln P(\mathbf{v},\mathbf{y}|\boldsymbol{\theta}_{mddbm}) = \ln\sum_{\mathbf{h}} e^{-E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})} - \ln\sum_{\mathbf{v},\mathbf{h},\mathbf{y}} e^{-E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})} \quad (4.9)$$

where $\mathbf{h}$ denotes the stack of hidden layers, $P(\mathbf{v},\mathbf{y}|\boldsymbol{\theta}_{mddbm})$ represents the probability assigned to the observation $(\mathbf{v},\mathbf{y})$ by the model parameter set $\boldsymbol{\theta}_{mddbm}$, and $E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})$ signifies the energy of the joint configuration $\{\mathbf{v},\mathbf{h},\mathbf{y}\}$ corresponding to the MDDBM model. The partition function of the model is defined as $Z = \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} e^{-E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})}$.

Since the parameter space of the model is quite large, obtaining the parameters that maximize (4.9) 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 $\Delta\boldsymbol{\theta}_{mddbm}^{t}$ based on the gradient of the log-likelihood. So, the update rule for the parameters is given by

$$\begin{aligned}\frac{\partial \ln \mathrm{L}(\boldsymbol{\theta}_{mddbm}|\mathbf{v},\mathbf{y})}{\partial\boldsymbol{\theta}_{mddbm}} &= \frac{\partial}{\partial\boldsymbol{\theta}_{mddbm}}\left(\ln\sum_{\mathbf{h}} e^{-E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})} - \ln\sum_{\mathbf{v},\mathbf{h},\mathbf{y}} e^{-E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})}\right)\\ &= -\sum_{\mathbf{h}} P(\mathbf{h}|\mathbf{v},\mathbf{y})\frac{\partial E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})}{\partial\boldsymbol{\theta}_{mddbm}} + \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} P(\mathbf{v},\mathbf{h},\mathbf{y})\frac{\partial E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y})}{\partial\boldsymbol{\theta}_{mddbm}}. \quad (4.10)\end{aligned}$$

So, the gradient of the log-likelihood function reduces to the difference between expectation of gradient of energy function under model distribution, which is termed as 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, the variational learning is employed to estimate the data-dependent expectation, whereas data-independent expectation is approximated by the stochastic approximation procedure. Hence, from (4.10), it can be observed that in order to learn the parameters of MDDBM model, the corresponding data-dependent and data-independent expectations are required to be estimated, which are described subsequently.

### 4.3.2 Estimation of Data-Dependent Expectations

In the section, the data-dependent expectations are estimated using the concept of variational learning [142]. In variational inference, 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,

$$\ln P(\mathbf{v},\mathbf{y}) = \ln \sum_{\mathbf{h}} P(\mathbf{v},\mathbf{h},\mathbf{y}) = \ln \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \frac{P(\mathbf{v},\mathbf{h},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})}. \tag{4.11}$$

Since logarithmic is a concave function, applying Jensen's inequality [31] in (4.11), we get

$$\ln P(\mathbf{v},\mathbf{y}) \geqslant \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \ln \frac{P(\mathbf{v},\mathbf{h},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})} = \mathcal{L}_v. \tag{4.12}$$

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

$$\ln P(\mathbf{v},\mathbf{y}) - \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \left\{ \ln P(\mathbf{v},\mathbf{y}) + \ln \frac{P(\mathbf{h}|\mathbf{v},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})} \right\} = KL(Q(\mathbf{h}|\mathbf{v},\mathbf{y})||P(\mathbf{h}|\mathbf{v},\mathbf{y})), \tag{4.13}$$

where $KL(Q(\mathbf{h}|\mathbf{v},\mathbf{y})||P(\mathbf{h}|\mathbf{v},\mathbf{y}))$ 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 $\ln P(\mathbf{v},\mathbf{y})$.

Let us consider the following factorized distribution as the approximate posterior distribution of (4.12):

$$Q(\mathbf{h}|\mathbf{v},\mathbf{y}) = \prod_{l=1}^{L_0} \prod_{m=1}^{M} \prod_{j=1}^{H^{lm}} q(h_j^{lm}|\mathbf{v},\mathbf{y}) \prod_{l=(L_0+1)}^{L} \prod_{j=1}^{H^l} q(h_j^l|\mathbf{v},\mathbf{y}); \tag{4.14}$$

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 $\mu_j$ 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 from (4.12), which is as follows:

$$\begin{aligned}\mathcal{L}_v &= \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y})\{\ln P(\mathbf{v},\mathbf{h},\mathbf{y}) - \ln Q(\mathbf{h}|\mathbf{v},\mathbf{y})\} \\ &= \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y})\{-E_{mddbm}(\mathbf{v},\mathbf{h},\mathbf{y}) - \ln Q(\mathbf{h}|\mathbf{v},\mathbf{y}) - \ln Z\}. \end{aligned} \tag{4.15}$$

Considering the energy function (4.8) of the MDDBM model and substituting the mean field distribution (4.14) in (4.15), the variational bound $\mathcal{L}_v$ for the proposed MDDBM model can be obtained as

$$\begin{aligned}\mathcal{L}_v = &\sum_{m=1}^{M}\sum_{i=1}^{V^m}\sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m}\mu_j^{1m} + \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{c=1}^{Y}\sum_{j=1}^{H^{lm}} y_c u_{cj}^{lm}\mu_j^{lm} + \sum_{l=(L_0+1)}^{L}\sum_{c=1}^{Y}\sum_{j=1}^{H^l} y_c u_{cj}^{l}\mu_j^{l} \\ &+ \sum_{l=1}^{L_0-1}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}}\sum_{k=1}^{H^{(l+1)m}} \mu_j^{lm} w_{jk}^{(l+1)m}\mu_k^{(l+1)m} + \sum_{m=1}^{M}\sum_{j=1}^{H^{L_0 m}}\sum_{k=1}^{H^{(L_0+1)}} \mu_j^{L_0 m} w_{jk}^{(L_0+1)m}\mu_k^{(L_0+1)} \\ &+ \sum_{l=(L_0+1)}^{L-1}\sum_{j=1}^{H^l}\sum_{k=1}^{H^{(l+1)}} \mu_j^{l} w_{jk}^{(l+1)}\mu_k^{(l+1)} + \sum_{m=1}^{M}\sum_{i=1}^{V^m} a_i^m v_i^m + \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}} b_j^{lm}\mu_j^{lm} + \sum_{c=1}^{Y} d_c y_c \\ &+ \sum_{l=(L_0+1)}^{L}\sum_{j=1}^{H^l} b_j^l\mu_j^l - \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}} \Big\{\mu_j^{lm}\ln\mu_j^{lm} + (1-\mu_j^{lm})\ln(1-\mu_j^{lm})\Big\} \\ &- \sum_{l=(L_0+1)}^{L}\sum_{j=1}^{H^l} \Big\{\mu_j^{l}\ln\mu_j^{l} + (1-\mu_j^{l})\ln(1-\mu_j^{l})\Big\} - \ln Z. \end{aligned} \tag{4.16}$$

Since the mean field parameters ($\boldsymbol{\mu}$) of $\mathcal{L}_v$ (4.16) define the equilibrium state of the model, they need to updated accordingly. In order to obtain the mean field parameters of the proposed MDDBM model, the bound $\mathcal{L}_v$ (4.16) is maximized with respect to $\boldsymbol{\mu}$ for a fixed parameter set $\boldsymbol{\theta}_{mddbm}$. The update rule for the nodes of first hidden layer corresponding to $m$-th modality is given by $\dfrac{\partial \mathcal{L}_v}{\partial \mu_j^{1m}} = 0$, which leads to

$$\mu_j^{1m} = \sigma\Bigg(\sum_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum_{k=1}^{H^{2m}} w_{jk}^{2m}\mu_k^{2m} + \sum_{c=1}^{Y} y_c u_{cj}^{1m} + b_j^{1m}\Bigg),\ \forall m \in \{1,2,\cdots,M\}; \tag{4.17}$$

where $\sigma(x) = \dfrac{1}{1+e^{-x}}$ is the sigmoid function. Similarly, the following update rules can be obtained:

$$\begin{aligned}\mu_j^{lm} = \sigma\Bigg(&\sum_{k=1}^{H^{(l-1)m}} \mu_k^{(l-1)m} w_{kj}^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m}\mu_k^{(l+1)m} + \sum_{c=1}^{Y} y_c u_{cj}^{lm} + b_j^{lm}\Bigg), \\ &\text{for } 1 < l < L_0 \text{ and } \forall m; \end{aligned} \tag{4.18}$$

$$\mu_j^{L_0 m} = \sigma\Bigg( \sum_{k=1}^{H^{(L_0-1)m}} \mu_k^{(L_0-1)m} w_{kj}^{L_0 m} + \sum_{k=1}^{H^{(L_0+1)}} w_{jk}^{(L_0+1)m} \mu_k^{(L_0+1)} + \sum_{c=1}^{Y} y_c u_{cj}^{L_0 m} + b_j^{L_0 m} \Bigg), \forall m; \tag{4.19}$$

$$\mu_j^{(L_0+1)} = \sigma\Bigg( \sum_{m=1}^{M} \sum_{k=1}^{H^{L_0 m}} \mu_k^{L_0 m} w_{kj}^{(L_0+1)m} + \sum_{c=1}^{Y} y_c u_{cj}^{(L_0+1)} + \sum_{k=1}^{H^{(L_0+2)}} w_{jk}^{(L_0+2)} \mu_k^{(L_0+2)} + b_j^{(L_0+1)} \Bigg); \tag{4.20}$$

$$\mu_j^{l} = \sigma\Bigg( \sum_{k=1}^{H^{(l-1)}} \mu_k^{(l-1)} w_{kj}^{l} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^{(l+1)} \mu_k^{(l+1)} + \sum_{c=1}^{Y} y_c u_{cj}^{l} + b_j^{l} \Bigg), \text{ for } (L_0+1) < l < L; \tag{4.21}$$

$$\text{and } \mu_j^{L} = \sigma\Bigg( \sum_{k=1}^{H^{(L-1)}} \mu_k^{(L-1)} w_{kj}^{L} + \sum_{c=1}^{Y} y_c u_{cj}^{L} + b_j^{L} \Bigg). \tag{4.22}$$

Thus, given a training data along with the corresponding class label $\{\mathbf{v}^m, \mathbf{y}\}$, the equilibrium state of the model is estimated using the concept of mean field theory. Now, based on the mean field parameters, obtained using (4.17)-(4.22), the parameter set $\boldsymbol{\theta}_{mddbm}$ of the proposed MDDBM architecture, corresponding to the data-dependent expectation, can be learned by maximizing $\mathcal{L}_v$ with respect to $\boldsymbol{\theta}_{mddbm}$ for the equilibrium mean field parameters $\boldsymbol{\mu}$, which are given by

$$\begin{aligned}
&\frac{\partial \mathcal{L}_v}{\partial w_{ij}^{1m}} = v_i^m \mu_j^{1m}; \quad \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{lm}} = \mu_j^{(l-1)m} \mu_k^{lm}, \text{ for } 1 < l \leqslant L_0; \quad \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{(L_0+1)m}} = \mu_j^{L_0 m} \mu_k^{(L_0+1)}; \\
&\frac{\partial \mathcal{L}_v}{\partial w_{jk}^{l}} = \mu_j^{(l-1)} \mu_k^{l}, \text{ for } (L_0+1) < l \leqslant L; \quad \frac{\partial \mathcal{L}_v}{\partial u_{cj}^{lm}} = y_c \mu_j^{lm}, \text{ for } 1 \leqslant l \leqslant L_0; \\
&\frac{\partial \mathcal{L}_v}{\partial u_{cj}^{l}} = y_c \mu_j^{l}, \text{ for } L_0 < l \leqslant L; \quad \frac{\partial \mathcal{L}_v}{\partial b_j^{lm}} = \mu_j^{lm}, \text{ for } 1 \leqslant l \leqslant L_0; \\
&\frac{\partial \mathcal{L}_v}{\partial a_i^{m}} = v_i^m; \quad \frac{\partial \mathcal{L}_v}{\partial b_j^{l}} = \mu_j^{l}, \text{ for } L_0 < l \leqslant L; \text{ and } \frac{\partial \mathcal{L}_v}{\partial d_c} = y_c; \ \forall m \in \{1, 2, \cdots, M\}.
\end{aligned} \tag{4.23}$$

So, the data-dependent expectations of (4.10) are estimated by the gradient ascent on the lower bound of the proposed MDDBM architecture.

### 4.3.3 Estimation of Data-Independent Expectations

Now, the energy gradient with respect to the model distribution is to be estimated. The Markov Chain Monte Carlo based stochastic approximation procedure [190] is considered to approximate the data-independent expectations. 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 $\boldsymbol{\theta}_{mddbm}$. The conditional distributions corresponding to the proposed MDDBM model are given by

$$P(\mathbf{h}^{1m}|\mathbf{v}^m,\mathbf{y},\mathbf{h}^{2m}) = \frac{P(\mathbf{h}^{1m},\mathbf{v}^m,\mathbf{y},\mathbf{h}^{2m})}{\sum\limits_{\mathbf{h}^{1m}} P(\mathbf{h}^{1m},\mathbf{v}^m,\mathbf{y},\mathbf{h}^{2m})}$$

$$= \frac{\prod\limits_{j=1}^{H^{1m}} e^{h_j^{1m}(\sum\limits_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum\limits_{c=1}^{Y} y_c u_{cj}^{1m} + \sum\limits_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + b_j^{1m})}}{\prod\limits_{j=1}^{H^{1m}} \sum\limits_{\mathbf{h}^{1m}} e^{h_j^{1m}(\sum\limits_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum\limits_{c=1}^{Y} y_c u_{cj}^{1m} + \sum\limits_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + b_j^{1m})}}, \ \forall m \in \{1,2,\cdots,M\}. \quad (4.24)$$

$$\text{Hence, } P(h_j^{1m}|\mathbf{v}^m,\mathbf{y},\mathbf{h}^{2m}) = \frac{e^{h_j^{1m}(\sum\limits_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum\limits_{c=1}^{Y} y_c u_{cj}^{1m} + \sum\limits_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + b_j^{1m})}}{\sum\limits_{\mathbf{h}^{1m}} e^{h_j^{1m}(\sum\limits_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum\limits_{c=1}^{Y} y_c u_{cj}^{1m} + \sum\limits_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + b_j^{1m})}}$$

$$= \sigma\left(\sum_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + \sum_{c=1}^{Y} y_c u_{cj}^{1m} + b_j^{1m}\right), \forall m. \quad (4.25)$$

Similarly, the subsequent conditional distributions can be obtained:

$$P(h_j^{lm}|\mathbf{h}^{(l-1)m},\mathbf{h}^{(l+1)m},\mathbf{y}) = \sigma\Bigg(\sum_{k=1}^{H^{(l-1)m}} h_k^{(l-1)m} w_{kj}^{lm} + \sum_{c=1}^{Y} y_c u_{cj}^{lm} + b_j^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m} h_k^{(l+1)m}\Bigg), \text{ for } 1 < l < L_0, \ \forall m; \quad (4.26)$$

$$P(h_j^{L_0m}|\mathbf{h}^{(L_0-1)m},\mathbf{h}^{(L_0+1)},\mathbf{y}) = \sigma\Bigg(\sum_{k=1}^{H^{(L_0-1)m}} h_k^{(L_0-1)m} w_{kj}^{L_0m} + \sum_{c=1}^{Y} y_c u_{cj}^{L_0m} + b_j^{L_0m} + \sum_{k=1}^{H^{(L_0+1)}} w_{jk}^{(L_0+1)m} h_k^{(L_0+1)}\Bigg), \ \forall m; \quad (4.27)$$

$$P(h_j^{(L_0+1)}|\mathbf{h}^{L_0m},\mathbf{h}^{(L_0+2)},\mathbf{y}) = \sigma\Bigg(\sum_{m=1}^{M}\sum_{k=1}^{H^{L_0m}} h_k^{L_0m} w_{kj}^{(L_0+1)m} + \sum_{c=1}^{Y} y_c u_{cj}^{(L_0+1)} + b_j^{(L_0+1)} + \sum_{k=1}^{H^{(L_0+2)}} w_{jk}^{(L_0+2)} h_k^{(L_0+2)}\Bigg); \quad (4.28)$$

$$P(h_j^{l}|\mathbf{h}^{(l-1)},\mathbf{h}^{(l+1)},\mathbf{y}) = \sigma\Bigg(\sum_{k=1}^{H^{(l-1)}} h_k^{(l-1)} w_{kj}^{l} + \sum_{c=1}^{Y} y_c u_{cj}^{l} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^{(l+1)} h_k^{(l+1)} + b_j^{l}\Bigg), \quad \text{for } (L_0+1) < l < L; \quad (4.29)$$

$$P(h_j^{L}|\mathbf{h}^{(L-1)},\mathbf{y}) = \sigma\left(\sum_{k=1}^{H^{(L-1)}} h_k^{(L-1)} w_{kj}^{L} + \sum_{c=1}^{Y} y_c u_{cj}^{L} + b_j^{L}\right); \quad (4.30)$$

$$P(v_i^m|\mathbf{h}^{1m}) = \sigma\left(\sum_{j=1}^{H^{1m}} w_{ij}^{1m} h_j^{1m} + a_i^m\right), \ \forall m; \quad (4.31)$$

$$P(y_c|\mathbf{h}^{11},\cdots,\mathbf{h}^{L_0M},\mathbf{h}^{L_0+1},\cdots,\mathbf{h}^{L}) = \frac{e^{X_c}}{\sum_{\bar{c}=1}^{Y} e^{X_{\bar{c}}}};$$

$$\text{where, } X_c = \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}} u_{cj}^{lm} h_j^{lm} + \sum_{l=(L_0+1)}^{L}\sum_{j=1}^{H^l} u_{cj}^{l} h_j^{l} + d_c. \tag{4.32}$$

It is to be mentioned here that if a Markov chain is run for sufficient number of steps, then it can be ensured that the chain will converge to an 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 (4.25)-(4.32). 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 superscript tilde (e.g., $\tilde{v}$);

$$\frac{\partial E_{mddbm}}{\partial w_{ij}^{1m}} = \tilde{v}_i^m \tilde{h}_j^{1m};\ \frac{\partial E_{mddbm}}{\partial w_{jk}^{lm}} = \tilde{h}_j^{(l-1)m}\tilde{h}_k^{lm},\ \text{for } 1 < l \leqslant L_0;\ \frac{\partial E_{mddbm}}{\partial w_{jk}^{(L_0+1)m}} = \tilde{h}_j^{L_0 m}\tilde{h}_k^{(L_0+1)};$$
$$\frac{\partial E_{mddbm}}{\partial w_{jk}^{l}} = \tilde{h}_j^{(l-1)}\tilde{h}_k^{l},\ \text{for } (L_0+1) < l \leqslant L;\ \ \frac{\partial E_{mddbm}}{\partial u_{cj}^{lm}} = \tilde{y}_c\tilde{h}_j^{lm},\ \text{for } 1 \leqslant l \leqslant L_0;$$
$$\frac{\partial E_{mddbm}}{\partial u_{cj}^{l}} = \tilde{y}_c\tilde{h}_j^{l},\ \text{for } L_0 < l \leqslant L;\ \ \frac{\partial E_{mddbm}}{\partial a_i^{m}} = \tilde{v}_i^{m};\ \ \frac{\partial E}{\partial b_j^{lm}} = \tilde{h}_j^{lm},\ \text{for } 1 \leqslant l \leqslant L_0;$$
$$\frac{\partial E_{mddbm}}{\partial b_j^{l}} = \tilde{h}_j^{l},\ \text{for } L_0 < l \leqslant L;\ \text{and}\ \ \frac{\partial E_{mddbm}}{\partial d_c} = \tilde{y}_c;\ \forall m \in \{1,2,\cdots,M\}. \tag{4.33}$$

Hence, the proposed model can be efficiently learned from data-dependent and data-independent estimates, obtained in (4.23) and (4.33), respectively.

### 4.3.4 Learning Rule of MDDBM Model Parameters

Let $N$, $S$, $t$, and $\eta$ 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 MDDBM architecture, required to perform gradient ascent on the log-likelihood function corresponding to the energy function of (4.8), is as follows:

$$\boldsymbol{\theta}_{mddbm}^{(t+1)} = \boldsymbol{\theta}_{mddbm}^{t} + \Delta\boldsymbol{\theta}_{mddbm}^{t}; \tag{4.34}$$

$$\text{where } \Delta\boldsymbol{\theta}_{mddbm}^{t} = \eta\left\{\frac{1}{N}\sum_{n=1}^{N}\left(\frac{\partial\mathcal{L}_v}{\partial\boldsymbol{\theta}_{mddbm}}\right)_n - \frac{1}{S}\sum_{s=1}^{S}\left(\frac{\partial E_{mddbm}}{\partial\boldsymbol{\theta}_{mddbm}}\right)_s\right\} - \rho\boldsymbol{\theta}_{mddbm}^{t} + \zeta\Delta\boldsymbol{\theta}_{mddbm}^{(t-1)}. \tag{4.35}$$

So, the energy function as well as the learning rule of the proposed MDDBM model are defined in such a way that the model encapsulates the underlying non-linear data

distribution of the given observations and ensures that similarity in the latent space implies similarity in the corresponding concepts. From (4.34) and (4.35), it can be observed that the update rules of the proposed MDDBM model follow the Hebbian rule, which is originally employed for the learning of standard Boltzmann machine [136]. 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 [8]. The learning algorithm of the proposed MDDBM model is illustrated in Algorithm 4.1.

**Algorithm 4.1** Learning of MDDBM Architecture.

**Input:** Set of $N$ training data vectors $\{\mathbf{v}^m\}_{n=1}^N$ for $M$ modalities along with the corresponding set of class labels $\{\mathbf{y}\}_{n=1}^N$, number of persistent chains ($S$), number of epochs ($\tau$), learning rate ($\eta$), weight decay ($\rho$), momentum ($\zeta$), and number of Gibbs steps ($\alpha$).
**Output:** Final parameter set $\boldsymbol{\theta}^{\tau}_{mddbm}$ of the architecture.

1: Perform greedy layer-wise pretraining to initialize the set of model parameters, $\boldsymbol{\theta}^{0}_{mddbm}$.
2: Randomly initialize $S$ Markov chains $\{\tilde{\mathbf{v}}^{m^0}, \tilde{\mathbf{y}}^0, \tilde{\mathbf{h}}^{1m^0}, \cdots, \tilde{\mathbf{h}}^{L_0m^0}, \tilde{\mathbf{h}}^{(L_0+1)^0}, \cdots, \tilde{\mathbf{h}}^{L^0}\}_{s=1}^S$.
3: **for** each epoch $t = 0$ to $\tau$ **do**
4: // Variational inference
5: **for** each training sample $n = 1$ to $N$ **do**
6: (i) Run mean field updates using (4.17)-(4.22) until convergence.
7: (ii) Save the obtained mean field parameter ($\boldsymbol{\mu}$) for the corresponding training sample, $\boldsymbol{\mu}_n = \boldsymbol{\mu}$.
8: **end for**
9: // Stochastic approximation
10: **for** each persistent chain $s = 1$ to $S$ **do**
11: Run the chain for $\alpha$-steps and sample the state $\{\tilde{\mathbf{v}}^{m^{t+1}}, \tilde{\mathbf{y}}^{t+1}, \tilde{\mathbf{h}}^{1m^{t+1}}, \cdots, \tilde{\mathbf{h}}^{L_0m^{t+1}}, \tilde{\mathbf{h}}^{(L_0+1)^{t+1}}, \cdots, \tilde{\mathbf{h}}^{L^{t+1}}\}$ from $\{\tilde{\mathbf{v}}^{m^t}, \tilde{\mathbf{y}}^t, \tilde{\mathbf{h}}^{1m^t}, \cdots, \tilde{\mathbf{h}}^{L_0m^t}, \tilde{\mathbf{h}}^{(L_0+1)^t}, \cdots, \tilde{\mathbf{h}}^{L^t}\}$ using (4.25)-(4.32).
12: **end for**
13: Update the parameters of the model from $\boldsymbol{\theta}^{t}_{mddbm}$ to $\boldsymbol{\theta}^{(t+1)}_{mddbm}$ using (4.34).
14: **end for**

## 4.4 Experimental Results and Discussions

In this section, the proficiency of the proposed MDDBM architecture is analyzed extensively and corresponding results are presented. Several state-of-the-art approaches, namely, rCCA [124], DCCA [9], DCCAE [202], MCCA [96], GMCCA [25], GMKCCA [25], LasCCA [53], DisCCA [53], MDBM [180], dMCCA [178], TOCCA [30], DACCA [44], DCCA-VG [193], and TDDCCA [37], are considered in the current study to establish the efficacy of the proposed model. In order to evaluate the performance of the existing algorithms as well as the proposed model, both training-testing and 10-fold cross-validation (CV) are performed. In case of training-testing, overall classification accuracy is considered, while for 10-fold 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.

### 4.4.1 Description of Data Sets

In this study, four benchmark databases, namely, CiteSeer [161], Cora [161], NUS-WIDE-OBJECT (NW-OBJECT) [27], and Reuters [7], and two cancer data sets are considered. CiteSeer and Cora (http://networkrepos- itory.com) consist of scientific publications with annotated labels, NW-OBJECT (https://lms.comp.nus.edu.sg/wp-content/uploads/2019/research/nuswide/NUS-WIDE.html) is image based database, whereas Reuters (http://archive.ics.uci.edu/ml/machine-learning-data- bases/00259) is a multilingual categorization data set, containing documents written in English language with the corresponding translations in four different languages.

Table 4.1: Description of Data Sets

| | Data Sets | Sample | Class | View | $V^1$ | $V^2$ | $V^3$ | $V^4$ | $V^5$ | $V^6$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Benchmark | 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 | - |
| Omics | CESC | 104 | 3 | 4 | 291368 | 192 | 174 | 12028 | - | - |
| | LGG | 374 | 3 | 5 | 293965 | 181 | 139 | 11973 | 6261 | - |

Different subtypes are identified for two real-life cancer data sets, which include cervical carcinoma (CESC) and lower grade glioma (LGG). These data sets are obtained from The Cancer Genome Atlas (TCGA) [194] (http://cancergenome.nih.gov). While CESC has four modalities, namely, DNA methylation (mDNA), protein expression (Protein), microRNA sequence (microRNA), and gene expression (RNA), LGG database has five modalities, namely, mDNA, Protein, microRNA, RNA, and copy number segmentation (CNS). A brief description of the data sets, which includes number of samples, number of classes, number of views, and number of features in each view, is presented in Table 4.1. It can be noted from Table 4.1 that CiteSeer and Cora databases have large dimension of the feature sets, NW-OBJECT data set has large number of samples with small number of features in each view, whereas Reuters database has large number of samples with large dimension of each of the feature sets. On the other hand, the omics data sets offer the problem of high dimensional feature sets with small number of samples. The performance of the proposed model as well as the existing algorithms are evaluated on these databases. For experimental purposes, each data set is randomly partitioned into two sets for training-testing and ten separate folds for 10-fold CV. In both the cases, the samples are equally distributed with respect to different classes. Detailed description of the data sets is reported in Appendix A.

### 4.4.2 Model Architecture and Implementation Details

In the proposed MDDBM architecture, two modality-specific hidden layers and one modality-free hidden layer are considered for all the experiments. Each of the modality-specific hidden layers consists of 50 hidden nodes, whereas the number of nodes in joint representation is considered to be 10 in the current study. The greedy layer-wise pretraining [163] is performed to initialize the parameters of the architecture sensibly by learning a stack of

modified restricted Boltzmann machines. The hidden nodes of the architecture are represented by the corresponding probability values and the parameters are updated based on the mini-batches formed from the given set 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 learning rate is initialized at 0.01 and then, gradually decreased with increase in number of epochs. For the estimation of data-independent expectations, 100 Gibbs steps and 20 separate Markov chains are considered. For the classification of samples into one of the known classes, maximum class probability is taken into consideration corresponding to the class nodes of the model.

### 4.4.3 Effectiveness of Proposed MDDBM Architecture

The performance of different aspects of the proposed MDDBM architecture is analyzed in this section, which includes significance of incorporating supervised information and efficacy of proposed method as feature extractor. The corresponding results are reported in Figure 4.2, and Table 4.2 and Table 4.3. The scatter plots of Figure 4.2 are depicted by considering the most relevant feature at $x$-axis and the corresponding most significant feature at $y$-axis, obtained using the concept of rough hypercuboid approach [126]. While the top row of Figure 4.2 corresponds to MDBM model, the plots of last row is obtained from the MDDBM architecture.

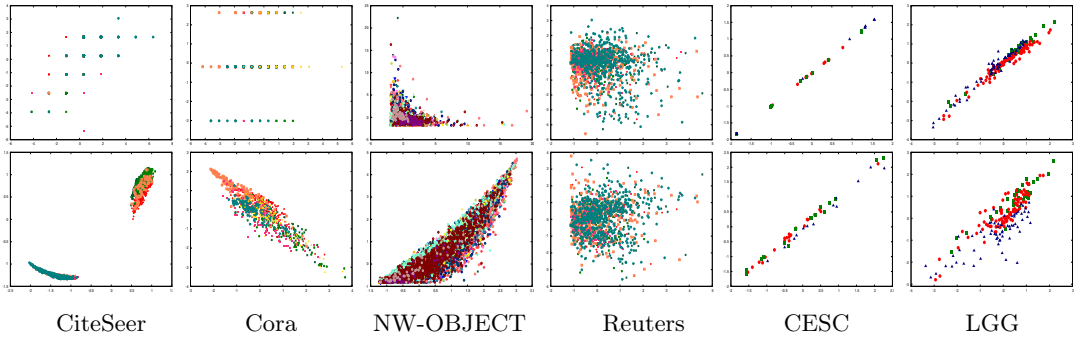



Figure 4.2: Scatter plots of MDBM (top row) and MDDBM (bottom row) for benchmark and omics data sets.

#### 4.4.3.1 Significance of Incorporating Supervised Information

The merits of DBM are judiciously integrated with the supervised information of sample categories to develop the proposed MDDBM model. While MDBM [179] is an unsupervised model, the proposed framework incorporates class nodes into the architecture to enhance the discriminative ability of the model. In this section, the significance of incorporating the supervised information into MDDBM architecture is established. It is to be mentioned here that the MDBM model considers the support vector machine (SVM) for classification of the samples into different categories, whereas in case of the proposed MDDBM model, the class labels of the samples are predicted from the architecture itself. The scatter plots of Figure 4.2, 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 4.2, where MDDBM performs significantly better than MDBM for training-testing on all the benchmark databases. Considering the graphs of Figure 4.2, corresponding to the omics data sets, it can be noticed that the samples from different cancer subtypes are better discriminated in case of MDDBM as compared to MDBM. Results reported in Table 4.3 also confirm that MDDBM outperforms MDBM, irrespective of experimental set-up and cancer data sets considered. Statistical significance analysis reveals that with reference to the MDBM model, the proposed model achieves significantly better p-values in all the 4 cases.

Table 4.2: Effectiveness of Proposed MDDBM Architecture on Benchmark Data

| Data Sets | MDBM+SVM | MDDBM for feature extraction | | MDDBM |
|---|---|---|---|---|
| | | SVM | Bayes | |
| CiteSeer | 17.80 | 65.76 | 67.03 | **71.93** |
| Cora | 10.99 | 60.27 | 52.94 | **64.37** |
| NW-OBJECT | 26.07 | 41.24 | **45.05** | 43.75 |
| Reuters | 46.84 | 59.85 | 59.67 | **61.25** |

Table 4.3: Effectiveness of Proposed MDDBM Architecture on Omics Data Sets

| Data Sets | Different Metrics | | MDBM+SVM | MDDBM for feature extraction | | MDDBM |
|---|---|---|---|---|---|---|
| | | | | SVM | Bayes | |
| CESC | Train-Test | | 48.08 | 55.77 | 51.92 | **67.31** |
| | 10-fold CV | Mean | 52.50 | 61.11 | **65.00** | 64.17 |
| | | Median | 54.17 | 62.96 | 62.50 | **66.67** |
| | | StdDev | 17.59 | 6.69 | 10.24 | **5.62** |
| | | Paired-$t$:p | 3.87E-02 | 1.49E-01 | 5.76E-01 | - |
| | | Wilcoxon:p | 2.92E-02 | 1.21E-01 | 6.40E-01 | - |
| LGG | Train-Test | | 65.05 | 76.34 | 74.19 | **79.57** |
| | 10-fold CV | Mean | 27.63 | 59.21 | 56.05 | **63.95** |
| | | Median | 18.42 | 60.53 | 52.63 | **63.16** |
| | | StdDev | 14.90 | 6.59 | **5.69** | 6.08 |
| | | Paired-$t$:p | 2.38E-06 | 7.33E-02 | 1.49E-02 | - |
| | | Wilcoxon:p | 2.45E-03 | 5.70E-02 | 1.07E-02 | - |

#### 4.4.3.2 Efficacy of Proposed Model as Feature Extractor

Apart from the classification purpose, the proposed MDDBM model can also be considered as feature extractor. The features extracted by the MDDBM architecture at the final layer are provided as input to various classifiers, namely, SVM and Bayes classifier, and the classification performance is studied in order to estimate the discriminative ability of the joint subspace. From the results reported in Table 4.2, it can be noted 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 Bayes

classifier as compared to the MDDBM architecture for NW-OBJECT data set. The results presented in Table 4.3 state that the Bayes classifier can suitably identify different subtypes of CESC data for 10-fold CV. However, considerable classification accuracy is achieved on all the data sets for both training-testing and 10-fold CV when the cancer subtypes are identified using the proposed architecture itself. Statistical significance analysis reveals that out of total 8 cases, the proposed MDDBM framework attains significantly better p-values in 2 cases, better but not significant p-values in 4 cases.

### 4.4.4 Performance Analysis for Pair of Modalities

In the existing literature, several approaches are present which compute linear on non-linear transformations of the given input pair of views. In this section, the performance of these existing methods is compared with that of the proposed MDDBM architecture on both benchmark and real-life cancer data sets. Two representative pairs of modalities, which include the pair of first and third modalities and the pair of third and fourth modalities, are chosen for each of the data sets and given as input to both state-of-the-art methods as well as the proposed architecture. The state-of-the-art approaches include rCCA [124], DCCA [9], and DCCAE [202]. While rCCA is a non-deep approach, both DCCA and DCCAE are based on deep architectures. In case of rCCA and DCCA, 50 features are extracted at final layer and 10 features are extracted at output layer for DCCAE, as suggested in the respective papers. The obtained features are then applied to the input of the SVM for classification purpose. However, in case of the proposed architecture, 10 features are considered for the joint representation, and the class labels are predicted from the architecture itself. The corresponding results are reported in Table 4.4 for benchmark databases and Table 4.5 for omics data sets.

Table 4.4: Comparative Performance Analysis for Pair of Modalities on Benchmark Data

| Data Sets | rCCA | DCCA | DCCAE | MDDBM | rCCA | DCCA | DCCAE | MDDBM |
|---|---|---|---|---|---|---|---|---|
| | $\mathbf{v}^1\mathbf{v}^3$ | | | | $\mathbf{v}^3\mathbf{v}^4$ | | | |
| CiteSeer | 41.05 | 58.13 | 51.95 | **66.85** | 47.50 | 46.41 | 45.41 | **70.66** |
| Cora | 38.73 | 31.96 | 48.17 | **52.83** | 43.06 | 44.51 | 46.28 | **57.16** |
| NW-OBJECT | 35.41 | 26.75 | 37.07 | **40.51** | 35.29 | 34.74 | 27.97 | **41.29** |
| Reuters | 40.15 | 42.51 | 45.50 | **54.50** | 33.88 | 38.78 | 39.26 | **55.22** |

From the results reported in Table 4.4, it can be observed that the proposed model outperforms the existing approaches on all the benchmark databases, irrespective of the pairs of modalities considered. The results presented in Table 4.5 demonstrates that the existing methods can efficiently identify subtypes of carcinoma samples for certain pairs of views, but fail to provide similar results for some other pairs. For example, the DCCAE method can identify over 75% of the samples correctly for input pair of views $\mathbf{v}^1\mathbf{v}^3$ on LGG data set for 10-fold CV, but it has failed to provide similar result for the input pair $\mathbf{v}^3\mathbf{v}^4$. Hence, the performance of these methods depends on the given input pair of modalities. However, the proposed method is not only able to achieve similar results for the given input pairs of views, but also attain highest classification accuracy on all the omics data sets for both training-testing and 10-fold CV. Statistical significance analysis demonstrates

Table 4.5: Comparative Performance Analysis for Pair of Modalities on Omics Data Sets

| Data Sets | Different Metrics | | rCCA | DCCA | DCCAE | MDDBM | rCCA | DCCA | DCCAE | MDDBM |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | $\mathbf{v}^1\mathbf{v}^3$ | | | | | | | |
| CESC | Train-Test | | 44.23 | 46.15 | 51.92 | **59.62** | 42.31 | 42.31 | 46.15 | **55.77** |
| | 10-fold CV | Mean | 50.00 | 49.17 | 53.33 | **63.33** | 46.67 | 46.67 | 45.83 | **60.83** |
| | | Median | 50.00 | 50.00 | 50.00 | **62.50** | 50.00 | 50.00 | 50.00 | **58.33** |
| | | StdDev | **3.93** | 7.30 | 8.96 | 11.92 | 9.78 | 9.78 | 10.58 | **5.62** |
| | | Paired-$t$:p | 2.28E-03 | 4.73E-03 | 6.67E-02 | - | 1.51E-03 | 1.51E-03 | 1.93E-03 | - |
| | | Wilcoxon:p | 5.76E-03 | 6.14E-03 | 5.46E-02 | - | 3.76E-03 | 3.76E-03 | 7.36E-03 | - |
| LGG | Train-Test | | 53.23 | 62.90 | 64.52 | **74.19** | 48.39 | 48.39 | 48.39 | **67.20** |
| | 10-fold CV | Mean | 47.37 | 66.32 | 75.79 | **77.63** | 47.37 | 47.37 | 47.37 | **62.11** |
| | | Median | 47.37 | 65.79 | **77.63** | **77.63** | 47.37 | 47.37 | 47.37 | **64.47** |
| | | StdDev | **0.00** | 8.30 | 12.33 | 6.11 | **0.00** | **0.00** | **0.00** | 6.93 |
| | | Paired-$t$:p | 3.87E-08 | 1.14E-03 | 3.51E-01 | - | 4.30E-05 | 4.30E-05 | 4.30E-05 | - |
| | | Wilcoxon:p | 2.46E-03 | 8.23E-03 | 3.61E-01 | - | 2.36E-03 | 2.36E-03 | 2.36E-03 | - |

that out of total 24 cases, the proposed approach achieves significantly better p-values for 20 cases, and better but not significant p-values for the rest 4 cases.

### 4.4.5 Comparative Performance Analysis

Finally, the classification performance of the proposed MDDBM architecture is studied with reference to several existing methods on benchmark as well as omics data sets, and the corresponding results are reported in Table 4.6 and Table 4.8 for benchmark databases, and Table 4.7 and Table 4.9 for real-life cancer data sets. The scatter plots of existing algorithms and the proposed model are presented in Figure 4.3 and Figure 4.4. 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 for classification purpose. The existing algorithms include multiset classical approaches as well as multi-view deep learning based methods.

#### 4.4.5.1 Performance of Existing Classical Approaches

In this section, the performance of the proposed method is analyzed with reference to several multiset classical approaches, namely, MCCA [96], GMCCA [25], GMKCCA [25], LasCCA [53], and DisCCA [53]. All the existing algorithms extract 25 features from the given input views to represent the joint subspace, which are then applied to the input of SVM for classification purpose. The scatter plots of Figure 4.3, obtained on benchmark databases, reveal that the separation between the samples of various categories has improved in case of MDDBM as compared to the existing approaches. Similar results can be noticed in Table 4.6, where the proposed method attains highest classification accuracy with respect to the existing multiset classical approaches on all the four benchmark databases. Considering the graphs of Figure 4.3 for omics data sets, it can be observed that the samples from different cancer subtypes are better discriminated in case of MDDBM with reference to the

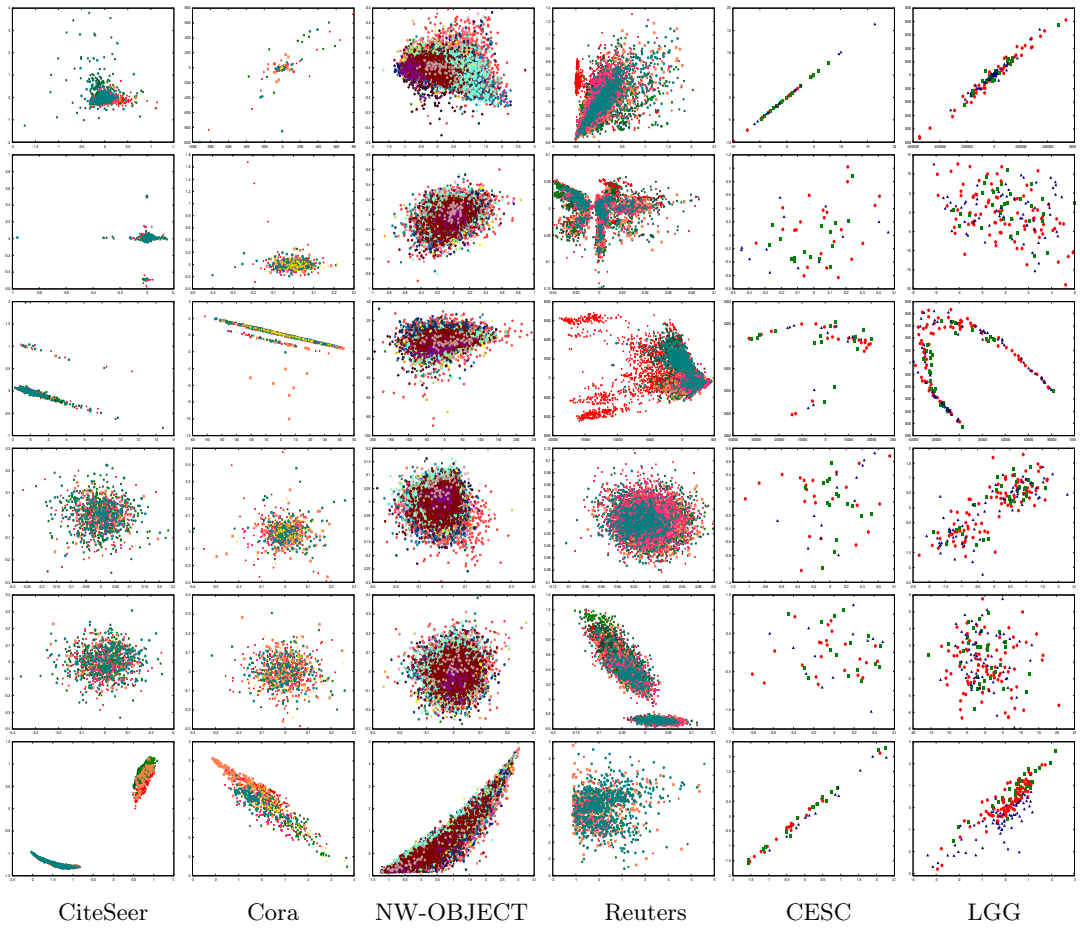



Figure 4.3: Scatter plots of existing multiset classical approaches on benchmark and omics data sets (from top to bottom row: MCCA, GMCCA, GMKCCA, LasCCA, DisCCA, and MDDBM, respectively).

classical approaches. Results reported in Table 4.7 also confirm that the proposed method outperforms the five multiset classical methods on all cancer data sets for both training-testing and 10-fold CV. Statistical significance tests reveal that the proposed architecture achieves significantly better p-values for all the 20 cases.

#### 4.4.5.2 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 [178], TOCCA [30], DACCA [44], DCCA-VG [193], and TDDCCA [37], and the corresponding results are presented in Figure 4.4, Table 4.8, and Table 4.9. For TOCCA, DACCA, DCCA-VG, and TDDCCA, 50, 80, 20, and 50 features are extracted, respectively, from the given views, which are then applied to the input of the SVM for classification. In case of dMCCA, 50 features are extracted at the final layer. Since the method is essentially a feed-forward network, it does not require any additional classifier for class label prediction. The architecture for each of the models follows the same as suggested in the corresponding papers.

Table 4.6: Performance Analysis of Classical Approaches on Benchmark Data Sets

| Data Sets | MCCA | GMCCA | GMKCCA | LasCCA | DisCCA | MDDBM |
|---|---|---|---|---|---|---|
| CiteSeer | 58.13 | 23.43 | 24.98 | 22.25 | 20.71 | **71.93** |
| Cora | 32.85 | 30.97 | 30.19 | 31.63 | 30.19 | **64.37** |
| NW-OBJECT | 30.34 | 4.56 | 6.43 | 7.40 | 10.93 | **43.75** |
| Reuters | 57.50 | 24.78 | 28.69 | 28.67 | 23.27 | **61.25** |

Table 4.7: Comparative Performance Analysis of Classical Approaches on Omics Data Sets

| Data Sets | Different Metrics | | MCCA | GMCCA | GMKCCA | LasCCA | DisCCA | MDDBM |
|---|---|---|---|---|---|---|---|---|
| CESC | Train-Test | | 38.46 | 42.31 | 44.23 | 42.31 | 36.54 | **67.31** |
| | 10-fold CV | Mean | 45.83 | 49.17 | 38.33 | 35.00 | 39.17 | **64.17** |
| | | Median | 50.00 | 50.00 | 41.67 | 33.33 | 33.33 | **66.67** |
| | | StdDev | 13.75 | 14.41 | 11.92 | 15.61 | 10.43 | **5.62** |
| | | Paired-$t$:p | 1.62E-03 | 2.56E-03 | 1.16E-04 | 2.76E-04 | 4.39E-05 | - |
| | | Wilcoxon:p | 8.47E-03 | 6.84E-03 | 2.46E-03 | 5.71E-03 | 3.63E-03 | - |
| LGG | Train-Test | | 39.78 | 33.33 | 38.71 | 44.09 | 29.03 | **79.57** |
| | 10-fold CV | Mean | 35.53 | 40.53 | 33.16 | 38.68 | 38.68 | **63.95** |
| | | Median | 34.21 | 40.79 | 31.58 | 36.84 | 38.16 | **63.16** |
| | | StdDev | 7.67 | 8.70 | **4.99** | 7.95 | 7.24 | 6.08 |
| | | Paired-$t$:p | 1.84E-06 | 7.85E-05 | 7.08E-09 | 1.18E-04 | 1.44E-05 | - |
| | | Wilcoxon:p | 2.52E-03 | 3.44E-03 | 2.50E-03 | 3.46E-03 | 2.53E-03 | - |

From the results reported in Table 4.8, it can be noticed that except for DCCA-VG method on Reuters database, the proposed method performs considerably better than the existing multi-view deep learning based methods on all the benchmark databases. Similar inference can be drawn from the scatter plots of benchmark databases, presented in Figure 4.4. For the omics data sets, results are depicted in Figure 4.4 and Table 4.9, which describe that the DCCA-VG and TDDCCA approaches perform better than the proposed model on CESC data for 10-fold CV. However, significant improvement in performance can be noted in case of the proposed MDDBM architecture as compared to the existing deep learning based models, irrespective of the experimental set-up and data sets considered. Statistical significance tests demonstrate that out of total 24 cases, the proposed model achieves significantly better p-values for 20 cases.

Table 4.8: Comparative Performance Analysis of Deep Models on Benchmark Data Sets

| Data Sets | dMCCA | TOCCA | DACCA | DCCA-VG | TDDCCA | MDDBM |
|---|---|---|---|---|---|---|
| CiteSeer | 21.16 | 39.60 | 55.22 | 37.33 | 40.42 | **71.93** |
| Cora | 30.19 | 52.39 | 50.06 | 44.62 | 53.05 | **64.37** |
| NW-OBJECT | 17.80 | 33.61 | 38.42 | 19.23 | 17.80 | **43.75** |
| Reuters | 51.72 | 57.38 | 56.38 | **64.38** | 57.27 | 61.25 |

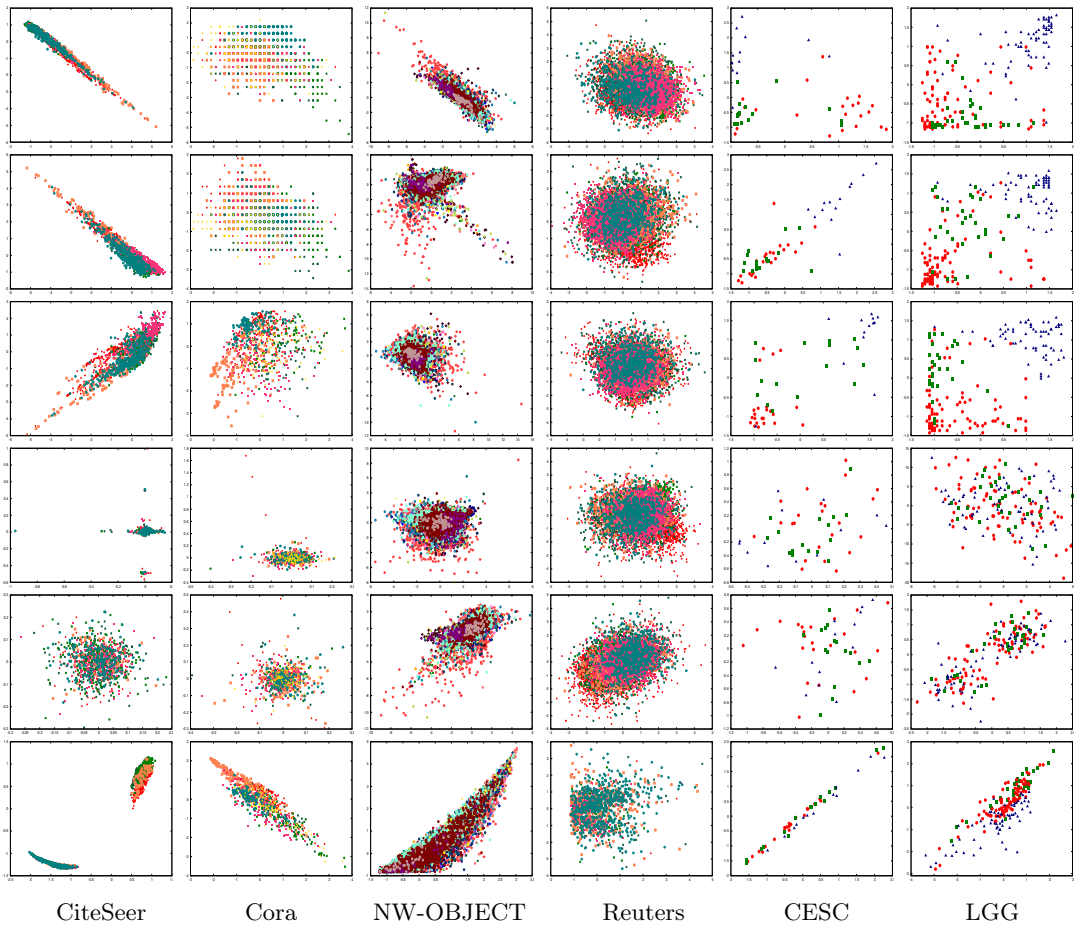



Figure 4.4: Scatter plots of existing deep models on benchmark and omics data sets (from top to bottom row: dMCCA, TOCCA, DACCA, DCCA-VG, TDDCCA, and MDDBM, respectively).

## 4.5 Conclusion

The major contribution of the paper is three-fold, namely, (a) developing MDDBM framework by judiciously integrating the theory of Boltzmann machine with the supervised information of sample categories; (b) demonstrating the effectiveness of the proposed architecture as feature extractor as well as classifier; and (c) illustrating the proficiency of the proposed method on different domains of applications, namely, object recognition, document classification, multilingual categorization, and cancer subtype identification.

The learning objective of the proposed architecture includes the merits of deep Boltzmann machines. It enables the network to figure out activations for the architecture nodes in each layer, which constitute a plausible explanation of how the observed data vectors would have been generated. 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 classifier. The proficiency of the proposed architecture is demonstrated on various multi-view data sets with reference to several classical as well as deep learning models, considering both

Table 4.9: Comparative Performance Analysis of Deep Architectures on Omics Data Sets

| Data Sets | Different Metrics | | dMCCA | TOCCA | DACCA | DCCA-VG | TDDCCA | MDDBM |
|---|---|---|---|---|---|---|---|---|
| CESC | Train-Test | | 51.92 | 36.54 | 47.12 | 65.38 | 53.98 | **67.31** |
| | 10-fold CV | Mean | 40.83 | 43.33 | 39.81 | 67.50 | **78.20** | 64.17 |
| | | Median | 41.67 | 50.00 | 39.42 | 70.83 | **78.20** | 66.67 |
| | | StdDev | 12.08 | 14.05 | 5.37 | 16.87 | **0.06** | 5.62 |
| | | Paired-$t$:p | 4.30E-05 | 1.00E-03 | 1.96E-07 | 7.05E-01 | 1.00E+00 | - |
| | | Wilcoxon:p | 3.30E-03 | 3.63E-03 | 2.53E-03 | 7.13E-01 | 9.98E-01 | - |
| LGG | Train-Test | | 62.37 | 45.70 | 65.59 | 77.96 | 66.85 | **79.57** |
| | 10-fold CV | Mean | 50.79 | 45.00 | 59.35 | 51.32 | 57.14 | **63.95** |
| | | Median | 47.37 | 46.05 | 58.87 | 51.32 | 57.00 | **63.16** |
| | | StdDev | 7.24 | 6.38 | 3.42 | 3.34 | **0.39** | 6.08 |
| | | Paired-$t$:p | 1.24E-04 | 5.80E-05 | 1.75E-02 | 5.80E-06 | 2.62E-03 | - |
| | | Wilcoxon:p | 3.37E-03 | 2.52E-03 | 1.42E-02 | 2.52E-03 | 6.26E-03 | - |

training-testing and ten-fold CV.

In the proposed MDDBM model, the joint subspace is learned from the individual modality-specific subspaces and class label information. However, it may so happen that the individual views correspond to completely different sources. For example, in case of cancer data sets, one view corresponds to DNA methylation data, whereas the other view refers to microRNA expression. In such a scenario, the individual hidden representations correspond to essentially different spaces. So, learning the joint subspace from the individual spaces may not be able to capture the cross-modal information. It is also reported in Table 4.9 that the existing TDDCCA method performs significantly better than the proposed model in categorizing the samples from CESC data set for 10-fold CV. This can possibly be explained by the fact that different views of the CESC data set are captured at various scales and so, it has become difficult for the proposed model to encapsulate the discriminative information from the individual spaces into the joint subspace. However, if the proposed model is learned in such a way that the modality-specific subspaces are highly correlated, then the inherent characteristics of the views can be efficiently modeled by the joint subspace. So, in the next chapter, the learning of the proposed architecture is defined in such a way that given the input views, the joint subspace is learned from maximally correlated subspaces.

# Chapter 5

# Discriminative Deep Canonical Correlation Analysis for Coherent Subspace Learning

## 5.1 Introduction

Advancement in information acquisition processes has entailed the development of predictive models for multi-view data analysis. The classification based on multi-view data has been exercised in numerous domains of applications, for example, object detection [114], tumor analysis [48], face recognition [81], 3D saliency detection [65], 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 diverse as well as coherent knowledge, corresponding to the given observations. Thus, judicious integration of information from multiple views, as evident in Chapter 3 and Chapter 4, can potentially provide a more comprehensive and discriminative representation of the data, as compared to each of the unimodal representations. In turn, it can stimulate the improvement in the proficiency of the multi-view learning models.

As mentioned in Chapter 4, several attempts have been made to characterize the inherent correlation across multiple views, of which the most acknowledged methods include multiset CCA (MCCA) [96], graph-regularized MCCA (GMCCA) [25], graph-regularized kernel MCCA (GMKCCA) [25], large-scale generalized CCA (LasCCA) [53], and distributed algorithm for generalized CCA (DisCCA) [53]. The regularized generalized CCA (RGCCA) has also been proposed in [187], which is a generalization of the regularized CCA for three or more sets of variables. It integrates the flexibility of partial least squares path modeling with the power of multiblock data analysis methods. In order to address with the singularity issue of the covariance matrices corresponding to the input views, an optimal shrinkage parameter is estimated for each view. As a result of which, the off-diagonal values of the covariance matrices are reduced, keeping the diagonal elements unchanged. Thus, the search space for obtaining the canonical variables will increase, which in turn, enhances the proficiency of the algorithm for multimodal data analysis. In order to obtain single

unified discriminant common space from the given multiple views by jointly optimizing the view-specific transforms corresponding to each of the input views, multi-view discriminant analysis (MvDA) [92] and MvDA with view-consistency (MvDA-VC) [93] have been developed which are discussed in Chapter 2 and Chapter 4 in details.

As reported in Chapter 4, a surging interest is noted in recent years for combining information from multiple modalities using deep learning based models, such as deep multiset CCA (dMCCA) [178], task optimal CCA (TOCCA) [30], multimodal deep Boltzmann machine (MDBM) [180], deep adversarial CCA (DACCA) [44], deep CCA with view generation (DCCA-VG) [193], and towards deep and discriminative CCA (TDDCCA) [37]. Rastegar et al. [157] have proposed a multimodal deep learning framework, termed as MDL-CW, which exploits the cross-weights between modality-specific representations and gradually learns interactions between the given modalities through a deep network. In this way, the modality that contains higher level information can help to find a better representation for other modalities. However, in MDL-CW model, the final representation of the multi-view data corresponds to concatenation of individual representations, obtained for each of the given input views. Although weights among the individual networks share information, the model lacks a joint representation learned over the space of multimodal inputs. In multimodal graph neural network (MMGNN) [54], an input data is represented as a graph, consisting of three sub-graphs. Then, three aggregators are introduced which guide the message passing from one graph to another to refine the nodes of the network. The initial representations of the nodes in the three sub-graphs are obtained from priors, learned from the deep convolution neural networks. In [209], multi-view generative adversarial network (MVGAN) have been developed to automatically expand the labelled multi-view samples, and the expanded dataset is then used to train the multistream convolutional neural network. While MMGNN and MVGAN frameworks concentrate on the view-specific information, they do not take into consideration the shared knowledge of different views.

From the existing literature, it can be observed that several deep architectures are considered to learn the joint representation of the given data from the input multiple modalities. The dMCCA model [178] is developed based on the framework of feed-forward network, whereas stacked autoencoder has been considered for MDL-CW [157]. While graph neural network is used for the development of MMGNN [54], generative adversarial network has been considered for the implementation of MVGAN [209]. 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 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, such as multimodal omics data or image and text as different views of the given data. The effectiveness of adversarial learning has a strong correlation with the distance between a test point and the manifold of training data embedded by the network. Consequently, the adversarial networks are more likely to be vulnerable to the blind-spot attacks, where the input observation resides in blind-spots or low density regions of the empirical distribution of training data but is still on the ground truth data manifold.

On the other hand, as discussed in Section 4.2.2 of Chapter 4, deep Boltzmann machine (DBM) [163] is an effective paradigm of undirected generative models that efficiently captures the non-linear dependencies between observed and latent variables by analyzing the energy landscape of the given observations. Unlike the feed-forward 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 a 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 in [163]. So, the learning procedure as well as the architecture of DBM compliment the psychological evidences. Furthermore, by following the gradient of variational lower bound on the likelihood objective, the parameters of all the layers in DBM can be jointly optimized, which is beneficial especially in case of learning models from heterogeneous data originating from different modalities. Hence, the joint representation in DBM based multi-view model is expected to encapsulate the underlying non-linear data distribution of the given observations. In this context, multimodal discriminative DBM (MDDBM) is developed in Chapter 4 to learn the joint subspace over the space of multimodal inputs. Also, incorporation of class nodes into the proposed deep architecture enhances the predictive ability of the model. Comparative performance analysis establishes the proficiency of the model in multi-view data analysis. However, in multi-view scenario, it may so happen that the modality-specific representations correspond to completely different spaces. So, learning the joint representation from the individual spaces may not be able to capture the cross-modal information appropriately. Hence, it is required that the learning objective of the multi-view model is defined in such a way that it can efficiently capture the correlated structures across several modalities.

In this regard, a novel architecture, termed as discriminative deep canonical correlation analysis (D2CCA), is proposed in this chapter by judiciously integrating the merits of MDDBM and the theory of CCA. 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 to maximally correlated subspaces. Hence, the joint representation, learned from the obtained subspaces, can efficiently capture the non-linear correlated structures across different modalities. Also, considering the class nodes in the architecture includes the supervised information of sample categories at each layer of the network, which in turn, allows the proposed D2CCA model to perform as feature extractor as well as classifier. Furthermore, the proposed framework is consolidated with corresponding convergence analysis. The proficiency of the D2CCA architecture is extensively studied and compared with numerous state-of-the-art methods on several benchmark and real-life cancer data sets considering both training-testing and ten-fold cross-validation. Some of the results of this chapter are reported in [104].

The rest of this chapter is organized as follows: Section 5.2 describes the architecture as well as the learning of the proposed D2CCA model. Various aspects of the proposed model are discussed in Section 5.3. The proficiency of the D2CCA framework is analyzed with reference to several state-of-the-art approaches on various benchmark and real-life cancer data sets in Section 5.4. Concluding remarks are provided in Section 5.5.

## 5.2 Discriminative Deep Canonical Correlation Analysis

In this section, a novel deep learning model, termed as discriminative deep CCA (D2CCA), is presented for multi-view data analysis. It judiciously integrates the theory of CCA and the merits of the proposed multimodal discriminative DBM (MDDBM), discussed in Chapter 4, to classify the given observations into different sample categories.

Let us assume that the proposed model has $M$ input views or modalities, where $\mathbf{v}^m = \{v_1^m, \cdots, v_i^m, \cdots\}$ represents the input view corresponding to the $m$-th modality and $\mathbf{y} = \{y_1, \cdots, y_p, \cdots\}$ 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 $\mathbf{h}^{lm} = \{h_1^{lm}, \cdots, h_j^{lm}, \cdots\}$, whereas the joint hidden representation, corresponding to the $l$-th layer, is referred to as $\mathbf{h}^l = \{h_1^l, \cdots, h_j^l, \cdots\}$. 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 $V^m$.

### 5.2.1 Objective Function of Proposed D2CCA Model

In MDDBM, the joint representation is learned from the individual modality-specific representations and class label information. However, it may so happen that the individual views correspond to entirely distinct 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 individual spaces may not be able to capture the cross-modal information. However, if the model is learned in such a way that the modality-specific subspaces 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 [78] is an effective statistical method in integrating information acquired from different views. It measures 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 $\omega_2 \in \Re^q$, from the two corresponding mean centred 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,

$$(\omega_1, \omega_2) = \underset{\|{X_1}^T\omega_1\|_2 = \|{X_2}^T\omega_2\|_2 = 1}{\arg \max} \{({X_1}^T\omega_1)^T({X_2}^T\omega_2)\}. \tag{5.1}$$

The objective of CCA is incorporated into the energy function $E_{d2cca}$ of the proposed

MDDBM model, which turns out to be

$$E_{d2cca}(\mathbf{v},\mathbf{h},\mathbf{y}) = -\sum_{m=1}^{M}\sum_{i=1}^{V^m}\sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m} h_j^{1m} - \sum_{l=1}^{L_0-1}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}}\sum_{k=1}^{H^{(l+1)m}} h_j^{lm} w_{jk}^{(l+1)m} h_k^{(l+1)m}$$
$$- \sum_{m=1}^{M}\sum_{j=1}^{H^{L_0m}}\sum_{k=1}^{H^{(L_0+1)}} h_j^{L_0m} w_{jk}^{(L_0+1)m} h_k^{(L_0+1)} - \sum_{l=(L_0+1)}^{L-1}\sum_{j=1}^{H^l}\sum_{k=1}^{H^{(l+1)}} h_j^l w_{jk}^{(l+1)} h_k^{(l+1)} - \sum_{c=1}^{Y} d_c y_c$$
$$- \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{c=1}^{Y}\sum_{j=1}^{H^{lm}} y_c u_{cj}^{lm} h_j^{lm} - \sum_{l=(L_0+1)}^{L}\sum_{c=1}^{Y}\sum_{j=1}^{H^l} y_c u_{cj}^{l} h_j^{l} - \sum_{m=1}^{M}\sum_{i=1}^{V^m} a_i^m v_i^m - \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}} b_j^{lm} h_j^{lm}$$
$$- \sum_{l=(L_0+1)}^{L}\sum_{j=1}^{H^l} b_j^l h_j^l - \sum_{m=1}^{M-1}\sum_{r=(m+1)}^{M}\sum_{j=1}^{H^{L_0m}} h_j^{L_0m} h_j^{L_0r} - \sum_{m=1}^{M}\lambda_m\left(1 - \sum_{j=1}^{H^{L_0m}} (h_j^{L_0m})^2\right). \quad (5.2)$$

Here, $\lambda_m$ is the Lagrange multiplier. It is assumed that $H^{L_0m}$ is same for all $m \in \{1,2,\cdots,M\}$. The criterion presented in (5.2) is termed as the sum of correlations, which is used to integrate more than two sets of multidimensional variables.

The architecture of the proposed D2CCA model follows the same as depicted in Figure 4.1 of Chapter 4. From the figure, it can be observed that 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 $l$-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 $u_{cj}^{lm}$ and $u_{cj}^{l}$ connect $c$-th class node to $j$-th hidden node of $l$-th modality-specific hidden layer from $m$-th modality, and $c$-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 $d_c$ 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 $c$-th class node, respectively.

Considering the energy function $E_{d2cca}(\mathbf{v},\mathbf{h},\mathbf{y})$ of (5.2), the parameter space $\boldsymbol{\theta}_{d2cca}$ of the model is defined by

$$\boldsymbol{\theta}_{d2cca} = \{w^{1m},\cdots,w^{(L_0+1)m},w^{(L_0+2)},\cdots,w^{L},u^{1m},\cdots,u^{L_0m},u^{(L_0+1)},\cdots,u^{L},a^m,b^{1m},$$
$$\cdots,b^{L_0m},b^{(L_0+1)},\cdots,b^{L},d,\lambda_m\},\quad \forall m \in \{1,2,\cdots,M\}.$$

Thus, the concept of CCA is incorporated into the learning objective of the proposed D2CCA architecture by including only $M$ number of $\lambda_m$ 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.

The learning of D2CCA model corresponds to estimating the parameter set $\boldsymbol{\theta}_{d2cca}$ that maximizes the probability of observing the given input data. Considering the input view

$(\mathbf{v}, \mathbf{y})$, the objective function of the D2CCA is given by the log-likelihood function, which is as follows:

$$\ln \mathrm{L}(\boldsymbol{\theta}_{d2cca}|\mathbf{v}, \mathbf{y}) = \ln P(\mathbf{v}, \mathbf{y}|\boldsymbol{\theta}_{d2cca}) = \ln \sum_{\mathbf{h}} e^{-E_{d2cca}(\mathbf{v},\mathbf{h},\mathbf{y})} - \ln \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} e^{-E_{d2cca}(\mathbf{v},\mathbf{h},\mathbf{y})}; \quad (5.3)$$

where $\mathbf{h}$ denotes the stack of hidden layers, $P(\mathbf{v}, \mathbf{y}|\boldsymbol{\theta}_{d2cca})$ represents the probability assigned to the observation $(\mathbf{v}, \mathbf{y})$ by the model parameter set $\boldsymbol{\theta}_{d2cca}$, and $E(\mathbf{v}, \mathbf{h}, \mathbf{y})$ signifies the energy of the joint configuration $\{\mathbf{v}, \mathbf{h}, \mathbf{y}\}$. The corresponding partition function can be defined as $Z = \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} e^{-E_{d2cca}(\mathbf{v},\mathbf{h},\mathbf{y})}$. Since the parameter space of the D2CCA model is quite large, the gradient ascent on the log-likelihood is commonly used to determine the optimal parameters of the model. So, the update rule for the parameters of the D2CCA model is given by

$$\frac{\partial \ln \mathrm{L}(\boldsymbol{\theta}_{d2cca}|\mathbf{v}, \mathbf{y})}{\partial \boldsymbol{\theta}_{d2cca}} = -\sum_{\mathbf{h}} P(\mathbf{h}|\mathbf{v}, \mathbf{y}) \frac{\partial E_{d2cca}(\mathbf{v}, \mathbf{h}, \mathbf{y})}{\partial \boldsymbol{\theta}_{d2cca}} + \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} P(\mathbf{v}, \mathbf{h}, \mathbf{y}) \frac{\partial E_{d2cca}(\mathbf{v}, \mathbf{h}, \mathbf{y})}{\partial \boldsymbol{\theta}_{d2cca}}. \quad (5.4)$$

Thus, the gradient of the log-likelihood function turns out to be the difference between the expectation of gradient of energy function under model distribution, referred to as data-independent expectation, and under the conditional distribution of hidden representation given the input views, termed as data-dependent expectation. Hence, in order to learn the parameters of D2CCA model, the corresponding data-dependent and data-independent expectations are required to be estimated, which are described subsequently.

### 5.2.2 Estimation of Data-Dependent Expectations

Now, the exact maximum likelihood learning is intractable, so the variational learning [142] is employed to estimate the data-dependent expectation. In variational inference, 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,

$$\ln P(\mathbf{v}, \mathbf{y}) = \ln \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v}, \mathbf{y}) \frac{P(\mathbf{v}, \mathbf{h}, \mathbf{y})}{Q(\mathbf{h}|\mathbf{v}, \mathbf{y})} \quad (5.5)$$

where $P(\mathbf{v}, \mathbf{h}, \mathbf{y}) = (1/Z)e^{-E_{d2cca}(\mathbf{v},\mathbf{h},\mathbf{y})}$ represents the probability associated with the joint configuration $\{\mathbf{v}, \mathbf{h}, \mathbf{y}\}$. Since logarithmic is a concave function, applying Jensen's inequality [31], we get

$$\ln P(\mathbf{v}, \mathbf{y}) \geqslant \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v}, \mathbf{y}) \ln \frac{P(\mathbf{v}, \mathbf{h}, \mathbf{y})}{Q(\mathbf{h}|\mathbf{v}, \mathbf{y})} = \mathcal{L}_v. \quad (5.6)$$

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

mean field theory, is given by

$$\begin{aligned}
&\ln P(\mathbf{v},\mathbf{y}) - \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \left\{ \ln P(\mathbf{v},\mathbf{y}) + \ln \frac{P(\mathbf{h}|\mathbf{v},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})} \right\} \\
&= \ln P(\mathbf{v},\mathbf{y}) - \ln P(\mathbf{v},\mathbf{y}) + \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \ln \frac{Q(\mathbf{h}|\mathbf{v},\mathbf{y})}{P(\mathbf{h}|\mathbf{v},\mathbf{y})} \\
&= KL(Q(\mathbf{h}|\mathbf{v},\mathbf{y})||P(\mathbf{h}|\mathbf{v},\mathbf{y})),
\end{aligned} \tag{5.7}$$

where $KL(Q(\mathbf{h}|\mathbf{v},\mathbf{y})||P(\mathbf{h}|\mathbf{v},\mathbf{y}))$ 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 $\ln P(\mathbf{v},\mathbf{y})$. Here, let the mean field distribution be defined as

$$Q(\mathbf{h}|\mathbf{v},\mathbf{y}) = \prod_{l=1}^{L_0} \prod_{m=1}^{M} \prod_{j=1}^{H^{lm}} q(h_j^{lm}|\mathbf{v},\mathbf{y}) \prod_{l=(L_0+1)}^{L} \prod_{j=1}^{H^l} q(h_j^l|\mathbf{v},\mathbf{y}); \tag{5.8}$$

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 $\mu_j$ denotes the probability of being the state of $h_j$ as 1. The definitions of $Q(\mathbf{h}|\mathbf{v},\mathbf{y})$, presented in (5.8), and $P(\mathbf{v},\mathbf{h},\mathbf{y})$, corresponding to the energy function obtained in (5.2), are substituted in (5.6) to obtain the final expression of $\mathcal{L}_v$, which is reported in (5.9).

$$\begin{aligned}
\mathcal{L}_v &= \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y})\{-E_{d2cca}(\mathbf{v},\mathbf{h},\mathbf{y}) - \ln Z\} - \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \ln Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \\
&= \sum_{m=1}^{M} \sum_{i=1}^{V^m} \sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m} \mu_j^{1m} + \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{c=1}^{Y} \sum_{j=1}^{H^{lm}} y_c u_{cj}^{lm} \mu_j^{lm} + \sum_{l=1}^{L_0-1} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} \sum_{k=1}^{H^{(l+1)m}} \mu_j^{lm} w_{jk}^{(l+1)m} \mu_k^{(l+1)m} \\
&+ \sum_{m=1}^{M} \sum_{j=1}^{H^{L_0 m}} \sum_{k=1}^{H^{(L_0+1)}} \mu_j^{L_0 m} w_{jk}^{(L_0+1)m} \mu_k^{(L_0+1)} + \sum_{l=(L_0+1)}^{L-1} \sum_{j=1}^{H^l} \sum_{k=1}^{H^{(l+1)}} \mu_j^l w_{jk}^{(l+1)} \mu_k^{(l+1)} + \sum_{m=1}^{M} \sum_{i=1}^{V^m} a_i^m v_i^m \\
&+ \sum_{l=(L_0+1)}^{L} \sum_{c=1}^{Y} \sum_{j=1}^{H^l} y_c u_{cj}^l \mu_j^l + \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} b_j^{lm} \mu_j^{lm} + \sum_{l=(L_0+1)}^{L} \sum_{j=1}^{H^l} b_j^l \mu_j^l + \sum_{c=1}^{Y} d_c y_c - \ln Z \\
&+ \sum_{m=1}^{M-1} \sum_{r=(m+1)}^{M} \sum_{j=1}^{H^{L_0 m}} \mu_j^{L_0 m} \mu_j^{L_0 r} - \sum_{l=1}^{L_0} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} \left\{ \mu_j^{lm} \ln \mu_j^{lm} + (1-\mu_j^{lm}) \ln(1-\mu_j^{lm}) \right\} \\
&+ \sum_{m=1}^{M} \lambda_m \left( 1 - \sum_{j=1}^{H^{L_0 m}} \mu_j^{L_0 m} \right) - \sum_{l=(L_0+1)}^{L} \sum_{j=1}^{H^l} \left\{ \mu_j^l \ln \mu_j^l + (1-\mu_j^l) \ln(1-\mu_j^l) \right\}.
\end{aligned} \tag{5.9}$$

Since, the mean field parameters ($\boldsymbol{\mu}$) of $\mathcal{L}_v$, presented in (5.9), define the equilibrium state of the model, they need to be updated accordingly. In order to obtain the mean field parameters of the proposed model, the variational bound $\mathcal{L}_v$ of (5.9) is maximized with respect to $\boldsymbol{\mu}$ for a fixed parameter set $\boldsymbol{\theta}_{d2cca}$. 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

$$\mu_j^{lm} = \sigma\Bigg( \sum_{k=1}^{H^{(l-1)m}} \mu_k^{(l-1)m} w_{kj}^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m} \mu_k^{(l+1)m} + \sum_{c=1}^{Y} y_c u_{cj}^{lm} + b_j^{lm} \Bigg), \quad \text{for } 1 \leqslant l < L_0 \text{ and } \forall j, m; \tag{5.10}$$

where $\sigma(x) = \dfrac{1}{1+e^{-x}}$ is the sigmoid function. Similarly, the following update rules can be obtained:

$$\mu_j^{L_0 m} = \sigma\Bigg( \sum_{k=1}^{H^{(L_0-1)m}} \mu_k^{(L_0-1)m} w_{kj}^{L_0 m} + \sum_{c=1}^{Y} y_c u_{cj}^{L_0 m} + b_j^{L_0 m} + \sum_{k=1}^{H^{(L_0+1)}} w_{jk}^{(L_0+1)m} \mu_k^{(L_0+1)} + \sum_{r \neq m=1}^{M} \mu_j^{L_0 r} - \lambda_m \Bigg),\ \forall j, m; \tag{5.11}$$

$$\mu_j^{l} = \sigma\Bigg( \sum_{k=1}^{H^{(l-1)}} \mu_k^{(l-1)} w_{kj}^{l} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^{(l+1)} \mu_k^{(l+1)} + \sum_{c=1}^{Y} y_c u_{cj}^{l} + b_j^{l} \Bigg),\ \text{for } L_0 < l \leqslant L,\ \forall j. \tag{5.12}$$

Thus, given the training data along with the corresponding class label $\{\mathbf{v}^m, \mathbf{y}\}$, the equilibrium state of the model is estimated using the concept of mean field theory. Now, based on the obtained mean field parameters, the parameter set $\boldsymbol{\theta}_{d2cca}$ of the proposed D2CCA architecture, corresponding to the data-dependent expectations, can be learned by maximizing the variational bound $\mathcal{L}_v$ with respect to $\boldsymbol{\theta}_{d2cca}$ for the equilibrium mean field parameters $\boldsymbol{\mu}$. The expressions for differentiation of $\mathcal{L}_v$ with respect to each of the model parameters are given by

$$\begin{aligned}
&\frac{\partial \mathcal{L}_v}{\partial w_{ij}^{1m}} = v_i^m \mu_j^{1m};\ \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{lm}} = \mu_j^{(l-1)m} \mu_k^{lm},\ \text{for } 1 < l \leqslant L_0;\ \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{(L_0+1)m}} = \mu_j^{L_0 m} \mu_k^{(L_0+1)};\\
&\frac{\partial \mathcal{L}_v}{\partial u_{cj}^{lm}} = y_c \mu_j^{lm},\ \text{for } 1 \leqslant l \leqslant L_0;\ \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{l}} = \mu_j^{(l-1)} \mu_k^{l},\ \text{for } (L_0+1) < l \leqslant L;\ \frac{\partial \mathcal{L}_v}{\partial a_i^m} = v_i^m;\\
&\frac{\partial \mathcal{L}_v}{\partial u_{cj}^{l}} = y_c \mu_j^{l},\ \text{for } L_0 < l \leqslant L;\ \frac{\partial \mathcal{L}_v}{\partial b_j^{lm}} = \mu_j^{lm},\ \text{for } 1 \leqslant l \leqslant L_0;\ \frac{\partial \mathcal{L}_v}{\partial b_j^{l}} = \mu_j^{l},\ \text{for } L_0 < l \leqslant L;\\
&\frac{\partial \mathcal{L}_v}{\partial d_c} = y_c;\ \text{and } \frac{\partial \mathcal{L}_v}{\partial \lambda_m} = \Bigg(1 - \sum_{j=1}^{H^{L_0 m}} \mu_j^{L_0 m}\Bigg);\ \forall m \in \{1, 2, \cdots, M\}.
\end{aligned} \tag{5.13}$$

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

### 5.2.3 Estimation of Data-Independent Expectations

Now, the second term of the gradient of log-likelihood function (5.4), that is, energy gradient with respect to the model distribution, is estimated using the Markov Chain Monte Carlo based stochastic approximation procedure [190]. 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 $\boldsymbol{\theta}_{d2cca}$. 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 corresponding to the proposed D2CCA model is given by

$$P(h_j^{1m}|\mathbf{v}^m, \mathbf{y}, \mathbf{h}^{2m}) = \sigma\left(\sum_{i=1}^{V^m} v_i^m w_{ij}^{1m} + \sum_{c=1}^{Y} y_c u_{cj}^{1m} + \sum_{k=1}^{H^{2m}} w_{jk}^{2m} h_k^{2m} + b_j^{1m}\right),\ \forall j, m; \tag{5.14}$$

$$P(h_j^{lm}|\mathbf{h}^{(l-1)m}, \mathbf{h}^{(l+1)m}, \mathbf{y}) = \sigma\Bigg(\sum_{k=1}^{H^{(l-1)m}} h_k^{(l-1)m} w_{kj}^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m} h_k^{(l+1)m} + \sum_{c=1}^{Y} y_c u_{cj}^{lm} + b_j^{lm}\Bigg),\ \text{for } 1 < l < L_0,\ \forall j, m; \tag{5.15}$$

$$P(h_j^{L_0 m}|\mathbf{h}^{(L_0-1)m}, \mathbf{h}^{(L_0+1)}, \mathbf{h}^{L_0 r}, \mathbf{y}) = \sigma\Bigg(\sum_{k=1}^{H^{(L_0-1)m}} h_k^{(L_0-1)m} w_{kj}^{L_0 m} + \sum_{c=1}^{Y} y_c u_{cj}^{L_0 m} + b_j^{L_0 m} + \sum_{k=1}^{H^{(L_0+1)}} w_{jk}^{(L_0+1)m} h_k^{(L_0+1)} + \sum_{r\neq m=1}^{M} h_j^{L_0 r} - \lambda_m\Bigg),\ \forall j, m; \tag{5.16}$$

$$P(h_j^{l}|\mathbf{h}^{(l-1)}, \mathbf{h}^{(l+1)}, \mathbf{y}) = \sigma\Bigg(\sum_{k=1}^{H^{(l-1)}} h_k^{(l-1)} w_{kj}^{l} + \sum_{c=1}^{Y} y_c u_{cj}^{l} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^{(l+1)} h_k^{(l+1)} + b_j^{l}\Bigg),\ \text{for } L_0 < l \leqslant L,\ \forall j; \tag{5.17}$$

$$P(v_i^m|\mathbf{h}^{1m}) = \sigma\left(\sum_{j=1}^{H^{1m}} w_{ij}^{1m} h_j^{1m} + a_i^m\right),\ \forall m; \tag{5.18}$$

$$P(y_c|\mathbf{h}^{11}, \cdots, \mathbf{h}^{L_0 M}, \mathbf{h}^{L_0+1}, \cdots, \mathbf{h}^{L}) = \frac{e^{X_c}}{\sum_{\bar{c}=1}^{Y} e^{X_{\bar{c}}}}; \tag{5.19}$$

$$\text{where } X_c = \sum_{l=1}^{L_0}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}} u_{cj}^{lm} h_j^{lm} + \sum_{l=(L_0+1)}^{L}\sum_{j=1}^{H^{l}} u_{cj}^{l} h_j^{l} + d_c. \tag{5.20}$$

Given that the convergence criteria are satisfied, which are to be discussed in Section 5.3.1, if a Markov chain is run for sufficient number of steps, then it can be ensured that the chain will converge to an 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 (5.14) - (5.19). 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 superscript tilde (e.g., $\tilde{v}$);

$$\frac{\partial E_{d2cca}}{\partial w_{ij}^{1m}} = \tilde{v}_i^m \tilde{h}_j^{1m};\ \frac{\partial E_{d2cca}}{\partial w_{jk}^{lm}} = \tilde{h}_j^{(l-1)m} \tilde{h}_k^{lm},\ \text{for } 1 < l \leqslant L_0;\ \frac{\partial E_{d2cca}}{\partial w_{jk}^{(L_0+1)m}} = \tilde{h}_j^{L_0 m} \tilde{h}_k^{(L_0+1)};$$
$$\frac{\partial E_{d2cca}}{\partial u_{cj}^{lm}} = \tilde{y}_c \tilde{h}_j^{lm},\ \text{for } 1 \leqslant l \leqslant L_0;\ \frac{\partial E_{d2cca}}{\partial w_{jk}^{l}} = \tilde{h}_j^{(l-1)} \tilde{h}_k^{l},\ \text{for } (L_0+1) < l \leqslant L;$$
$$\frac{\partial E_{d2cca}}{\partial u_{cj}^{l}} = \tilde{y}_c \tilde{h}_j^{l},\ \text{for } L_0 < l \leqslant L;\ \frac{\partial E_{d2cca}}{\partial b_{j}^{lm}} = \tilde{h}_j^{lm},\ \text{for } 1 \leqslant l \leqslant L_0;\ \frac{\partial E_{d2cca}}{\partial a_{i}^{m}} = \tilde{v}_i^{m};$$
$$\frac{\partial E_{d2cca}}{\partial b_{j}^{l}} = \tilde{h}_j^{l},\ \text{for } L_0 < l \leqslant L;\ \frac{\partial E_{d2cca}}{\partial d_{c}} = \tilde{y}_c;$$
$$\text{and } \frac{\partial E_{d2cca}}{\partial \lambda_m} = \left(1 - \sum_{j=1}^{H^{L_0 m}} (\tilde{h}_j^{L_0 m})^2\right);\ \forall m \in \{1, 2, \cdots, M\}. \tag{5.21}$$

Hence, the proposed model can be efficiently learned from data-dependent and data-independent estimates, obtained in (5.13) and (5.21), respectively.

### 5.2.4 Learning Rule of D2CCA Model Parameters

Let $N$, $S$, $t$, and $\eta$ be the number of training samples, number of persistent Markov chains, current epoch, and learning rate, respectively. Thus, the update rule for the parameters of the proposed D2CCA architecture, required to perform gradient ascent on the log-likelihood function corresponding to the energy function of (5.2), is as follows:

$$\boldsymbol{\theta}_{d2cca}^{(t+1)} = \boldsymbol{\theta}_{d2cca}^{t} + \Delta\boldsymbol{\theta}_{d2cca}^{t}; \tag{5.22}$$
$$\text{where } \Delta\boldsymbol{\theta}_{d2cca}^{t} = \eta\left\{\frac{1}{N}\sum_{n=1}^{N}\left(\frac{\partial \mathcal{L}_v}{\partial \boldsymbol{\theta}_{d2cca}}\right)_n - \frac{1}{S}\sum_{s=1}^{S}\left(\frac{\partial E_{d2cca}}{\partial \boldsymbol{\theta}_{d2cca}}\right)_s\right\} - \rho\boldsymbol{\theta}_{d2cca}^{t} + \zeta\Delta\boldsymbol{\theta}_{d2cca}^{(t-1)}. \tag{5.23}$$

From (5.22) and (5.23), 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 standard Boltzmann machine [136]. 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 [8]. It can be observed that all the parameters of the model are learned simultaneously using (5.22) which is considered to be beneficial in case of learning joint subspace from heterogeneous data. Also, subtracting the data-independent expectations from the corresponding data-dependent terms in (5.22) basically stabilizes the distribution of parameters of the proposed model in order to propagate uncertainties associated with ambiguous inputs. The learning algorithm of the proposed D2CCA model is illustrated in Algorithm 5.1.

**Algorithm 5.1** Learning of D2CCA Model.

**Input:** Set of $N$ training data vectors $\{\mathbf{v}^m\}_{n=1}^N$ for $M$ modalities along with the corresponding set of class labels $\{\mathbf{y}\}_{n=1}^N$, number of persistent chains ($S$), number of epochs ($\tau$), learning rate ($\eta$), weight decay ($\rho$), momentum ($\zeta$), and number of Gibbs steps ($\alpha$).
**Output:** Final parameter set $\boldsymbol{\theta}_{d2cca}^{\tau}$ of the architecture.

Perform greedy layer-wise pretraining to initialize the set of model parameters, $\boldsymbol{\theta}_{d2cca}^{0}$.
Randomly initialize $S$ Markov chains $\{\tilde{\mathbf{v}}^{m^0}, \tilde{\mathbf{y}}^0, \tilde{\mathbf{h}}^{1m^0}, \cdots, \tilde{\mathbf{h}}^{L_0 m^0}, \tilde{\mathbf{h}}^{(L_0+1)^0}, \cdots, \tilde{\mathbf{h}}^{L^0}\}_{s=1}^{S}$.
**for** each epoch $t = 0$ to $\tau$ **do**
    // Variational inference
    **for** each training sample $n = 1$ to $N$ **do**
        (i) Run mean field updates using (5.10)-(5.12) until convergence.
        (ii) Save the obtained mean field parameter ($\boldsymbol{\mu}$) for the corresponding training sample, $\boldsymbol{\mu}_n = \boldsymbol{\mu}$.
    **end for**
    // Stochastic approximation
    **for** each persistent chain $s = 1$ to $S$ **do**
        Run the chain for $\alpha$-steps and sample the state $\{\tilde{\mathbf{v}}^{m^{t+1}}, \tilde{\mathbf{y}}^{t+1}, \tilde{\mathbf{h}}^{1m^{t+1}}, \cdots, \tilde{\mathbf{h}}^{L_0 m^{t+1}}, \tilde{\mathbf{h}}^{(L_0+1)^{t+1}}, \cdots, \tilde{\mathbf{h}}^{L^{t+1}}\}$ from $\{\tilde{\mathbf{v}}^{m^t}, \tilde{\mathbf{y}}^t, \tilde{\mathbf{h}}^{1m^t}, \cdots, \tilde{\mathbf{h}}^{L_0 m^t}, \tilde{\mathbf{h}}^{(L_0+1)^t}, \cdots, \tilde{\mathbf{h}}^{L^t}\}$ using (5.14)-(5.19).
    **end for**
    Update the parameters of the model from $\boldsymbol{\theta}_{d2cca}^{t}$ to $\boldsymbol{\theta}_{d2cca}^{(t+1)}$ using (5.22).
**end for**

## 5.3 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.

### 5.3.1 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 (5.6) of the model. Given a particular state, the parameter set $\boldsymbol{\theta}_{d2cca}$ 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 (5.2), is given by

$$\nabla\mathcal{L}_v(\theta_{d2cca}) = \Bigg[\frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial w_{ij}^{1m}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial w_{jk}^{(L_0+1)m}} \; \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial w_{jk}^{L_0+2}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial w_{jk}^{L}} \; \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial u_{cj}^{1m}}$$
$$\cdots \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial u_{cj}^{L_0 m}} \; \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial u_{cj}^{L_0+1}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial u_{cj}^{L}} \; \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial a_i^m} \; \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial b_j^{1m}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2cca})}{\partial b_j^{L_0 m}}$$

$$\frac{\partial \mathcal{L}_v(\theta_{d2cca})}{\partial b_j^{L_0+1}} \cdots \frac{\partial \mathcal{L}_v(\theta_{d2cca})}{\partial b_j^{L}} \quad \frac{\partial \mathcal{L}_v(\theta_{d2cca})}{\partial d_c} \quad \frac{\partial \mathcal{L}_v(\theta_{d2cca})}{\partial \lambda_m} \Bigg]^T ; \ \forall i, j, k, c, m, \tag{5.24}$$

where $T$ denotes the transpose operator. The gradient of $\mathcal{L}_v(\theta_{d2cca})$ with respect to each of the parameters can be obtained using (5.13), which makes it clear that the gradient function $\nabla\mathcal{L}_v(\theta_{d2cca})$ is independent of $\theta_{d2cca}$, that is, $\nabla\mathcal{L}_v(\theta_{d2cca\_1}) = \nabla\mathcal{L}_v(\theta_{d2cca\_2})$, $\forall\theta_{d2cca\_1}$, $\theta_{d2cca\_2} \in \boldsymbol{\theta}_{d2cca}$. The independence of $\nabla\mathcal{L}_v(\theta_{d2cca})$ with respect to the parameters of D2CCA model is evident since each parameter of the model is included in the corresponding energy function $E_{d2cca}(\mathbf{v}, \mathbf{h}, \mathbf{y})$, presented in (5.2) as an additive term. So, it can be said that $\mathcal{L}_v$ is a differential function having $\beta$-Lipschitz continuous gradient for some $\beta \geqslant 0$, that is, $\|\nabla\mathcal{L}_v(\theta_{d2cca\_1}) - \nabla\mathcal{L}_v(\theta_{d2cca\_2})\|_2 \leqslant \beta\|\theta_{d2cca\_1} - \theta_{d2cca\_2}\|_2$. For a function having $\beta$-Lipschitz gradient, it is known that $\forall\theta_{d2cca\_1}, \theta_{d2cca\_2} \in \boldsymbol{\theta}_{d2cca}$,

$$\begin{aligned}\mathcal{L}_v(\theta_{d2cca\_1}) \ \leqslant \mathcal{L}_v(\theta_{d2cca\_2}) &+ \nabla\mathcal{L}_v(\theta_{d2cca\_2})^T(\theta_{d2cca\_1} - \theta_{d2cca\_2}) \\ &+ \frac{1}{2}\beta\|\theta_{d2cca\_1} - \theta_{d2cca\_2}\|_2^2.\end{aligned} \tag{5.25}$$

Let, $\theta_{d2cca\_1} = \theta_{d2cca}^t$ and $\theta_{d2cca\_2} = \theta_{d2cca}^{(t+1)}$, where $t$ denotes the current epoch. Substituting the values of $\theta_{d2cca\_1}$ and $\theta_{d2cca\_2}$, and rearranging the terms in (5.25), we get

$$\mathcal{L}_v(\theta_{d2cca}^{(t+1)}) \geqslant \mathcal{L}_v(\theta_{d2cca}^t) + \nabla\mathcal{L}_v(\theta_{d2cca}^{(t+1)})^T(\theta_{d2cca}^{(t+1)} - \theta_{d2cca}^t) - \frac{1}{2}\beta\|\theta_{d2cca}^{(t+1)} - \theta_{d2cca}^t\|_2^2.$$

Since gradient ascent algorithm on $\mathcal{L}_v$ is employed in the proposed model to learn the parameter values of $\boldsymbol{\theta}_{d2cca}$, it is obvious that $\theta_{d2cca}^{(t+1)} = \theta_{d2cca}^t + \eta\nabla\mathcal{L}_v(\theta_{d2cca}^t)$, where $\eta$ denotes the learning rate. Now, considering the independence property of $\nabla\mathcal{L}_v(\theta_{d2cca})$ in the proposed model, that is, $\nabla\mathcal{L}_v(\theta_{d2cca}^{(t+1)}) = \nabla\mathcal{L}_v(\theta_{d2cca}^t)$, we have

$$\mathcal{L}_v(\theta_{d2cca}^{(t+1)}) \geqslant \mathcal{L}_v(\theta_{d2cca}^t) + \eta(1 - \frac{1}{2}\beta\eta)\|\nabla\mathcal{L}_v(\theta_{d2cca}^t)\|_2^2.$$

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

$$\mathcal{L}_v(\theta_{d2cca}^{(t+1)}) \geqslant \mathcal{L}_v(\theta_{d2cca}^t) + \frac{1}{2}\eta\|\nabla\mathcal{L}_v(\theta_{d2cca}^t)\|_2^2. \tag{5.26}$$

Let $\theta_{d2cca}^*$ be the optimal parameter set that maximizes the lower bound, or equivalently maximizes the log-likelihood function of the proposed model in such a way that, $\mathcal{L}_v(\theta_{d2cca}^*) \geqslant \mathcal{L}_v(\theta_{d2cca}), \forall\theta_{d2cca} \in \boldsymbol{\theta}_{d2cca}$. Now, since $\mathcal{L}_v(\theta_{d2cca})$ is defined as a linear function of $\theta_{d2cca}$ in (5.9), we have

$$\mathcal{L}_v(\theta_{d2cca}^t) = \mathcal{L}_v(\theta_{d2cca}^*) + \nabla\mathcal{L}_v(\theta_{d2cca}^t)^T(\theta_{d2cca}^t - \theta_{d2cca}^*). \tag{5.27}$$

Using (5.27) in (5.26), we get

$$\mathcal{L}_v(\theta^*_{d2cca}) - \mathcal{L}_v(\theta^{(t+1)}_{d2cca}) \leqslant -\nabla\mathcal{L}_v(\theta^t_{d2cca})^T(\theta^t_{d2cca} - \theta^*_{d2cca}) - \frac{1}{2}\eta\|\nabla\mathcal{L}_v(\theta^t_{d2cca})\|_2^2$$

$$\Rightarrow \mathcal{L}_v(\theta^*_{d2cca}) - \mathcal{L}_v(\theta^{(t+1)}_{d2cca}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^t_{d2cca} - \theta^*_{d2cca}\|_2^2 - \|\theta^t_{d2cca} - \theta^*_{d2cca}\|_2^2 - \|\eta\nabla\mathcal{L}_v(\theta^t_{d2cca})\|_2^2 - 2\eta\nabla\mathcal{L}_v(\theta^t_{d2cca})^T(\theta^t_{d2cca} - \theta^*_{d2cca})\Big\}$$

$$\Rightarrow \mathcal{L}_v(\theta^*_{d2cca}) - \mathcal{L}_v(\theta^{(t+1)}_{d2cca}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^t_{d2cca} - \theta^*_{d2cca}\|_2^2 - \|\theta^{t+1}_{d2cca} - \theta^*_{d2cca}\|_2^2\Big\}.$$

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

$$\sum_{t=0}^{T}\Big\{L_v(\theta^*_{d2cca}) - L_v(\theta^{t+1}_{d2cca})\Big\} \leqslant \frac{1}{2\eta}\sum_{t=0}^{T}\Big\{\|\theta^t_{d2cca} - \theta^*_{d2cca}\|_2^2 - \|\theta^{t+1}_{d2cca} - \theta^*_{d2cca}\|_2^2\Big\}$$

$$\Rightarrow \tau\mathcal{L}_v(\theta^*_{d2cca}) - \sum_{t=0}^{\tau-1}\mathcal{L}_v(\theta^{t+1}_{d2cca}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^0_{d2cca} - \theta^*_{d2cca}\|_2^2 - \|\theta^\tau_{d2cca} - \theta^*_{d2cca}\|_2^2\Big\}.$$

Since $\mathcal{L}_v(\theta_{d2cca})$ is an increasing function of $\theta_{d2cca}$, we can replace $\sum_{t=0}^{\tau-1}\mathcal{L}_v(\theta^{(t+1)}_{d2cca})$ with $\tau\mathcal{L}_v(\theta^\tau_{d2cca})$ and the inequality will still hold. Thus, we get

$$\begin{aligned}\mathcal{L}_v(\theta^*_{d2cca}) - \mathcal{L}_v(\theta^\tau_{d2cca}) &\leqslant \frac{1}{2\eta\tau}\Big\{\|\theta^0_{d2cca} - \theta^*_{d2cca}\|_2^2 - \|\theta^\tau_{d2cca} - \theta^*_{d2cca}\|_2^2\Big\} \\ &\leqslant \frac{1}{2\eta\tau}\|\theta^0_{d2cca} - \theta^*_{d2cca}\|_2^2. \end{aligned} \tag{5.28}$$

From (5.28), it can be concluded that the variational learning algorithm in the proposed D2CCA model converges with a rate $\mathcal{O}(\frac{1}{\tau})$ after $\tau$ iterations, if the learning rate is considered to be small enough, that is, $\eta \leqslant \frac{1}{\beta}$.

Now, 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 [218, 220]. One necessary condition requires the learning rate ($\eta$) 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 $\eta_t = \frac{a}{b+t}$, for constants $a > 0$ and $b > 0$. Also, in practice, the sequence $|\theta^t_{d2cca}|$ is bounded and the Markov chain is ergodic which, along with the condition on learning rate, establish the convergence of stochastic approximation procedure. Together with the condition on the variational learning (5.28), this ensures the convergence of the proposed D2CCA model.

### 5.3.2 D2CCA Model 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 apriori.

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 in 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 nodes 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 estimation of data-dependent expectations, described in Section 5.2.2, 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 (5.9), mean field equations from (5.10) to (5.12), and gradient of lower bound in (5.13) only for $c \in \{1, \cdots, Y_0\}$ and will have a zero value for $c \in \{Y_0 + 1, \cdots, Y\}$. 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 rest of the nodes.

However, this is not the case for the estimation of data-independent expectations, presented in Section 5.2.3. For the computation of conditional distribution $P(y_c|\mathbf{h}^{11}, \cdots, \mathbf{h}^{L_0 M}, \mathbf{h}^{L_0+1}, \cdots, \mathbf{h}^{L})$ in (5.19), the value of $X_c$ depends on the weights and biases associated with the node $y_c$, which may result into prediction of class labels corresponding to the invalid class nodes. In order to overcome such situation, the update rule for the weight and bias parameters related to the class nodes, that is, $u_{cj}^{lm}$, $u_{cj}^{l}$, and $d_c$, which is earlier demonstrated in (5.23), can be modified as follows:

$$\Delta u_{cj}^{lm^t} = \begin{cases} -u_{cj}^{lm^t} \;\; \text{if} \;\; \sum_{n=1}^{N} y_c \mu_j^{lm} = 0; \\ \eta \left\{ \frac{1}{N} \sum_{n=1}^{N} (y_c \mu_j^{lm}) - \frac{1}{S} \sum_{s=1}^{S} (\tilde{y}_c \tilde{h}_j^{lm}) \right\} - \rho u_{cj}^{lm^t} + \zeta \Delta u_{cj}^{lm^{(t-1)}} \;\; \text{otherwise;} \end{cases}$$
$$\text{for } 1 \leqslant l \leqslant L_0; \qquad (5.29)$$

$$\Delta u_{cj}^{l^t} = \begin{cases} -u_{cj}^{l^t} \;\; \text{if} \;\; \sum_{n=1}^{N} y_c \mu_j^{l} = 0; \\ \eta \left\{ \frac{1}{N} \sum_{n=1}^{N} (y_c \mu_j^{l}) - \frac{1}{S} \sum_{s=1}^{S} (\tilde{y}_c \tilde{h}_j^{l}) \right\} - \rho u_{cj}^{l^t} + \zeta \Delta u_{cj}^{l^{(t-1)}} \;\; \text{otherwise;} \end{cases}$$
$$\text{for } L_0 < l \leqslant L; \qquad (5.30)$$

$$\Delta d_c^t = \begin{cases} -d_c^t \;\; \text{if} \;\; \sum_{n=1}^{N} y_c = 0; \\ \eta \left\{ \frac{1}{N} \sum_{n=1}^{N} y_c - \frac{1}{S} \sum_{s=1}^{S} \tilde{y}_c \right\} - \rho d_c^t + \zeta \Delta d_c^{(t-1)} \;\; \text{otherwise.} \end{cases} \tag{5.31}$$

Thus, the update rules (5.29)-(5.31), considered with (5.22), 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. Thus, the class labels of the unknown samples can be predicted from the set of valid class labels when the total number of input classes is unknown apriori.

## 5.4 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 [187], MCCA [96], GMCCA [25], GMKCCA [25], LasCCA [53], DisCCA [53], MvDA [92], MvDA-VC [93], MDBM [180], dMCCA [178], TOCCA [30], DACCA [44], DCCA-VG [193], TDD-CCA [37], MDL-CW [157], MMGNN [54], and MVGAN [209]. Both training-testing and 10-fold cross-validation (CV) are performed to evaluate the performance of the proposed model as well as existing algorithms. In case of training-testing, overall classification accuracy is considered, while for 10-fold 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.

### 5.4.1 Description of Data Sets

In this study, six benchmark databases, namely, Digits [86], Caltech [114], CiteSeer [161], Cora [161], NUS-WIDE-OBJECT (NW-OBJECT) [27], and Reuters [7], and three cancer data sets are considered to evaluate the performance of different algorithms. Digits, Caltech, and NW-OBJECT are image based databases, CiteSeer and Cora consist of scientific publications with annotated labels, whereas Reuters is a multilingual categorization data set. Different subtypes are identified for three real-life cancer data sets, which include cervical carcinoma (CESC), lower grade glioma (LGG), and lung carcinoma (LUNG). These data sets are obtained from The Cancer Genome Atlas (TCGA) [194].

It is to be noted here that while Digits and Caltech data sets exhibit large variance in the dimensionality of the input feature sets, CiteSeer and Cora databases have large number of features in the corresponding feature sets. The NW-OBJECT data set has large number of samples with small number of features in each view, whereas Reuters database has large number of samples with large dimension of each of the feature sets. On the other hand, the omics data sets offer the problem of high dimensional feature sets with small number of samples. A brief description of the CiteSeer, Cora, NW-OBJECT, Reuters, CESC, and

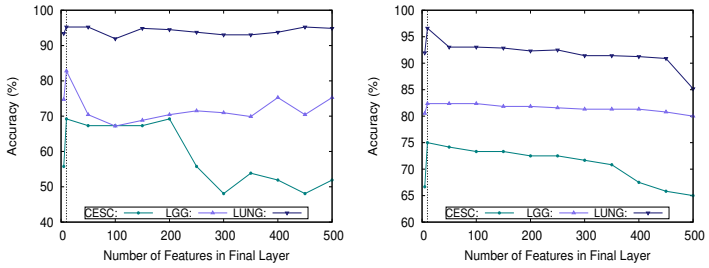
Table 5.1: Description of Data Sets

| 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 |
| Omics | LUNG | 546 | 2 | 5 | 294668 | 180 | 216 | 20502 | 49230 | - |

LGG data sets is presented in Chapter 4. In addition to the aforementioned data sets, the Digits, Caltech, and LUNG databases are also considered in this chapter for performance analysis of different algorithms. The number of samples, number of classes, number of views, and number of features in each view corresponding to these three databases, are presented in Table 5.1. Each data set is randomly partitioned into two sets for training-testing and ten separate folds for 10-fold CV. In both the cases, the samples are equally distributed with respect to different classes. Detailed description of all the data sets is reported in Appendix A.

### 5.4.2 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, extensive experiment is carried out on three 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 10-fold CV. The corresponding results are reported in Figure 5.1 considering three cancer data sets. 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 classification accuracy, achieved by the model, remains almost constant or even decreases in some cases. However, if the number of nodes in joint layer is fixed at 10, considerable classification accuracy can be achieved irrespective of the experimental set-up and data sets considered. Thus, the number of nodes in joint representation is considered to be 10 for the rest of the study.




(a) Training-Testing (b) 10-fold CV

Figure 5.1: Variation of classification accuracy with respect to number of features in final layer for omics data sets.

The greedy layer-wise pretraining [163] 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 learning rate is initialized at 0.01 and then, gradually decreased with increase in number of epochs. For the estimation of data-independent expectations, 100 Gibbs steps and 20 separate Markov chains are considered. For the classification of samples into one of the known classes, maximum class probability is taken into consideration corresponding to the class nodes of the model.

### 5.4.3 Choice of Deep Model

In existing literature, several deep models have been considered to learn the joint subspace 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 generative adversarial network has been considered for the implementation of MVGAN. However, the proposed D2CCA architecture has been developed based on the framework of 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 Table 5.2 and Table 5.3 for benchmark and omics databases, respectively.

Table 5.2: Effectiveness of D2CCA 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** |

From the results presented in Table 5.2, 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 data sets. However, highest classification accuracy is attained by the proposed model on all the six databases. In case of omics data sets, corresponding to the results reported in Table 5.3, significant improvement in performance can be noted for the proposed D2CCA model as compared to different deep architectures for both training-testing and 10-fold CV. Statistical significance analysis demonstrates that the proposed model achieves significantly better p-values for 20 cases and better but not significant p-values for the rest of the 4 cases.

### 5.4.4 Effectiveness of Proposed D2CCA Architecture

In this section, the effectiveness of different aspects of the proposed D2CCA architecture is demonstrated, which includes significance of integrating CCA and effectiveness of proposed

Table 5.3: Effectiveness of D2CCA Over Different Deep Models on Omics Data Sets

| 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 | - |
| 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 | - |

method as feature extractor. The corresponding results are reported in Figure 5.2, and Table 5.4 and Table 5.5. The scatter plots of Figure 5.2 are depicted by considering the most relevant feature at $x$-axis and the corresponding most significant feature at $y$-axis, obtained using the concept of rough hypercuboid approach [126]. While the top row of Figure 5.2 corresponds to MDDBM model, the plots of last rows are obtained from D2CCA architecture, respectively.

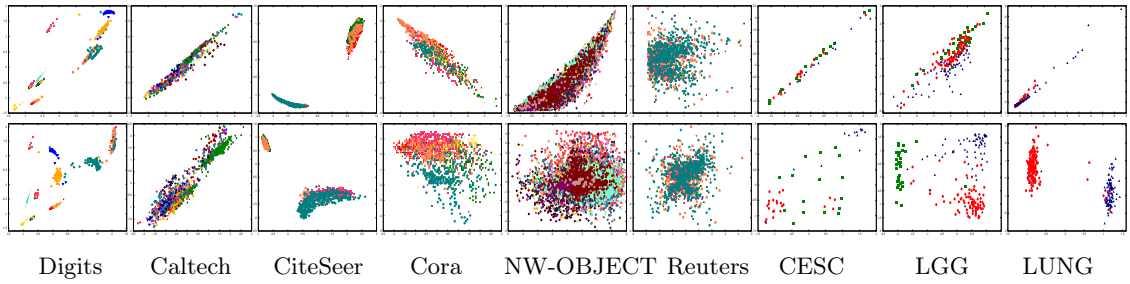



Figure 5.2: Scatter plots of MDDBM (top row) and D2CCA (bottom row) for benchmark and omics data sets.

#### 5.4.4.1 Significance of Integrating CCA

In this chapter, a novel deep learning framework is proposed for multimodal data classification. The D2CCA model is developed by integrating the theory of CCA with MD-

Table 5.4: Effectiveness of Proposed Architecture on Benchmark Data

| Data | MDDBM | D2CCA+SVM | D2CCA+Bayes | D2CCA |
|---|---|---|---|---|
| Digits | 85.60 | 87.50 | 85.90 | **91.00** |
| Caltech | 74.14 | 83.90 | 71.74 | **84.54** |
| CiteSeer | 71.93 | 63.85 | **94.10** | 73.12 |
| Cora | 64.37 | 63.49 | 66.26 | **80.47** |
| NW-OBJECT | 43.75 | 33.56 | 34.16 | **52.21** |
| Reuters | 61.25 | 84.64 | **85.47** | 84.60 |

DBM 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 MDDBM architecture, the performance of D2CCA is compared with that of MDDBM on both benchmark and omics data sets. Considering the scatter plots of Figure 5.2, corresponding to the benchmark databases, it can be observed that the separation between the samples of different classes is improved in case of D2CCA as compared to MDDBM. This result is also reflected in Table 5.4, where the classification accuracy has improved for D2CCA in comparison to MDDBM on the benchmark data for training-testing.

The scatter plots of Figure 5.2, corresponding to each of the omics data sets, reveal that all the given classes are well separated for D2CCA. In case of MDDBM, although the samples from different classes can be distinguished for LUNG data, they tend to overlap for CESC and LGG data sets. This observation can also be validated from the results reported in Table 5.5, where it can be observed that MDDBM performs better on LUNG data, in comparison to rest of the omics data sets. However, the highest classification accuracy is achieved by D2CCA architecture on all the data sets, for both training-testing and 10-fold CV. Statistical significance analysis demonstrates that the D2CCA model attains significantly better p-values for all the cases.

#### 5.4.4.2 Effectiveness of D2CCA as Feature Extractor

Apart from multimodal data classification, the proposed D2CCA architecture can be considered as 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 5.4, it can be 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 Bayes classifier as compared to the D2CCA architecture for CiteSeer and Reuters data sets. The results presented in Table 5.5 state that the samples from CESC data set can be efficiently recognized by the SVM for training-testing, while the Bayes classifier can suitably identify different subtypes of LGG data for training-testing and CESC data for 10-fold CV. However, considerable classification accuracy is achieved on all the omics data sets for both training-testing and 10-fold CV using the proposed architecture itself. Statistical significance analysis reveals that out of total 12 cases, the proposed D2CCA

Table 5.5: Effectiveness of Proposed Architecture on Omics Data Sets

| Data | Different Metrics | | MDDBM | D2CCA +SVM | D2CCA +Bayes | D2CCA |
|---|---|---|---|---|---|---|
| CESC | Train-Test | | 67.31 | **75.00** | 69.23 | 69.23 |
| | 10-fold CV | Mean | 64.17 | 54.17 | **80.00** | 75.00 |
| | | Median | 66.67 | 58.33 | **79.17** | 75.00 |
| | | StdDev | **5.62** | 18.94 | 8.05 | 6.80 |
| | | Paired-*t*:p | 3.13E-03 | 8.68E-03 | 9.16E-01 | - |
| | | Wilcoxon:p | 7.31E-03 | 1.89E-02 | 8.58E-01 | - |
| LGG | Train-Test | | 79.57 | 67.74 | **94.62** | 82.80 |
| | 10-fold CV | Mean | 63.95 | 18.42 | 69.74 | **82.37** |
| | | Median | 63.16 | 18.42 | 69.74 | **82.89** |
| | | StdDev | 6.08 | 0.00 | 6.71 | **5.41** |
| | | Paired-*t*:p | 2.42E-05 | 1.75E-11 | 1.62E-04 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.49E-03 | 2.53E-03 | - |
| LUNG | Train-Test | | 94.51 | 92.67 | 90.48 | **95.24** |
| | 10-fold CV | Mean | 90.18 | 67.68 | 91.61 | **96.61** |
| | | Median | 91.96 | 62.50 | 92.86 | **97.32** |
| | | StdDev | 7.49 | 26.45 | 5.19 | **3.41** |
| | | Paired-*t*:p | 5.00E-03 | 2.35E-03 | 1.01E-04 | - |
| | | Wilcoxon:p | 3.98E-03 | 2.52E-03 | 2.52E-03 | - |

framework attains significantly better p-values in 10 cases.

### 5.4.5 Comparative Performance Analysis

Finally, the classification performance of the proposed D2CCA architecture is compared with that of several existing methods on benchmark as well as omics data sets, and the corresponding results are reported in Table 5.6, Table 5.7, Table 5.8, and Table 5.9. The scatter plots of existing algorithms and the proposed model are presented in Figure 5.3 and Figure 5.4. 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 for classification purpose. The existing algorithms include multiset CCA based methods, multi-view discriminative analysis based methods, and multi-view deep learning based methods.

#### 5.4.5.1 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 [187], MCCA [96], GMCCA [25], GMKCCA [25], LasCCA [53], and DisCCA [53]. The scatter plots corresponding to the MCCA, GMCCA, GMKCCA, LasCCA, and DisCCA approaches are presented in Figure 4.3 of Chapter 4, whereas the scatter plots for RGCCA method are depicted in Figure 5.3 on both benchmark and omics databases. All the existing algorithms extract 25

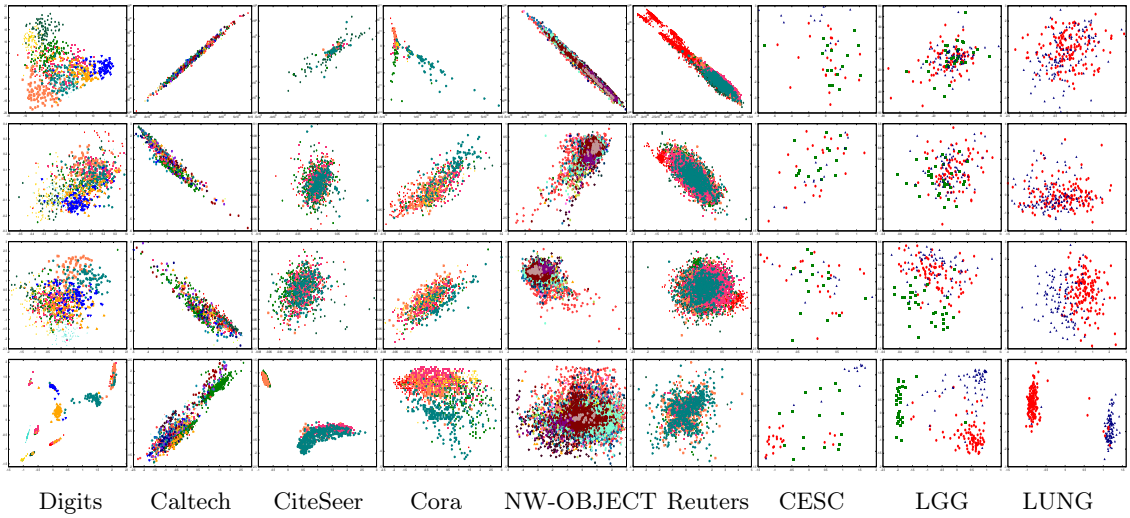



Figure 5.3: Scatter plots of classical approaches on benchmark and omics data sets (from top to bottom row: RGCCA, MvDA, MvDA-VC, and D2CCA, respectively).

Table 5.6: Comparative Performance Analysis of Classical Approaches on Benchmark Data

| Data | RGCCA | MCCA | GMCCA | GMKCCA | LasCCA | DisCCA | MvDA | MvDA-VC | D2CCA |
|---|---|---|---|---|---|---|---|---|---|
| Digits | 90.30 | 87.00 | 11.20 | 6.60 | 10.20 | 5.60 | 92.40 | **93.50** | 91.00 |
| Caltech | 33.71 | 41.83 | 4.82 | 7.48 | 4.06 | 3.17 | 76.30 | 75.29 | **84.54** |
| CiteSeer | 27.61 | 58.13 | 23.43 | 24.98 | 22.25 | 20.71 | 37.69 | 43.51 | **73.12** |
| Cora | 52.16 | 32.85 | 30.97 | 30.19 | 31.63 | 30.19 | 53.94 | 55.72 | **80.47** |
| NW-OBJECT | 18.91 | 30.34 | 4.56 | 6.43 | 7.40 | 10.93 | 29.03 | 28.62 | **52.21** |
| Reuters | 55.26 | 57.50 | 24.78 | 28.69 | 28.67 | 23.27 | 56.01 | 55.15 | **84.60** |

features from the given input views to represent the joint subspace, which are then applied to the input of SVM for classification purpose. From the plots presented in Figure 4.3 and Figure 5.3, and the results reported in Table 5.6, it can be observed that although RGCCA and MCCA achieve satisfactory results for Digits data set, they have failed to obtain similar results for other benchmark databases. However, the proposed method attains highest classification accuracy with respect to the existing CCA based methods on all the six benchmark databases. In case of omics data sets, from Figure 4.3, Figure 5.3, and Table 5.7, it can be observed that the proposed method outperforms all the six multiset CCA based methods on three cancer data sets for both training-testing and 10-fold CV, except on CESC data set for 10-fold CV, where RGCCA achieves similar classification accuracy. Statistical significance tests reveal that out of total 36 cases, the proposed architecture achieves significantly better p-values for 34 cases.

#### 5.4.5.2 Performance of Discriminative Analysis Based Methods

Various state-of-the-art multi-view discriminative analysis based methods, namely, MvDA [92] and MvDA-VC [93], are considered for performance evaluation. For each of the methods, 25 features are extracted, and then given as input to the SVM for classification pur-

Table 5.7: Comparative Performance Analysis of Classical Approaches on Omics Data

| Data | Different Metrics | | RGCCA | MCCA | GMCCA | GMKCCA | LasCCA | DisCCA | MvDA | MvDA-VC | D2CCA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CESC | Train-Test | | 61.54 | 38.46 | 42.31 | 44.23 | 42.31 | 36.54 | 42.31 | 40.38 | **69.23** |
| | 10-fold CV | Mean | **75.00** | 45.83 | 49.17 | 38.33 | 35.00 | 39.17 | 46.67 | 50.00 | **75.00** |
| | | Median | **79.17** | 50.00 | 50.00 | 41.67 | 33.33 | 33.33 | 41.67 | 50.00 | 75.00 |
| | | StdDev | 13.03 | 13.75 | 14.41 | 11.92 | 15.61 | 10.43 | 15.32 | 14.16 | **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 | - |
| | | 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 | - |
| LGG | Train-Test | | 41.40 | 39.78 | 33.33 | 38.71 | 44.09 | 29.03 | 75.81 | 73.12 | **82.80** |
| | 10-fold CV | Mean | 45.00 | 35.53 | 40.53 | 33.16 | 38.68 | 38.68 | 75.79 | 81.05 | **82.37** |
| | | Median | 47.37 | 34.21 | 40.79 | 31.58 | 36.84 | 38.16 | 76.32 | 78.95 | **82.89** |
| | | StdDev | 10.99 | 7.67 | 8.70 | **4.99** | 7.95 | 7.24 | 8.02 | 7.83 | 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 | - |
| | | 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 | - |
| LUNG | Train-Test | | 87.91 | 46.52 | 68.86 | 86.08 | 82.42 | 47.62 | 92.31 | 91.58 | **95.24** |
| | 10-fold CV | Mean | 87.68 | 51.43 | 68.39 | 86.07 | 85.18 | 50.71 | 94.82 | 95.54 | **96.61** |
| | | Median | 86.61 | 50.89 | 69.64 | 87.50 | 85.71 | 48.21 | 96.43 | 95.54 | **97.32** |
| | | StdDev | 4.16 | **3.13** | 7.48 | 8.32 | 6.68 | 8.84 | 4.16 | 3.29 | 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 | - |
| | | 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 | - |

Table 5.8: Comparative Performance Analysis of Deep Models on Benchmark Data Sets

| Data | dMCCA | TOCCA | DACCA | DCCA-VG | TDDCCA | MDL-CW | MMGNN | MVGAN | D2CCA |
|---|---|---|---|---|---|---|---|---|---|
| Digits | 10.00 | **96.50** | 84.60 | 89.00 | 85.90 | 86.40 | 88.90 | 82.70 | 91.00 |
| Caltech | 33.71 | 80.23 | 73.89 | 49.68 | 74.52 | 54.25 | 74.27 | 77.31 | **84.54** |
| CiteSeer | 21.16 | 39.60 | 55.22 | 37.33 | 40.42 | 42.05 | 41.78 | 46.32 | **73.12** |
| Cora | 30.19 | 52.39 | 50.06 | 44.62 | 53.05 | 41.40 | 42.06 | 44.95 | **80.47** |
| NW-OBJECT | 17.80 | 33.61 | 38.42 | 19.23 | 17.80 | 18.20 | 37.87 | 27.61 | **52.21** |
| Reuters | 51.72 | 57.38 | 56.38 | 64.38 | 57.27 | 62.69 | 59.16 | 57.60 | **84.60** |

pose. The scatter plots corresponding to the MvDA and MvDA-VC methods are depicted in Figure 5.3 on both benchmark and omics databases. Considering the graphs depicted in Figure 5.3 and the results reported in Table 5.6, 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 Digits database. For omics data sets, the plots presented in Figure 5.3 and results are reported in Table 5.7 which demonstrate that although MvDA and MvDA-VC achieve similar classification accuracy, the proposed architecture outperforms both the methods on all the omics data sets. From the p-values obtained through two statistical significance tests, it can be observed that out of total 12 cases, the proposed model attains significantly better p-values for 7 cases and better but not significant p-values for 5 cases.

#### 5.4.5.3 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 [178], TOCCA [30], DACCA [44], DCCA-VG [193], TDDCCA [37], MDL-CW [157], MMGNN [54], and

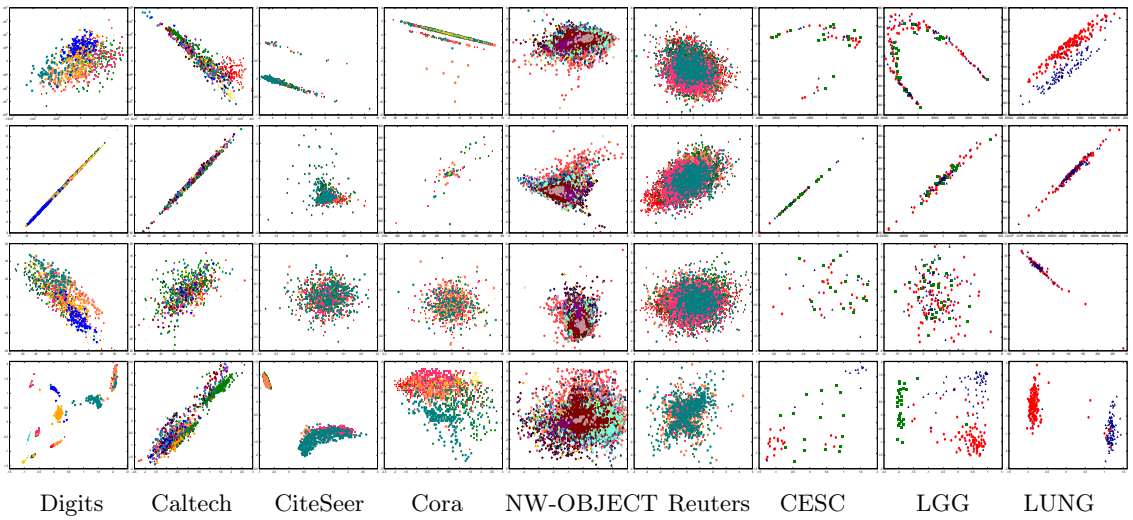



Figure 5.4: Scatter plots of deep approaches on benchmark and omics data sets (from top to bottom row: MDL-CW, MMGNN, MVGAN, and D2CCA, respectively).

MVGAN [209], and the corresponding results are reported in Table 5.8 and Table 5.9. The scatter plots corresponding to the dMCCA, TOCCA, DACCA, DCCA-VG, and TDDCCA models are presented in Figure 4.4 of Chapter 4, whereas the scatter plots for MDL-CW, MMGNN, and MVGAN models are depicted in Figure 5.4 on both benchmark and omics databases. For TOCCA, DACCA, DCCA-VG, TDDCCA, and MDL-CW, 50, 80, 20, 50, and 600 features are extracted, respectively, from the given views, which are then applied to the input of the SVM for classification. In 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 classifier for class label prediction. The architecture for each of the models follows the same as suggested in the corresponding papers. From the plots presented in Figure 4.4 and Figure 5.4, and the results reported in Table 5.8, it can be noticed that except for TOCCA method on Digits database, the proposed method performs considerably better than the existing multi-view deep learning based methods on the benchmark databases. For the omics data sets, the graphs are depicted in Figure 4.4 and Figure 5.4, and the results are presented in Table 5.9, which describes that the TDDCCA approach performs better than the proposed model on CESC data for 10-fold CV. However, significant improvement in performance can be noted in case of the proposed D2CCA architecture as compared to the existing deep learning based models, irrespective of the experimental set-up and data sets considered. Statistical significance tests demonstrate that out of total 48 cases, the proposed model achieves significantly better p-values for 38 cases, and better but not significant p-values for 8 cases.

## 5.5 Conclusion

The major contribution of this chapter is four-fold, namely, (a) introducing D2CCA model based on the theory of CCA and MDDBM; (b) consolidating the theory of the proposed framework with convergence analysis; (c) demonstrating the effectiveness of the proposed

Table 5.9: Comparative Performance Analysis of Deep Architectures on Omics Data

| Data | Different Metrics | | dMCCA | TOCCA | DACCA | DCCA-VG | TDDCCA | MDL-CW | MMGNN | MVGAN | D2CCA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CESC | Train-Test | | 51.92 | 36.54 | 47.12 | 65.38 | 53.98 | 65.38 | 55.77 | 57.69 | **69.23** |
| | 10-fold CV | Mean | 40.83 | 43.33 | 39.81 | 67.50 | **78.20** | 69.17 | 69.17 | 61.67 | 75.00 |
| | | Median | 41.67 | 50.00 | 39.42 | 70.83 | **78.20** | 66.67 | 66.67 | 58.33 | 75.00 |
| | | StdDev | 12.08 | 14.05 | 5.37 | 16.87 | **0.06** | 14.72 | 13.64 | 12.55 | 6.80 |
| | | Paired-$t$:p | 3.17E-05 | 1.42E-04 | 3.17E-07 | 9.67E-02 | 9.15E-01 | 1.66E-01 | 1.36E-01 | 3.16E-03 | - |
| | | Wilcoxon:p | 2.45E-03 | 2.49E-03 | 2.52E-03 | 7.63E-02 | 8.34E-01 | 1.42E-01 | 1.05E-01 | 6.97E-03 | - |
| LGG | Train-Test | | 62.37 | 45.70 | 65.59 | 77.96 | 66.85 | 73.12 | 71.51 | 74.73 | **82.80** |
| | 10-fold CV | Mean | 50.79 | 45.00 | 59.35 | 51.32 | 57.14 | 76.84 | 72.11 | 76.84 | **82.37** |
| | | Median | 47.37 | 46.05 | 58.87 | 51.32 | 57.00 | 77.63 | 72.37 | 77.63 | **82.89** |
| | | StdDev | 7.24 | 6.38 | 3.42 | 3.34 | **0.39** | 7.63 | 3.96 | 7.63 | 5.41 |
| | | Paired-$t$:p | 2.19E-06 | 1.96E-07 | 6.98E-08 | 4.76E-08 | 6.16E-08 | 3.50E-03 | 1.45E-05 | 3.50E-03 | - |
| | | Wilcoxon:p | 2.50E-03 | 2.52E-03 | 2.53E-03 | 2.49E-03 | 2.53E-03 | 8.09E-03 | 2.50E-03 | 8.09E-03 | - |
| LUNG | Train-Test | | 59.34 | 57.14 | 95.05 | 89.74 | 67.42 | 93.77 | 90.84 | 92.67 | **95.24** |
| | 10-fold CV | Mean | 60.71 | 57.14 | 95.71 | 94.11 | 67.72 | 93.93 | 82.14 | 94.11 | **96.61** |
| | | Median | 58.93 | 57.14 | 95.60 | 95.54 | 67.65 | 95.54 | 93.75 | 94.64 | **97.32** |
| | | StdDev | 5.05 | 0.00 | 1.17 | 4.69 | **0.38** | 4.31 | 19.32 | 3.95 | 3.41 |
| | | Paired-$t$:p | 1.99E-10 | 2.13E-11 | 2.53E-01 | 1.48E-02 | 5.71E-10 | 2.33E-03 | 1.75E-02 | 1.47E-02 | - |
| | | Wilcoxon:p | 2.52E-03 | 2.43E-03 | 2.88E-01 | 1.76E-02 | 2.53E-03 | 5.71E-03 | 2.17E-02 | 1.76E-02 | - |

architecture as feature extractor as well as classifier; and (d) illustrating the proficiency of the proposed method on different domains of applications, namely, object recognition, document classification, multilingual categorization, and cancer subtype identification.

The learning objective of the proposed architecture includes the merits of multi-view discriminative deep Boltzmann machines which allows the model to estimate the activations for the hidden nodes of the framework in such a way that a plausible explanation of how the observed data vectors would have been generated can be constituted. Considering supervised information into the objective function enhances the discriminative ability of the joint representation of the model, which in turn, entitles the D2CCA model to serve as the feature extractor as well as classifier. In order to capture the non-linear 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 data sets. Significant improvement in performance is noticed in case of the proposed model as compared to the existing approaches for both training-testing and 10-fold CV.

The proposed framework is developed primarily based on the consensus principle. However, 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. Additionally, a data-specific architecture of the proposed deep model needs to be estimated instead of considering an uniform architecture for all the given databases. It can be noted from Figure 5.1 that despite of the extensive experimentation carried out on all the data sets, a trade-off has to be made between the classification accuracy achieved on different databases in order to obtain the uniform architecture of the D2CCA framework. So, it is necessary to determine the optimal number of layers and the number of hidden nodes at

a particular layer for each data set in such a way that the challenges offered by different databases can be efficiently characterized. In this regard, a novel deep learning model is presented in next chapter, based on the concept of Hilbert-Schmidt independence criterion, to capture the relevant cross-view information from the given multi-view data. An upper bound on the error probability of the proposed deep model is estimated in terms of the model architecture. It facilitates determining the optimal architecture of the proposed model for each database considered.

# Chapter 6

# Discriminative Deep Generalized Dependency Analysis for Cross-View Learning

## 6.1 Introduction

The primary objective of a predictive model is to identify and analyze the inherent structures or patterns of the data, which are relevant to categorize the given samples or observations into different groups. With the recent advent of data acquisition processes, a surging interest is noted for the development of predictive models for the analysis of multi-view data in numerous domains of applications. However, due to the presence of noise and disparity among the sources of different views, it is primarily assumed that each view has a fundamentally distinct representation of the underlying data distribution. Hence, it is essential for the predictive models to discover the intrinsic patterns and relationship shared between each pair of given input views. Concatenation of all the views suffers from loss of the individual statistical properties of each of the input views, which is predominant in case of heterogeneous data. Hence, information from various sources needs to be consolidated appropriately in order to enhance the proficiency of the multimodal predictive models.

In existing literature, two significant principles are generally considered for judicious integration of information from multiple sources, namely, consensus and complementary. In multi-view environment, the input views are expected to agree upon the inherent latent distribution from where the views are assumed to be generated. The consensus principle focuses on maximizing the agreement on different views [96]. In [33], the connection between the consensus of two hypotheses on two views and their corresponding error rates is established. Theoretical analysis demonstrates that the probability of a disagreement of two independent hypotheses upper bounds the error rate of either hypothesis. Thus, by maximizing the agreement between the two hypotheses, the error rate of each hypothesis can be minimized. On the other hand, the complementary principle states that each view of the given data may provide some knowledge which is distinct from the rest of the views [93]. Therefore, the underlying complementary information of different views can be

suitably exploited to learn the diverse knowledge regarding the latent inherent structure of the data.

Following the consensus principle, various classical as well as deep learning models have been developed to characterize the correlated structures across different views, namely, regularized generalized CCA (RGCCA) [187], multiset CCA (MCCA) [96], graph-regularized MCCA (GMCCA) [25], graph-regularized kernel MCCA (GMKCCA) [25], large-scale generalized CCA (LasCCA) [53], distributed algorithm for CCA (DisCCA) [53], deep CCA with view generation (DCCA-VG) [193], and multimodal deep learning framework (MDL-CW) [157], which have already been discussed in Chapter 2 and Chapter 4 in details. Several multi-view algorithms based on the complementary principle have also been formulated over the past few years, which include multi-view discriminant analysis (MvDA) [92], MvDA with view-consistency (MvDA-VC) [93], multi-view generative adversarial network (MVGAN) [209], multi-view common component discriminant analysis (MvCCDA) [217], and multi-view linear discriminant analysis network (MvLDAN) [79].

The MvCCDA approach is proposed in [217] to handle view discrepancy, discriminability, and non-linearity in a joint manner. It incorporates the supervised information of sample categories into the common component extraction process to learn a discriminant joint subspace corresponding to the given input multi-view data. Hu et al. [79] have formulated MvLDAN method by seeking a non-linear discriminant and view-invariant representation shared among multiple views. It employs multiple feed-forward neural networks corresponding to each view and a novel eigenvalue-based objective function to encapsulate the discriminative variance into the shared subspace. Since MvCCDA is developed based on paired observations, it does not take into consideration the optimization of all the input views simultaneously. The MvLDAN model concentrates on the view-specific information, but they does not take into consideration the shared knowledge of different views.

In recent years, an escalation of interest is noted for developing models that can efficiently capture both consensus and complementary information from the given multi-view data. In this context, towards deep and discriminative CCA (TDDCCA) [37], multimodal graph neural network (MMGNN) [54], and deep adversarial CCA (DACCA) [44] models have already been discussed in Chapter 2, Chapter 4, and Chapter 5 in details. Inheriting the advantages of both structural graph learning and multi-view generative models, multi-view graph restricted Boltzmann machine (mgRBM) is developed in [221]. In mgRBM, the latent representations of each view are determined by considering the structure information of other views. It also adaptively balances the consensus and complementary information of different views to disentangle the hidden representations of multi-view data. The state-of-the-art multi-view learning approaches consider only pair-wise correlations and ignores the higher-order correlations among multiple views for identifying latent distribution of the given observations. In order to explore the consensus as well as complementary information, and simultaneously explore the high-order statistical relationships, a novel deep convolutional network, which is referred to as tensor canonical correlation analysis (TCCA), is developed in [213]. The multi-view filter kernels are learned by decomposing a higher-order covariance tensor [125]. Though TDDCCA [37] concentrates on learning the correlated subspaces, it eventually disregards the underlying data distribution. Since the DACCA [44] 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.

Hence, a deep learning model needs to be developed which can efficiently learn a joint

representation over the space of multimodal inputs. In order to address the multi-view classification problem, it is required that the similarity in the latent space implies the similarity in the corresponding concepts. Non-linearity is another important aspect in multi-view data analysis. It is also expected in multimodal learning that the joint representation is learned from the given input views in such a way that the view-specific as well as the cross-view information are preserved properly. In the context of cross-view dependency, it is essential that both view-consistency and view-discrepancy are addressed simultaneously. In existing literature, a unified approach, based on either consensus and/or complementary principles, is considered to represent view-consistency or view-discrepency in the joint subspace. However, in the proposed approach, it is assumed that each view has a fundamentally distinct representation of the underlying data distribution and so, the relationship between each pair of views is considered to be unique. Hence, instead of considering a unified approach among all the pairs, a view-pair specific method should be employed for efficient representation of cross-view information in the joint subspace. It can be observed in Chapter 5 that extensive experimentation needs to be carried out on all the data sets and a trade-off has to be made between the classification accuracy achieved on different databases, in order to obtain the uniform architecture of the D2CCA framework. So, it is necessary to determine the optimal number of layers and the number of hidden nodes at a particular layer for each data set in such a way that the challenges offered by different databases can be efficiently characterized.

In this regard, a novel deep learning model, termed as discriminative deep generalized dependency analysis (D2GDA), is developed based on the framework of multimodal discriminative deep Boltzmann machine (MDDBM), which is discussed in Chapter 4 in details. Thus, the proposed model can efficiently encapsulate the underlying non-linear data distribution of the given observations. The MDDBM framework considers supervised information of sample categories at each layer of the network, as a result of which the discriminative ability of the D2GDA model is enhanced. Also, no additional classifier needs to employed in case of the proposed D2GDA model for classifying the given observations into different categories. Based on the concept of Hilbert-Schmidt independence criterion, a loss function is proposed to efficiently capture the cross-view dependency across several views. The proposed model is developed based on the hypothesis that the relation between each pair of views is unique. Hence, a view-pair specific constraint is incorporated in the loss function to extract the relevant cross-view information in terms of consensus and/or complementary knowledge from the given input pairs of views. An upper bound on the error probability of the proposed deep model is estimated in terms of the model architecture. Hence, instead of heuristically determining the architecture of the proposed model, an optimal architecture is estimated for each given database based on the Bayes error analysis of the network. While the number of layers is obtained from the total error probability of the model, the number of nodes at each layer is computed based on the Hilbert-Schmidt independence criterion. An analytic formulation demonstrates that the proposed model is the generalization of several state-of-the-art feature extraction techniques. The proposed approach is further consolidated with the convergence analysis. Finally, the proficiency of the proposed model is studied on numerous domain of applications with reference to several state-of-the-arts multi-view classification algorithms. Some of the results of this chapter are reported in [105].

The rest of this chapter is organized as follows: Section 6.2 describes the concept of

Hilbert-Schmidt independence criterion. The proposed deep model is discussed in details in Section 6.3. Different aspects of the proposed model which include error analysis, generalization ability, and convergence analysis, are discussed in Section 6.4. The efficacy of the D2GDA framework is analyzed with reference to several state-of-the-art approaches on various benchmark and real-life cancer data sets in Section 6.5. Concluding remarks are provided in Section 6.6.

## 6.2 Basics of Hilbert-Schmidt Independence Criterion

A vector space with the inner-product $\langle \cdot, \cdot \rangle$ operation is referred to as inner-product space. An important property regarding inner-product space is given by [17]:

**Property 6.1.** *Any finite dimensional inner-product space is a Hilbert space.*

Let us consider, $k$ represents a kernel in Hilbert space. The reproducing property of Hilbert space is given by the following theorem.

**Theorem 6.1.** *If $k$ is a positive definite kernel, then there exists a unique reproducing kernel Hilbert space (RKHS) $\mathcal{F}$ whose kernel is $k$.*

The Hilbert-Schmidt independence criterion (HSIC) [59] efficiently measures the dependency between two random variables, by mapping the variables into RKHS such that the correlations measured in that space correspond to higher-order joint moments between the original distributions. The advantage of the HSIC over state-of-the-art dependency measures is that it provides an empirical definition to compute the dependency between two random variables without estimating the joint distribution.

Consider two random variables $X$ and $Y$, which are defined over two input spaces $\mathcal{X}$ and $\mathcal{Y}$, respectively, with a joint distribution $p_{xy}$. Let us define a mapping $\phi(x)$ from $x \in \mathcal{X}$ to RKHS $\mathcal{F}$, such that the inner product between two vectors in $\mathcal{F}$ is given by a kernel function $k^1(x, x') = \langle \phi(x), \phi(x') \rangle$. Let $\mathcal{G}$ be another RKHS defined on input space $\mathcal{Y}$ with mapping $\varphi(y)$ and kernel function $k^2(y, y') = \langle \varphi(y), \varphi(y') \rangle$. The linear cross-covariance operator $C_{xy} : \mathcal{G} \rightarrow \mathcal{F}$ between two feature maps is defined as

$$C_{xy} = \mathbb{E}[(\phi(x) - \mu_x) \otimes (\varphi(y) - \mu_y)], \tag{6.1}$$

where $\mathbb{E}[\cdot]$ denotes the expectation operator, $\mu_x = \mathbb{E}[\phi(x)]$ and $\mu_y = \mathbb{E}[\varphi(y)]$ represent the mean values of $\phi(x)$ and $\varphi(y)$, respectively, and $\otimes$ signifies the tensor product. Using Riesz's representation theorem [158], it can be shown that if $\mathbb{E}[k^1(x, x')]$ and $\mathbb{E}[k^2(y, y')]$ are finite, then $C_{xy}$ exists and is unique.

Given two separable RKHSs $\mathcal{F}$, $\mathcal{G}$, and the joint distribution $p_{xy}$, the HSIC between two variables $X$ and $Y$ is defined as the Hilbert-Schmidt norm of the cross-covariance operator:

$$\text{HSIC}(p_{xy}, \mathcal{F}, \mathcal{G}) = \|C_{xy}\|^2_{\text{HS}}. \tag{6.2}$$

Let $\mathcal{Z} = \{(x_1, y_1), \cdots, (x_N, y_N)\} \subseteq \mathcal{X} \times \mathcal{Y}$ be a set of $N$ independent observations drawn from $p_{xy}$. The empirical estimate of the HSIC is given by

$$\text{HSIC}(\mathcal{Z}, \mathcal{F}, \mathcal{G}) = \frac{1}{(N-1)^2}\mathbf{tr}(K^1 D K^2 D), \tag{6.3}$$

where $\mathbf{tr}(\cdot)$ denotes the trace operator, $K^1, K^2, D \in \mathbb{R}^{N \times N}$, $K^1$ and $K^2$ are Gram matrices corresponding to the kernels $k^1$ and $k^2$, respectively, where $K^1_{i,j} = k^1(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$, $K^2_{i,j} = k^2(y_i, y_j) = \langle \varphi(y_i), \varphi(y_j) \rangle$, and $D_{i,j} = \delta_{i,j} - N^{-1}$ centers the Gram matrix to have zero mean in the feature space. In [59], it has been shown that $\text{HSIC}(\mathcal{Z}, \mathcal{F}, \mathcal{G})$ converges to $\text{HSIC}(p_{xy}, \mathcal{F}, \mathcal{G})$ at a rate of $\mathcal{O}(N^{-1/2})$ with a bias of $\mathcal{O}(N^{-1})$.

**Theorem 6.2.** *Denote by $\mathcal{F}$, $\mathcal{G}$ RKHSs with universal kernels $k^1$, $k^2$ on the compact domains $\mathcal{X}$ and $\mathcal{Y}$, respectively. Assume without loss of generality that $\|f\|_\infty \leqslant 1$ and $\|g\|_\infty \leqslant 1$ for all functional $f \in \mathcal{F}$ and $g \in \mathcal{G}$. Then, $\|C_{xy}\|_{HS} = 0$ if and only if $X$ and $Y$ are independent* [59].

In general, if the covariance between the variables $X$ and $Y$ is zero, then it does not imply that the variables are independent of each other. If $X$ and $Y$ are non-linearly related, then it will not be reflected in the corresponding covariance value. However, a zero HSIC value does imply independence of the associated variables. Hence, it can be said that the HSIC takes higher-order moments into account while measuring the dependency between two random variables.

## 6.3 Proposed Model

In this section, the proposed D2GDA model, along with its learning, is discussed in details. At first, the concept of generalized dependency analysis (GDA) is proposed in the current study to capture the cross-modal information from the given view-specific representations. Then, the architecture of the proposed D2GDA model is discussed to encapsulate the underlying data distribution over the space of multimodal inputs. Finally, the learning of the D2GDA model is developed by judiciously incorporating the objective of GDA into the proposed deep framework.

### 6.3.1 Proposed Generalized Dependency Analysis

Let us consider that the given input modalities $\{\mathbf{v}^m\}$ are transformed into respective modality-specific subspaces $\{\mathbf{h}^m\}$. Now, the joint subspace $\mathbf{h}$, learned from the $\{\mathbf{h}^m\}$, should be able to capture the modality-specific characteristics as well as the cross-modal information of the data across various modalities. The pictorial representation of the above concept is depicted in Figure 6.1, where $M$ denotes the total number of input modalities.

Now, the relevant cross-modal information can be embedded in the correlated structures or complementary knowledge of different views. Since each view has a fundamentally distinct representation of the underlying data distribution, the relationship between each pair of views is assumed to be unique. Hence, instead of considering an unified approach

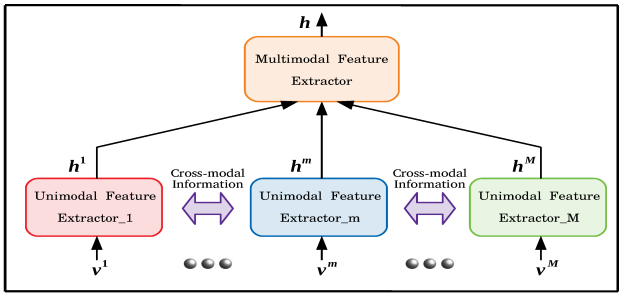



Figure 6.1: Illustration of proposed multi-view data analysis framework.

among all the pairs, a view-pair specific method should be employed for efficient representation of cross-modal information in the joint subspace. In this regard, an objective function is proposed in this chapter, based on the concept of HSIC, that not only quantifies dependency among multiple modalities, but also facilitates to identify relevant cross-modal information in terms of coherent structures or complementary knowledge from the given input pair of views.

In the proposed approach, it is assumed that the vector space spanned by each modality-specific representation is $\mathbb{R}^H$, where $H$ is the dimension of $\mathbf{h}^m$, $\forall_{m=1}^{M}$. Hence, it is essentially a Hilbert space of dimension $H$ since it holds Property 6.1.

Let, $K^m$ be the Gram matrix corresponding to the kernel $k^m$ associated with the Hilbert space spanned by $\mathbf{h}^m$. In the proposed method, $K^m$ is defined as

$$K^m = (\mathbf{h}^m - \underline{\mathbf{h}}^m) \otimes (\mathbf{h}^m - \underline{\mathbf{h}}^m), \quad \forall m \in \{1, 2, \cdots, M\} \tag{6.4}$$

where $\underline{\mathbf{h}}^m$ represents the mean vector of $\mathbf{h}^m$. So, $K^m$ is defined to be a cross-covariance matrix.

**Property 6.2.** *$K^m$ is always positive semi-definite.*

The positive definiteness of $K^m$ can be ensured by restricting the variance value equal to 1 using Lagrange multiplier. So, by Theorem 6.1, it can be said that the vector space spanned by the corresponding modality-specific subspace is RKHS. Hence, based on the $K^m$ considered in the current study, the value of HSIC between the modality-specific subspaces $\mathbf{h}^m$ and $\mathbf{h}^r$ is computed as

$$\begin{aligned}
\text{HSIC}(\mathbf{h}^m, \mathbf{h}^r) &= \frac{1}{(N-1)^2}\Big\{\mathbf{tr}(K^m K^r)\Big\} \\
&= \frac{1}{(N-1)^2}\left\{\mathbf{tr}\left(\Big[\Big(\sum_{n=1}^{N}(h_{nj}^m - \underline{h}_j^m)(h_{nk}^m - \underline{h}_k^m)\Big)_{j,k}\Big]_{H\times H}\Big[\Big(\sum_{n=1}^{N}(h_{nj}^r - \underline{h}_j^r)(h_{nk}^r - \underline{h}_k^r)\Big)_{j,k}\Big]_{H\times H}\right)\right\} \\
&= \frac{1}{(N-1)^2}\left\{\mathbf{tr}\left(\Big[\Big(\sum_{n=1}^{N}\sum_{i=1}^{H}(h_{nj}^m - \underline{h}_j^m)(h_{nk}^r - \underline{h}_k^r)(h_{ni}^m - \underline{h}_i^m)(h_{ni}^r - \underline{h}_i^r)\Big)_{j,k}\Big]_{H\times H}\right)\right\}
\end{aligned}$$

$$= \frac{1}{(N-1)^2}\left\{\sum_{n=1}^{N}\sum_{j=1}^{H}\sum_{i=1}^{H}(h_{nj}^m - \underline{h}_j^m)(h_{nj}^r - \underline{h}_j^r)(h_{ni}^m - \underline{h}_i^m)(h_{ni}^r - \underline{h}_i^r)\right\}$$

$$= \frac{1}{(N-1)^2}\sum_{n=1}^{N}\left\{\sum_{j=1}^{H}\left((h_{nj}^m - \underline{h}_j^m)(h_{nj}^r - \underline{h}_j^r)\right)^2 + 2\sum_{j=1}^{H}\sum_{i=(j+1)}^{H}(h_{nj}^m - \underline{h}_j^m)(h_{nj}^r - \underline{h}_j^r)(h_{ni}^m - \underline{h}_i^m)(h_{ni}^r - \underline{h}_i^r)\right\}$$

$$= \frac{1}{(N-1)^2}\sum_{n=1}^{N}\left\{\sum_{j=1}^{H}(h_{nj}^m - \underline{h}_j^m)(h_{nj}^r - \underline{h}_j^r)\right\}^2. \tag{6.5}$$

Thus, from the definition of $K^m$, presented in (6.4), the expression of HSIC$(\mathbf{h}^m, \mathbf{h}^r)$ is obtained in (6.5), which is to be considered for the rest of the current study. Here, the centering matrices of (6.3) are ignored since $K^m$ in (6.4) is already defined to be mean centered. Based on the above analysis, the following theorem is introduced for HSIC$(\mathbf{h}^m, \mathbf{h}^r)$, $\forall m, r$, corresponding to the Gram matrix $K^m$ defined in (6.4).

**Theorem 6.3.** *The value of HSIC$(\boldsymbol{h}^m, \boldsymbol{h}^r)$ between $\boldsymbol{h}^m$ and $\boldsymbol{h}^r$ lies within the range of [0,1], if $H \leqslant \frac{4(N-1)}{\sqrt{N}}$ and $h_{nj}^m \in$ [0,1], $\forall n, j, m$.*

*Proof*: It is assumed that $h_{nj}^m \in [0,1]$, $\forall n, j, m$.

$$\text{So, } (h_{nj}^m - \underline{h}_j^m) \in [-\frac{1}{2}, \frac{1}{2}], \ \forall n, j, m$$

$$\Rightarrow (h_{nj}^m - \underline{h}_j^m)(h_{nj}^r - \underline{h}_j^r) \in [-\frac{1}{4}, \frac{1}{4}], \ \forall n, j, m, r$$

$$\Rightarrow \left\{\sum_{j=1}^{H}(h_{nj}^m - \underline{h}_j^m)(h_{nj}^r - \underline{h}_j^r)\right\}^2 \in [0, \frac{|H^2|}{16}], \ \forall n, m, r$$

$$\Rightarrow \text{HSIC}(\mathbf{h}^m, \mathbf{h}^r) \in [0, \frac{N|H^2|}{16(N-1)^2}], \ \forall m, r.$$

$$\text{So, if } H \leqslant \frac{4(N-1)}{\sqrt{N}}, \text{HSIC}(\mathbf{h}^m, \mathbf{h}^r) \in [0,1], \forall m, r. \square$$

Thus, Theorem 6.3 provides an upper bound on the dimensionality of modality-specific representations $\mathbf{h}^m$, $\forall_{m=1}^M$ based on the number of given observations $N$. It is important to note here that the bound obtained in Theorem 6.3 becomes particularly effective in cases where the dimension or the number of features in a representation is heuristically determined. In the proposed approach, the definition of HSIC, obtained from (6.5), is employed to quantify the cross-modal information between each pair of modality-specific representations. As discussed in Section 6.2, the higher value of HSIC indicates higher dependency between two random variables, while the zero value of HSIC implies independence between the associated variables. In order to capture the non-linear dependency among multiple modalities, a new measure, termed as balanced HSIC (BHSIC), is introduced by incorporating a balance parameter between each pair of views as follows:

$$\text{BHSIC}(\mathbf{h}^1, \cdots, \mathbf{h}^m, \cdots, \mathbf{h}^M) = \frac{2}{M(M-1)} \sum_{m=1}^{M-1} \sum_{r=(m+1)}^{M} |\gamma_{mr}| \text{HSIC}(\mathbf{h}^m, \mathbf{h}^r). \tag{6.6}$$

Here, $\gamma_{mr}$ denotes the balance parameter between a pair of views $\mathbf{v}^m$ and $\mathbf{v}^r$. The value of $\gamma_{mr}$ in (6.6) signifies the contribution of dependency between $\mathbf{h}^m$ and $\mathbf{h}^r$ in the overall cross-modal information of the given input data. An important property of the proposed BHSIC measure is to be noted here.

**Property 6.3.** *Given HSIC($\boldsymbol{h}^m, \boldsymbol{h}^r$) ∈ [0,1] and $\gamma_{mr} \in$ [-1,1], $\forall_{m,r=1}^{M}$, the value of BHSIC($\boldsymbol{h}^1$, $\cdots, \boldsymbol{h}^m, \cdots, \boldsymbol{h}^M$) lies within the range of [0,1]. A zero value of BHSIC denotes that all the modality-specific representations are completely independent of each other, while the higher value of BHSIC signifies higher dependency between the given representations.*

Based on the BHSIC measure, defined in (6.6), a loss function is proposed to learn view-consistency and view-discrepancy simultaneously across several modalities, which is as follows:

$$\begin{aligned} E_{GDA}(\mathbf{h}^1, \cdots, \mathbf{h}^m, \cdots, \mathbf{h}^M) &= -\sum_{m=1}^{M} \lambda_m \Big(1 - \mathbf{tr}(K^m)\Big) - \Big\{ \sum_{m=1}^{M-1} \sum_{r=(m+1)}^{M} \gamma_{mr} \mathbf{tr}(K^m K^r) \Big\} \\ &= -\sum_{n=1}^{N} \Bigg[ \sum_{m=1}^{M} \lambda_m \Big(1 - \sum_{j=1}^{H} (h_{nj}^m - \underline{h}_j^m)^2\Big) + \sum_{m=1}^{M-1} \sum_{r=(m+1)}^{M} \gamma_{mr} \Big\{ \sum_{j=1}^{H} (h_{nj}^m - \underline{h}_j^m)(h_{nj}^r - \underline{h}_j^r) \Big\}^2 \Bigg], \end{aligned} \tag{6.7}$$

where $\gamma_{mr} \in [-1, 1]$ and $\lambda_m$ represents the Lagrange multiplier. The first term in (6.7) ensures that the variance value of $\mathbf{h}^m$ is equal to 1, or equivalently, $K^m$, $\forall m$, is always positive definite. The second term computes the weighted dependency value between each pair of given input modalities. Thus, for $\gamma_{mr} \in (0, 1]$, minimizing $E_{GDA}$ will ensure that the BHSIC value and correspondingly, the dependency between the modality-specific subspaces are maximized. As a consequence, the coherent knowledge between the views will be reflected in the joint subspace. However, if $\gamma_{mr} \in [-1, 0)$, minimization of $E_{GDA}$ corresponds to the minimization of the BHSIC value. Hence, the independence between the associated pair of views will be maximized, which in turn, enhances the complementary information of the individual views in the joint representation.

**Property 6.4.** *If $\gamma_{mr} \in (0, 1]$, then the dependency between $\boldsymbol{h}^m$ and $\boldsymbol{h}^r$ is maximized; if $\gamma_{mr} \in [-1, 0)$, then the independence between $\boldsymbol{h}^m$ and $\boldsymbol{h}^r$ is maximized; and if $\gamma_{mr} = 0$, then the dependency between the corresponding view-pair is not taken under consideration in order to minimize $E_{GDA}$.*

The loss function, defined in (6.7), and Property 6.4 ensure that a view-pair specific method is developed which can appropriately capture cross-modal information across multiple modalities in terms of correlated or complementary structures of the data, based on

the appropriate values of balance parameters. In order to learn the optimal value of $\gamma_{mr}$, a deep learning model is proposed next.

### 6.3.2 Architecture of Proposed D2GDA Model

In multi-view classification problem, it is expected that the non-linear structures embedded in the given input views, along with the supervised information of sample categories, are suitably reflected in the shared subspace. Hence, the architecture of the proposed D2GDA model is developed based on the framework of multimodal discriminative deep Boltzmann machine (MDDBM), introduced in Chapter 4. Therefore, the joint subspace, learned from the D2GDA model, is expected to encapsulate the underlying non-linear data distribution of the given observations. Since the class nodes are included into the framework, the hidden subspaces of the proposed model contain the supervised information of sample categories, which in turn enhances the discriminative ability of the model. Also, it allows the model to predict the class label of the given observations, without employing any additional classifier.

In the proposed D2GDA model, let the input view corresponding to the $m$-th modality be represented by $\mathbf{v}^m = \{v_1^m, \cdots, v_i^m, \cdots\}$ and $\mathbf{y} = \{y_1, \cdots, y_c, \cdots\}$ denotes the class label information. Let us assume that $L_1 > 0$ signifies the number of modality-specific hidden layers in the architecture and $L_2 > 0$ refers to the number of joint hidden layers. While $\mathbf{h}^{lm} = \{h_1^{lm}, \cdots, h_j^{lm}, \cdots\}$ represents the $l$-th modality-specific hidden representation of the $m$-th modality, the joint hidden representation, corresponding to the $l$-th layer, is referred to as $\mathbf{h}^l = \{h_1^l, \cdots, h_j^l, \cdots\}$. 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 $V^m$.

Here, the bidirectional weight parameters $w_{ij}^{1m}$, $w_{jk}^{(l+1)m}$, $w_{jk}^{(L_1+1)m}$, and $w_{jk}^{(l+1)}$ connect the $i$-th input 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 $l$-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 $L_1$ 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 $u_{cj}^{lm}$ and $u_{cj}^{l}$ connect the $c$-th class node to the $j$-th hidden node of the $l$-th modality-specific hidden layer from $m$-th modality, and $c$-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 $d_c$ are associated with the $i$-th visible node of the $m$-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 $c$-th class node, respectively. Thus, the supervised information of sample categories can be appropriately incorporated at each layer of the architecture through proper learning of the set of parameters associated with the class nodes, which in turn, enhances the proficiency of the proposed framework.

### 6.3.3 Learning D2GDA Model Using Generalized Dependency Analysis

The objective function of GDA is judiciously integrated with the learning objective of MDDBM architecture to develop the proposed D2GDA model. The deep framework is not only able to learn the intrinsic characteristics associated with each of the given input modalities, but also identifies the relevant cross-modal information across different views.

The overall objective function and learning of the parameters corresponding to the D2GDA model are discussed in details in this section.

#### 6.3.3.1 Objective Function of Proposed Model

The proper learning of the MDDBM framework ensures that the joint subspace suitably represents the underlying inherent characteristics of the given modalities as well as supervised information of the sample categories. In order to encapsulate the cross-view dependency across several modalities in the shared subspace, the loss function, proposed in (6.7), is considered in the D2GDA model. The principle of GDA is integrated with the objective of the MDDBM model since the following properties hold.

- In MDDBM, $h_j^{L_1 m}$, $\forall j, m$, represents the state of the $j$-th hidden node of modality-specific hidden layer $L_1$ from modality $m$, which is essentially a real value. Hence, $\mathbf{h}^{L_1 m}$ spans $\mathbb{R}^H$, where $H$ denotes the dimension of $\mathbf{h}^{L_1 m}$, $\forall m$, that is, $H^{L_1 1} = H^{L_1 2} = \cdots H^{L_1 M} = H$. So, Property 6.1 holds.
- The Gram matrix $K^m$ is a variance-covariance matrix corresponding to $\mathbf{h}^{L_1 m}$ of the MDDBM architecture. In effect, it satisfies Property 6.2.
- In MDDBM framework, $h_j^{L_1 m} \in \{0, 1\}$, $\forall j, m$. Considering $H \leqslant \dfrac{4(N-1)}{\sqrt{N}}$ and $\gamma_{mr} \in [-1, 1]$, it can be ensured that both HSIC($\mathbf{h}^{L_1 m}, \mathbf{h}^{L_1 r}$) and BHSIC($\mathbf{h}^{L_1 1}, \cdots, \mathbf{h}^{L_1 M}$), $\forall_{m,r=1}^{M}$, lie within the range of [0,1]. Thus, Theorem 6.3 and Property 6.3 are satisfied.

So, the loss function of GDA, presented in (6.7), corresponding to each given observation, can be efficiently combined with the energy function (4.8) of the MDDBM architecture. Hence, the overall objective of the D2GDA model turns out be

$$
\begin{aligned}
&E_{d2gda}(\mathbf{v}, \mathbf{h}, \mathbf{y}) = E_{mddbm}(\mathbf{v}, \mathbf{h}, \mathbf{y}) + E_{gda}(\mathbf{h}^{L_1 1}, \cdots, \mathbf{h}^{L_1 M}) \\
&= -\sum_{m=1}^{M}\sum_{i=1}^{V^m}\sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m} h_j^{1m} - \sum_{l=1}^{L_1-1}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}}\sum_{k=1}^{H^{(l+1)m}} h_j^{lm} w_{jk}^{(l+1)m} h_k^{(l+1)m} - \sum_{m=1}^{M}\sum_{i=1}^{V^m} a_i^m v_i^m \\
&- \sum_{m=1}^{M}\sum_{j=1}^{H^{L_1 m}}\sum_{k=1}^{H^1} h_j^{L_1 m} w_{jk}^{(L_1+1)m} h_k^1 - \sum_{l=1}^{L_1}\sum_{m=1}^{M}\sum_{c=1}^{Y}\sum_{j=1}^{H^{lm}} y_c u_{cj}^{lm} h_j^{lm} - \sum_{l=1}^{L_2}\sum_{c=1}^{Y}\sum_{j=1}^{H^l} y_c u_{cj}^{l} h_j^{l} \\
&- \sum_{l=1}^{L_2-1}\sum_{j=1}^{H^l}\sum_{k=1}^{H^{(l+1)}} h_j^l w_{jk}^l h_k^{(l+1)} - \sum_{l=1}^{L_1}\sum_{m=1}^{M}\sum_{j=1}^{H^{lm}} b_j^{lm} h_j^{lm} - \sum_{m=1}^{M} \lambda_m \left(1 - \sum_{j=1}^{H^{L_1 m}} (h_j^{L_1 m} - \underline{h}_j^{L_1 m})^2\right) \\
&- \sum_{m=1}^{M-1}\sum_{r=(m+1)}^{M} \gamma_{mr} \left\{\sum_{j=1}^{H^{L_1 m}} (h_j^{L_1 m} - \underline{h}_j^{L_1 m})(h_j^{L_1 r} - \underline{h}_j^{L_1 r})\right\}^2 - \sum_{l=1}^{L_2}\sum_{j=1}^{H^l} b_j^l h_j^l - \sum_{c=1}^{Y} d_c y_c; \quad (6.8)
\end{aligned}
$$

where $E_{gda}$ represents the loss function $E_{GDA}$ of (6.7) for each given observation. The parameter space of the proposed model is defined by $\boldsymbol{\theta}_{d2gda} = \boldsymbol{\theta}_{mddbm} \cup \boldsymbol{\theta}_{gda}$; where $\boldsymbol{\theta}_{gda} = \{\lambda_m, \gamma_{mr}\}$, $\forall_{m,r=1}^{M}$. The learning of D2GDA model corresponds to estimating the model

parameter set $\boldsymbol{\theta}_{d2gda}$ that maximizes the probability of observing the given input data. Considering the input view $\{\mathbf{v}, \mathbf{y}\}$, the objective function of the proposed model is given by the log-likelihood function, which is as follows:

$$\ln \mathrm{L}(\boldsymbol{\theta}_{d2gda}|\mathbf{v}, \mathbf{y}) = \ln P(\mathbf{v}, \mathbf{y}|\boldsymbol{\theta}_{d2gda}) = \ln \sum_{\mathbf{h}} e^{-E_{d2gda}(\mathbf{v},\mathbf{h},\mathbf{y})} - \ln \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} e^{-E_{d2gda}(\mathbf{v},\mathbf{h},\mathbf{y})}; \quad (6.9)$$

where $\mathbf{h}$ denotes the stack of hidden layers, $P(\mathbf{v}, \mathbf{y}|\boldsymbol{\theta}_{d2gda})$ represents the probability assigned to the observation $(\mathbf{v}, \mathbf{y})$ by the model parameter set $\boldsymbol{\theta}_{d2gda}$, and $E_{d2gda}(\mathbf{v}, \mathbf{h}, \mathbf{y})$ signifies the energy of the joint configuration $\{\mathbf{v}, \mathbf{h}, \mathbf{y}\}$. The corresponding partition function can be defined as $Z = \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} e^{-E_{d2gda}(\mathbf{v},\mathbf{h},\mathbf{y})}$.

The values of $M$ number of $\lambda_m$ and $\binom{M}{2}$ number of $\gamma_{mr}$ parameters, along with the other parameters of D2GDA framework, can be obtained through proper learning of the model. If $\gamma_{mr}$ is learned to be positive, then the energy function in (6.8) decreases with increase in the dependency between $\mathbf{h}^{L_1 m}$ and $\mathbf{h}^{L_1 r}$. However, if $\gamma_{mr}$ is learned to be negative, then $E_{d2gda}(\mathbf{v}, \mathbf{h}, \mathbf{y})$ in (6.8) decreases as the corresponding modality-specific representations become more independent of each other. Since the parameter space of the D2GDA model is quite large, the gradient ascent on the log-likelihood is commonly used to determine the optimal parameters of the model. So, the update rule for the parameters of the D2GDA model is given by

$$\frac{\partial \ln \mathrm{L}(\boldsymbol{\theta}_{d2gda}|\mathbf{v}, \mathbf{y})}{\partial \boldsymbol{\theta}_{d2gda}} = -\sum_{\mathbf{h}} P(\mathbf{h}|\mathbf{v}, \mathbf{y}) \frac{\partial E_{d2gda}(\mathbf{v}, \mathbf{h}, \mathbf{y})}{\partial \boldsymbol{\theta}_{d2gda}} + \sum_{\mathbf{v},\mathbf{h},\mathbf{y}} P(\mathbf{v}, \mathbf{h}, \mathbf{y}) \frac{\partial E_{d2gda}(\mathbf{v}, \mathbf{h}, \mathbf{y})}{\partial \boldsymbol{\theta}_{d2gda}}. \quad (6.10)$$

Thus, the gradient of the log-likelihood function turns out to be the difference between the expectation of gradient of energy function under model distribution, referred to as data-independent expectation, and under the conditional distribution of hidden representation given the input views, termed as data-dependent expectation.

#### 6.3.3.2 Estimation of Data-Dependent Expectations

Now, the exact maximum likelihood learning is intractable, so the variational learning [142] is employed to estimate the data-dependent expectation. In variational inference, 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, it can be observed that

$$\ln P(\mathbf{v}, \mathbf{y}) = \ln \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v}, \mathbf{y}) \frac{P(\mathbf{v}, \mathbf{h}, \mathbf{y})}{Q(\mathbf{h}|\mathbf{v}, \mathbf{y})} \quad (6.11)$$

where $P(\mathbf{v}, \mathbf{h}, \mathbf{y}) = (1/Z)e^{-E_{d2cca}(\mathbf{v},\mathbf{h},\mathbf{y})}$ represents the probability associated with the joint configuration $\{\mathbf{v}, \mathbf{h}, \mathbf{y}\}$. Since logarithmic is a concave function, applying Jensen's inequal-

ity [31], we get

$$\ln P(\mathbf{v},\mathbf{y}) \geqslant \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \ln \frac{P(\mathbf{v},\mathbf{h},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})} = \mathcal{L}_v. \tag{6.12}$$

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

$$\begin{aligned}
&\ln P(\mathbf{v},\mathbf{y}) - \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \left\{ \ln P(\mathbf{v},\mathbf{y}) + \ln \frac{P(\mathbf{h}|\mathbf{v},\mathbf{y})}{Q(\mathbf{h}|\mathbf{v},\mathbf{y})} \right\} \\
&= \ln P(\mathbf{v},\mathbf{y}) - \ln P(\mathbf{v},\mathbf{y}) + \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \ln \frac{Q(\mathbf{h}|\mathbf{v},\mathbf{y})}{P(\mathbf{h}|\mathbf{v},\mathbf{y})} \\
&= KL(Q(\mathbf{h}|\mathbf{v},\mathbf{y})||P(\mathbf{h}|\mathbf{v},\mathbf{y})),
\end{aligned} \tag{6.13}$$

where $KL(Q(\mathbf{h}|\mathbf{v},\mathbf{y})||P(\mathbf{h}|\mathbf{v},\mathbf{y}))$ 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 $\ln P(\mathbf{v},\mathbf{y})$. Here, let the mean field distribution be defined as

$$Q(\mathbf{h}|\mathbf{v},\mathbf{y}) = \prod_{l=1}^{L_1} \prod_{m=1}^{M} \prod_{j=1}^{H^{lm}} q(h_j^{lm}|\mathbf{v},\mathbf{y}) \prod_{l=1}^{L_2} \prod_{j=1}^{H^l} q(h_j^l|\mathbf{v},\mathbf{y}); \tag{6.14}$$

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 $\mu_j$ denotes the probability of being the state of $h_j$ as 1. The definitions of $Q(\mathbf{h}|\mathbf{v},\mathbf{y})$, presented in (6.14), and $P(\mathbf{v},\mathbf{h},\mathbf{y})$, corresponding to the energy function obtained in (6.8), are substituted in (6.12) to obtain the final expression of $\mathcal{L}_v$, which is reported in (6.15).

$$\begin{aligned}
\mathcal{L}_v &= \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \Big\{ \ln P(\mathbf{v},\mathbf{h},\mathbf{y}) - \ln Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \Big\} \\
&= \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \Big\{ -E_{d2gda}(\mathbf{v},\mathbf{h},\mathbf{y}) - \ln Z \Big\} - \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \ln Q(\mathbf{h}|\mathbf{v},\mathbf{y}) \\
&= \sum_{m=1}^{M} \sum_{i=1}^{V^m} \sum_{j=1}^{H^{1m}} v_i^m w_{ij}^{1m} \mu_j^{1m} + \sum_{l=1}^{L_1-1} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} \sum_{k=1}^{H^{(l+1)m}} \mu_j^{lm} w_{jk}^{(l+1)m} \mu_k^{(l+1)m} + \sum_{m=1}^{M} \sum_{i=1}^{V^m} a_i^m v_i^m \\
&+ \sum_{m=1}^{M} \sum_{j=1}^{H^{L_1 m}} \sum_{k=1}^{H^1} \mu_j^{L_1 m} w_{jk}^{(L_1+1)m} \mu_k^1 + \sum_{l=1}^{L_1} \sum_{m=1}^{M} \sum_{c=1}^{Y} \sum_{j=1}^{H^{lm}} y_c u_{cj}^{lm} \mu_j^{lm} + \sum_{l=1}^{L_2} \sum_{c=1}^{Y} \sum_{j=1}^{H^l} y_c u_{cj}^{l} \mu_j^{l} \\
&+ \sum_{l=1}^{L_2-1} \sum_{j=1}^{H^l} \sum_{k=1}^{H^{(l+1)}} \mu_j^l w_{jk}^l \mu_k^{(l+1)} + \sum_{l=1}^{L_1} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} b_j^{lm} \mu_j^{lm} + \sum_{l=1}^{L_2} \sum_{j=1}^{H^l} b_j^l \mu_j^l + \sum_{c=1}^{Y} d_c y_c
\end{aligned}$$

$$
\begin{aligned}
&+ \sum_{m=1}^{M} \lambda_m \left[ 1 - \sum_{j=1}^{H^{L_1 m}} \left\{ \mu_j^{L_1 m} - 2\mu_j^{L_1 m} \underline{h}_j^{L_1 m} + (\underline{h}_j^{L_1 m})^2 \right\} \right] - \ln Z \\
&+ \sum_{m=1}^{M-1} \sum_{r=(m+1)}^{M} \sum_{j=1}^{H^{L_1 m}} \gamma_{mr} \left\{ \mu_j^{L_1 m} - 2\mu_j^{L_1 m} \underline{h}_j^{L_1 m} + (\underline{h}_j^{L_1 m})^2 \right\} \left\{ \mu_j^{L_1 r} - 2\mu_j^{L_1 r} \underline{h}_j^{L_1 r} + (\underline{h}_j^{L_1 r})^2 \right\} \\
&+ 2 \sum_{m=1}^{M-1} \sum_{r=(m+1)}^{M} \sum_{j=1}^{H^{L_1 m}-1} \sum_{k=(j+1)}^{H^{L_1 m}} \gamma_{mr} (\mu_j^{L_1 m} - \underline{h}_j^{L_1 m})(\mu_j^{L_1 r} - \underline{h}_j^{L_1 r})(\mu_k^{L_1 m} - \underline{h}_k^{L_1 m})(\mu_k^{L_1 r} - \underline{h}_k^{L_1 r}) \\
&- \sum_{l=1}^{L_1} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} \left\{ \mu_j^{lm} \ln \mu_j^{lm} + (1 - \mu_j^{lm}) \ln(1 - \mu_j^{lm}) \right\} - \sum_{l=1}^{L_2} \sum_{j=1}^{H^l} \left\{ \mu_j^{l} \ln \mu_j^{l} + (1 - \mu_j^{l}) \ln(1 - \mu_j^{l}) \right\}.
\end{aligned} \tag{6.15}
$$

Since, the mean field parameters ($\boldsymbol{\mu}$) of $\mathcal{L}_v$, presented in (6.15) define the equilibrium state of the model, they need to be updated accordingly. In order to obtain the mean field parameters of the proposed model, the variational bound $\mathcal{L}_v$ of (6.15) is maximized with respect to $\boldsymbol{\mu}$ for a fixed parameter set $\boldsymbol{\theta}_{d2gda}$. Considering $H^{0m} = V^m$, $\boldsymbol{\mu}^{0m} = \mathbf{v}^m$, $H^0 = H^{L_1 m}$, $\mu_k^0 w_{kj}^1 = \sum_{m=1}^{M} \mu_k^{L_1 m} w_{kj}^{(L_1+1)m}$, and $\boldsymbol{\mu}^l = 0,\ \forall l > L_2$, the update rules for the nodes of the hidden layers, corresponding to the proposed D2GDA model are given by

$$
\begin{aligned}
\mu_j^{lm} &= \sigma\Bigg( \sum_{k=1}^{H^{(l-1)m}} \mu_k^{(l-1)m} w_{kj}^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m} \mu_k^{(l+1)m} + \sum_{c=1}^{Y} y_c u_{cj}^{lm} + b_j^{lm} \Bigg), \\
&\qquad \text{for } 1 \leqslant l < L_1 \text{ and } \forall j, m;
\end{aligned} \tag{6.16}
$$

$$
\begin{aligned}
\mu_j^{L_1 m} &= \sigma\Bigg( \sum_{k=1}^{H^{(L_1-1)m}} \mu_k^{(L_1-1)m} w_{kj}^{L_1 m} + \sum_{k=1}^{H^1} w_{jk}^{(L_1+1)m} \mu_k^1 + \sum_{c=1}^{Y} y_c u_{cj}^{L_1 m} + b_j^{L_1 m} \\
&- \lambda_m (1 - 2\underline{h}_j^{L_1 m}) + \sum_{r \neq m=1}^{M} \gamma_{mr} (1 - 2\underline{h}_j^{L_1 m}) \{ \mu_j^{L_1 r} - 2\mu_j^{L_1 r} \underline{h}_j^{L_1 r} + (\underline{h}_j^{L_1 r})^2 \} \\
&+ 2 \sum_{r \neq m=1}^{M} \gamma_{mr} \sum_{k \neq j=1}^{H^{L_1 m}} (\mu_j^{L_1 r} - \underline{h}_j^{L_1 r})(\mu_k^{L_1 m} - \underline{h}_k^{L_1 m})(\mu_k^{L_1 r} - \underline{h}_k^{L_1 r}) \Bigg), \ \forall j, m;
\end{aligned} \tag{6.17}
$$

$$
\mu_j^{l} = \sigma\Bigg( \sum_{k=1}^{H^{(l-1)}} \mu_k^{(l-1)} w_{kj}^{(l-1)} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^{l} \mu_k^{(l+1)} + \sum_{c=1}^{Y} y_c u_{cj}^{l} + b_j^{l} \Bigg), \ \text{for } 1 \leqslant l \leqslant L_2, \ \forall j; \tag{6.18}
$$

where $\sigma(x) = \dfrac{1}{1 + e^{-x}}$ is the sigmoid function. Given the equilibrium state of the model, the parameter set $\boldsymbol{\theta}_{d2gda}$ of the proposed architecture, corresponding to the data-dependent expectation, can be learned by maximizing $\mathcal{L}_v$ with respect to $\boldsymbol{\theta}_{d2gda}$ for the equilibrium mean field parameters $\boldsymbol{\mu}$, obtained using (6.16)-(6.18). The expressions for differentiation of $\mathcal{L}_v$ with respect to each of the parameters are given by

$$\frac{\partial \mathcal{L}_v}{\partial w_{ij}^{1m}} = v_i^m \mu_j^{1m};\ \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{lm}} = \mu_j^{(l-1)m}\mu_k^{lm},\ \text{for } 1 < l \leqslant L_1;\ \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{(L_1+1)m}} = \mu_j^{L_1 m}\mu_k^1;$$
$$\frac{\partial \mathcal{L}_v}{\partial w_{jk}^{l}} = \mu_j^{l}\mu_k^{(l+1)},\ \text{for } 1 \leqslant l < L_2;\ \frac{\partial \mathcal{L}_v}{\partial u_{cj}^{lm}} = y_c \mu_j^{lm},\ \text{for } 1 \leqslant l \leqslant L_1;$$
$$\frac{\partial \mathcal{L}_v}{\partial u_{cj}^{l}} = y_c \mu_j^{l},\ \text{for } 1 \leqslant l \leqslant L_2;\ \frac{\partial \mathcal{L}_v}{\partial a_i^m} = v_i^m;\ \frac{\partial \mathcal{L}_v}{\partial b_j^{lm}} = \mu_j^{lm},\ \text{for } 1 \leqslant l \leqslant L_1;\ \frac{\partial \mathcal{L}_v}{\partial d_c} = y_c;$$
$$\frac{\partial \mathcal{L}_v}{\partial b_j^{l}} = \mu_j^{l},\ \text{for } 1 \leqslant l \leqslant L_2;\ \frac{\partial \mathcal{L}_v}{\partial \lambda_m} = \left(1 - \sum_{j=1}^{H^{L_1 m}} \left\{\mu_j^{L_1 m} - 2\mu_j^{L_1 m}\underline{h}_j^{L_1 m} + (\underline{h}_j^{L_1 m})^2\right\}\right);$$
$$\frac{\partial \mathcal{L}_v}{\partial \gamma_{mr}} = \sum_{j=1}^{H^{L_1 m}} \left\{\mu_j^{L_1 m} - 2\mu_j^{L_1 m}\underline{h}_j^{L_1 m} + (\underline{h}_j^{L_1 m})^2\right\}\left\{\mu_j^{L_1 r} - 2\mu_j^{L_1 r}\underline{h}_j^{L_1 r} + (\underline{h}_j^{L_1 r})^2\right\}$$
$$+ 2\sum_{j=1}^{H^{L_1 m}-1}\sum_{k=(j+1)}^{H^{L_1 m}} (\mu_j^{L_1 m} - \underline{h}_j^{L_1 m})(\mu_j^{L_1 r} - \underline{h}_j^{L_1 r})(\mu_k^{L_1 m} - \underline{h}_k^{L_1 m})(\mu_k^{L_1 r} - \underline{h}_k^{L_1 r}) \quad \forall m, r. \tag{6.19}$$

Thus, the data-dependent expectations are estimated by the gradient ascent on the lower bound of the log-likelihood function. These expectations are employed in (6.27), discussed in Section 6.3.3.4, to update the parameters of the proposed architecture, which exhibits that the parameter updation rule follows Hebb's postulate. The Hebbian learning rule is not only considered to be one of the most acknowledged learning rules, but also compliments the corresponding psychological evidences.

#### 6.3.3.3 Estimation of Data-Independent Expectations

In order to obtain the gradient of log-likelihood function, the energy gradient with respect to the model distribution needs to be estimated. The Markov Chain Monte Carlo based stochastic approximation procedure [190] is considered to approximate the data-independent expectations. 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 $\boldsymbol{\theta}_{d2gda}$. Considering $\mathbf{h}^l = 0,\ \forall l > L_2$, $h_k^0 w_{kj}^0 = \sum_{m=1}^{M} h_k^{L_1 m} w_{kj}^{(L_1+1)m}$, and $H^{0m}$, $\mathbf{h}^{0m}$, and $H^0$ are defined as in Section 6.3.3.2, the conditional distributions corresponding to the proposed D2GDA model are given by

$$P(h_j^{lm}|\mathbf{h}^{(l-1)m}, \mathbf{h}^{(l+1)m}, \mathbf{y}) = \sigma\Bigg(\sum_{k=1}^{H^{(l-1)m}} h_k^{(l-1)m} w_{kj}^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m} h_k^{(l+1)m} + \sum_{c=1}^{Y} y_c u_{cj}^{lm} + b_j^{lm}\Bigg) \text{ for } 1 \leqslant l < L_1, \forall j, m; \tag{6.20}$$

$$P(h_j^{L_1 m}|\mathbf{h}^{(L_1-1)m}, \mathbf{h}^1, \mathbf{h}_{-j}^{L_1 m}, \mathbf{h}^{L_1 r}, \mathbf{y}) = \sigma\Bigg( \sum_{k=1}^{H^{(L_1-1)m}} h_k^{(L_1-1)m} w_{kj}^{L_1 m} + \sum_{k=1}^{H^1} w_{jk}^{(L_1+1)m} h_k^1 + \sum_{c=1}^{Y} y_c u_{cj}^{L_1 m} + b_j^{L_1 m} - \lambda_m (1 - 2\underline{h}_j^{L_1 m}) + \sum_{r \neq m=1}^{M} \gamma_{mr} (1 - 2\underline{h}_j^{L_1 m}) \Big( h_j^{L_1 r} - \underline{h}_j^{L_1 r} \Big)^2 + 2 \sum_{r \neq m=1}^{M} \sum_{k \neq j=1}^{H^{L_1 m}} \gamma_{mr} (h_j^{L_1 r} - \underline{h}_j^{L_1 r})(h_k^{L_1 m} - \underline{h}_k^{L_1 m})(h_k^{L_1 r} - \underline{h}_k^{L_1 r}) \Bigg), \ \forall j, m; \tag{6.21}$$

where $\mathbf{h}_{-j}^{L_1 m}$ represents modality-specific hidden representation, corresponding to $L_1$-th layer of $m$-th modality, consisting values of all the nodes except $h_j^{L_1 m}$. Similarly, the following update rules can be obtained.

$$P(h_j^l|\mathbf{h}^{(l-1)}, \mathbf{h}^{(l+1)}, \mathbf{y}) = \sigma\Bigg( \sum_{k=1}^{H^{(l-1)}} h_k^{(l-1)} w_{kj}^{(l-1)} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^l h_k^{(l+1)} + \sum_{c=1}^{Y} y_c u_{cj}^l + b_j^l \Bigg), \quad \text{for } 1 \leqslant l \leqslant L_2, \ \forall j; \tag{6.22}$$

$$P(v_i^m|\mathbf{h}^{1m}) = \sigma\left( \sum_{j=1}^{H^{1m}} w_{ij}^{1m} h_j^{1m} + a_i^m \right), \ \forall i, m; \tag{6.23}$$

$$P(y_c|\mathbf{h}^{11}, \cdots, \mathbf{h}^{L_1 M}, \mathbf{h}^1, \cdots, \mathbf{h}^{L_2}) = \frac{e^{X_c}}{\sum_{\bar{c}=1}^{Y} e^{X_{\bar{c}}}};$$

$$\text{where} \quad X_c = \sum_{l=1}^{L_c} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} u_{cj}^{lm} h_j^{lm} + \sum_{l=1}^{L_2} \sum_{j=1}^{H^l} u_{cj}^l h_j^l + d_c, \ \forall c. \tag{6.24}$$

Given that the convergence criteria, discussed in Section 6.4.2, are satisfied, if a Markov chain is run for sufficient number of steps, then it can be ensured that the chain will converge to an 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 (6.20)-(6.24). Thus, 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 superscript tilde (e.g., $\tilde{v}$):

$$\frac{\partial E_{d2gda}}{\partial w_{ij}^{1m}} = \tilde{v}_i^m \tilde{h}_j^{1m}; \ \frac{\partial E_{d2gda}}{\partial w_{jk}^{lm}} = \tilde{h}_j^{(l-1)m} \tilde{h}_k^{lm}, \ \text{for } 1 < l \leqslant L_1; \ \frac{\partial E_{d2gda}}{\partial w_{jk}^{(L_1+1)m}} = \tilde{h}_j^{L_1 m} \tilde{h}_k^1;$$

$$\frac{\partial E_{d2gda}}{\partial w_{jk}^l} = \tilde{h}_j^l \tilde{h}_k^{(l+1)}, \ \text{for } 1 \leqslant l < L_2; \ \frac{\partial E_{d2gda}}{\partial u_{cj}^{lm}} = \tilde{y}_c \tilde{h}_j^{lm}, \ \text{for } 1 \leqslant l \leqslant L_1; \ \frac{\partial E_{d2gda}}{\partial a_i^m} = \tilde{v}_i^m;$$

$$\frac{\partial E_{d2gda}}{\partial u_{cj}^l} = \tilde{y}_c \tilde{h}_j^l, \ \text{for } 1 \leqslant l \leqslant L_2; \ \frac{\partial E_{d2gda}}{\partial b_j^{lm}} = \tilde{h}_j^{lm}, \ \text{for } 1 \leqslant l \leqslant L_1; \ \frac{\partial E_{d2gda}}{\partial d_c} = \tilde{y}_c;$$

$$\frac{\partial E_{d2gda}}{\partial b_j^l} = \tilde{h}_j^l, \text{ for } 1 \leqslant l \leqslant L_2; \frac{\partial E_{d2gda}}{\partial \lambda_m} = \left\{ 1 - \sum_{j=1}^{H^{L_1 m}} \left( \tilde{h}_j^{L_1 m} - \underline{\tilde{h}}_j^{L_1 m} \right)^2 \right\}; \text{ and}$$

$$\frac{\partial E_{d2gda}}{\partial \gamma_{mr}} = \left\{ \sum_{j=1}^{H^{L_1 m}} (\tilde{h}_j^{L_1 m} - \underline{\tilde{h}}_j^{L_1 m})(\tilde{h}_j^{L_1 r} - \underline{\tilde{h}}_j^{L_1 r}) \right\}^2 \forall m, r. \tag{6.25}$$

The data-independent estimates, obtained in (6.25), are considered in (6.27) for learning the parameter values of the proposed model. As in case of data-dependent estimates, similar observation can be noticed for data-independent estimates as well, where the learning rule for the parameters of the architecture follows Hebb's postulates. Hence, the proposed model can efficiently learn both the data-dependent and data-independent estimates, using (6.19) and (6.25), respectively.

#### 6.3.3.4 Learning Rule of D2GDA Model Parameters

Let us assume that $t$, $\eta$, $N$, $S$, $\rho$, and $\zeta$ represent the current epoch, learning rate, number of training observations, number of persistent Markov chains, weight decay, and momentum constant, respectively. Thus, the learning rule for different parameters of the proposed D2GDA model, required to perform gradient ascent on the log-likelihood function corresponding to the energy function of (6.8), can be defined as:

$$\boldsymbol{\theta}_{d2gda}^{(t+1)} = F\big(\boldsymbol{\theta}_{d2gda}^{t} + \Delta\boldsymbol{\theta}_{d2gda}^{t}\big); \tag{6.26}$$

$$\text{where } \Delta\boldsymbol{\theta}_{d2gda}^{t} = \eta \left\{ \frac{1}{N} \sum_{n=1}^{N} \left( \frac{\partial \mathcal{L}_v}{\partial \boldsymbol{\theta}_{d2gda}} \right)_n - \frac{1}{S} \sum_{s=1}^{S} \left( \frac{\partial E_{d2gda}}{\partial \boldsymbol{\theta}_{d2gda}} \right)_s \right\} - \rho \boldsymbol{\theta}_{d2gda}^{t} + \zeta \Delta\boldsymbol{\theta}_{d2gda}^{(t-1)}, \tag{6.27}$$

and $F(\cdot)$ denotes hyperbolic tangent function in case of balance parameter $\gamma_{mr}$, and identity function for the rest of the parameters belonging to $\boldsymbol{\theta}_{d2gda}$. Thus, it can be noted here that in the proposed D2GDA model, the values of $\gamma_{mr}$ are learned in such a way that $\gamma_{mr} \in [-1, 1]$, and hence, Property 6.4, presented in Section 6.3.1, is satisfied. It can also be observed that all the parameters of the model are learned simultaneously using (6.26), and the modification in the parameter values of the D2GDA model at any epoch depends on the values of pre-synaptic and post-synaptic nodes of the parameter, which is in accordance with the Hebbian learning rule. Also, subtracting the data-independent expectations from the corresponding data-dependent terms in (6.26) basically stabilizes the distribution of parameters as well as allows the proposed model to propagate uncertainties associated with ambiguous inputs. The learning algorithm of the proposed model is illustrated in Algorithm 6.1.

## 6.4 Different Aspects of Proposed Model

In this section, different aspects of the proposed model are studied, which include error analysis, convergence criterion, and generalization ability of the D2GDA model. The de-

**Algorithm 6.1** Learning of Proposed D2GDA Model.

**Input:** Set of $N$ training data vectors $\{\mathbf{v}^m\}_{n=1}^N$ for $M$ modalities along with the corresponding set of class labels $\{\mathbf{y}\}_{n=1}^N$, number of persistent chains ($S$), number of epochs ($\tau$), learning rate ($\eta$), weight decay ($\rho$), momentum ($\zeta$), and number of Gibbs steps ($\alpha$).
**Output:** Final parameter set $\boldsymbol{\theta}_{d2gda}^{\tau}$ of the architecture.

1: Perform greedy layer-wise pretraining to initialize the set of model parameters, $\boldsymbol{\theta}_{d2gda}^{0}$.
2: Randomly initialize $S$ Markov chains $\{\tilde{\mathbf{v}}^{m^0}, \tilde{\mathbf{y}}^0, \tilde{\mathbf{h}}^{1m^0}, \cdots, \tilde{\mathbf{h}}^{L_0m^0}, \tilde{\mathbf{h}}^{(L_0+1)^0}, \cdots, \tilde{\mathbf{h}}^{L^0}\}_{s=1}^S$.
3: **for** each epoch $t = 0$ to $\tau$ **do**
4: // Variational inference
5: **for** each training sample $n = 1$ to $N$ **do**
6: (i) Run mean field updates using (6.16)-(6.18) until convergence.
7: (ii) Save the obtained mean field parameter ($\boldsymbol{\mu}$) for the corresponding training sample, $\boldsymbol{\mu}_n = \boldsymbol{\mu}$.
8: **end for**
9: // Stochastic approximation
10: **for** each persistent chain $s = 1$ to $S$ **do**
11: Run the chain for $\alpha$-steps and sample the state $\{\tilde{\mathbf{v}}^{m^{t+1}}, \tilde{\mathbf{y}}^{t+1}, \tilde{\mathbf{h}}^{1m^{t+1}}, \cdots, \tilde{\mathbf{h}}^{L_0m^{t+1}}, \tilde{\mathbf{h}}^{(L_0+1)^{t+1}}, \cdots, \tilde{\mathbf{h}}^{L^{t+1}}\}$ from $\{\tilde{\mathbf{v}}^{m^t}, \tilde{\mathbf{y}}^t, \tilde{\mathbf{h}}^{1m^t}, \cdots, \tilde{\mathbf{h}}^{L_0m^t}, \tilde{\mathbf{h}}^{(L_0+1)^t}, \cdots, \tilde{\mathbf{h}}^{L^t}\}$ using (6.20)-(6.24).
12: **end for**
13: Update the parameters of the model from $\boldsymbol{\theta}_{d2gda}^{t}$ to $\boldsymbol{\theta}_{d2gda}^{(t+1)}$ using (6.26).
14: **end for**

tailed analysis of each of these aspects is discussed next.

### 6.4.1 Error Analysis of Proposed Model

In the current study, the D2GDA model is developed to classify the observations of the given multi-view data into different categories. Now, it is well known that Bayes discriminant function provides the optimal solution to any classification problem. In order to analyze the discriminative ability of the proposed framework, the mean-squared error between the prediction rule (6.24) of the D2GDA model and Bayes decision rule is studied. Thus, reduction in the corresponding error will indicate better approximation of Bayes discriminant function by the proposed model, which in turn, will ensure better discriminative ability of the model.

Let, $\mathbf{v}$ be the input visible vector, $\omega_c$ represents the $c$-th input class, where $C$ represents the total number of classes, that is, $C = Y$, $\Omega_c$ denotes the set of all possible visible vectors, and $\Omega = \bigcup_{c=1}^{C} \Omega_c$ signifies the input space. In the proposed D2GDA model, the class label for the input vector $\mathbf{v}$ is predicted as $\arg\max_c P(y_c = 1|\mathbf{v})$, where $P(y_c = 1|\mathbf{v})$ is obtained using (6.24). Now, the Bayes optimal discriminant functions are given by $g_c(\mathbf{v}) = P(\omega_c|\mathbf{v})$, $\forall c \in \{1, 2, \cdots, C\}$, and the corresponding decision rule is $\mathbf{v} \in \omega_c$ if $g_c(\mathbf{v}) \geqslant g_k(\mathbf{v})$, $\forall k \neq c$. This decision rule is termed as minimum error decision rule as it provides the minimum probability of error.

In order to establish that the prediction criterion (6.24) of the proposed architecture approximates the Bayes decision rule, it is to be demonstrated that the following error criterion is minimized through learning of the architecture:

$$\epsilon^2(\boldsymbol{\theta}_{d2gda}) = \sum_{c=1}^{C} \int_{\Omega} \Big[ \ln P(y_c = 1|\mathbf{v}) - g_c(\mathbf{v}) \Big]^2 p(\mathbf{v}) d\mathbf{v}. \tag{6.28}$$

In the proposed framework, learning of the model (6.9) refers to estimating the parameter values for which the probability of observing the given samples is maximized, that is,

$$\max_{\boldsymbol{\theta}_{d2gda}} \left\{ \frac{1}{N} \sum_{\mathbf{v},\mathbf{y}} \ln P(\mathbf{v},\mathbf{y}) \right\} = \max_{\boldsymbol{\theta}_{d2gda}} \left\{ \frac{N_1}{N} \frac{1}{N_1} \sum_{\mathbf{v}\in\Omega_1} \ln P(\mathbf{v}, y_1 = 1) + \cdots + \frac{N_C}{N} \frac{1}{N_C} \sum_{\mathbf{v}\in\Omega_C} \ln P(\mathbf{v}, y_C = 1) \right\} \tag{6.29}$$

where $N_c$ denotes the number of training observations corresponding to the class $\omega_c$. Now, the number of feature vectors drawn from $p(\mathbf{v})$ for any given class is proportional to the a priori probability of that class. Considering a class with non-zero probability of occurrence, $N \to \infty$ will imply $N_c \to \infty$. So, by strong law of large numbers, (6.29) can be written as

$$\begin{aligned}
&\max_{\boldsymbol{\theta}_{d2gda}} \left\{ \frac{1}{N} \sum_{\mathbf{v},\mathbf{y}} \ln P(\mathbf{v},\mathbf{y}) \right\} = \max_{\boldsymbol{\theta}_{d2gda}} \int_{\Omega} \Big\{ P(\omega_1) p(\mathbf{v}|\omega_1) \ln P(\mathbf{v}, y_1 = 1) + \cdots \\
&\qquad + P(\omega_C) p(\mathbf{v}|\omega_C) \ln P(\mathbf{v}, y_C = 1) \Big\} d\mathbf{v} \\
&= \max_{\boldsymbol{\theta}_{d2gda}} \sum_{c=1}^{C} \int_{\Omega} P(\omega_c) p(\mathbf{v}|\omega_c) \ln P(\mathbf{v}, y_c = 1) d\mathbf{v} \\
&= \max_{\boldsymbol{\theta}_{d2gda}} \left\{ \sum_{c=1}^{C} \int_{\Omega} P(\omega_c) p(\mathbf{v}|\omega_c) \ln P(y_c = 1|\mathbf{v}) d\mathbf{v} + \sum_{c=1}^{C} \int_{\Omega} P(\omega_c) p(\mathbf{v}|\omega_c) \ln p(\mathbf{v}) d\mathbf{v} \right\} \\
&= \max_{\boldsymbol{\theta}_{d2gda}} \left\{ \sum_{c=1}^{C} \int_{\Omega} p(\mathbf{v}) P(\omega_c|\mathbf{v}) \ln P(y_c = 1|\mathbf{v}) d\mathbf{v} \right\} + \int_{\Omega} p(\mathbf{v}) \ln p(\mathbf{v}) d\mathbf{v} \\
&= \max_{\boldsymbol{\theta}_{d2gda}} \left\{ \sum_{c=1}^{C} \int_{\Omega} p(\mathbf{v}) g_c(\mathbf{v}) \ln P(y_c = 1|\mathbf{v}) d\mathbf{v} \right\} + \int_{\Omega} p(\mathbf{v}) \ln p(\mathbf{v}) d\mathbf{v} \\
&= \min_{\boldsymbol{\theta}_{d2gda}} \left\{ \sum_{c=1}^{C} \int_{\Omega} \Big[ \big\{ \ln P(y_c = 1|\mathbf{v}) \big\}^2 - 2 g_c(\mathbf{v}) \ln P(y_c = 1|\mathbf{v}) + \big\{ g_c(\mathbf{v}) \big\}^2 \Big] p(\mathbf{v}) d\mathbf{v} \right. \\
&\left. - \sum_{c=1}^{C} \int_{\Omega} \big\{ \ln P(y_c = 1|\mathbf{v}) \big\}^2 p(\mathbf{v}) d\mathbf{v} \right\} - \sum_{c=1}^{C} \int_{\Omega} \big\{ g_c(\mathbf{v}) \big\}^2 p(\mathbf{v}) d\mathbf{v} + \int_{\Omega} p(\mathbf{v}) \ln p(\mathbf{v}) d\mathbf{v}
\end{aligned}$$

$$= \min_{\boldsymbol{\theta}_{d2gda}} \left\{ \epsilon^2(\boldsymbol{\theta}_{d2gda}) - \sum_{c=1}^{C} \int_{\Omega} \left\{ \ln P(y_c = 1|\mathbf{v}) \right\}^2 p(\mathbf{v}) d\mathbf{v} \right\} - \sum_{c=1}^{C} \int_{\Omega} \left\{ g_c(\mathbf{v}) \right\}^2 p(\mathbf{v}) d\mathbf{v} + \int_{\Omega} p(\mathbf{v}) \ln p(\mathbf{v}) d\mathbf{v}. \tag{6.30}$$

Hence, it can be observed that learning of the proposed architecture with prediction criterion (6.24) attempts to provide a classifier which is mean-squared error approximation to the Bayes optimal classifier. So, minimization of the mean-squared error depends on the efficient learning of the model parameters, which in turn, depends on the model architecture. Thus, by modifying the architecture of the model or parameter values, the discriminative ability of the model can be varied. Hence, there must exist a relation between the model architecture and the values of the parameters with the error probability of the proposed D2GDA model.

Because of the resemblance of prediction criterion between the proposed method and Bayes decision rule, the error probability of the D2GDA model can be defined in accordance with the Bayes multi-class classifier [169], which is given by

$$\begin{aligned} P_e &= \mathbb{E}[P(e|\mathbf{v})] = \mathbb{E}[1 - \max_c P(y_c = 1|\mathbf{v})] \\ &\leqslant 2 \sum_{\mathbf{v} \in \Omega} p(\mathbf{v}) \sum_{c=1}^{C-1} \sum_{k=(c+1)}^{C} P(y_c = 1|\mathbf{v}) P(y_k = 1|\mathbf{v}). \end{aligned} \tag{6.31}$$

In the current study, $P(y_c = 1|\mathbf{v})$ is defined as conditional distribution in (6.24), which can be replaced in (6.31) and the corresponding upper bound on the error probability of the D2GDA model can be obtained as

$$\begin{aligned} P_e &\leqslant \sum_{\mathbf{v} \in \Omega} p(\mathbf{v}) \left\{ 2 \sum_{c=1}^{C-1} \sum_{k=(c+1)}^{C} \frac{e^{X_c}}{\sum_{\bar{c}=1}^{C} e^{X_{\bar{c}}}} \frac{e^{X_k}}{\sum_{\bar{k}=1}^{C} e^{X_{\bar{k}}}} \right\} \\ \Rightarrow P_e &\leqslant \sum_{\mathbf{v} \in \Omega} p(\mathbf{v}) \left\{ \frac{\sum_{c=1}^{C} e^{2X_c} + 2 \sum_{c=1}^{C-1} \sum_{k=(c+1)}^{C} e^{X_c} e^{X_k} - \sum_{c=1}^{C} e^{2X_c}}{\left( \sum_{\bar{c}=1}^{C} e^{X_{\bar{c}}} \right)^2} \right\} \\ \Rightarrow P_e &\leqslant \sum_{\mathbf{v} \in \Omega} p(\mathbf{v}) \left\{ 1 - \frac{\sum_{c=1}^{C} \left( e^{X_c} \right)^2}{\left( \sum_{\bar{c}=1}^{C} e^{X_{\bar{c}}} \right)^2} \right\} \end{aligned} \tag{6.32}$$

$$\text{where } X_c = \sum_{l=1}^{L_1} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} u_{cj}^{lm} h_j^{lm} + \sum_{l=1}^{L_2} \sum_{j=1}^{H^l} u_{cj}^{l} h_j^{l} + d_c. \tag{6.33}$$

So, an upper bound of the error probability $P_e$ is achieved in terms of $X_c$, which depends on both architecture as well as parameters of the model. Through proper learning of the

model parameters, the upper bound on the error probability can be minimized. Also, by suitably varying the model architecture, a tighter bound on $P_e$ can be achieved.

Let, $u_c^1 = \min_{j,l,m}\{u_{cj}^{lm}\}$, $h^1 = \min_{j,l,m}\{h_j^{lm}\}$, $u_c^2 = \min_{j,l}\{u_{cj}^{l}\}$, $h^2 = \min_{j,l}\{h_j^{l}\}$, $H^1 = \min_{l,m}\{H^{lm}\}$, and $H^2 = \min_{l}\{H^{l}\}$. So,

$$X_c \leqslant L_1 M H^1 u_c^1 h^1 + L_2 H^2 u_c^2 h^2. \tag{6.34}$$

If the value of $X_c$ in (6.33) is substituted with the formulation of (6.34), the inequality will still hold, which is given by

$$P_e \leqslant \sum_{\mathbf{v}\in\Omega} p(\mathbf{v}) \left\{ 1 - \frac{\sum_{c=1}^{C} e^{2L_1 M H^1 u_c^1 h^1 + 2L_2 H^2 u_c^2 h^2}}{\left( \sum_{c=1}^{C} e^{L_1 M H^1 u_c^1 h^1 + L_2 H^2 u_c^2 h^2} \right)^2} \right\}. \tag{6.35}$$

It can be observed from (6.35) that instead of heuristically determining the architecture of the proposed framework, an optimal deep architecture can be obtained for the analysis of the given multi-view data. Apart from the model parameters and architecture of the proposed D2GDA framework, the error probability also depends on the nature and complexity of the given classification problem.

### 6.4.2 Convergence Analysis

In the proposed D2GDA model, variational learning is employed to estimate the data-dependent expectations inSection 6.3.3.2. It provides a lower bound $\mathcal{L}_v : \mathbb{R}^{|\boldsymbol{\theta}_{d2gda}|} \mapsto \mathbb{R}$ on the log-likelihood function (6.15) of the proposed model. Given an equilibrium state, the parameter set $\boldsymbol{\theta}_{d2gda}$ 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 (6.8), can be expressed as

$$\begin{aligned}
\nabla\mathcal{L}_v(\theta_{d2gda}) = \Bigg[ & \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial w_{ij}^{1m}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial w_{jk}^{(L_1+1)m}} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial w_{jk}^{1}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial w_{jk}^{(L_2-1)}} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial u_{cj}^{1m}} \\
& \cdots \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial u_{cj}^{L_1 m}} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial u_{cj}^{1}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial u_{cj}^{L_2}} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial a_i^m} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial b_j^{1m}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial b_j^{L_1 m}} \\
& \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial b_j^{1}} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial b_j^{L_2}} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial d_c} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial \lambda_m} \frac{\partial\mathcal{L}_v(\theta_{d2gda})}{\partial \gamma_{mr}} \Bigg]^T ; \forall i,j,k,c,m,r,
\end{aligned} \tag{6.36}$$

where $T$ denotes the transpose operator. The gradient of $\mathcal{L}_v(\theta_{d2gda})$ with respect to each of the parameters can be obtained using (6.19), which makes it clear that the gradient function $\nabla\mathcal{L}_v(\theta_{d2gda})$ is independent of $\theta_{d2gda}$, that is, $\nabla\mathcal{L}_v(\theta_{d2gda\_1}) = \nabla\mathcal{L}_v(\theta_{d2gda\_2})$, $\forall\theta_{d2gda\_1}$, $\theta_{d2gda\_2} \in \boldsymbol{\theta}_{d2gda}$. The independence of $\nabla\mathcal{L}_v(\theta_{d2gda})$ with respect to the parameters of D2GDA model is evident since each parameter of the model is included in the corre-

sponding energy function $E_{d2gda}(\mathbf{v}, \mathbf{h}, \mathbf{y})$ as an additive term. So, it can be said that $\mathcal{L}_v$ is a differential function having $\beta$-Lipschitz continuous gradient for some $\beta \geqslant 0$, that is, $\|\nabla\mathcal{L}_v(\theta_{d2gda\_1}) - \nabla\mathcal{L}_v(\theta_{d2gda\_2})\|_2 \leqslant \beta\|\theta_{d2gda\_1} - \theta_{d2gda\_2}\|_2$. For a function having $\beta$-Lipschitz gradient, it is known that $\forall\theta_{d2gda\_1}, \theta_{d2gda\_2} \in \boldsymbol{\theta}_{d2gda}$,

$$\begin{aligned}\mathcal{L}_v(\theta_{d2gda\_1}) \quad \leqslant \mathcal{L}_v(\theta_{d2gda\_2}) + \nabla\mathcal{L}_v(\theta_{d2gda\_2})^T(\theta_{d2gda\_1} - \theta_{d2gda\_2}) \\ + \frac{1}{2}\beta\|\theta_{d2gda\_1} - \theta_{d2gda\_2}\|_2^2. \end{aligned} \tag{6.37}$$

Let, $\theta_{d2gda\_1} = \theta^t_{d2gda}$ and $\theta_{d2gda\_2} = \theta^{(t+1)}_{d2gda}$, where $t$ denotes the current epoch. Now, substituting the values of $\theta_{d2gda\_1}$ and $\theta_{d2gda\_2}$, and rearranging the terms in (6.37), we get

$$\mathcal{L}_v(\theta^{(t+1)}_{d2gda}) \geqslant \mathcal{L}_v(\theta^t_{d2gda}) + \nabla\mathcal{L}_v(\theta^{(t+1)}_{d2gda})^T(\theta^{(t+1)}_{d2gda} - \theta^t_{d2gda}) - \frac{1}{2}\beta\|\theta^{(t+1)}_{d2gda} - \theta^t_{d2gda}\|_2^2.$$

Since gradient ascent algorithm on $\mathcal{L}_v$ is employed in the proposed model to learn the parameter values of $\boldsymbol{\theta}_{d2gda}$, it is obvious that $\theta^{(t+1)}_{d2gda} = \theta^t_{d2gda} + \eta\nabla\mathcal{L}_v(\theta^t_{d2gda})$, where $\eta$ denotes the learning rate. Now, considering the independence property of $\nabla\mathcal{L}_v(\theta_{d2gda})$ in the proposed model, that is, $\nabla\mathcal{L}_v(\theta^{(t+1)}_{d2gda}) = \nabla\mathcal{L}_v(\theta^t_{d2gda})$, we have

$$\mathcal{L}_v(\theta^{(t+1)}_{d2gda}) \geqslant \mathcal{L}_v(\theta^t_{d2gda}) + \eta(1 - \frac{1}{2}\beta\eta)\|\nabla\mathcal{L}_v(\theta^t_{d2gda})\|_2^2.$$

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

$$\mathcal{L}_v(\theta^{(t+1)}_{d2gda}) \geqslant \mathcal{L}_v(\theta^t_{d2gda}) + \frac{1}{2}\eta\|\nabla\mathcal{L}_v(\theta^t_{d2gda})\|_2^2. \tag{6.38}$$

Let $\theta^*_{d2gda}$ be the optimal parameter set that maximizes the lower bound, or equivalently maximizes the log-likelihood function of the proposed model in such a way that, $\mathcal{L}_v(\theta^*_{d2gda}) \geqslant \mathcal{L}_v(\theta_{d2gda}), \forall\theta_{d2gda} \in \boldsymbol{\theta}_{d2gda}$. Now, since $\mathcal{L}_v(\theta_{d2gda})$ is defined as a linear function of $\theta_{d2gda}$ in (6.15), we have

$$\mathcal{L}_v(\theta^t_{d2gda}) = \mathcal{L}_v(\theta^*_{d2gda}) + \nabla\mathcal{L}_v(\theta^t_{d2gda})^T(\theta^t_{d2gda} - \theta^*_{d2gda}). \tag{6.39}$$

Using (6.39) in (6.38), we get

$$\begin{aligned}&\mathcal{L}_v(\theta^*_{d2gda}) - \mathcal{L}_v(\theta^{(t+1)}_{d2gda}) \leqslant -\nabla\mathcal{L}_v(\theta^t_{d2gda})^T(\theta^t_{d2gda} - \theta^*_{d2gda}) - \frac{1}{2}\eta\|\nabla\mathcal{L}_v(\theta^t_{d2gda})\|_2^2 \\ &\Rightarrow \quad \mathcal{L}_v(\theta^*_{d2gda}) - \mathcal{L}_v(\theta^{(t+1)}_{d2gda}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^t_{d2gda} - \theta^*_{d2gda}\|_2^2 - \|\theta^t_{d2gda} - \theta^*_{d2gda}\|_2^2 - \|\eta\nabla\mathcal{L}_v(\theta^t_{d2gda})\|_2^2 \\ &\qquad\qquad - 2\eta\nabla\mathcal{L}_v(\theta^t_{d2gda})^T(\theta^t_{d2gda} - \theta^*_{d2gda})\Big\} \\ &\Rightarrow \quad \mathcal{L}_v(\theta^*_{d2gda}) - \mathcal{L}_v(\theta^{(t+1)}_{d2gda}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^t_{d2gda} - \theta^*_{d2gda}\|_2^2 - \|\theta^{t+1}_{d2gda} - \theta^*_{d2gda}\|_2^2\Big\}.\end{aligned}$$

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

$$\sum_{t=0}^{T}\Big\{L_v(\theta^*_{d2gda}) - L_v(\theta^{t+1}_{d2gda})\Big\} \leqslant \frac{1}{2\eta}\sum_{t=0}^{T}\Big\{\|\theta^t_{d2gda} - \theta^*_{d2gda}\|_2^2 - \|\theta^{t+1}_{d2gda} - \theta^*_{d2gda}\|_2^2\Big\}$$

$$\Rightarrow\ \tau\mathcal{L}_v(\theta^*_{d2gda}) - \sum_{t=0}^{\tau-1}\mathcal{L}_v(\theta^{t+1}_{d2gda}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^0_{d2gda} - \theta^*_{d2gda}\|_2^2 - \|\theta^\tau_{d2gda} - \theta^*_{d2gda}\|_2^2\Big\}.$$

Since $\mathcal{L}_v(\theta_{d2gda})$ is an increasing function of $\theta_{d2gda}$, we can replace $\sum_{t=0}^{\tau-1}\mathcal{L}_v(\theta^{(t+1)}_{d2gda})$ with $\tau\mathcal{L}_v(\theta^\tau_{d2gda})$ and the inequality will still hold. Thus, we get

$$\begin{aligned}\mathcal{L}_v(\theta^*_{d2gda}) - \mathcal{L}_v(\theta^\tau_{d2gda}) &\leqslant \frac{1}{2\eta\tau}\Big\{\|\theta^0_{d2gda} - \theta^*_{d2gda}\|_2^2 - \|\theta^\tau_{d2gda} - \theta^*_{d2gda}\|_2^2\Big\}\\ &\leqslant \frac{1}{2\eta\tau}\|\theta^0_{d2gda} - \theta^*_{d2gda}\|_2^2. \end{aligned} \tag{6.40}$$

From (6.40), it can be concluded that the variational learning algorithm in the proposed D2GDA model converges with a rate $\mathcal{O}(\frac{1}{\tau})$ after $\tau$ iterations, if the learning rate is considered to be small enough, that is, $\eta \leqslant \frac{1}{\beta}$.

Now, 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 [218, 220]. One necessary condition requires the learning rate ($\eta$) 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 $\eta_t = \frac{a}{b+t}$, for constants $a > 0$ and $b > 0$. Also, in practice, the sequence $|\theta^t_{d2gda}|$ is bounded and the Markov chain is ergodic which, along with the condition on learning rate, establish the convergence of stochastic approximation procedure. Together with the condition on the variational learning (6.40), this ensures the convergence of the proposed D2GDA model.

### 6.4.3 Generalization Ability of Proposed Model

In this section, various state-of-the-art approaches, such as canonical correlation analysis (CCA) [78], generalized multi-view principal component analysis (GMPCA) [171], and partial least squares (PLS) [207], are demonstrated as special cases of the energy function $E_{gda}$, proposed in (6.8).

#### 6.4.3.1 Canonical Correlation Analysis

In CCA [78], the main objective is to maximize the correlation between each pair of given input modalities. Let, $\gamma_{mr} = 1$ and $\underline{\mathbf{h}}^{L_1 m} = 0,\ \forall_{m,r=1}^{M}$. So, the energy function $E_{gda}(\mathbf{h})$,

presented in (6.8), reduces to

$$E_{cca}(\mathbf{h}) = -\sum_{m=1}^{M-1}\sum_{r=(m+1)}^{M}\left\{\sum_{j=1}^{H^{L_1m}} h_j^{L_1m}h_j^{L_1r}\right\}^2 - \sum_{m=1}^{M}\lambda_m\left(1-\sum_{j=1}^{H^{L_1m}}(h_j^{L_1m})^2\right). \quad (6.41)$$

Here, the first term $\sum_{j=1}^{H^{L_1m}} h_j^{L_1m}h_j^{L_1r}$ corresponds to the trace of covariance between $\mathbf{h}^{L_1m}$ and $\mathbf{h}^{L_1r}$, and the second term represents the constraint that the variance of $\mathbf{h}^{L_1m}$ is equal to 1. So, in order to minimize the energy function $E_{cca}(\mathbf{h})$ in (6.41), $\mathbf{tr}(\text{cov}(\mathbf{h}^{L_1m}, \mathbf{h}^{L_1r}))$ is to be maximized subject to the constraint that $\text{var}(\mathbf{h}^{L_1m}) = 1$. So, the energy function $E_{cca}(\mathbf{h})$ in (6.41) is essentially the Lagrangian of the CCA. Now, if $E_{cca}(\mathbf{h})$, obtained in (6.41), is considered in the energy function $E_{d2gda}(\mathbf{v}, \mathbf{h}, \mathbf{y})$ of (6.8), then the joint representation of the MDDBM architecture will be learned from maximally correlated modality-specific subspaces, and hence, the corresponding model is referred to as MDDBM_CCA. For efficient learning of the parameters of the MDDBM_CCA model, the first term of variational lower bound $\mathcal{L}_v$, the update rule for $\mu_j^{L_1m}$, and the conditional distribution $P(h_j^{L_1m}|\mathbf{h}^{(L_1-1)m}, \mathbf{h}^1, \mathbf{h}_{-j}^{L_1m}, \mathbf{h}^{L_1r}, \mathbf{y})$ are required to be modified by suitably replacing the values of $\gamma_{mr}$ and $\underline{\mathbf{h}}^{L_1m}$ in (6.15), (6.17), and (6.21), respectively.

#### 6.4.3.2 Generalized Multi-View Principal Component Analysis

The GMPCA [171] aims to determine the direction where the variance of each given modality as well as the covariance between every pair of modalities are maximized. Suppose, $\gamma_{mr} = 1$, $\lambda_m = -1$, and $\underline{\mathbf{h}}^{L_1m} = 0$, $\forall m, r$. In this case, the energy function $E_{gda}(\mathbf{h})$ of (6.8) is reduced to

$$E_{gmpca}(\mathbf{h}) = -\sum_{m=1}^{M-1}\sum_{r=(m+1)}^{M}\left\{\sum_{j=1}^{H^{L_1m}} h_j^{L_1m}h_j^{L_1r}\right\}^2 - \sum_{m=1}^{M}\sum_{j=1}^{H^{L_1m}}(h_j^{L_1m})^2. \quad (6.42)$$

From (6.42), it can be observed that the first term represents the squared value of trace of covariance between every pair of modality-specific hidden representations, while the second term denotes the variance of $\mathbf{h}^{L_1m}$, $\forall m$. So, in order to detect the optimal minima in the given energy landscape, both the variance and covariance terms need to be maximized, which is primarily the objective of the GMPCA. The deep framework obtained by considering $E_{gmpca}(\mathbf{h})$ of (6.42) in the energy function of (6.8) is termed as MDDBM_GMPCA. The update rule for the model parameters can be determined by substituting $\gamma_{mr}$, $\lambda_m$, and $\underline{\mathbf{h}}^{L_1m}$ values in (6.15), (6.17), and (6.21), respectively.

#### 6.4.3.3 Partial Least Squares

The objective of PLS [207] is to maximize the covariance between each pair of input modalities. Assume, $\gamma_{mr} = 1$, $\lambda_m = 0$, and $\underline{\mathbf{h}}^{L_1m} = 0$, $\forall m, r$. In such a scenario, the energy

function $E_{gda}(\mathbf{h})$ of (6.8) turns out to be

$$E_{pls}(\mathbf{h}) = -\sum_{m=1}^{M-1}\sum_{r=(m+1)}^{M}\left\{\sum_{j=1}^{H^{L_1 m}} h_j^{L_1 m} h_j^{L_1 r}\right\}^2. \tag{6.43}$$

It is evident from (6.43) that $E_{pls}(\mathbf{h})$ is the negative of the squared value of $\mathbf{tr}(\text{cov}(\mathbf{h}^{L_1 m}, \mathbf{h}^{L_1 r}))$, $\forall m, r$. Hence, in order to minimize $E_{pls}(\mathbf{h})$ in (6.43), the covariance between each pair of modality-specific hidden representations is required to be maximized, which is clearly the objective of PLS. Now, if the energy function $E_{pls}(\mathbf{h})$ of (6.43) is employed in the overall energy function $E_{d2gda}(\mathbf{v}, \mathbf{h}, \mathbf{y})$ of (6.8), then the joint subspace of the MDDBM architecture will be formed in such a way that the covariance between the modality-specific representations in the projected space is maximum. The deep framework, obtained using the energy function of (6.43), is termed as MDDBM_PLS, which can be learned by replacing $\gamma_{mr}$, $\lambda_m$, and $\underline{\mathbf{h}}^{L_1 m}$ values in (6.15), (6.17), and (6.21), respectively.

Thus, the proposed loss function is the generalization of the three acknowledged feature extraction techniques, namely, CCA, GMPCA, and PLS. In this context, it is to be mentioned here that, if $\gamma_{mr} = 0$, $\lambda_m = 0$, and $\underline{\mathbf{h}}^{L_1 m} = 0$, $\forall m, r$, then the D2GDA model boils down to the MDDBM model.

## 6.5 Experimental Results and Discussions

In this section, the performance of the proposed D2GDA model is extensively studied and the corresponding results are reported. In order to evaluate the efficacy of the proposed architecture, several existing algorithms are considered, which include RGCCA [187], MCCA [96], GMCCA [25], GMKCCA [25], LasCCA [53], DisCCA [53], MvDA [92], MvDA-VC [93], MvCCDA [217], MDBM [180], dMCCA [178], TOCCA [30], DACCA [44], DCCA-VG [193], TDDCCA [37], MDL-CW [157], MMGNN [54], MvLDAN [79], mgRBM [221], TCCA [213], and MVGAN [209]. The performance of the proposed approach as well as existing methods is demonstrated in terms of both training-testing and 10-fold cross-validation (CV). In case of training-testing, overall classification accuracy is considered, while 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 for 10-fold CV.

### 6.5.1 Description of Data Sets

In order to evaluate the performance of different algorithms, seven benchmark databases, namely, Digits [63], Caltech [114], CiteSeer [161], Cora [161], NUS-WIDE-OBJECT (NW-OBJECT) [27], Reuters [7], and Animals with Attributes (AwA) [208], and five cancer data sets are considered. The Digits, Caltech, NW-OBJECT, and AwA are image based databases, CiteSeer and Cora consist of scientific publications with annotated labels, whereas Reuters is a multilingual categorization data set, containing documents written in English language with the corresponding translations in four different languages. Five real-life omics data sets, corresponding to cervical carcinoma (CESC), colorectal carcinoma

(CRC), kidney carcinoma (KIDNEY), lower grade glioma (LGG), and lung carcinoma (LUNG), are obtained from The Cancer Genome Atlas [194].

Table 6.1: Description of Data Sets

| Data | | Sample | Class | View | $V^1$ | $V^2$ | $V^3$ | $V^4$ | $V^5$ | $V^6$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Benchmark | AwA | 30475 | 50 | 6 | 2688 | 2000 | 252 | 2000 | 2000 | 2000 |
| Omics | CRC | 261 | 2 | 4 | 293526 | 222 | 236 | 13465 | - | - |
| | KIDNEY | 305 | 2 | 5 | 300451 | 174 | 209 | 20502 | 9059 | - |

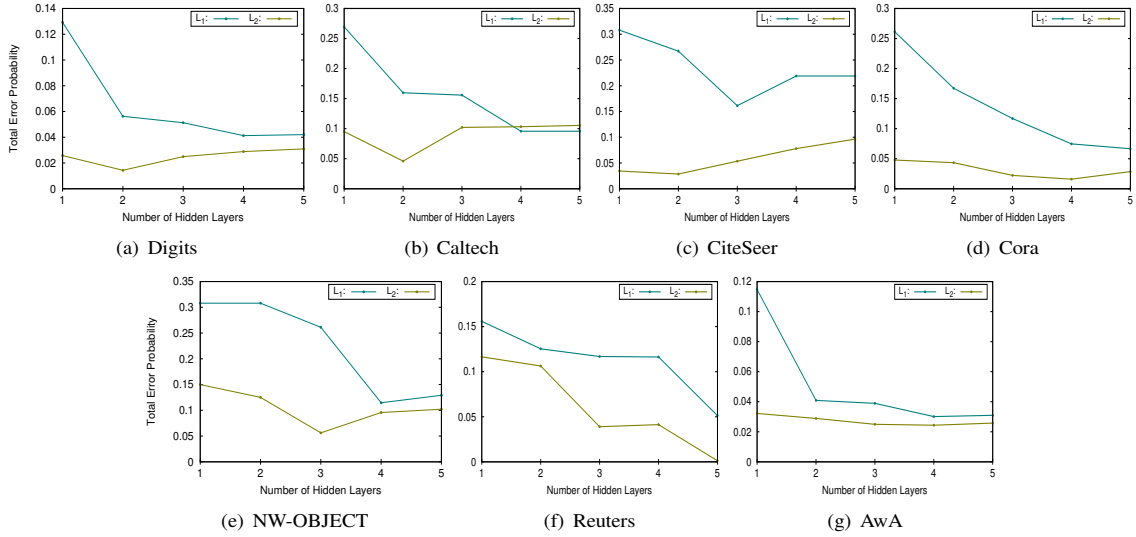
It is to be noted here that while Digits and Caltech data sets exhibit large variance in the dimensionality of the input feature sets, CiteSeer and Cora databases have large number of features in the corresponding feature sets. The NW-OBJECT data set has large number of samples with small number of features in each view, whereas Reuters and AwA databases have large number of samples with large dimension of each of the feature sets. On the other hand, the omics data sets offer the problem of high dimensional feature sets with small number of samples. A brief description of the CiteSeer, Cora, NW-OBJECT, Reuters, CESC, and LGG data sets is presented in Chapter 4, whereas the description of Digits, Caltech, and LUNG databases is illustrated in Chapter 5. In addition to the aforementioned data sets, the AwA, CRC, and KIDNEY databases are also considered in this chapter for performance analysis of different algorithms. The number of samples, number of classes, number of views, and number of features in each view corresponding to these three databases, are tabulated in Table 6.1. Each data set is randomly partitioned into two sets for training-testing and ten separate folds for 10-fold CV. In both the cases, the samples are equally distributed with respect to given classes. Detailed description of all the data sets is reported in Appendix A.

### 6.5.2 Model Architecture Based on Error Bound

In existing literature, the architecture of a deep framework is heuristically determined for all the databases under consideration. Hence, it does not take into account the diversities present in the nature of the problem as well as the complexities associated with the given data sets. However, in the proposed method, an upper bound on the error probability (6.35) is estimated in terms of the architecture of the model, which enables the framework to select an optimal architecture for the analysis of the given multi-view data. In the current study, greedy layer-wise pretraining [163] is performed to initialize the model parameters sensibly.

In order to determine the optimal number of layers in the proposed D2GDA model, extensive experiments are carried out on both benchmark and omics data sets. During the pretraining, the number of modality-specific hidden layers ($L_1$) is varied from 1 to 5 for each of the data sets, keeping the number of joint hidden layers ($L_2$) fixed at 1 and the corresponding values of error bound are noted. Then, $L_1$ is fixed at the value for which the error bound has achieved the minimum value, while $L_2$ is varied from 1 to 5 and the variation in the error bound is observed. The value of $L_2$ for which error bound attains the minimum value is considered for the analysis of the particular data set. The variation of error bound with respect to $L_1$ and $L_2$, corresponding to the benchmark and

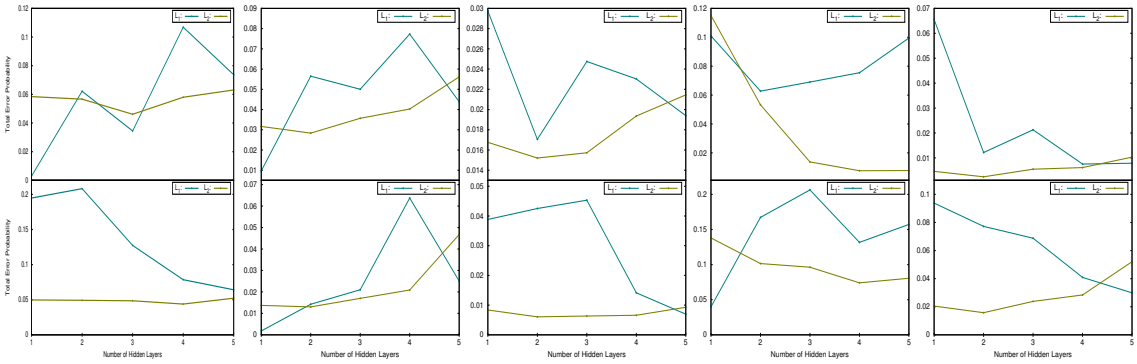
omics databases, are presented in Figure 6.2 and Figure 6.3, respectively, for both training-testing and 10-fold CV. The optimal values of $L_1$ and $L_2$, obtained from the corresponding error plots, are tabulated in Table 6.2. It establishes the fact that different number of layers is required by the proposed deep architecture to address the challenges offered by each of the data sets for various experimental set-up.




(a) Digits (b) Caltech (c) CiteSeer (d) Cora

(e) NW-OBJECT (f) Reuters (g) AwA

Figure 6.2: Variation of error bound with respect to the architecture of the proposed D2GDA model on benchmark data.




(a) CESC (b) CRC (c) KIDNEY (d) LGG (e) LUNG

Figure 6.3: Variation of error bound with respect to the architecture of the proposed D2GDA framework (top row: training-testing and bottom row: 10-fold CV on omics data).

The hidden nodes of the architecture are represented by the corresponding probability values and the parameters are updated based on the mini-batches formed from the given set of training samples. The number of hidden nodes is upper bounded by Theorem 6.3, presented in Section 6.3.1. Each of the $L_1$ layers consists of 25 hidden nodes, whereas the $L_2$ joint layers have 10 hidden nodes each. The number of epochs, and the values of momentum and weight decay are considered to be 100, 0.5, and 0.0005, respectively. The value of learning rate is initialized at 0.01 and then, gradually decreased with the increase

Table 6.2: Optimal Number of Layers for D2GDA Model Based on Estimated Error Bound

| Different Metrices | Different Data Sets | | Number of Layers $(L_1, L_2)$ |
|---|---|---|---|
| Training-Testing | Benchmark | Digits | 4,2 |
| | | Caltech | 4,2 |
| | | CiteSeer | 3,2 |
| | | Cora | 5,4 |
| | | NW-OBJECT | 4,3 |
| | | Reuters | 5,5 |
| | | AwA | 4,4 |
| | Omics | CESC | 1,3 |
| | | CRC | 1,2 |
| | | KIDNEY | 2,2 |
| | | LGG | 2,4 |
| | | LUNG | 4,2 |
| 10-fold CV | Omics | CESC | 5,4 |
| | | CRC | 1,2 |
| | | KIDNEY | 5,2 |
| | | LGG | 1,4 |
| | | LUNG | 5,2 |

in number of epochs. For the estimation of data-independent expectations, 100 Gibbs steps and 20 separate Markov chains are considered.

### 6.5.3 Effectiveness of Proposed D2GDA Model

Various state-of-the-art feature extraction methods, namely CCA [78], GMPCA [171], and PLS [207], can be expressed as special cases of the proposed loss function employed in the D2GDA model. So, for different values of $\gamma_{mr}, \lambda_m,$ and $\underline{\mathbf{h}}^{L_1 m}$, the energy function of the proposed D2GDA framework reduces to the objective function of MDDBM, MDDBM_CCA, MDDBM_GMPCA, and MDDBM_PLS models. In order to establish the effectiveness of the proposed model, the performance of the D2GDA framework is extensively studied in comparison to its different variants on seven benchmark databases and five omics data sets, considering both training-testing and 10-fold CV. The corresponding results are reported in Figure 6.4, Table 6.3, and Table 6.4. It is to be mentioned here that the architecture remains the same for the proposed model as well as the variants corresponding to each of the data sets. Only the learning objectives are changed based on the values of $\gamma_{mr}, \lambda_m,$ and $\overline{\mathbf{h}}^{L_1 m}$, in (6.15), (6.17), and (6.21), respectively. The scatter plots of Figure 6.4 are depicted by considering the most relevant feature at $x$-axis and the corresponding most significant feature at $y$-axis, obtained using the concept of rough hypercuboid approach [126]. The first row of Figure 6.4 corresponds to MDDBM framework, the second, third, and fourth rows refer to MDDBM_CCA, MDDBM_GMPCA, and MDDBM_PLS models, respectively, while the plots of last row are obtained from the proposed D2GDA framework.

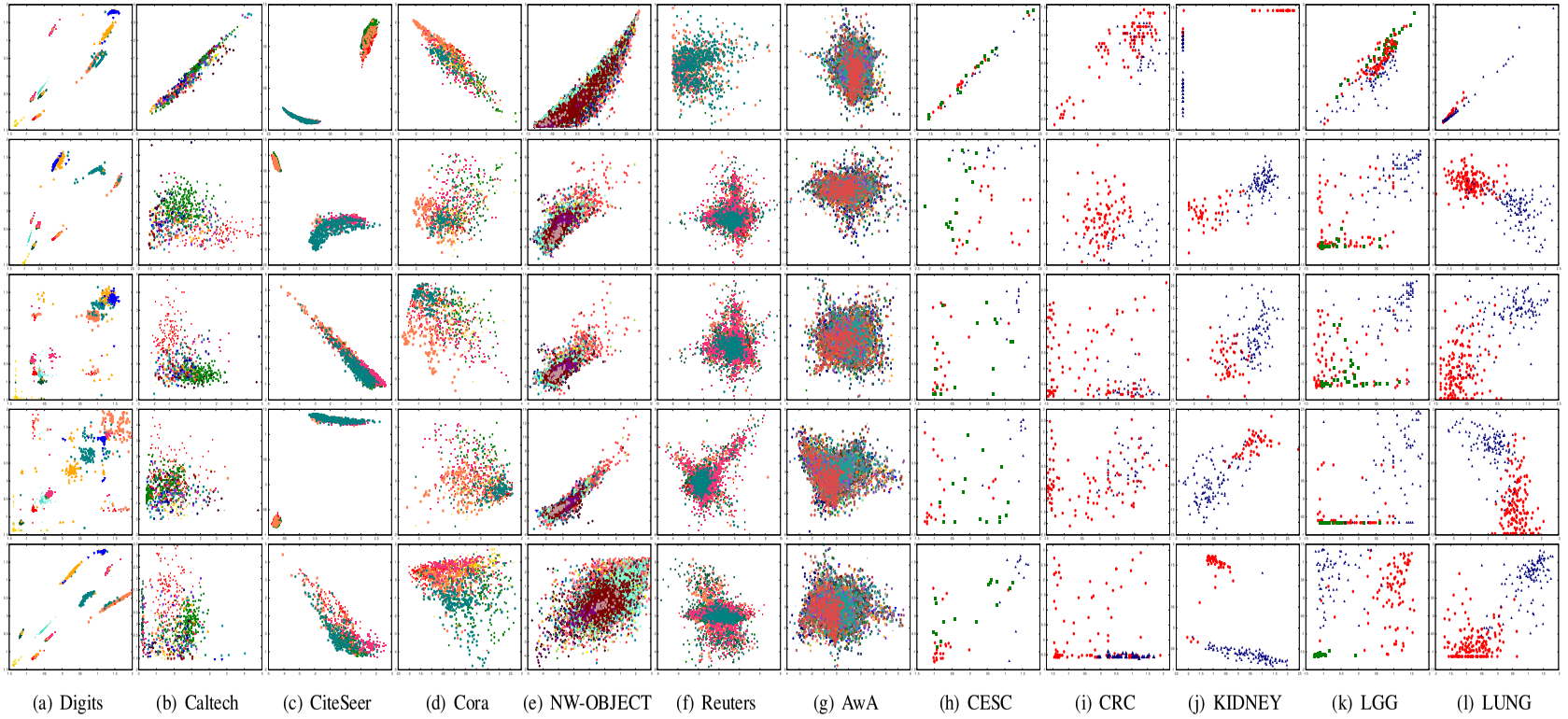



Figure 6.4: Scatter plots of different variants of proposed architecture on benchmark and omics data sets (1st row: MDDBM; 2nd row: MDDBM_CCA; 3rd row: MDDBM_GMPCA; 4th row: MDDBM_PLS; and 5th row: D2GDA).

Considering the scatter plots of Figure 6.4 corresponding to the benchmark databases, it can be observed that all the classes of Digits data set can be properly identified by the different variants of the proposed model. For Caltech, AwA, and Reuters databases, although the samples from different classes tend to overlap for MDDBM architecture, the separation between the classes has improved for rest of the models. However, in case of CiteSeer and Cora data sets, almost similar plots are obtained for all the models. This result is also reflected in Table 6.3, where the classification accuracy on benchmark databases is presented for training-testing. From the results reported in Table 6.3, it can be observed that significant improvement in the performance can be noted for all the models in comparison to MDDBM framework on Caltech, AwA, and Reuters data sets, whereas comparable classification accuracy is achieved by the frameworks in case of CiteSeer and Cora databases. Although MDDBM_GMPCA and MDDBM_PLS perform better than the proposed model in CiteSeer and Cora databases, respectively, the D2GDA model performs significantly better on Digits, Caltech, NW-OBJECT, Reuters, and AwA data sets.

Table 6.3: Comparative Performance Analysis of Different Variants of Proposed D2GDA Model on Benchmark Database

| Data | MDDBM | MDDBM _CCA | MDDBM _GMPCA | MDDBM _PLS | D2GDA |
|---|---|---|---|---|---|
| Digits | 85.60 | 91.60 | 88.80 | 90.00 | **97.30** |
| Caltech | 74.14 | 83.78 | 82.51 | 84.28 | **92.52** |
| CiteSeer | 71.93 | 73.02 | **73.48** | 71.30 | 73.12 |
| Cora | 64.37 | 82.13 | 81.69 | **83.02** | 82.46 |
| NW-OBJECT | 43.75 | 46.97 | 46.44 | 49.38 | **55.39** |
| Reuters | 61.25 | 67.45 | 62.74 | 74.26 | **86.70** |
| AwA | 56.86 | 59.76 | 65.40 | 60.04 | **68.57** |

The scatter plots of Figure 6.4 corresponding to the omics data sets demonstrate that all the given classes can be efficiently identified by the proposed architecture. While the classes are well separated for all the variants on KIDNEY and LUNG data, the separation between the samples of different categories has detoriated in case of rest of the data sets. This observations can be validated from the results reported in Table 6.4, where it can be noted that considerable classification accuracy is obtained by almost all the variants of the proposed model on KIDNEY and LUNG databases. Although the highest classification accuracy is achieved by the proposed framework on all the cancer data sets for both training-testing and 10-fold CV, significant improvement in the performance of the D2GDA model is noted on CESC, CRC and LGG data sets. The statistical significance analysis reveals that out of the total 40 cases, the proposed model attains significantly better p-values in 30 cases and better but not significant p-values for the rest 10 cases.

### 6.5.4 Comparative Performance Analysis

Finally, the performance of the proposed D2GDA model is studied on seven benchmark and five omics data sets with reference to several existing approaches, which include consensus principle based methods, complementary principle based approaches, and both consensus

and complementary principles based approaches. The corresponding results are reported in Table 6.5, Table 6.6, Table 6.7, Table 6.8, Table 6.9, and Table 6.10. The scatter plots of the existing approaches as well as the proposed D2GDA model are depicted in Figure 6.5. It is to be mentioned here that, in case of proposed framework, the class labels of input samples are predicted from the architecture itself. Hence, no classifier is employed in the proposed method for classification purpose.

Table 6.4: Performance Analysis of Different Variants of Proposed Model on Omics Data

| Data | Different Metrics | | MDDBM | MDDBM_CCA | MDDBM_GMPCA | MDDBM_PLS | D2GDA |
|---|---|---|---|---|---|---|---|
| CESC | Train-Test | | 67.31 | 61.54 | 59.62 | 57.69 | **78.85** |
| | 10-fold CV | Mean | 64.17 | 70.00 | 72.50 | 70.00 | **84.17** |
| | | Median | 66.67 | 66.67 | 75.00 | 70.83 | **83.33** |
| | | StdDev | **5.62** | 8.96 | 13.64 | 11.92 | 6.15 |
| | | Paired-$t$:p | 1.62E-04 | 2.14E-03 | 1.24E-02 | 4.73E-03 | - |
| | | Wilcoxon:p | 3.48E-03 | 7.23E-03 | 2.11E-02 | 1.20E-02 | - |
| CRC | Train-Test | | 78.46 | 80.00 | 77.69 | 79.23 | **85.50** |
| | 10-fold CV | Mean | 71.11 | 83.33 | 84.07 | 82.59 | **87.04** |
| | | Median | 74.07 | 81.48 | 85.19 | 81.48 | **88.89** |
| | | StdDev | 15.20 | 5.59 | 6.31 | 5.25 | **4.00** |
| | | Paired-$t$:p | 6.26E-03 | 1.15E-02 | 7.63E-02 | 6.50E-03 | - |
| | | Wilcoxon:p | 2.34E-03 | 7.97E-03 | 8.59E-02 | 1.36E-02 | - |
| KIDNEY | Train-Test | | 98.03 | 98.03 | 96.05 | 94.74 | **98.68** |
| | 10-fold CV | Mean | 90.32 | 99.35 | 99.35 | 98.10 | **99.68** |
| | | Median | 91.94 | **100.00** | **100.00** | 98.21 | **100.00** |
| | | StdDev | 8.60 | 1.36 | 1.36 | **0.42** | 1.02 |
| | | Paired-$t$:p | 3.84E-03 | 2.96E-01 | 1.72E-01 | 1.53E-03 | - |
| | | Wilcoxon:p | 3.68E-03 | 2.82E-01 | 1.59E-01 | 6.06E-03 | - |
| LGG | Train-Test | | 79.57 | 75.00 | 71.81 | 77.13 | **90.32** |
| | 10-fold CV | Mean | 63.95 | 86.32 | 84.47 | 81.58 | **98.68** |
| | | Median | 63.16 | 85.53 | 82.89 | 81.58 | **98.68** |
| | | StdDev | 6.08 | 7.42 | 6.50 | 3.51 | **1.39** |
| | | Paired-$t$:p | 1.28E-08 | 2.90E-04 | 2.13E-05 | 1.51E-08 | - |
| | | Wilcoxon:p | 2.50E-03 | 3.42E-03 | 2.50E-03 | 2.45E-03 | - |
| LUNG | Train-Test | | 94.51 | 95.24 | 94.87 | 94.51 | **96.34** |
| | 10-fold CV | Mean | 90.18 | 96.96 | 96.61 | 96.79 | **97.32** |
| | | Median | 91.96 | 97.32 | 96.43 | **98.21** | **98.21** |
| | | StdDev | 7.49 | **2.92** | 3.09 | 3.24 | 2.95 |
| | | Paired-$t$:p | 3.24E-03 | 2.22E-01 | 1.84E-02 | 9.67E-02 | - |
| | | Wilcoxon:p | 3.79E-03 | 2.98E-01 | 3.17E-02 | 2.46E-01 | - |

Table 6.5: Comparative Performance Analysis of Consensus Principle Based Methods on Benchmark Databases

| Data | RGCCA | MCCA | GMCCA | GMKCCA | LasCCA | DisCCA | DCCA-VG | MDL-CW | D2GDA |
|---|---|---|---|---|---|---|---|---|---|
| Digits | 90.30 | 87.00 | 11.20 | 6.60 | 10.20 | 5.60 | 89.00 | 86.40 | **97.20** |
| Caltech | 33.71 | 41.83 | 4.82 | 7.48 | 4.06 | 3.17 | 49.68 | 54.25 | **92.52** |
| CiteSeer | 27.61 | 58.13 | 23.43 | 24.98 | 22.25 | 20.71 | 37.33 | 42.05 | **73.12** |
| Cora | 52.16 | 32.85 | 30.97 | 30.19 | 31.63 | 30.19 | 44.62 | 41.40 | **82.46** |
| NW-OBJECT | 18.91 | 30.34 | 4.56 | 6.43 | 7.40 | 10.93 | 19.23 | 18.20 | **55.39** |
| Reuters | 55.26 | 57.50 | 24.78 | 28.69 | 28.67 | 23.27 | 64.38 | 62.69 | **86.70** |
| AwA | 6.90 | 15.08 | 1.58 | 3.09 | 1.59 | 1.94 | 46.59 | 59.58 | **68.57** |

Table 6.6: Performance Analysis of Consensus Principle Based Approaches on Omics Data

| Data | Different Metrics | | RGCCA | MCCA | GMCCA | GMKCCA | LasCCA | DisCCA | DCCA-VG | MDL-CW | D2GDA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CESC | Train-Test | | 61.54 | 38.46 | 42.31 | 44.23 | 42.31 | 36.54 | 65.38 | 65.38 | **78.85** |
| | 10-fold CV | Mean | 75.00 | 45.83 | 49.17 | 38.33 | 35.00 | 39.17 | 67.50 | 69.17 | **84.17** |
| | | Median | 79.17 | 50.00 | 50.00 | 41.67 | 33.33 | 33.33 | 70.83 | 66.67 | **83.33** |
| | | StdDev | 13.03 | 13.75 | 14.41 | 11.92 | 15.61 | 10.43 | 16.87 | 14.72 | **6.15** |
| | | Paired-$t$:p | 1.21E-02 | 1.55E-05 | 6.79E-05 | 1.23E-07 | 9.46E-07 | 2.45E-07 | 2.94E-03 | 5.00E-03 | - |
| | | Wilcoxon:p | 1.55E-02 | 2.49E-03 | 2.49E-03 | 2.47E-03 | 2.50E-03 | 2.47E-03 | 8.81E-03 | 1.24E-02 | - |
| CRC | Train-Test | | 83.85 | 73.85 | 83.85 | 50.77 | 78.46 | 73.08 | 77.69 | 76.15 | **85.50** |
| | 10-fold CV | Mean | 81.85 | 60.74 | 78.52 | 54.07 | 78.52 | 62.59 | 76.30 | 77.04 | **87.04** |
| | | Median | 83.33 | 62.96 | 81.48 | 55.56 | 77.78 | 68.52 | 75.93 | 77.78 | **88.89** |
| | | StdDev | 5.64 | 10.36 | 7.57 | 8.59 | 4.20 | 13.23 | 4.68 | **3.83** | 4.00 |
| | | Paired-$t$:p | 8.24E-03 | 5.10E-05 | 2.74E-03 | 9.26E-07 | 3.10E-04 | 6.43E-05 | 1.87E-04 | 1.04E-05 | - |
| | | Wilcoxon:p | 1.41E-02 | 2.53E-03 | 6.14E-03 | 2.52E-03 | 2.50E-03 | 2.50E-03 | 3.79E-03 | 2.38E-03 | - |
| KIDNEY | Train-Test | | 91.45 | 55.92 | 85.53 | 82.89 | 74.34 | 58.55 | 93.42 | 92.76 | **98.68** |
| | 10-fold CV | Mean | 92.90 | 60.97 | 75.81 | 81.61 | 79.03 | 60.97 | 96.45 | 95.48 | **99.68** |
| | | Median | 91.94 | 61.29 | 77.42 | 83.87 | 77.42 | 64.52 | 96.77 | 95.16 | **100.00** |
| | | StdDev | 4.76 | 6.17 | 6.32 | 7.91 | 10.99 | 8.93 | 2.82 | 3.12 | **1.02** |
| | | Paired-$t$:p | 9.15E-04 | 7.22E-09 | 3.05E-07 | 2.92E-05 | 1.42E-04 | 1.48E-07 | 4.23E-03 | 1.86E-03 | - |
| | | Wilcoxon:p | 5.61E-03 | 2.34E-03 | 2.45E-03 | 2.50E-03 | 2.39E-03 | 2.46E-03 | 1.16E-02 | 7.88E-03 | - |
| LGG | Train-Test | | 41.40 | 39.78 | 33.33 | 38.71 | 44.09 | 29.03 | 77.96 | 73.12 | **90.32** |
| | 10-fold CV | Mean | 45.00 | 35.53 | 40.53 | 33.16 | 38.68 | 38.68 | 51.32 | 76.84 | **98.68** |
| | | Median | 47.37 | 34.21 | 40.79 | 31.58 | 36.84 | 38.16 | 51.32 | 77.63 | **98.68** |
| | | StdDev | 10.99 | 7.67 | 8.70 | 4.99 | 7.95 | 7.24 | 3.34 | 7.63 | **1.39** |
| | | Paired-$t$:p | 6.57E-08 | 1.10E-09 | 2.85E-09 | 2.97E-12 | 7.39E-10 | 7.39E-10 | 1.43E-11 | 2.97E-06 | - |
| | | Wilcoxon:p | 2.47E-03 | 2.46E-03 | 2.50E-03 | 2.50E-03 | 2.47E-03 | 2.50E-03 | 2.49E-03 | 2.50E-03 | - |
| LUNG | Train-Test | | 87.91 | 46.52 | 68.86 | 86.08 | 82.42 | 47.62 | 89.74 | 93.77 | **96.34** |
| | 10-fold CV | Mean | 87.68 | 51.43 | 68.39 | 86.07 | 85.18 | 50.71 | 94.11 | 93.93 | **97.32** |
| | | Median | 86.61 | 50.89 | 69.64 | 87.50 | 85.71 | 48.21 | 95.54 | 95.54 | **98.21** |
| | | StdDev | 4.16 | 3.13 | 7.48 | 8.32 | 6.68 | 8.84 | 4.69 | 4.31 | **2.95** |
| | | Paired-$t$:p | 8.00E-05 | 2.72E-12 | 2.45E-07 | 8.54E-04 | 1.01E-04 | 1.92E-08 | 5.99E-03 | 2.00E-04 | - |
| | | Wilcoxon:p | 2.50E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.52E-03 | 2.53E-03 | 5.81E-03 | 3.30E-03 | - |

#### 6.5.4.1 Performance of Consensus Principle Based Methods

In this section, several state-of-the-art consensus principle based methods are considered for performance evaluation of the proposed D2GDA model, namely RGCCA [187], MCCA [96], GMCCA [25], GMKCCA [25], LasCCA [53], DisCCA [53], DCCA-VG [193], and MDL-CW [157]. Out of these methods, RGCCA, MCCA, GMCCA, GMKCCA, LasCCA, and DisCCA are classical approaches, whereas DCCA-VG and MDL-CW are deep learning models. The scatter plots corresponding to the MCCA, GMCCA, GMKCCA, LasCCA,

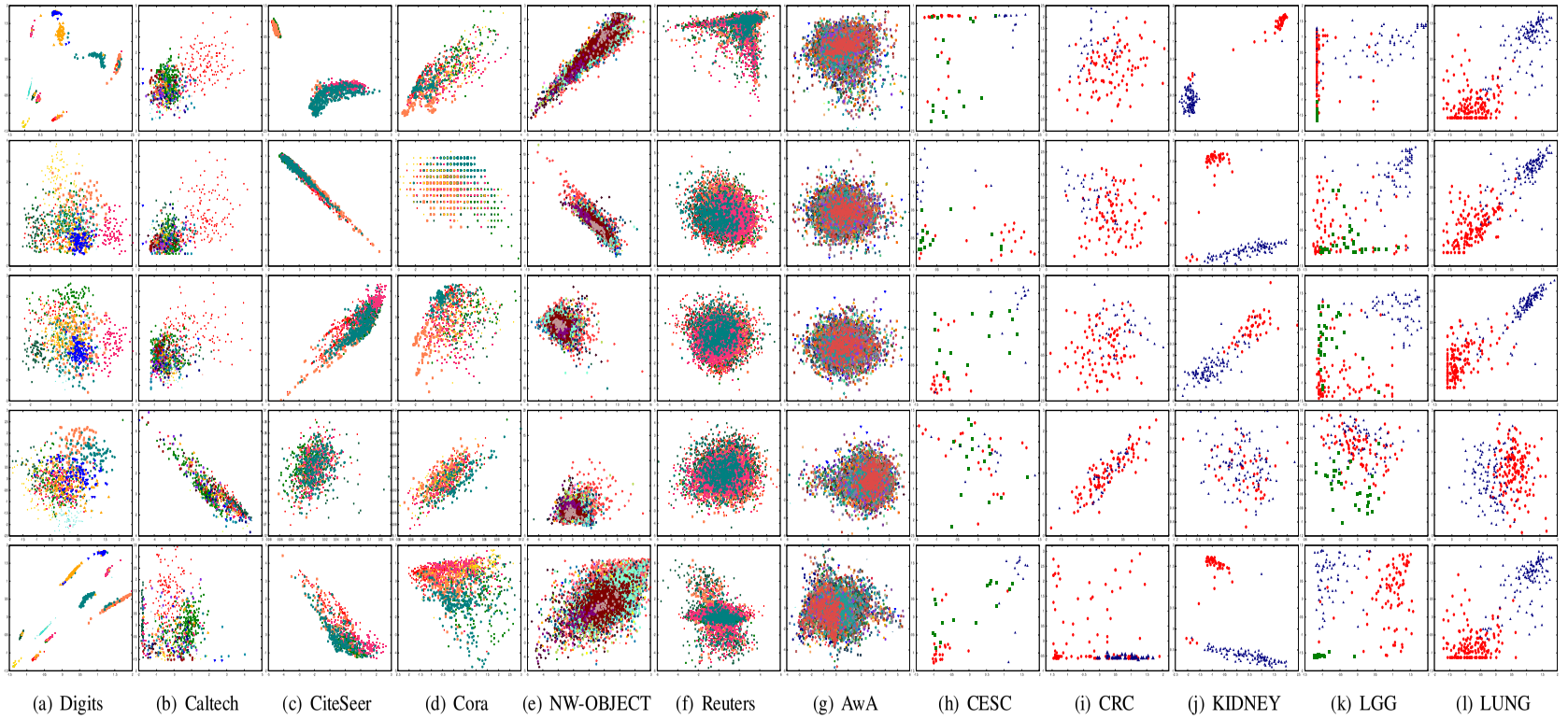



Figure 6.5: Scatter plots of existing and proposed algorithms on benchmark and omics data sets (from top to bottom row: MvCCDA, mgRBM, MvLDAN, TCCA, and D2GDA, respectively).

and DisCCA approaches are presented in Figure 4.3 of Chapter 4, whereas the scatter plots for RGCCA method are depicted in Figure 5.3 on both benchmark and omics databases. The plots corresponding to the DCCA-VG and MDL-CW models are illustrated in Figure 4.4 of Chapter 4 and Figure 5.4 of Chapter 5, respectively. The scatter plots corresponding to the proposed D2GDA framework are presented in Figure 6.5. Each of the existing classical algorithms considers 25 features to represent the joint subspace, whereas DCCA-VG and MDL-CW models consider 20 and 600 features, respectively, in the shared subspace. The architecture for each of these methods follows the same as described in the corresponding papers. For the existing algorithms, the extracted features are applied to the input of support vector machine (SVM) [197] for classification purpose. From the results reported in Table 6.5, it can be observed that although RGCCA and MCCA achieve considerable classification accuracy on Digits data set, they fail to attain similar results on rest of the benchmark databases. However, the proposed model exhibits significantly better performance with respect to the existing consensus principle based methods on all the seven benchmark databases. The results reported in Table 6.6 corresponding to the omics data sets demonstrate that the proposed method outperforms all the eight multiset consensus principle based methods on five cancer data sets for both training-testing and 10-fold CV. The statistical significance test reveals that the proposed architecture achieves significantly better p-values for all the 80 cases.

Table 6.7: Comparative Performance Analysis of Complementary Principle Based Approaches on Benchmark Databases

| Data | MvDA | MvDA-VC | MvCCDA | MvLDAN | MVGAN | D2GDA |
|---|---|---|---|---|---|---|
| Digits | 92.40 | 93.50 | 92.80 | 90.70 | 82.70 | **97.20** |
| Caltech | 76.30 | 75.29 | 76.68 | 79.47 | 77.31 | **92.52** |
| CiteSeer | 37.69 | 43.51 | 45.69 | 49.05 | 46.32 | **73.12** |
| Cora | 53.94 | 55.72 | 58.71 | 63.26 | 44.95 | **82.46** |
| NW-OBJECT | 29.03 | 28.62 | 37.33 | 36.27 | 27.61 | **55.39** |
| Reuters | 56.01 | 55.15 | 55.36 | 53.36 | 57.60 | **86.70** |
| AwA | 16.55 | 15.37 | 62.67 | 47.41 | **69.06** | 68.57 |

#### 6.5.4.2 Performance of Complementary Principle Based Approaches

Here, the performance of the proposed model is analyzed with reference to several complementary principle based approaches, namely MvDA [92], MvDA-VC [93], MvCCDA [217], MvLDAN [79], and MVGAN [209]. Out of these methods, MvDA, MvDA-VC, and MvCCDA are classical approaches, whereas MvLDAN and MVGAN are deep learning models. The scatter plots corresponding to the MvDA and MvDA-VC approaches are presented in Figure 4.3 of Chapter 4, whereas the scatter plots for MvCCDA method are depicted in Figure 5.3 on both benchmark and omics databases. The plots corresponding to the MVGAN model are illustrated in Figure 5.4 of Chapter 5. The scatter plots corresponding to the MvLDAN model as well as proposed D2GDA framework are presented in Figure 6.5. Each of the existing classical algorithms considers 25 features to represent the joint subspace,

Table 6.8: Comparative Performance Analysis of Complementary Principle Based Approaches on Omics Data Sets

| Data | Different Metrics | | MvDA | MvDA-VC | MvCCDA | MvLDAN | MVGAN | D2GDA |
|---|---|---|---|---|---|---|---|---|
| CESC | Train-Test | | 42.31 | 40.38 | 59.62 | 65.38 | 57.69 | **78.85** |
| | 10-fold CV | Mean | 46.67 | 50.00 | 61.67 | 65.83 | 61.67 | **84.17** |
| | | Median | 41.67 | 50.00 | 58.33 | 66.67 | 58.33 | **83.33** |
| | | StdDev | 15.32 | 14.16 | 12.55 | 10.72 | 12.55 | **6.15** |
| | | Paired-$t$:p | 4.39E-05 | 1.85E-05 | 7.22E-04 | 1.00E-03 | 7.23E-04 | - |
| | | Wilcoxon:p | 2.52E-03 | 2.47E-03 | 3.61E-03 | 5.76E-03 | 3.53E-03 | - |
| CRC | Train-Test | | 80.77 | 83.08 | 82.31 | 80.77 | 79.23 | **86.15** |
| | 10-fold CV | Mean | 83.70 | 86.67 | 82.59 | 80.00 | 81.85 | **87.04** |
| | | Median | 85.19 | 88.89 | 81.48 | 81.48 | 81.48 | **88.89** |
| | | StdDev | 5.00 | 6.34 | 3.92 | 3.98 | **3.68** | 4.00 |
| | | Paired-$t$:p | 2.07E-02 | 4.36E-01 | 2.56E-03 | 3.59E-04 | 2.63E-04 | - |
| | | Wilcoxon:p | 1.01E-01 | 8.07E-01 | 8.47E-03 | 2.45E-03 | 3.76E-03 | - |
| KIDNEY | Train-Test | | 92.76 | 94.74 | 95.39 | 96.71 | 95.39 | **98.68** |
| | 10-fold CV | Mean | 93.23 | 94.19 | 95.16 | 87.74 | 81.61 | **99.68** |
| | | Median | 93.55 | 93.55 | 93.55 | 85.48 | 77.42 | **100.00** |
| | | StdDev | 2.38 | 2.54 | 3.80 | 8.30 | 12.64 | **1.02** |
| | | Paired-$t$:p | 1.43E-05 | 1.54E-04 | 1.30E-03 | 9.70E-04 | 6.94E-04 | - |
| | | Wilcoxon:p | 2.32E-03 | 3.19E-03 | 7.03E-03 | 8.68E-03 | 5.71E-03 | - |
| LGG | Train-Test | | 75.81 | 73.12 | 77.96 | 75.81 | 74.73 | **90.32** |
| | 10-fold CV | Mean | 75.79 | 81.05 | 77.63 | 76.84 | 76.84 | **98.68** |
| | | Median | 76.32 | 78.95 | 77.63 | 77.63 | 77.63 | **98.68** |
| | | StdDev | 8.02 | 7.83 | 6.11 | 7.63 | 7.63 | **1.39** |
| | | Paired-$t$:p | 3.87E-06 | 2.56E-05 | 4.74E-07 | 2.97E-06 | 2.97E-06 | - |
| | | Wilcoxon:p | 2.50E-03 | 2.47E-03 | 2.52E-03 | 2.50E-03 | 2.50E-03 | - |
| LUNG | Train-Test | | 92.31 | 91.58 | 93.77 | 90.48 | 92.67 | **96.34** |
| | 10-fold CV | Mean | 94.82 | 95.54 | 96.96 | 95.18 | 94.11 | **97.32** |
| | | Median | 96.43 | 95.54 | 98.21 | 94.64 | 94.64 | **98.21** |
| | | StdDev | 4.16 | 3.29 | 3.37 | 3.77 | 3.95 | **2.95** |
| | | Paired-$t$:p | 2.76E-02 | 7.48E-03 | 2.22E-01 | 9.00E-03 | 2.56E-03 | - |
| | | Wilcoxon:p | 4.63E-02 | 1.21E-02 | 2.97E-01 | 8.24E-03 | 8.88E-03 | - |

whereas MvLDAN and MVGAN models consider 20 and 50 features, respectively, in the shared subspace. The architecture for each of these methods follows the same as described in the corresponding papers. For the existing classical algorithms, the extracted features are applied to the input of SVM for classification purpose. The results corresponding to Table 6.7 demonstrate that the complementary principle based methods achieve considerable accuracy on all the seven benchmark databases. However, the highest classification accuracy is attained by the proposed model in all the cases, except for AwA data set where the MVGAN model attains the highest classification accuracy. From the results reported in Table 6.8, it can be observed that the proposed model performs considerably better than all the existing approaches on all the five cancer data sets for both training-testing and

10-fold CV. The p-values obtained from the two statistical significance tests indicate that out of total 50 cases, the proposed model attains significantly better p-values for 45 cases and better but not significant p-values for 4 cases.

Table 6.9: Comparative Performance Analysis of Both Consensus and Complementary Principle Based Approaches on Benchmark Databases

| Data | MDBM | mgRBM | DACCA | TCCA | TDDCCA | MMGNN | D2GDA |
|---|---|---|---|---|---|---|---|
| Digits | 10.00 | 88.60 | 84.60 | **97.80** | 85.90 | 88.90 | 97.20 |
| Caltech | 2.53 | 78.75 | 73.89 | 87.58 | 74.52 | 74.27 | **92.52** |
| CiteSeer | 17.8 | 63.23 | 55.22 | 56.40 | 40.42 | 41.78 | **73.12** |
| Cora | 10.99 | 58.16 | 50.06 | 56.94 | 53.05 | 42.06 | **82.46** |
| NW-OBJECT | 26.07 | 32.16 | 38.42 | 33.61 | 17.80 | 37.87 | **55.39** |
| Reuters | 46.84 | 58.13 | 56.38 | 75.97 | 57.27 | 59.16 | **86.70** |
| AwA | 27.16 | 53.5 | **69.20** | 64.63 | 53.32 | 44.96 | 68.57 |

#### 6.5.4.3 Performance of Consensus and Complementary Principles Based Approaches

Finally, the proficiency of the proposed framework is compared with that of several state-of-the-art methods which are based on both the consensus and complementary principles. These methods include MDBM [180], mgRBM [221], DACCA [44], TDDCCA [37], TCCA [213], and MMGNN [54]. Out of these methods, mgRBM is a classical approaches, whereas MDBM, DACCA, TDDCCA, TCCA, and MMGNN are deep learning models. The scatter plots corresponding to the MDBM model are presented in Figure 4.2, whereas the scatter plots for DACCA and TDDCCA models are depicted in Figure 4.4 of Chapter 4 on both benchmark and omics databases. The plots corresponding to the MMGNN model are illustrated in Figure 5.4 of Chapter 5. The scatter plots corresponding to the rest of the existing algorithms, namely mgRBM and TCCA as well as proposed D2GDA framework are presented in Figure 6.5. The shared subspace is represented with 2048, 50, 80, 50, $64|V|B$, and 50 features by MDBM, mgRBM, DACCA, TDDCCA, TCCA, and MMGNN, respectively, where $|V|$ and $B$ denote the number of input views and total number of batches, considered for a particular data set. The architecture for each of these models follows the same as described in the corresponding papers. From the results reported in Table 6.9, it is evident that significant improvement in classification accuracy is achieved by the proposed architecture in comparison to the existing multi-view approaches based on both consensus and complementary principles on all the benchmark databases, except for TCCA and DACCA models on Digits and AwA databases, respectively. In case of omics data sets, the results are presented in Table 6.10, which signify that the proposed model outperforms the six existing methods for all the five real-life cancer data sets considered. Statistical significance analysis reveals that out of total 60 cases, the proposed model achieves significantly better p-values for 56 cases and better but not significant p-values in the rest 4 cases.

Table 6.10: Comparative Performance Analysis of Both Consensus and Complementary Principle Based Approaches on Omics Data Sets

| Data | Different Metrics | | MDBM | mgRBM | DACCA | TCCA | TDDCCA | MMGNN | D2GDA |
|---|---|---|---|---|---|---|---|---|---|
| CESC | Train-Test | | 48.08 | 63.46 | 47.12 | 61.54 | 53.98 | 55.77 | **78.85** |
| | 10-fold CV | Mean | 52.50 | 63.33 | 39.81 | 60.19 | 78.20 | 69.17 | **84.17** |
| | | Median | 54.17 | 62.50 | 39.42 | 61.54 | 78.20 | 66.67 | **83.33** |
| | | StdDev | 17.59 | 9.78 | 5.37 | 5.74 | **0.06** | 13.64 | 6.15 |
| | | Paired-$t$:p | 3.15E-04 | 7.68E-05 | 7.72E-08 | 1.38E-06 | 6.80E-03 | 9.36E-03 | - |
| | | Wilcoxon:p | 3.98E-03 | 3.58E-03 | 2.52E-03 | 2.53E-03 | 6.23E-03 | 1.30E-02 | - |
| CRC | Train-Test | | 26.15 | 76.15 | 85.77 | **86.92** | 72.50 | 74.62 | 85.50 |
| | 10-fold CV | Mean | 54.07 | 70.37 | 81.69 | 82.92 | 48.80 | 80.00 | **87.04** |
| | | Median | 70.37 | 74.07 | 81.54 | 83.85 | 48.76 | 81.48 | **88.89** |
| | | StdDev | 24.33 | 8.37 | 3.12 | 3.55 | **0.30** | 4.35 | 4.00 |
| | | Paired-$t$:p | 1.25E-03 | 2.41E-04 | 7.28E-03 | 3.39E-02 | 1.60E-10 | 9.58E-05 | - |
| | | Wilcoxon:p | 2.38E-03 | 2.47E-03 | 1.09E-02 | 2.97E-02 | 2.53E-03 | 3.56E-03 | - |
| KIDNEY | Train-Test | | 69.08 | 86.84 | 96.71 | 96.71 | 51.59 | 89.47 | **98.68** |
| | 10-fold CV | Mean | 70.97 | 91.94 | 96.64 | 96.58 | 42.48 | 95.16 | **99.68** |
| | | Median | 67.74 | 91.94 | 96.38 | 96.71 | 42.43 | 93.55 | **100.00** |
| | | StdDev | 10.20 | 7.17 | 1.09 | 1.19 | **0.22** | 3.80 | 1.02 |
| | | Paired-$t$:p | 4.45E-06 | 4.82E-03 | 2.79E-05 | 1.62E-04 | 2.20E-16 | 1.30E-03 | - |
| | | Wilcoxon:p | 1.95E-03 | 8.57E-03 | 2.49E-03 | 3.31E-03 | 2.52E-03 | 7.03E-03 | - |
| LGG | Train-Test | | 65.05 | 72.04 | 65.59 | 81.18 | 66.85 | 71.51 | **90.32** |
| | 10-fold CV | Mean | 27.63 | 70.26 | 59.35 | 77.89 | 57.14 | 72.11 | **98.68** |
| | | Median | 18.42 | 68.42 | 58.87 | 77.63 | 57.00 | 72.37 | **98.68** |
| | | StdDev | 14.90 | 12.09 | 3.42 | 4.51 | **0.39** | 3.96 | 1.39 |
| | | Paired-$t$:p | 4.59E-08 | 1.89E-05 | 2.00E-11 | 2.61E-08 | 2.31E-15 | 1.92E-09 | - |
| | | Wilcoxon:p | 2.34E-03 | 2.52E-03 | 2.53E-03 | 2.46E-03 | 2.53E-03 | 2.40E-03 | - |
| LUNG | Train-Test | | 87.91 | 87.55 | 95.05 | 93.41 | 67.42 | 90.84 | **96.34** |
| | 10-fold CV | Mean | 61.25 | 86.96 | 95.71 | 95.42 | 67.72 | 82.14 | **97.32** |
| | | Median | 42.86 | 90.18 | 95.60 | 95.24 | 67.65 | 93.75 | **98.21** |
| | | StdDev | 23.81 | 11.67 | 1.17 | 1.00 | **0.38** | 19.32 | 2.95 |
| | | Paired-$t$:p | 3.24E-04 | 2.95E-03 | 9.29E-02 | 5.71E-02 | 1.44E-10 | 1.59E-02 | - |
| | | Wilcoxon:p | 2.49E-03 | 2.52E-03 | 1.01E-01 | 5.71E-02 | 2.53E-03 | 1.03E-02 | - |

## 6.6 Conclusion

The primary contributions of the current study include (a) formulation of a loss function, based on the concept of HSIC, to capture the relevant cross-view dependency in terms of coherent and/or complementary knowledge from the given pair of input views; (b) integrating the principle of cross-view dependency learning with the objective of MDDBM framework; (c) determining the data specific architecture of D2GDA model based on the estimated error bound; (d) demonstrating the generalization ability of the D2GDA model both theoretically as well as experimentally; and finally, (e) illustrating the efficacy of the D2GDA model on different domains of application, namely, object recognition, document classification, multilingual categorization, and cancer subtype identification.

In this study, a loss function is developed to efficiently represent the cross-modal depen-

dency across several modalities in terms of coherent as well as complementary structures of the given multi-view data. The MDDBM architecture includes the modality-specific characteristics as well as supervised information of sample categories into the joint subspace. Incorporating the loss function, corresponding to the proposed cross-view dependency analysis, into the learning objective of MDDBM architecture enables the D2GDA model to encapsulate the latent probability distribution of the given multimodal data as well as predict class labels of the given observations. The error analysis, generalization ability, and convergence analysis establish the efficacy of the proposed model. The comparative performance analysis demonstrates the proficiency of the proposed model on several multi-view data sets, considering both training-testing and 10-fold CV.

Combining information from multiple modalities is particularly challenging when the input modalities involve both image and non-image information. It is primarily due to the fact that as opposed to the non-image counterparts, the image modalities embody neighbourhood information which needs to be encapsulated properly for efficient representation of the given input data. A $B_1 \times B_2$ image can be visualized as a point on the hypercube in $\Re^{B_1 B_2}$ where each pixel is realized as a coordinate axis of the hypercube. So, all the $B_1 \times B_2$ images can be considered as different points on the hypercube. Now, only some specific points on the hypercube form the meaningful naturally occurring images. Consequently, if few points are uniformly sampled from the hypercube, it is most likely that noisy images will be obtained. Hence it can be said that the points lie on the low dimensional manifold, embedded in high dimensional pixel space ($\Re^{B_1 B_2}$). So, for proper characterization of the images, the corresponding manifold needs to be appropriately modelled from the given input images such that moving into the latent space results into moving on the manifold. Now, Laplacian eigenmap builds a graph incorporating the intrinsic geometry of the given data and considers eigenvectors corresponding to the Laplacian of the graph to obtain the neighbourhood preserving low dimensional embedding of the high dimensional data. Thus, in the next chapter, the concept of Laplacian eigenmap is judicially incorporated into the architecture of MDDBM model for efficient representation and classification of images into multiple categories.

# Chapter 7

# Discriminative Deep Joint Laplacian Embedding for Spatial Proximity Analysis

## 7.1 Introduction

Analysis of multi-view data becomes difficult when the relationships between the given multiple views are required to be explored in case of input heterogeneous data. The heterogeneous nature of data originates due to the wide range of variations in data types, formats, or even acquisition sensors, considered in different views of the input data. However, it can be noted from Chapter 5 and Chapter 6 that the consistency and diversity of different view representations can be effectively utilized to obtain a comprehensive and descriptive joint representation of the given input data, which in turn, stimulates the proficiency of the multi-view predictive models. Consolidating information from multiple views is particularly challenging when the input views involve both image and non-image information. It is primarily due to the fact that as opposed to the non-image counterparts, the image modalities embody neighbourhood information which requires to be encapsulated properly for efficient representation of the input data. In such scenario, different feature sets are commonly considered to represent the innate properties of the input image while learning joint subspace from the given multiple image and non-image modalities. This results into not only loss in information but also affects the classification performance of the overall system. Hence, combining one-dimensional non-image modalities with the multi-dimensional image modalities is a promising research problem with numerous applicability.

A substantial amount of research work has been carried over the past few years for efficient characterization and classification of images into multiple categories. Representing color and textural properties of an image through several local and global features has already been discussed in Section 3.1 of Chapter 3. However, in case of the appearance based methods, an image of dimension $B_1 \times B_2$ is represented by a vector in $B_1 \times B_2$ dimensional space. In practice, the $B_1 \times B_2$ dimensional space is too large to enable robust prediction of the corresponding images into different categories. A conventional approach

to resolve this issue is to consider dimensionality reduction techniques [14,113,116,170,195]. Out of these methods, two of the most acknowledged techniques are principal component analysis (PCA) [195] and linear discriminant analysis (LDA) [14].

PCA is an eigenvector method developed to model linear variation in high-dimensional data. PCA performs dimensionality reduction by projecting the original $B$-dimensional data onto the $b(<< B)$-dimensional linear subspace spanned by the leading eigenvectors of the covariance matrix of the given input data. The goal of the method is to find a set of mutually orthogonal basis functions that capture the directions of maximum variance in the data and for which the coefficients are pairwise decorrelated. For linearly embedded manifolds, PCA is guaranteed to discover the dimensionality of the manifold and produces a compact representation. On the other hand, LDA is a supervised learning algorithm. It searches for the projection axes on which the data points of different classes are far from each other while requiring data points of the same class to be close to each other. Unlike PCA which encodes information in an orthogonal linear space, LDA encodes discriminating information in a linearly separable space using bases that are not necessarily orthogonal. However, both PCA and LDA effectively consider only the Euclidean structure. They fail to discover the underlying structure, if the corresponding images lie on a non-linear submanifold hidden in the image space.

Over the last few years, various methods have been developed to discover the non-linear structure of the manifold from the high-dimensional input space, which include Laplacian eigenmaps [16], locally linear embedding [162], and isomap [186]. Geometric deep learning is a relatively nascent field that has attracted significant attention in the recent years, since it integrates the concept of aforementioned manifold learning methods with the representation ability of deep frameworks. In order to identify the low-dimensional manifold embedded in high-dimensional ambient space, graph regularized restricted Boltzmann machine (GraphRBM) has been developed in [24]. It incorporates a graph regularized term into the energy function of the restricted Boltzmann machine (RBM). It is extended to deep model by learning a stack of GraphRBM based modules, which is termed as full GraphRBM-based deep belief network (fGraphDBN). In [83], a Riemannian network architecture, referred to as SPDNet, has been introduced by incorporating the advantages of symmetric positive definite (SPD) matrix based non-linear learning into a deep framework. The convolutional restricted Boltzmann machine (CRBM) is developed in [152], which incorporates the translation invariance property of convolution operation into the energy function of RBM to reduce the number of parameters needed to be learned for proper training of the model. These non-linear methods do yield impressive results on some benchmark artificial data sets. However, they produce maps that are defined only on the training data points and the evaluation procedure of the maps on unknown test data points remains unclear [68].

Hence, a geometrically motivated deep predictive model is required to be developed, which can process multiple image and non-image modalities simultaneously. In case of multi-view analysis, it is essential that descriptive and comprehensive information is efficiently extracted from all the views of the given input data and appropriately reflected in the joint subspace. If one or more input views correspond to image modalities, then it should be ensured that the innate topological properties of each of the input image modalities are properly preserved. Hence, for proper characterization of a particular image view, the corresponding manifold needs to be appropriately modelled from the given input im-

ages in such a way that moving into the latent space results into moving on the manifold. In case of multiple image views, the intrinsic geometric structures of the corresponding image manifolds need to be consolidate appropriately in the joint subspace. In order to address the multi-view classification problem, it is necessary that similarity in the latent space implies similarity in the corresponding concepts. Non-linearity is another important aspect that needs to be addressed in multi-view learning.

In this regard, a novel deep learning model, termed as discriminative deep joint Laplacian embedding (D2JLE), is developed based on the framework of multimodal discriminative deep Boltzmann machine (MDDBM), introduced in Chapter 4. In order to recognize and represent the latent geometric structures of a particular image view, an objective function is developed based on the theory of Laplacian eigenmap, which can efficiently encapsulate the underlying low-dimensional embedding from the input high-dimensional pixel space. As two graph Laplacian matrices are not comparable even if they correspond to the same set of observations, an objective function is formulated based on the concept of simultaneous diagonalization of Laplacians to consolidate the topological properties of multiple image views. In the proposed D2JLE model, these two objective functions are judiciously integrated with the learning objective of the MDDBM to capture the imperative information from both image and non-image views. The proposed model is developed based on the hypothesis that not all the modalities contribute uniformly in providing the discriminative information of sample categories. Hence, the relevance of each modality is evaluated based on the discrimination criterion of the corresponding modality and the joint subspace is learned from the weighted combination of the individual subspaces. An upper bound on the error probability of the proposed model is estimated in terms of the model architecture, which allows the model to determine the optimal architecture of the model for each database considered. The proposed D2JLE model is further consolidated with convergence analysis. As the proposed model is developed based on the MDDBM framework, it can efficiently encapsulate the underlying non-linear data distribution over the space of multimodal inputs. The MDDBM framework considers supervised information of sample categories at each layer of the network, as a result of which the discriminative ability of the D2JLE model is enhanced. Also, no additional classifier is required to be employed in case of the proposed D2JLE model for classifying the given observations into different categories. The proficiency of the proposed model is demonstrated on four benchmark databases, three HEp-2 cell image data sets, and the real-life Virus database for both training-testing and 10-fold cross validation (CV).

The rest of this chapter is organized as follows: In Section 7.2, the basic notions in spectral geometry of manifolds and graphs are summarized. Section 7.3 describes the architecture as well as the learning of the proposed D2JLE model. Different aspects of the proposed model, which include error analysis and convergence analysis, are discussed in Section 7.4. In Section 7.5, the proficiency of the D2JLE framework is analyzed with reference to several state-of-the-art approaches on various benchmark, HEp-2 cell, and Virus data sets. Concluding remarks are provided in Section 7.6.

## 7.2 Basics of Graph Laplacian Eigenmap

A Riemannian manifold is a differentiable manifold whose tangent spaces are equipped with an inner product operation. Let, $\mathcal{M}$ be a compact, $b$-dimensional Riemannian manifold, which is embedded into a $B(>> b)$-dimensional Euclidean space. The structure of the manifold can be studied by means of the Laplace-Beltrami operator $\Delta_{\mathcal{M}}$ on $\mathcal{M}$ [159, 196], defined axiomatically through the Stokes identity as

$$\int_{\mathcal{M}} f \Delta_{\mathcal{M}} g d\zeta = \int_{\mathcal{M}} \langle \Delta_{\mathcal{M}} f, \Delta_{\mathcal{M}} g \rangle d\zeta, \tag{7.1}$$

where $f, g : \mathcal{M} \mapsto \Re$ are smooth scalar fields on the manifold $\mathcal{M}$, $d\zeta$ is a volume element, $\Delta_{\mathcal{M}}$ is the intrinsic gradient, and $\langle \cdot, \cdot \rangle$ represents the Riemannian metric, which is the inner product on the tangent space. The Laplace-Beltrami operator reduces to the common Laplacian operator $\Delta f = \sum_{j=1}^{B} \frac{\partial^2 \hat{f}}{\partial x_j^2}, \hat{f} : \mathcal{M} \mapsto \Re$, where $\hat{f}$ is twice differentiable and $x_j$ being the Euclidean coordinates, if the manifold is an open subset of the Euclidean space. The eigenfunction $\nu_j$ on $\Delta_{\mathcal{M}}$, satisfying the Laplacian eigenvalue problem $\Delta_{\mathcal{M}} \nu_j = \lambda_j \nu_j$, is often referred to as manifold harmonic. The eigenvalues of $\nu_j$ are analogous to frequencies in the Euclidean case and the eigenfunctions to the basis functions related to sine and cosine terms [196]. Low frequency eigenfunctions, corresponding to the smallest eigenvalues, describe the global structure of the manifold, while the high frequency eigenfunctions capture the details [159].

A common way to discretize a manifold is by a graph structure. An undirected weighted graph is an ordered pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, consisting of the set of nodes $\mathcal{V} = \{\upsilon_1, \cdots, \upsilon_p, \cdots, \upsilon_N\}$ and the set of edges $\mathcal{E}$, which are unordered pairs of elements of $\mathcal{V}$. Such a graph is constructed from a data set by assigning each data point sampled on an underlying manifold to a node. For instance, in case of analysis of a given set of images, each node of the graph represents a particular image and the number of nodes in the graph equals the number of the images in the input set [229]. The edge weights represent a notion of similarity between data points. Two nodes $\upsilon_p$, $\upsilon_r$ are connected with an edge $a_{pr} \geqslant \Theta, \Theta \in \Re_0^+$. A popular choice for the graph edges is to use a Gaussian kernel, $a_{pr} = \exp(-\frac{||\upsilon_p - \upsilon_r||_F^2}{2\sigma^2})$, where $F$ denotes the Frobenius norm and $\sigma > 0$ is the kernel bandwidth [16]. However, various other kernels can also be considered depending on the problem definition [230].

The eigenvectors of a graph Laplacian discretize the eigenfunctions of the Laplace-Beltrami operator [16]. Let us consider $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-\frac{1}{2}}$ represents the symmetric normalized graph Laplacian. Here, the matrix $\mathbf{A} = [a_{pr}]_{N \times N}$ is referred to as the adjacency matrix where $N$ denotes the total number of input observations and $\mathbf{D} = \text{diag}(\sum_{\text{r} \neq \text{p}} \text{a}_{\text{pr}})$ represents the diagonal degree matrix, containing the degree of each node, that is, the sum of weights connected to that node.

**Property 7.1.** $\mathbf{L}$ *is always positive semi-definite.*

From Property 7.1, it is evident that $\mathbf{L}$ is always symmetric. Since $\mathbf{L}$ is a real matrix, it is also a Hermitian matrix. The spectral theorem of Hermitian matrices follows next.

**Theorem 7.1.** *Every Hermitian matrix is unitarily diagonalizable [77].*

Hence, $\mathbf{L}$ can be decomposed as $\mathbf{L} = \mathbf{\Phi}^T\mathbf{\Lambda}\mathbf{\Phi}$, such that $\mathbf{\Phi}^T\mathbf{\Phi} = \mathbf{I}_N$, where $\mathbf{I}_N$ denotes the identity matrix of order $N$, $\mathbf{\Phi}$ signifies the matrix of column eigenvectors of $\mathbf{L}$, $\mathbf{\Lambda} = \text{Diag}(\gamma_1, \gamma_2, \cdots, \gamma_N)$ represents the diagonal matrix of corresponding eigenvalues $0 = \gamma_1 \leqslant \gamma_2 \leqslant \cdots \leqslant \gamma_N$, and $T$ indicates the transpose operator. The eigenvectors of the Laplacian $\mathbf{L}$ can be considered as a discretization of eigenfunctions $\nu_j$ of the continuous operator $\Delta_{\mathcal{M}}$ on $\mathcal{M}$. It is shown in [203] that under certain conditions on the discretization of the Laplacian, they converge to the continuous counterparts.

Several methods have been developed over the years for geometric construction of the underlying manifolds with eigenvectors and eigenvalues of the Laplacian [15, 162, 186]. A popular spectral embedding technique is Laplacian eigenmap (LE) algorithm [16]. It captures the intrinsic low-dimensional structure of a manifold by finding an optimal embedding, which preserves the neighbourhood property of the input data. This can be posed as the minimization problem

$$\underset{\mathbf{H}\in\Re^{N\times b}}{\arg\min} \ \mathbf{tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}) \ \text{ such that } \ \mathbf{H}^T\mathbf{D}\mathbf{H} = \mathbf{I}_b \tag{7.2}$$

where $\mathbf{tr}(\cdot)$ symbolizes the trace operator. Now, the problem in (7.2) has an analytic solution $\mathbf{H} = (\mathbf{h}_1, \cdots, \mathbf{h}_b)$ containing the first $b$ eigenvectors of $\mathbf{L}$. Thus, effectively embedding of the data signifies the smallest eigenvectors of the graph Laplacian. Such an embedding is referred to as LE. The neighbourhood-preserving property of the eigenmaps is related to the fact that the smallest or low-frequency eigenvectors of the Laplacian vary smoothly on the manifold.

By using the objective function of (7.2), two neighboring points $\upsilon_p$ and $\upsilon_r$ in the original space, represented with high value of $a_{pr}$, incur a penalty if they are mapped far apart in the embedding space. Therefore, minimization of (7.2) ensures that the local neighborhood properties are preserved. For $b$-dimensional embedding problem, the constraint, presented in (7.2), prevents the solution to collapse onto a subspace having dimension less than $b-1$. ($b$ in cases where orthogonality to the constant vector is required). Let $\mathbf{1}$ be the constant function taking 1 at each vertex. It is evident that $\mathbf{1}$ is an eigenvector with zero eigenvalue. If the graph is connected, $\mathbf{1}$ is the only eigenvector for zero eigenvalue. The diagonal degree matrix $\mathbf{D}$ provides a natural measure on the vertices of the graph. The higher value of $\mathbf{D}_{pp}$, corresponding to the $p$-th vertex, signifies greater importance of the vertex. It is shown in [16] that solution of the minimization problem is equivalent to that of the generalized eigenvalue problem of the graph Laplacian, $\mathbf{L}\vartheta = \chi\mathbf{D}\vartheta$, whose eigenvectors give the optimal embedding, with the diagonal degree matrix $\mathbf{D}$. Here, $\chi$ symbolizes the eigenvalue corresponding to the eigenvector $\vartheta$ of $\mathbf{L}$.

## 7.3 Proposed Model

In this section, the geometrically motivated deep predictive model, termed as D2JLE, is proposed. It can appropriately process multiple image and non-image modalities simultaneously, and obtain a joint subspace reflecting the inherent characteristics of each of the given input modalities. At first, the architecture of the proposed D2JLE model is described. The proposed approach for efficient extraction and propagation of the locality preserving properties of an input image modality is then discussed. The proposed method

of combining the topological properties of multiple image modalities is discussed next. A discriminative relevance measure is introduced based on which the imperative information from all the modalities is consolidated properly in the joint subspace. Finally, the learning objective of the proposed D2JLE model is outlined for proper characterization as well as classification of the given observation into various categories.

### 7.3.1 Architecture of Proposed D2JLE Model

In case of multimodal data analysis, it is assumed that each view has a fundamentally distinct representation of the underlying data distribution. So, the joint subspace should be able to reflect the non-linear structures embedded in the given input modalities, along with the supervised information of sample categories. Hence, the architecture of the proposed D2JLE model is developed based on the framework of multimodal discriminative deep Boltzmann machine (MDDBM), introduced in [104] and presented in Chapter 4. Therefore, the joint subspace, learned from the D2JLE model, is expected to encapsulate the underlying non-linear data distribution of the given observations. Since the class nodes are included into the framework, the hidden subspaces of the proposed model contain the supervised information of sample categories, which in turn, enhances the discriminative ability of the model. Also, it allows the model to predict the class label of the given observations, without employing any additional classifier.

In the proposed D2JLE model, let $M$ be the total number of input views, out of which $M_1 > 1$ views corresponds to images, $M_2 > 0$ views corresponds to non-image information, and $M_1 + M_2 = M$. While the input image views are represented by the normalized symmetric Laplacian matrices $\{\mathbf{L}^1, \cdots, \mathbf{L}^{M_1}\}$, the input non-image views are denoted by $\{\mathbf{v}^1, \cdots, \mathbf{v}^{M_2}\}$. Also, $\mathbf{y}_p = \{y_{p1}, \cdots, y_{pc}, \cdots\}$ denotes the input class label information of $p$-th observation. Let us assume that $L_1 > 0$ signifies the number of modality-specific hidden layers in the architecture and $L_2 > 0$ refers to the number of joint hidden layers. While $\mathbf{h}_p^{lm} = \{h_{p1}^{lm}, \cdots, h_{pj}^{lm}, \cdots\}$ represents the $l$-th modality-specific hidden representation of the $m$-th modality for $p$-th observation, the joint hidden representation, corresponding to the $l$-th layer of $p$-th observation, is referred to as $\mathbf{h}_p^l = \{h_{p1}^l, \cdots, h_{pj}^l, \cdots\}$. Here, the number of nodes in a representation is expressed by the corresponding capital letter. For example, the number of nodes in $\mathbf{h}^{lm}$ is denoted by $H^{lm}$. The architecture of the proposed D2JLE model is depicted in Figure 7.1.

The bidirectional weight parameters $w_{jk}^{(l+1)m}$, $w_{jk}^{(L_1+1)m}$, and $w_{jk}^{(l+1)}$ connect $j$-th hidden node of $l$-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 $L_1$ 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 $u_{cj}^{lm}$ and $u_{cj}^{l}$ connect the $c$-th class node to the $j$-th hidden node of the $l$-th modality-specific hidden layer from $m$-th modality, and $c$-th class node to $j$-th hidden node of $l$-th joint hidden layer, respectively. The bias parameters $b_j^{lm}$, $b_j^l$, and $d_c$ are associated with the $j$-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 $c$-th class node, respectively. Thus, the supervised information of sample categories can be appropriately incorporated at each layer of the architecture through proper learning of the set of parameters associated with the class nodes, which in turn, enhances the proficiency of the proposed framework.

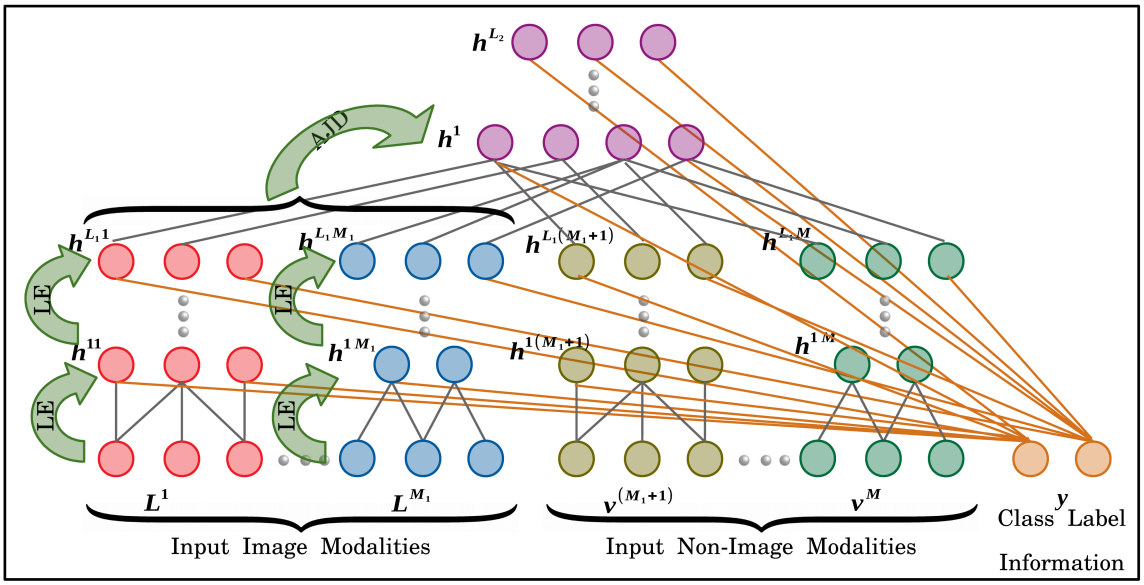



Figure 7.1: Illustration of proposed D2JLE framework.

### 7.3.2 Encapsulating Image Manifold

In this section, the objective function considered in the proposed model for the analysis of an image modality is discussed in details. A $B_1 \times B_2$ image can be visualized as a point on the hypercube in $\Re^{B_1 B_2}$, where each pixel is realized as a coordinate axis of the hypercube. Let us consider a normalized image with three pixels, then $(0.5, 0.5, 0.5)$ denotes the middle point in the hypercube, $(0, 1, 0)$ represents a vertex, and $(0.5, 1, 1)$ signifies an edge of the hypercube. So, all the $B_1 \times B_2$ images can be considered as different points on the hypercube. Now, only some specific points on the hypercube form the meaningful naturally occurring images. Consequently, if few points are uniformly sampled from the hypercube, it is most likely that noisy images will be obtained. Let us consider two images of a person with different facial expressions which are represented by two different points in the hypercube. If a linear path is drawn between the two points, then the intermediate points on the path may not correspond to the faces of that person, which signify that the transition is not a smooth one. However, if a specific path is followed between the two points such that the intermediate points correspond to the faces of the person, then the transition of the faces will be smooth. Hence, it can be said that the points lie on the low dimensional face manifold, embedded in high dimensional pixel space ($\Re^{B_1 B_2}$). Traversing between any two points while remaining on the manifold ensures smooth transition between the images. Therefore, for proper characterization and classification of the images, the corresponding manifold is required to be appropriately modelled from the given input images such that moving into the latent space results into moving on the manifold in the original space. Now, similar images tend to have similar spatial properties in a neighbourhood. Hence, the mapping from the high-dimensional pixel space to low-dimensional latent space should preserve the intrinsic geometric structure of the data, that is, nearby points in the original space should remain close in the latent space as well.

In [15], Belkin and Neogi have developed the non-linear data representation technique,

termed as LE algorithm, to obtain a neighbourhood preserving low-dimensional embedding of the image manifold. The concept of LE is considered in the proposed model for the analysis of the input image modality of the given data. The justification for considering the LE algorithm comes from the role of the Laplace-Beltrami operator in providing an optimal embedding for the manifold. The manifold is approximated by the adjacency graph computed from the data points. The Laplace-Beltrami operator is approximated by the weighted Laplacian of the adjacency graph with weights chosen appropriately. Thus, the embedding for the data basically approximates the eigenmaps of the Laplace-Beltrami operator, which are intrinsically defined on the corresponding manifold. It is shown in [16] that by preserving the spatial properties in the embedding, the algorithm implicitly emphasizes the natural categories in the data. It is also required to be mentioned here that the biological perceptual apparatus is confronted with high-dimensional stimuli from which it recovers the low-dimensional structure. If the approach, considered for recovering such low-dimensional structure, is inherently local, as in case of the LE algorithm, then it may serve as the basis for the emergence of categories in biological perception [15]. Furthermore, the neighbourhood preserving characteristics of the LE algorithm makes it relatively insensitive to outliers and noise.

In the proposed model, the weighted adjacency graph $\mathbf{A} = [a_{pr}]_{N\times N}$ is constructed incorporating the neighbourhood information of the given data where each input image is considered as a node of the graph. Based on the notion of the Laplacian $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-\frac{1}{2}}$ of the graph, a low-dimensional embedding of the data is computed using (7.2). The objective function corresponding to the optimization problem of (7.2) is defined as

$$E_{LE}(\mathbf{L},\mathbf{H}) = \sum_{p=1}^{N-1}\sum_{r=1,r\neq p}^{N} a_{pr}\left\{\sum_{j=1}^{b}\Big(h_{pj}-h_{rj}\Big)^2\right\} + \sum_{j=1}^{b}\kappa_j\left\{1-\sum_{p=1}^{N}\Big(\sum_{r=1,r\neq p}^{N}a_{pr}\Big){h_{pj}}^2\right\} - \sum_{j=1}^{b-1}\sum_{k=(j+1)}^{b}\psi_{jk}\left\{\sum_{p=1}^{N}\Big(\sum_{r=1,r\neq p}^{N}a_{pr}\Big)h_{pj}h_{pk}\right\}, \tag{7.3}$$

where $\mathbf{h}_p = \{h_{p1},\cdots,h_{pj},\cdots,h_{pb}\}$ and $\mathbf{h}_r$ denote the $b$-dimensional embedding corresponding to $p$-th and $r$-th observations, respectively. Here, $\kappa_j$ and $\psi_{jk}$ represent the Lagrange multipliers. The embedding obtained using (7.2) may be viewed as a discrete approximation to a continuous map that naturally arises from the intrinsic geometry of the image manifold.

There exist various proficient methods [28, 29] in literature to solve the objective function in (7.3) and obtain the optimal embedding $\mathbf{h}_p$ for the $p$-th observation of the input data. However, the current study deals with the classification problem. Hence, it is important that, in addition to the geometric structures of the image manifolds, the discriminative information of the sample categories is also reflected in the corresponding embedding. Now, the architecture and learning objective of multimodal discriminative deep Boltzmann machine (MDDBM), introduced in Chapter 4, have already been described to encapsulate the underlying non-linear data distribution of the given observations. The class nodes of the MDDBM framework incorporates the supervised information at each layer of the model, which not only enhances the discriminative ability of the latent subspaces, but also allows the model to predict the class labels of the observations without employing any additional classifiers.

Now, the MDDBM model is developed for the analysis of multi-view data. However, this section addresses the problem of efficient characterization and classification of a particular image modality. So, the energy function of MDDBM model, presented in (4.8), is required to be defined to capture the intrinsic characteristics of a given particular modality. This can be achieved by considering $M = 1$ and $\mathbf{h}^l = 0,\ \forall l > L_0$; in (4.8). The corresponding model is termed as discriminative deep Boltzmann machine (DDBM) and the energy function is denoted by $E_{ddbm}(\mathbf{v}, \mathbf{h}, \mathbf{y})$. In order to encapsulate the locality preserving properties of a given image modality as well as the supervised information of sample categories, the theory of LE is judiciously integrated with the DDBM framework and the resultant predictive model is referred to as discriminative deep architecture for image analysis (D2AIA). The learning objective of the proposed D2AIA model can be obtained by incorporating the objective function of the LE algorithm, presented in (7.3), corresponding to each given observation, into the energy function of DDBM model, which turns out to be

$$
\begin{aligned}
&E_{d2aia}(\mathbf{L}_p, \mathbf{h}_p, \mathbf{y}_p) = E_{le}(\mathbf{L}_p, \mathbf{h}_p) + E_{ddbm}(\mathbf{L}_p, \mathbf{h}_p, \mathbf{y}_p) \\
&= \sum_{l=1}^{L_1} \sum_{r=1, r\neq p}^{N} a_{pr} \left\{ \sum_{j=1}^{H^l} \left( h_{pj}^l - h_{rj}^l \right)^2 \right\} + \sum_{l=1}^{L_1} \sum_{j=1}^{H^l} \kappa_j^l \left\{ 1 - \Big( \sum_{r=1, r\neq p}^{N} a_{pr} \Big) (h_{pj}^l)^2 \right\} \\
&- \sum_{l=1}^{L_1} \sum_{j=1}^{H^l - 1} \sum_{k=(j+1)}^{H^l} \psi_{jk}^l \left\{ \Big( \sum_{r=1, r\neq p}^{N} a_{pr} \Big) h_{pj}^l h_{pk}^l \right\} - \sum_{l=1}^{L_1} \sum_{j=1}^{H^l} \sum_{k=1}^{H^{(l+1)}} h_{pj}^l w_{jk}^{(l+1)} h_{pk}^{(l+1)} \\
&- \sum_{l=1}^{L_1} \sum_{c=1}^{Y} \sum_{j=1}^{H^l} y_{pc} u_{cj}^l h_{pj}^l - \sum_{l=1}^{L_1} \sum_{j=1}^{H^l} b_j^l h_{pj}^l - \sum_{c=1}^{Y} d_c y_{pc}. \qquad (7.4)
\end{aligned}
$$

Here, $E_{LE}(\mathbf{L}, \mathbf{H}) = \sum_{p=1}^{N} E_{le}(\mathbf{L}_p, \mathbf{h}_p)$, $E_{d2aia}(\mathbf{a}_p, \mathbf{h}_p, \mathbf{y}_p)$ signifies the energy of the state $\{\mathbf{a}_p, \mathbf{h}_p, \mathbf{y}_p\}$ of the D2AIA model, corresponding to the $p$-th image of the given data, $L_1$ denotes the total number of layers considered in the model, $\mathbf{h}_p^l$ represent the embedding at $l$-th layer of the model corresponding to the $p$-th input image, and $H^l$ indicates the number of nodes at $l$-th layer of the model. The parameter set of the proposed D2AIA model is given by $\theta_{d2aia} = \{a_{pr}, w_{jk}^l, u_{cj}^l, b_j^l, d_c\}$. Thus, based on the energy function, defined in (7.4), the proposed model can efficiently encapsulate the underlying image manifold, where the topological properties of the input images are properly preserved, enhance the discriminative ability of the obtained embeddings by incorporating supervised information of the sample categories, and propagate the spatial as well as discriminative properties through each layer of the model.

### 7.3.3 Combining Multiple Image Manifolds

In case of multi-view data analysis, it is required that the inherent characteristics of each of the input views of the given data are precisely preserved in the joint subspace. With two or more image modalities of the input data, the image manifolds, represented by the eigenvectors of the corresponding graph Laplacians, must be consolidated properly in the joint subspace such that the underlying non-linear data distribution of the given observations can be encapsulated efficiently. Now, two graph Laplacian matrices are not

comparable even if they correspond to the same set of observations [41]. This can possibly be explained by the following arguments. First, they do not share the same intrinsic structure, that is, they are not isometric. More generally, for non-isometric manifold, which is usually the case in real applications, the Laplacian eigenvectors can differ drastically, causing the information from different image modalities mutually incomparable. Second, the Laplacian eigenvectors are defined only up to a sign, that is, if $\phi$ is an eigenvector of $\mathbf{L}$, then -$\phi$ is also an eigenvector with the same eigenvalue. Third, if the multiplicity of an eigenvalue is greater than one, the eigenvectors corresponding to the eigenvalue may be in a different order or subject to an orthonormal transformation. A common solution is to find eigenbases of the Laplacians simultaneously [42]. An important theorem regarding the simultaneous diagonalization of two matrices, say $\mathbf{A}_1$ and $\mathbf{A}_2$, is given by next theorem.

**Theorem 7.2.** *Two Hermitian matrices* $\mathbf{A}_1$ *and* $\mathbf{A}_2$ *can be diagonalized simultaneously, if and only if* $\mathbf{A}_1\mathbf{A}_2 = \mathbf{A}_2\mathbf{A}_1$ *[77].*

So, from Theorem 7.2, it is evident the graph Laplacians will be simultaneously diagonalizable, iff they commute. Simultaneous diagonalization implies that there exists a single set of orthonormal vectors $\underline{\mathrm{H}}$, which is referred to as joint eigenvectors, such that $\underline{\mathbf{H}}^T\mathbf{L}^m\underline{\mathbf{H}} = \mathbf{\Lambda}^m = \mathrm{diag}(\gamma_1^{\mathrm{m}}, \gamma_2^{\mathrm{m}}, \cdots, \gamma_{\mathrm{N}}^{\mathrm{m}})$, is the diagonal matrix of eigenvalues corresponding to the Laplacian matrix $\mathbf{L}^m$ of $m$-th image modality. Thus, the joint diagonalization allows to remove the ambiguities and incompatibilities between the given modalities. However, due to the differences between the modalities, discretization, and presence of noise, the Laplacian matrices rarely commute and have a joint eigenbasis. An approximate joint diagonalization [12] is possible by solving the following optimization problem:

$$\min_{\underline{\mathbf{H}} \in \Re^{N \times H}} \sum_{m=1}^{M_1} \|\mathrm{off}(\underline{\mathbf{H}}^{\mathrm{T}}\mathbf{L}^{\mathrm{m}}\underline{\mathbf{H}})\|_{\mathrm{F}}^2 \;\; \text{such that} \;\; \underline{\mathbf{H}}^{\mathrm{T}}\underline{\mathbf{H}} = \mathbf{I}_{\mathrm{H}}; \tag{7.5}$$

where $\mathrm{off}(\mathbf{X}) = \|\mathbf{X} - \mathrm{Diag}(\mathbf{X})\|_{\mathrm{F}}^2$ is the sum of squared off-diagonal elements. Here, $\mathrm{Diag}(\mathbf{X})$ denotes a diagonal matrix containing only the diagonal elements of $\mathbf{X}$, $M_1$ denotes the total number of input image modalities, and $H$ represents the number of smallest eigenvectors considered. For a symmetric matrix $\mathbf{L}^m$, the optimization problem, presented in (7.5), achieves the minimum value of zero with a minimizer $\underline{\mathbf{H}}$ containing the eigenvectors of $\mathbf{L}^m$. The objective function corresponding to the optimization problem of (7.5) is defined as

$$\begin{aligned} E_{AJD}(\mathbf{L}^1, \cdots, \mathbf{L}^{M_1}, \underline{\mathbf{H}}) &= H(H-1) \sum_{p=1}^{N-1} \sum_{r=(p+1)}^{N} \sum_{m=1}^{M_1} \left(a_{pr}^m\right)^2 \left\{ \prod_{j=1}^{H} (h_{pj} - h_{rj})^2 \right\} \\ &+ \lambda \sum_{p=1}^{N} \left(1 - \sum_{j=1}^{H} h_{pj}^2\right) + 2 \sum_{m=1}^{M_1} \sum_{p=1}^{N-1} \sum_{r=(p+1)}^{N} \sum_{s=p}^{N-1} \sum_{t=(r+1)}^{N} a_{pr}^m a_{st}^m \left\{ \prod_{j=1}^{H} (h_{pj} - h_{rj})(h_{sj} - h_{tj}) \right\}. \end{aligned} \tag{7.6}$$

Here, $\{\mathbf{L}^1, \cdots, \mathbf{L}^{M_1}\}$ represents the Laplacian matrices corresponding to the $M_1$ input image modalities. Thus, the topological properties of each of the image modalities can

be consolidated properly and included in the joint subspace using the objective function $E_{AJD}(\mathbf{L}^1, \cdots, \mathbf{L}^{M_1}, \underline{\mathbf{H}})$, presented in (7.6). In order to incorporate the supervised information of sample categories into the joint subspace and to capture the non-linear data distribution over the space of multimodal inputs, the proposed D2JLE model is considered in the current study. The learning objective of the proposed model is described in Section 7.3.5.

### 7.3.4 Computation of Relevance

In multimodal environment, all the modalities may not contribute equally in providing the discriminative information of sample categories. Hence, learning joint subspace from the modality-specific subspaces with equal weightage may degrade the overall performance of the model. So, the relevance of each of the modalities is required to be evaluated based on the discrimination ability of that particular modality. Subsequently, the joint subspace is learned from the weighted combination of individual subspaces. In this context, a relevance measure is proposed in the current study to evaluate each of the given input modalities, which is defined as follows:

$$\Gamma^m = \frac{\sum_{p=1}^{N-1} \sum_{r=p+1}^{N} \sum_{j=1}^{H^m} (h_{pj}^m - h_{rj}^m)^2 (1 - \sum_{c=1}^{Y} y_{pc} y_{rc})}{1 + \sum_{p=1}^{N-1} \sum_{r=p+1}^{N} \sum_{j=1}^{H^m} (h_{pj}^m - h_{rj}^m)^2 (\sum_{c=1}^{Y} y_{pc} y_{rc})}, \quad \forall m. \tag{7.7}$$

Here, $\Gamma_m$ represents the relevance of the $m$-th modality. The $\mathbf{h}_p^m$ and $\mathbf{h}_r^m$ signify the modality-specific subspaces corresponding to the $p$-th and $r$-th input observations of the $m$-th modality, respectively, whereas $\mathbf{y}_p$ and $\mathbf{y}_r$ denote the class vector for $p$-th and $r$-th input observations, respectively. The $H^m$, $N$, and $Y$ represent the dimension of modality-specific subspace corresponding to the $m$-th modality, the number of observations of the given input data, and the number of class labels, respectively. Now, $\Gamma_m$ is defined in (7.7) such that the numerator will have non-zero values only for observations belonging to different classes. In case of the denominator, the term under summation yields non-zero values only for samples belonging to the same class and remains quiescent otherwise. Thus, a high value of $\Gamma_m$ signifies high inter-class separability and low intra-class distance between the given observations, which implies better discriminative ability of the corresponding modality. The low $\Gamma_m$ value indicates low inter-class separability and high intra-class difference, which denotes overlap between observations of different classes. Therefore, incorporating the imperative information from all the input modalities into the joint subspace, based on the proposed relevance measure basically, emphasizes the modalities with high discriminability, which in turn, enhances the efficacy of the joint subspace in categorizing the observations into different classes.

### 7.3.5 Learning D2JLE Model Using Laplacian Eigenmap

In this section, the learning objective of the proposed D2JLE model, developed for efficient characterization as well as classification of the given image and non-image modalities of the input data, is discussed in details.

#### 7.3.5.1 Objective Function of Proposed Model

Since the proposed D2JLE model is developed based on the architecture of the MDDBM model, introduced in Chapter 4, the learning objective of the D2JLE model includes the energy function of the MDDBM model, presented in (4.8) of Chapter 4. Hence, the joint subspace of the D2JLE model can efficiently encapsulate the underlying non-linear data distribution over the space of multimodal inputs. Also, due to the presence of the class nodes, the supervised information of sample categories is incorporated into the hidden representation at each layer of the model, which not only enhances the discriminability of the model but also allows the model to predict the class label of the given observations. In order to capture the low-dimensional manifold, embedded into the high-dimensional pixel space, the objective function of (7.3), corresponding to the optimization problem of the LE algorithm, is included into the learning objective of the D2JLE model in such a way that moving into the modality-specific subspaces results into moving on the corresponding manifolds. Combining the objective function of LE algorithm with the energy function of the MDDBM model, it can be ensured that the topological properties of each modality are not only extracted efficiently from the given input images, but also propagated properly through the deep layers of the proposed model. Generally, the information from different modalities is mutually incomparable as the Laplacian eigenvectors, corresponding to the image modalities, differ drastically. Hence, the objective function of approximate joint diagonalization of Laplacians, presented in (7.6), is considered in the learning objective of the proposed model, so that the intrinsic geometric structures of the corresponding image manifolds can be consolidated properly and reflected in the joint subspace. Furthermore, the imperative information from all the modalities is combined in the joint subspace based on the discriminative relevance measure, defined in (7.7). Hence, the overall objective function of the proposed D2JLE model, corresponding to the $p$-th observation, is given by

$$
\begin{aligned}
&E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p) = E_{le}(\mathbf{L}_p, \mathbf{h}_p) + E_{ajd}(\mathbf{L}_p, \mathbf{h}_p) + E_{mddbm}(\mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p) \\
&= \sum_{m=1}^{M_1}\sum_{l=1}^{L_1}\sum_{r=1,r\neq p}^{N} a_{pr}^{m}\left\{\sum_{j=1}^{H^{lm}}\left(h_{pj}^{lm}-h_{rj}^{lm}\right)^2\right\} + \sum_{m=1}^{M_1}\sum_{l=1}^{L_1}\sum_{j=1}^{H^{lm}}\kappa_j^{lm}\left\{1-\Big(\sum_{r=1,r\neq p}^{N} a_{pr}^{m}\Big)(h_{pj}^{lm})^2\right\} \\
&\quad - \sum_{m=1}^{M_1}\sum_{l=1}^{L_1}\sum_{j=1}^{H^{lm}-1}\sum_{k=(j+1)}^{H^{lm}}\psi_{jk}^{lm}\left\{\Big(\sum_{r=1,r\neq p}^{N} a_{pr}^{m}\Big)h_{pj}^{lm}h_{pk}^{lm}\right\} \\
&\quad + H^1(H^1-1)\sum_{m=1}^{M_1}\Gamma_m\sum_{r=1,r\neq p}^{N}\left(a_{pr}^{m}\right)^2\left\{\prod_{j=1}^{H^1}(h_{pj}^{1}-h_{rj}^{1})^2\right\} + \lambda\Big(1-\sum_{j=1}^{H^1}(h_{pj}^{1})^2\Big) \\
&\quad + 2\sum_{m=1}^{M_1}\Gamma_m\sum_{r=1,r\neq p}^{N}\sum_{s=p}^{N-1}\sum_{t=(r+1)}^{N} a_{pr}^{m}a_{st}^{m}\left\{\prod_{j=1}^{H^1}(h_{pj}^{1}-h_{rj}^{1})(h_{sj}^{1}-h_{tj}^{1})\right\} - \sum_{m=1}^{M_2}\sum_{i=1}^{V^m}\sum_{j=1}^{H^{1m}} v_{pi}^{m}w_{ij}^{1m}h_{pj}^{1m} \\
&\quad - \sum_{m=1}^{M}\sum_{l=1}^{L_1-1}\sum_{j=1}^{H^{lm}}\sum_{k=1}^{H^{(l+1)m}} h_{pj}^{lm}w_{jk}^{(l+1)m}h_{pk}^{(l+1)m} - \sum_{m=1}^{M_1}\sum_{l=1}^{L_1}\sum_{c=1}^{Y}\sum_{j=1}^{H^{lm}} y_{pc}u_{cj}^{lm}h_{pj}^{lm} \\
&\quad - \sum_{m=1}^{M}\Gamma_m\sum_{j=1}^{H^{L_1m}}\sum_{k=1}^{H^1} h_{pj}^{L_1m}w_{jk}^{(L_1+1)m}h_{pk}^{1} - \sum_{l=1}^{L_2-1}\sum_{j=1}^{H^l}\sum_{k=1}^{H^{(l+1)}} h_{pj}^{l}w_{jk}^{l}h_{pk}^{(l+1)} - \sum_{l=1}^{L_2}\sum_{c=1}^{Y}\sum_{j=1}^{H^l} y_{pc}u_{cj}^{l}h_{pj}^{l}
\end{aligned}
$$

$$- \sum_{m=1}^{M} \sum_{l=1}^{L_1} \sum_{j=1}^{H^{lm}} b_j^{lm} h_{pj}^{lm} - \sum_{l=1}^{L_2} \sum_{j=1}^{H^l} b_j^l h_{pj}^l - \sum_{c=1}^{Y} d_c y_{pc}, \tag{7.8}$$

where $E_{ajd}(\mathbf{L}_p, \mathbf{h}_p)$ represents the loss function $E_{AJD}(\mathbf{L}^1, \cdots, \mathbf{L}^{M_1}, \underline{\mathbf{H}})$ of (7.6) for each given observation. The parameter space of the proposed model is defined by $\boldsymbol{\theta}_{d2jle} = \{w^{1m}, \cdots, w^{(L_1+1)m}, w^1, \cdots, w^{L_2-1}, u^{1m}, \cdots, u^{L_1 m}, u^1, \cdots, u^{L_2}, b^{1m}, \cdots, b^{L_1 m}, b^1, \cdots, b^{L_2}, d, \kappa^{lm}, \psi^{lm}, \lambda\}, \ \forall l, m$. It is to be noted here that for $M_1 = 1$, the energy function of the proposed model $E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)$ deduces to the summation of the objective function, corresponding to the LE algorithm, $E_{le}(\mathbf{L}_p, \mathbf{h}_p)$ and the energy function of the MDDBM model $E_{mddbm}(\mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)$. The learning of D2JLE model corresponds to estimating the model parameter set $\boldsymbol{\theta}_{d2jle}$ that maximizes the probability of observing the given input data. Considering the input view $\{\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p\}$, corresponding to the $p$-th observation, the objective function of the proposed D2JLE model is given by the log-likelihood function, which is as follows:

$$\begin{aligned} &\ln \mathrm{L}(\boldsymbol{\theta}_{d2jle} | \mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p) = \ln P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p | \boldsymbol{\theta}_{d2jle}) \\ &= \ln \sum_{\mathbf{h}} e^{-E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)} - \ln \sum_{\mathbf{L}, \mathbf{v}, \mathbf{h}, \mathbf{y}} e^{-E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)}; \end{aligned} \tag{7.9}$$

where $\mathbf{h}$ denotes the stack of hidden layers, $P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p | \boldsymbol{\theta}_{d2jle})$ represents the probability assigned to the $p$-th input observation $\{\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p\}$ by the model parameter set $\boldsymbol{\theta}_{d2jle}$, and $E(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)$ signifies the energy of the joint configuration $\{\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p\}$. The corresponding partition function can be defined as $Z = \sum_{\mathbf{L}, \mathbf{v}, \mathbf{h}, \mathbf{y}} e^{-E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)}$. Since the parameter space of the D2JLE model is quite large, the gradient ascent on the log-likelihood is commonly used to determine the optimal parameters of the model. So, the update rule for the parameters of the D2JLE model is given by

$$\begin{aligned} \frac{\partial \ln \mathrm{L}(\boldsymbol{\theta}_{d2jle} | \mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)}{\partial \boldsymbol{\theta}_{d2jle}} &= - \sum_{\mathbf{h}} P(\mathbf{h}_p | \mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \frac{\partial E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)}{\partial \boldsymbol{\theta}_{d2jle}} \\ &+ \sum_{\mathbf{L}, \mathbf{v}, \mathbf{h}, \mathbf{y}} P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p) \frac{\partial E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)}{\partial \boldsymbol{\theta}_{d2jle}}. \end{aligned} \tag{7.10}$$

Thus, the gradient of the log-likelihood function turns out to be the difference between the expectation of gradient of energy function under model distribution, referred to as data-independent expectation, and under the conditional distribution of hidden representation given the input views, termed as data-dependent expectation. Hence, in order to learn the parameters of D2JLE model, the corresponding data-dependent and data-independent expectations are required to be estimated, which are described subsequently.

#### 7.3.5.2 Estimation of Data-Dependent Expectations

Now, the exact maximum likelihood learning is intractable. So, the variational learning [142] is employed to estimate the data-dependent expectation. In variational inference, the posterior distribution $P(\mathbf{h}_p | \mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)$ is approximated with a tractable mean field

distribution $Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \approx P(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)$. Now,

$$\ln P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) = \ln \sum_{\mathbf{h}} Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \frac{P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)}{Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)} \tag{7.11}$$

where $P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p) = (1/Z)e^{-E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)}$ represents the probability associated with the joint configuration $\{\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p\}$ corresponding to the $p$-th input observation. Since logarithmic is a concave function, applying Jensen's inequality [31], we get

$$\ln P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \geqslant \sum_{\mathbf{h}} Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \ln \frac{P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)}{Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)} = \mathcal{L}_v. \tag{7.12}$$

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

$$\begin{aligned}
&\ln P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) - \sum_{\mathbf{h}} Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \left\{ \ln P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) + \ln \tfrac{P(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)}{Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)} \right\} \\
&= \ln P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) - \ln P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) + \sum_{\mathbf{h}} Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \ln \tfrac{Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)}{P(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)} \\
&= KL(Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)||P(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)),
\end{aligned} \tag{7.13}$$

where $KL(Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)||P(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p))$ is the Kullback-Leibler divergence between the two distributions $P$ and $Q$. So, better approximation of $P(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)$ implies tighter bound on $\ln P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)$. Let the mean field distribution be defined as

$$Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) = \prod_{m=1}^{M} \prod_{l=1}^{L_1} \prod_{j=1}^{H^{lm}} q(h_{pj}^{lm}|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \prod_{l=1}^{L_2} \prod_{j=1}^{H^l} q(h_{pj}^{l}|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p); \tag{7.14}$$

where the hidden units $\{h_{pj}\}$ are considered to be Bernoulli variables with $q(h_{pj}|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) = \mu_{pj}^{\{h_{pj}=1\}}(1-\mu_{pj})^{\{h_{pj}=0\}}$ and $\mu_{pj}$ denotes the probability of being the state of $h_{pj}$ as 1. The definitions of $Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)$, presented in (7.14), and $P(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)$, corresponding to the energy function obtained in (7.8), are substituted in (7.12) to obtain the final expression of $\mathcal{L}_v$, which is as follows:

$$\begin{aligned}
\mathcal{L}_v &= \sum_{\mathbf{h}} Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p)\{-E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p) - \ln Z\} \\
&\quad - \sum_{\mathbf{h}} Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \ln Q(\mathbf{h}_p|\mathbf{L}_p, \mathbf{v}_p, \mathbf{y}_p) \\
&= -\sum_{m=1}^{M_1} \sum_{l=1}^{L_1} \sum_{r=1, r\neq p}^{N} a_{pr}^{m} \left\{ \sum_{j=1}^{H^{lm}} \left( \mu_{pj}^{lm} + \mu_{rj}^{lm} - 2\mu_{pj}^{lm}\mu_{rj}^{lm} \right) \right\} + \sum_{m=1}^{M_2} \sum_{i=1}^{V^m} \sum_{j=1}^{H^{1m}} v_{pi}^{m} w_{ij}^{1m} \mu_{pj}^{1m} \\
&\quad - \sum_{m=1}^{M_1} \sum_{l=1}^{L_1} \sum_{j=1}^{H^{lm}} \kappa_j^{lm} \left\{ 1 - \left( \sum_{r=1, r\neq p}^{N} a_{pr}^{m} \right) \mu_{pj}^{lm} \right\} + \sum_{m=1}^{M} \sum_{l=1}^{L_1-1} \sum_{j=1}^{H^{lm}} \sum_{k=1}^{H^{(l+1)m}} \mu_{pj}^{lm} w_{jk}^{(l+1)m} \mu_{pk}^{(l+1)m}
\end{aligned}$$

$$+\sum_{m=1}^{M_1}\sum_{l=1}^{L_1}\sum_{j=1}^{H^{lm}-1}\sum_{k=(j+1)}^{H^{lm}}\psi_{jk}^{lm}\left\{\Big(\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big)\mu_{pj}^{lm}\mu_{pk}^{lm}\right\}+\sum_{m=1}^{M_1}\sum_{l=1}^{L_1}\sum_{c=1}^{Y}\sum_{j=1}^{H^{lm}}y_{pc}u_{cj}^{lm}\mu_{pj}^{lm}$$
$$-H^1(H^1-1)\sum_{m=1}^{M_1}\Gamma_m\sum_{r=1,r\neq p}^{N}\left(a_{pr}^{m}\right)^2\left\{\prod_{j=1}^{H^1}(\mu_{pj}^{1}+\mu_{rj}^{1}-2\mu_{pj}^{1}\mu_{rj}^{1})\right\}-\lambda\Big(1-\sum_{j=1}^{H^1}\mu_{pj}^{1}\Big)$$
$$-2\sum_{m=1}^{M_1}\Gamma_m\sum_{r=1,r\neq p}^{N}\sum_{s=p}^{N-1}\sum_{t=(r+1)}^{N}a_{pr}^{m}a_{st}^{m}\left\{\prod_{j=1}^{H^1}(\mu_{pj}^{1}-\mu_{rj}^{1})(\mu_{sj}^{1}-\mu_{tj}^{1})\right\}+\sum_{l=1}^{L_2}\sum_{c=1}^{Y}\sum_{j=1}^{H^l}y_{pc}u_{cj}^{l}\mu_{pj}^{l}$$
$$+\sum_{m=1}^{M}\Gamma_m\sum_{j=1}^{H^{L_1m}}\sum_{k=1}^{H^1}\mu_{pj}^{L_1m}w_{jk}^{(L_1+1)m}\mu_{pk}^{1}+\sum_{l=1}^{L_2-1}\sum_{j=1}^{H^l}\sum_{k=1}^{H^{(l+1)}}\mu_{pj}^{l}w_{jk}^{l}\mu_{pk}^{(l+1)}$$
$$+\sum_{m=1}^{M}\sum_{l=1}^{L_1}\sum_{j=1}^{H^{lm}}b_{j}^{lm}\mu_{pj}^{lm}+\sum_{l=1}^{L_2}\sum_{j=1}^{H^l}b_{j}^{l}\mu_{pj}^{l}+\sum_{c=1}^{Y}d_c y_{pc}-\ln Z. \tag{7.15}$$

Since, the mean field parameters ($\boldsymbol{\mu}_p$) of $\mathcal{L}_v$, presented in (7.15), define the equilibrium state of the model corresponding to the $p$-th observation, they need to be updated accordingly. In order to obtain the mean field parameters of the proposed D2JLE model, the variational bound $\mathcal{L}_v$ of (7.15) is maximized with respect to $\boldsymbol{\mu}_p$ for a fixed parameter set $\boldsymbol{\theta}_{d2jle}$. The update rule for the nodes of first hidden layer corresponding to $m$-th modality is obtained by setting $\dfrac{\partial\mathcal{L}_v}{\partial\mu_{pj}^{lm}}=0$, which leads to

$$\mu_{pj}^{1m}=\sigma\Bigg(\sum_{i=1}^{V^m}v_{pi}^{m}w_{ij}^{1m}+\sum_{k=1}^{H^{2m}}w_{jk}^{2m}\mu_{pk}^{2m}+\sum_{c=1}^{Y}y_{pc}u_{cj}^{1m}+b_{pj}^{1m}-\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big(1-2\mu_{rj}^{1m}\Big)$$
$$-\kappa_j^{1m}\left\{1-\Big(\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big)\mu_{pj}^{1m}\right\}-\sum_{k=1,k\neq j}^{H^{1m}}\psi_{jk}^{1m}\left\{\Big(\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big)\mu_{pj}^{1m}\mu_{pk}^{1m}\right\}\Bigg),\ \forall m; \tag{7.16}$$

where $\sigma(x)=\dfrac{1}{1+e^{-x}}$ is the sigmoid function. Similarly, the following update rules can be obtained:

$$\mu_{pj}^{lm}=\sigma\Bigg(\sum_{k=1}^{H^{(l-1)m}}\mu_{pk}^{(l-1)m}w_{kj}^{lm}+\sum_{k=1}^{H^{(l+1)m}}w_{jk}^{(l+1)m}\mu_{pk}^{(l+1)m}+\sum_{c=1}^{Y}y_{pc}u_{cj}^{lm}+b_{pj}^{lm}$$
$$-\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big(1-2\mu_{rj}^{lm}\Big)-\kappa_j^{lm}\left\{1-\Big(\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big)\mu_{pj}^{lm}\right\}$$
$$-\sum_{k=1,k\neq j}^{H^{lm}}\psi_{jk}^{lm}\left\{\Big(\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big)\mu_{pj}^{lm}\mu_{pk}^{lm}\right\}\Bigg),\ \text{for } 1<l<L_1 \text{ and } \forall m; \tag{7.17}$$

$$\mu_{pj}^{L_1 m} = \sigma\Bigg( \sum_{k=1}^{H^{(L_1-1)m}} \mu_{pk}^{(L_1-1)m} w_{kj}^{L_1 m} + \Gamma_m \sum_{k=1}^{H^1} w_{jk}^{(L_1+1)m} \mu_{pk}^{1} + \sum_{c=1}^{Y} y_{pc} u_{cj}^{L_1 m} + b_{pj}^{L_1 m}$$
$$- \sum_{r=1, r\neq p}^{N} a_{pr}^{m} \Big(1 - 2\mu_{rj}^{L_1 m}\Big) - \kappa_j^{L_1 m} \Bigg\{ 1 - \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^{m} \Big) \mu_{pj}^{L_1 m} \Bigg\}$$
$$- \sum_{k=1, k\neq j}^{H^{L_1 m}} \psi_{jk}^{L_1 m} \Bigg\{ \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^{m} \Big) \mu_{pj}^{L_1 m} \mu_{pk}^{L_1 m} \Bigg\} \Bigg), \ \forall m; \tag{7.18}$$

$$\mu_{pj}^{1} = \sigma\Bigg( \sum_{m=1}^{M} \Gamma_m \sum_{k=1}^{H^{L_1 m}} \mu_{pk}^{L_1 m} w_{kj}^{(L_1+1)m} + \sum_{k=1}^{H^2} w_{jk}^{1} \mu_{pk}^{2} + \sum_{c=1}^{Y} y_{pc} u_{cj}^{1} + b_{pj}^{1}$$
$$- H^1(H^1 - 1) \sum_{m=1}^{M_1} \Gamma_m \sum_{r=1, r\neq p}^{N} \big(a_{pr}^{m}\big)^2 (1 - 2\mu_{rj}^{1}) + \lambda$$
$$- 2 \sum_{m=1}^{M_1} \Gamma_m \sum_{r=1, r\neq p}^{N} \sum_{s=p}^{N-1} \sum_{t=(r+1)}^{N} a_{pr}^{m} a_{st}^{m} (\mu_{sj}^{1} - \mu_{tj}^{1}) \Bigg); \tag{7.19}$$

$$\mu_{pj}^{l} = \sigma\Bigg( \sum_{k=1}^{H^{(l-1)}} \mu_{pk}^{(l-1)} w_{kj}^{l} + \sum_{k=1}^{H^{(l+1)}} w_{jk}^{(l+1)} \mu_{pk}^{(l+1)} + \sum_{c=1}^{Y} y_{pc} u_{cj}^{l} + b_{pj}^{l} \Bigg), \text{for} 1 < l < L_2; \tag{7.20}$$

$$\text{and} \ \ \mu_{pj}^{L_2} = \sigma\Bigg( \sum_{k=1}^{H^{(L_2-1)}} \mu_{pk}^{(L_2-1)} w_{kj}^{L_2} + \sum_{c=1}^{Y} y_{pc} u_{cj}^{L_2} + b_{pj}^{L_2} \Bigg). \tag{7.21}$$

Thus, given the training data along with the corresponding class label $\{\mathbf{L}_p^m, \mathbf{v}_p^m, \mathbf{y}_p\}$ for the $p$-th observation, the equilibrium state of the model is estimated using the concept of mean field theory. Now, based on the obtained mean field parameters, the parameter set $\boldsymbol{\theta}_{d2jle}$ of the proposed D2JLE architecture, corresponding to the data-dependent expectations, can be learned by maximizing the variational bound $\mathcal{L}_v$ with respect to $\boldsymbol{\theta}_{d2jle}$ for the equilibrium mean field parameters $\boldsymbol{\mu}_p$. The expressions for differentiation of $\mathcal{L}_v$ with respect to each of the model parameters are given by

$$\frac{\partial \mathcal{L}_v}{\partial w_{ij}^{1m}} = v_{pi}^{m} \mu_{pj}^{1m}; \ \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{lm}} = \mu_{pj}^{(l-1)m} \mu_{pk}^{lm}, \ \text{for} \ 1 < l \leqslant L_1; \ \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{(L_1+1)m}} = \mu_{pj}^{L_1 m} \mu_{pk}^{1};$$
$$\frac{\partial \mathcal{L}_v}{\partial u_{cj}^{lm}} = y_{pc} \mu_{pj}^{lm}, \ \text{for} \ 1 \leqslant l \leqslant L_1; \ \frac{\partial \mathcal{L}_v}{\partial w_{jk}^{l}} = \mu_{pj}^{l} \mu_{pk}^{(l+1)}, \ \text{for} \ 1 \leqslant l < L_2;$$
$$\frac{\partial \mathcal{L}_v}{\partial u_{cj}^{l}} = y_{pc} \mu_{pj}^{l}, \ \text{for} \ 1 \leqslant l \leqslant L_2; \ \frac{\partial \mathcal{L}_v}{\partial b_{pj}^{lm}} = \mu_{pj}^{lm}, \ \text{for} \ 1 \leqslant l \leqslant L_1; \ \frac{\partial \mathcal{L}_v}{\partial b_{pj}^{l}} = \mu_{pj}^{l}, \ \text{for} \ 1 \leqslant l \leqslant L_2;$$
$$\frac{\partial \mathcal{L}_v}{\partial d_c} = y_{pc}; \ \frac{\partial \mathcal{L}_v}{\partial \kappa_j^{lm}} = \Bigg\{ 1 - \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^{m} \Big) \mu_{pj}^{lm} \Bigg\}, \ \text{for} \ 1 \leqslant l \leqslant L_1;$$
$$\frac{\partial \mathcal{L}_v}{\partial \psi_{jk}^{lm}} = \Bigg\{ \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^{m} \Big) \mu_{pj}^{lm} \mu_{pk}^{lm} \Bigg\}, \ \text{for} \ 1 \leqslant l \leqslant L_1; \ \text{and} \ \ \frac{\partial \mathcal{L}_v}{\partial \lambda} = \Big( 1 - \sum_{j=1}^{H^1} \mu_{pj}^{1} \Big); \ \forall p, m. \tag{7.22}$$

So, the data-dependent expectations are estimated by the gradient ascent on the lower

bound of the proposed architecture.

#### 7.3.5.3 Estimation of Data-Independent Expectations

Now, the second term of the gradient of log-likelihood function (7.10), that is, energy gradient with respect to the model distribution, is estimated using the Markov Chain Monte Carlo based stochastic approximation procedure [190]. 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 $\boldsymbol{\theta}_{d2jle}$. The conditional distributions corresponding to the proposed D2JLE model are given by

$$P(h_{pj}^{1m}|\mathbf{L}_p^m, \mathbf{v}_p^m, \mathbf{y}_p, \mathbf{h}_p^{2m}) = \sigma\Bigg( \sum_{i=1}^{V^m} v_{pi}^m w_{ij}^{1m} + \sum_{k=1}^{H^{2m}} w_{jk}^{2m} h_{pk}^{2m} + \sum_{c=1}^{Y} y_{pc} u_{cj}^{1m} + b_{pj}^{1m} - \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big(1 - 2h_{rj}^{1m}\Big) - \kappa_j^{1m} \Bigg\{ 1 - \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big) h_{pj}^{1m} \Bigg\} - \sum_{k=1, k\neq j}^{H^{1m}} \psi_{jk}^{1m} \Bigg\{ \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big) h_{pj}^{1m} h_{pk}^{1m} \Bigg\} \Bigg), \ \forall m. \tag{7.23}$$

Similarly, the subsequent conditional distributions can be obtained:

$$P(h_{pj}^{lm}|\mathbf{L}_p^m, \mathbf{h}_p^{(l-1)m}, \mathbf{h}_p^{(l+1)m}, \mathbf{y}_p) = \sigma\Bigg( \sum_{k=1}^{H^{(l-1)m}} h_{pk}^{(l-1)m} w_{kj}^{lm} + \sum_{k=1}^{H^{(l+1)m}} w_{jk}^{(l+1)m} h_{pk}^{(l+1)m} + \sum_{c=1}^{Y} y_{pc} u_{cj}^{lm} + b_{pj}^{lm} - \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big(1 - 2h_{rj}^{lm}\Big) - \kappa_j^{lm} \Bigg\{ 1 - \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big) h_{pj}^{lm} \Bigg\} - \sum_{k=1, k\neq j}^{H^{lm}} \psi_{jk}^{lm} \Bigg\{ \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big) h_{pj}^{lm} h_{pk}^{lm} \Bigg\} \Bigg), \text{ for } 1 < l < L_1, \ \forall m; \tag{7.24}$$

$$P(h_{pj}^{L_1 m}|\mathbf{L}_p^m, \mathbf{h}_p^{(L_1-1)m}, \mathbf{h}_p^1, \mathbf{y}_p) = \sigma\Bigg( \sum_{k=1}^{H^{(L_1-1)m}} h_{pk}^{(L_1-1)m} w_{kj}^{L_1 m} + \Gamma_m \sum_{k=1}^{H^1} w_{jk}^{(L_1+1)m} h_{pk}^1 + \sum_{c=1}^{Y} y_{pc} u_{cj}^{L_1 m} + b_{pj}^{L_1 m} - \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big(1 - 2h_{rj}^{L_1 m}\Big) - \kappa_j^{L_1 m} \Bigg\{ 1 - \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big) h_{pj}^{L_1 m} \Bigg\} - \sum_{k=1, k\neq j}^{H^{L_1 m}} \psi_{jk}^{L_1 m} \Bigg\{ \Big( \sum_{r=1, r\neq p}^{N} a_{pr}^m \Big) h_{pj}^{L_1 m} h_{pk}^{L_1 m} \Bigg\} \Bigg), \ \forall m; \tag{7.25}$$

$$P(h_{pj}^1|\mathbf{L}_p^m, \mathbf{h}_p^{L_1 m}, \mathbf{h}_p^2, \mathbf{y}_p) = \sigma\Bigg( \sum_{m=1}^{M} \Gamma_m \sum_{k=1}^{H^{L_1 m}} h_{pk}^{L_1 m} w_{kj}^{(L_1+1)m} + \sum_{k=1}^{H^2} w_{jk}^1 h_{pk}^2 + \sum_{c=1}^{Y} y_{pc} u_{cj}^1 + b_{pj}^1 - H^1(H^1-1) \sum_{m=1}^{M_1} \Gamma_m \sum_{r=1, r\neq p}^{N} \big(a_{pr}^m\big)^2 (1 - 2h_{rj}^1) + \lambda - 2 \sum_{m=1}^{M_1} \Gamma_m \sum_{r=1, r\neq p}^{N} \sum_{s=p}^{N-1} \sum_{t=(r+1)}^{N} a_{pr}^m a_{st}^m (h_{sj}^1 - h_{tj}^1) \Bigg); \tag{7.26}$$

$$P(h_{pj}^{l}|\mathbf{h}_{p}^{(l-1)},\mathbf{h}_{p}^{(l+1)},\mathbf{y}_{p})=\sigma\Bigg(\sum_{k=1}^{H^{(l-1)}}h_{pk}^{(l-1)}w_{kj}^{l}+\sum_{k=1}^{H^{(l+1)}}w_{jk}^{(l+1)}h_{pk}^{(l+1)}+\sum_{c=1}^{Y}y_{pc}u_{cj}^{l}+b_{pj}^{l}\Bigg),$$
$$\text{for } 1<l<L_2; \tag{7.27}$$

$$P(h_{pj}^{L_2}|\mathbf{h}_{p}^{(L_2-1)},\mathbf{y}_{p})=\sigma\Bigg(\sum_{k=1}^{H^{(L_2-1)}}h_{pk}^{(L_2-1)}w_{kj}^{L_2}+\sum_{c=1}^{Y}y_{pc}u_{cj}^{L_2}+b_{pj}^{L_2}\Bigg); \tag{7.28}$$

$$P(v_{pi}^{m}|\mathbf{h}_{p}^{1m})=\sigma\Bigg(\sum_{j=1}^{H^{1m}}w_{ij}^{1m}h_{pj}^{1m}\Bigg),\ \forall m; \tag{7.29}$$

$$P(y_{pc}|\mathbf{h}_{p}^{11},\cdots,\mathbf{h}_{p}^{L_1M},\mathbf{h}_{p}^{1},\cdots,\mathbf{h}_{p}^{L_2})=\frac{e^{X_{pc}}}{\sum_{\bar{c}=1}^{Y}e^{X_{p\bar{c}}}};$$

$$\text{where } X_{pc}=\sum_{m=1}^{M}\sum_{l=1}^{L_1}\sum_{j=1}^{H^{lm}}u_{cj}^{lm}h_{pj}^{lm}+\sum_{l=1}^{L_2}\sum_{j=1}^{H^{l}}u_{cj}^{l}h_{pj}^{l}+d_{pc}. \tag{7.30}$$

Given that the convergence criteria are satisfied, which are to be discussed in Section 7.4.2, if a Markov chain is run for sufficient number of steps, then it can be ensured that the chain will converge to an 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 (7.23) - (7.30). 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 superscript tilde (e.g., $\tilde{v}$);

$$\frac{\partial E_{d2jle}}{\partial w_{ij}^{1m}}=\tilde{v}_{pi}^{m}\tilde{h}_{pj}^{1m};\ \frac{\partial E_{d2jle}}{\partial w_{jk}^{lm}}=\tilde{h}_{pj}^{(l-1)m}\tilde{h}_{pk}^{lm},\ \text{for } 1<l\leqslant L_1;\ \frac{\partial E_{d2jle}}{\partial w_{jk}^{(L_1+1)m}}=\tilde{h}_{pj}^{L_1m}\tilde{h}_{pk}^{1};$$
$$\frac{\partial E_{d2jle}}{\partial u_{cj}^{lm}}=\tilde{y}_{pc}\tilde{h}_{pj}^{lm},\ \text{for } 1\leqslant l\leqslant L_1;\ \frac{\partial E_{d2jle}}{\partial w_{jk}^{l}}=\tilde{h}_{pj}^{l}\tilde{h}_{pk}^{(l+1)},\ \text{for } 1\leqslant l<L_2;$$
$$\frac{\partial E_{d2jle}}{\partial u_{cj}^{l}}=\tilde{y}_{pc}\tilde{h}_{pj}^{l},\ \text{for } 1\leqslant l\leqslant L_2;\ \frac{\partial E_{d2jle}}{\partial b_{j}^{lm}}=\tilde{h}_{pj}^{lm},\ \text{for } 1\leqslant l\leqslant L_1;\ \frac{\partial E_{d2jle}}{\partial d_{c}}=\tilde{y}_{pc};$$
$$\frac{\partial E_{d2jle}}{\partial b_{j}^{l}}=\tilde{h}_{pj}^{l},\ \text{for } 1\leqslant l\leqslant L_2;\ \frac{\partial E_{d2jle}}{\partial \kappa_{j}^{lm}}=\Bigg\{1-\Big(\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big)(h_{pj}^{lm})^2\Bigg\},\ \text{for } 1\leqslant l\leqslant L_1;$$
$$\frac{\partial E_{d2jle}}{\partial \psi_{jk}^{lm}}=\Bigg\{\Big(\sum_{r=1,r\neq p}^{N}a_{pr}^{m}\Big)h_{pj}^{lm}h_{pk}^{lm}\Bigg\},\ \text{for } 1\leqslant l\leqslant L_1;\ \text{and}$$
$$\frac{\partial E_{d2jle}}{\partial \lambda}=\Big(1-\sum_{j=1}^{H^1}(h_{pj}^{1})^2\Big);\ \forall p,m. \tag{7.31}$$

Hence, the proposed model can be efficiently learned from the data-dependent and data-independent estimates, obtained in (7.22) and (7.31), respectively.

#### 7.3.5.4 Learning Rule of D2JLE Model Parameters

Let $N$, $S$, $t$, and $\eta$ be the number of training samples, number of persistent Markov chains, current epoch, and learning rate, respectively. Thus, the update rule for the parameters of the proposed D2JLE architecture, required to perform gradient ascent on the log-likelihood function corresponding to the energy function of (7.8), is as follows:

$$\boldsymbol{\theta}_{d2jle}^{(t+1)} = \boldsymbol{\theta}_{d2jle}^{t} + \Delta\boldsymbol{\theta}_{d2jle}^{t}; \tag{7.32}$$

$$\text{where } \Delta\boldsymbol{\theta}_{d2jle}^{t} = \eta\left\{\frac{1}{N}\sum_{p=1}^{N}\left(\frac{\partial\mathcal{L}_v}{\partial\boldsymbol{\theta}_{d2jle}}\right)_p - \frac{1}{S}\sum_{s=1}^{S}\left(\frac{\partial E_{d2jle}}{\partial\boldsymbol{\theta}_{d2jle}}\right)_s\right\} - \rho\boldsymbol{\theta}_{d2jle}^{t} + \zeta\Delta\boldsymbol{\theta}_{d2jle}^{(t-1)}. \tag{7.33}$$

From (7.32) and (7.33), it can be observed that the update rules of the proposed D2JLE model follow the Hebbian rule, which is originally employed for the learning of standard Boltzmann machine [136]. 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 [8]. It can be observed that all the parameters of the model are learned simultaneously using (7.32) which is considered to be beneficial in case of learning joint subspace from heterogeneous data. Also, subtracting the data-independent expectations from the corresponding data-dependent terms in (7.32) basically stabilizes the distribution of parameters of the proposed model in order to propagate uncertainties associated with ambiguous inputs. The learning algorithm of the proposed D2JLE model is outlined in Algorithm 7.1.

## 7.4 Different Aspects of Proposed Model

In this section, different aspects of the proposed model are studied, which include error analysis and convergence analysis of the D2JLE model. The detailed analysis of each of these aspects is discussed next.

### 7.4.1 Error Analysis of Proposed Model

In the current study, the D2JLE model is developed to classify the given observations into different categories. Now, it is well known that Bayes discriminant function provides the optimal solution to any classification problem. In order to analyze the discriminative ability of the proposed framework, the mean-squared error between the prediction rule (7.30) of the D2JLE model and Bayes decision rule is studied. Thus, reduction in the corresponding error will indicate better approximation of Bayes discriminant function by the proposed model, which in turn, will ensure better discriminative ability of the model.

Following the similar steps as demonstrated in Section 6.4.1 of Chapter 6, it can be established that the learning of the proposed D2JLE architecture with prediction criterion (7.30) attempts to provide a classifier, which is mean-squared error approximation to the Bayes optimal classifier. So, minimization of the mean-squared error depends on the efficient learning of the model parameters, which in turn, depends on the model architecture.

**Algorithm 7.1** Learning of D2JLE Model.

**Input:** Set of Laplacian matrices $\{\mathbf{L}^m\}$ corresponding to $M_1$ image modalities and set of data vectors $\{\mathbf{v}^m\}$ for $M_2$ non-image modalities, along with the corresponding set of class labels $\{\mathbf{y}\}$, number of persistent chains ($S$), number of epochs ($\tau$), learning rate ($\eta$), weight decay ($\rho$), momentum ($\zeta$), and number of Gibbs steps ($\alpha$).

**Output:** Final parameter set $\boldsymbol{\theta}^{\tau}_{d2jle}$ of the architecture.

Perform greedy layer-wise pretraining to initialize the set of model parameters, $\boldsymbol{\theta}^{0}_{d2jle}$.
Randomly initialize $S$ Markov chains $\{\tilde{\mathbf{L}}^{m^0}, \tilde{\mathbf{v}}^{m^0}, \tilde{\mathbf{y}}^{0}, \tilde{\mathbf{h}}^{1m^0}, \cdots, \tilde{\mathbf{h}}^{L_1m^0}, \tilde{\mathbf{h}}^{1^0}, \cdots, \tilde{\mathbf{h}}^{L_2{}^0}\}_{s=1}^{S}$.
**for** each epoch $t = 0$ to $\tau$ **do**
    // Variational inference
    **for** each training sample $p = 1$ to $N$ **do**
        (i) Run mean field updates using (7.16)-(7.21) until convergence.
        (ii) Save the obtained mean field parameter ($\boldsymbol{\mu}$) for the corresponding training sample, $\boldsymbol{\mu}_p = \boldsymbol{\mu}$.
    **end for**
    // Stochastic approximation
    **for** each persistent chain $s = 1$ to $S$ **do**
        Run the chain for $\alpha$-steps and sample the state $\{\tilde{\mathbf{L}}^{m^{t+1}}, \tilde{\mathbf{v}}^{m^{t+1}}, \tilde{\mathbf{y}}^{t+1}, \tilde{\mathbf{h}}^{1m^{t+1}}, \cdots, \tilde{\mathbf{h}}^{L_1m^{t+1}}, \tilde{\mathbf{h}}^{1^{t+1}}, \cdots, \tilde{\mathbf{h}}^{L_2{}^{t+1}}\}$ from $\{\tilde{\mathbf{L}}^{m^t}, \tilde{\mathbf{v}}^{m^t}, \tilde{\mathbf{y}}^{t}, \tilde{\mathbf{h}}^{1m^t}, \cdots, \tilde{\mathbf{h}}^{L_1m^t}, \tilde{\mathbf{h}}^{1^t}, \cdots, \tilde{\mathbf{h}}^{L_2{}^t}\}$ using (7.23)-(7.30).
    **end for**
    Update the parameters of the model from $\boldsymbol{\theta}^{t}_{d2jle}$ to $\boldsymbol{\theta}^{(t+1)}_{d2jle}$ using (7.32).
**end for**

Thus, by modifying the architecture of the model or parameter values, the discriminative ability of the model can be varied. Hence, there must exist a relation between the model architecture and the values of the parameters with the error probability of the proposed D2JLE model.

Because of the resemblance of prediction criterion between the proposed method and Bayes decision rule, the error probability of the D2JLE model can be defined in accordance with the Bayes multi-class classifier [169], which is given by

$$\begin{aligned} P_e &= \mathbb{E}[P(e|\mathbf{L},\mathbf{v})] = \mathbb{E}[1 - \max_c P(y_c = 1|\mathbf{L},\mathbf{v})] \\ &\leqslant 2 \sum_{\mathbf{L},\mathbf{v}\in\Omega} p(\mathbf{L},\mathbf{v}) \sum_{c=1}^{C-1} \sum_{k=(c+1)}^{C} P(y_c = 1|\mathbf{L},\mathbf{v}) P(y_k = 1|\mathbf{L},\mathbf{v}). \end{aligned} \tag{7.34}$$

Substituting the conditional distribution $P(y_{pc} = 1|\mathbf{L}_p, \mathbf{v}_p)$ of (7.30) in (7.34), the upper

bound on the error probability of the D2JLE model can be obtained, which is given by

$$P_e \leqslant \sum_{\mathbf{L}_p, \mathbf{v}_p \in \Omega} p(\mathbf{L}_p, \mathbf{v}_p) \left\{ 2 \sum_{c=1}^{C-1} \sum_{k=(c+1)}^{C} \frac{e^{X_{pc}}}{\sum_{\bar{c}=1}^{C} e^{X_{\bar{p}c}}} \frac{e^{X_{pk}}}{\sum_{\bar{k}=1}^{C} e^{X_{\bar{p}k}}} \right\}$$

$$\Rightarrow P_e \leqslant \sum_{\mathbf{L}_p, \mathbf{v}_p \in \Omega} p(\mathbf{L}_p, \mathbf{v}_p) \left\{ \frac{\sum_{c=1}^{C} e^{2X_{pc}} + 2 \sum_{c=1}^{C-1} \sum_{k=(c+1)}^{C} e^{X_{pc}} e^{X_{pk}} - \sum_{c=1}^{C} e^{2X_{pc}}}{\left( \sum_{\bar{c}=1}^{C} e^{X_{\bar{p}c}} \right)^2} \right\}$$

$$\Rightarrow P_e \leqslant \sum_{\mathbf{L}_p, \mathbf{v}_p \in \Omega} p(\mathbf{L}_p, \mathbf{v}_p) \left\{ 1 - \frac{\sum_{c=1}^{C} \left( e^{X_{pc}} \right)^2}{\left( \sum_{\bar{c}=1}^{C} e^{X_{\bar{p}c}} \right)^2} \right\} \tag{7.35}$$

$$\text{where } X_{pc} = \sum_{l=1}^{L_1} \sum_{m=1}^{M} \sum_{j=1}^{H^{lm}} u_{cj}^{lm} h_{pj}^{lm} + \sum_{l=1}^{L_2} \sum_{j=1}^{H^l} u_{cj}^{l} h_{pj}^{l} + d_{pc}. \tag{7.36}$$

So, an upper bound of the error probability $P_e$ is achieved in terms of $X_{pc}$, which depends on both architecture as well as parameters of the model. Through proper learning of the model parameters, the upper bound on the error probability can be minimized. Also, by suitably varying the model architecture, a tighter bound on $P_e$ can be achieved.

Let, $u_c^1 = \min_{j,l,m}\{u_{cj}^{lm}\}$, $h_p^1 = \min_{j,l,m}\{h_{pj}^{lm}\}$, $u_c^2 = \min_{j,l}\{u_{cj}^{l}\}$, $h_p^2 = \min_{j,l}\{h_{pj}^{l}\}$, $H^1 = \min_{l,m}\{H^{lm}\}$, and $H^2 = \min_{l}\{H^{l}\}$. So,

$$X_{pc} \leqslant L_1 M H^1 u_c^1 h_p^1 + L_2 H^2 u_c^2 h_p^2. \tag{7.37}$$

If the value of $X_{pc}$ in (7.36) is substituted with the formulation of (7.37), the inequality will still hold, which is given by

$$P_e \leqslant \sum_{\mathbf{v} \in \Omega} p(\mathbf{v}) \left\{ 1 - \frac{\sum_{c=1}^{C} e^{2L_1 M H^1 u_c^1 h^1 + 2L_2 H^2 u_c^2 h^2}}{\left( \sum_{c=1}^{C} e^{L_1 M H^1 u_c^1 h^1 + L_2 H^2 u_c^2 h^2} \right)^2} \right\}. \tag{7.38}$$

It can be observed from (7.38) that instead of heuristically determining the architecture of the proposed framework, an optimal deep architecture can be obtained for the analysis of the given multi-view data. Apart from the model parameters and architecture of the proposed D2JLE framework, the error probability also depends on the nature and complexity of the given classification problem.

### 7.4.2 Convergence Analysis of the Proposed D2JLE Model

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 (7.12) of the model. Given a particular state, the parameter set $\boldsymbol{\theta}_{d2jle}$ 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 (7.8), is given by

$$\begin{aligned}\nabla\mathcal{L}_v(\theta_{d2jle}) = \Bigg[ & \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial w_{ij}^{1m}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial w_{jk}^{(L_1+1)m}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial w_{jk}^{1}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial w_{jk}^{L_2}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial u_{cj}^{1m}} \\ & \cdots \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial u_{cj}^{L_1m}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial u_{cj}^{1}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial u_{cj}^{L_2}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial b_{j}^{1m}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial b_{j}^{L_1m}} \\ & \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial b_{j}^{1}} \cdots \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial b_{j}^{L_2}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial d_{c}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial \kappa_{j}^{1m}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial \kappa_{j}^{L_1m}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial \psi_{jk}^{1m}} \\ & \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial \psi_{jk}^{L_1m}} \quad \frac{\partial\mathcal{L}_v(\theta_{d2jle})}{\partial \lambda} \Bigg]^T ; \ \forall i,j,k,c,m, \end{aligned} \tag{7.39}$$

where $T$ denotes the transpose operator. The gradient of $\mathcal{L}_v(\theta_{d2jle})$ with respect to each of the parameters can be obtained using (7.22), which makes it clear that the gradient function $\nabla\mathcal{L}_v(\theta_{d2jle})$ is independent of $\theta_{d2jle}$, that is, $\nabla\mathcal{L}_v(\theta_{d2jle\_1}) = \nabla\mathcal{L}_v(\theta_{d2jle\_2})$, $\forall\theta_{d2jle\_1}$, $\theta_{d2jle\_2} \in \boldsymbol{\theta}_{d2jle}$. The independence of $\nabla\mathcal{L}_v(\theta_{d2jle})$ with respect to the parameters of D2JLE model is evident since each parameter of the model is included in the corresponding energy function $E_{d2jle}(\mathbf{L}_p, \mathbf{v}_p, \mathbf{h}_p, \mathbf{y}_p)$, presented in (7.8) as an additive term. So, it can be said that $\mathcal{L}_v$ is a differential function having $\beta$-Lipschitz continuous gradient for some $\beta \geqslant 0$, that is, $\|\nabla\mathcal{L}_v(\theta_{d2jle\_1}) - \nabla\mathcal{L}_v(\theta_{d2jle\_2})\|_2 \leqslant \beta\|\theta_{d2jle\_1} - \theta_{d2jle\_2}\|_2$. For a function having $\beta$-Lipschitz gradient, it is known that $\forall\theta_{d2jle\_1}, \theta_{d2jle\_2} \in \boldsymbol{\theta}_{d2jle}$,

$$\begin{aligned}\mathcal{L}_v(\theta_{d2jle\_1}) \quad \leqslant \mathcal{L}_v(\theta_{d2jle\_2}) + \nabla\mathcal{L}_v(\theta_{d2jle\_2})^T(\theta_{d2jle\_1} - \theta_{d2jle\_2}) \\ + \frac{1}{2}\beta\|\theta_{d2jle\_1} - \theta_{d2jle\_2}\|_2^2. \end{aligned} \tag{7.40}$$

Let, $\theta_{d2jle\_1} = \theta_{d2jle}^t$ and $\theta_{d2jle\_2} = \theta_{d2jle}^{(t+1)}$, where $t$ denotes the current epoch. Substituting the values of $\theta_{d2jle\_1}$ and $\theta_{d2jle\_2}$, and rearranging the terms in (7.40), we get

$$\mathcal{L}_v(\theta_{d2jle}^{(t+1)}) \geqslant \quad \mathcal{L}_v(\theta_{d2jle}^{t}) + \nabla\mathcal{L}_v(\theta_{d2jle}^{(t+1)})^T(\theta_{d2jle}^{(t+1)} - \theta_{d2jle}^{t}) - \frac{1}{2}\beta\|\theta_{d2jle}^{(t+1)} - \theta_{d2jle}^{t}\|_2^2.$$

Since gradient ascent algorithm on $\mathcal{L}_v$ is employed in the proposed model to learn the parameter values of $\boldsymbol{\theta}_{d2jle}$, it is obvious that $\theta_{d2jle}^{(t+1)} = \theta_{d2jle}^{t} + \eta\nabla\mathcal{L}_v(\theta_{d2jle}^{t})$, where $\eta$ denotes the learning rate. Now, considering the independence property of $\nabla\mathcal{L}_v(\theta_{d2jle})$ in the

proposed model, that is, $\nabla\mathcal{L}_v(\theta^{(t+1)}_{d2jle}) = \nabla\mathcal{L}_v(\theta^{t}_{d2jle})$, we have

$$\mathcal{L}_v(\theta^{(t+1)}_{d2jle}) \geqslant \mathcal{L}_v(\theta^{t}_{d2jle}) + \eta(1 - \frac{1}{2}\beta\eta)\|\nabla\mathcal{L}_v(\theta^{t}_{d2jle})\|_2^2.$$

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

$$\mathcal{L}_v(\theta^{(t+1)}_{d2jle}) \geqslant \mathcal{L}_v(\theta^{t}_{d2jle}) + \frac{1}{2}\eta\|\nabla\mathcal{L}_v(\theta^{t}_{d2jle})\|_2^2. \tag{7.41}$$

Let $\theta^{*}_{d2jle}$ be the optimal parameter set that maximizes the lower bound, or equivalently maximizes the log-likelihood function of the proposed model in such a way that, $\mathcal{L}_v(\theta^{*}_{d2jle}) \geqslant \mathcal{L}_v(\theta_{d2jle}), \forall\theta_{d2jle} \in \boldsymbol{\theta}_{d2jle}$. Since $\mathcal{L}_v(\theta_{d2jle})$ is defined as a linear function of $\theta_{d2jle}$ in (7.15), we have

$$\mathcal{L}_v(\theta^{t}_{d2jle}) = \mathcal{L}_v(\theta^{*}_{d2jle}) + \nabla\mathcal{L}_v(\theta^{t}_{d2jle})^T(\theta^{t}_{d2jle} - \theta^{*}_{d2jle}). \tag{7.42}$$

Using (7.42) in (7.41), we get

$$\begin{aligned}
&\mathcal{L}_v(\theta^{*}_{d2jle}) - \mathcal{L}_v(\theta^{(t+1)}_{d2jle}) \leqslant -\nabla\mathcal{L}_v(\theta^{t}_{d2jle})^T(\theta^{t}_{d2jle} - \theta^{*}_{d2jle}) - \frac{1}{2}\eta\|\nabla\mathcal{L}_v(\theta^{t}_{d2jle})\|_2^2 \\
\Rightarrow\ &\mathcal{L}_v(\theta^{*}_{d2jle}) - \mathcal{L}_v(\theta^{(t+1)}_{d2jle}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^{t}_{d2jle} - \theta^{*}_{d2jle}\|_2^2 - \|\theta^{t}_{d2jle} - \theta^{*}_{d2jle}\|_2^2 - \|\eta\nabla\mathcal{L}_v(\theta^{t}_{d2jle})\|_2^2 \\
&\qquad\qquad - 2\eta\nabla\mathcal{L}_v(\theta^{t}_{d2jle})^T(\theta^{t}_{d2jle} - \theta^{*}_{d2jle})\Big\} \\
\Rightarrow\ &\mathcal{L}_v(\theta^{*}_{d2jle}) - \mathcal{L}_v(\theta^{(t+1)}_{d2jle}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^{t}_{d2jle} - \theta^{*}_{d2jle}\|_2^2 - \|\theta^{t+1}_{d2jle} - \theta^{*}_{d2jle}\|_2^2\Big\}.
\end{aligned}$$

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

$$\begin{aligned}
&\sum_{t=0}^{T}\Big\{\mathcal{L}_v(\theta^{*}_{d2jle}) - \mathcal{L}_v(\theta^{t+1}_{d2jle})\Big\} \leqslant \frac{1}{2\eta}\sum_{t=0}^{T}\Big\{\|\theta^{t}_{d2jle} - \theta^{*}_{d2jle}\|_2^2 - \|\theta^{t+1}_{d2jle} - \theta^{*}_{d2jle}\|_2^2\Big\} \\
\Rightarrow\ &\tau\mathcal{L}_v(\theta^{*}_{d2jle}) - \sum_{t=0}^{\tau-1}\mathcal{L}_v(\theta^{t+1}_{d2jle}) \leqslant \frac{1}{2\eta}\Big\{\|\theta^{0}_{d2jle} - \theta^{*}_{d2jle}\|_2^2 - \|\theta^{\tau}_{d2jle} - \theta^{*}_{d2jle}\|_2^2\Big\}.
\end{aligned}$$

Since $\mathcal{L}_v(\theta_{d2jle})$ is an increasing function of $\theta_{d2jle}$, we can replace $\sum_{t=0}^{\tau-1}\mathcal{L}_v(\theta^{(t+1)}_{d2jle})$ with $\tau\mathcal{L}_v(\theta^{\tau}_{d2jle})$ and the inequality will still hold. Thus, we get

$$\begin{aligned}
\mathcal{L}_v(\theta^{*}_{d2jle}) - \mathcal{L}_v(\theta^{\tau}_{d2jle}) &\leqslant \frac{1}{2\eta\tau}\Big\{\|\theta^{0}_{d2jle} - \theta^{*}_{d2jle}\|_2^2 - \|\theta^{\tau}_{d2jle} - \theta^{*}_{d2jle}\|_2^2\Big\} \\
&\leqslant \frac{1}{2\eta\tau}\|\theta^{0}_{d2jle} - \theta^{*}_{d2cca}\|_2^2.
\end{aligned} \tag{7.43}$$

From (7.43), it can be concluded that the variational learning algorithm in the proposed D2JLE model converges with a rate $\mathcal{O}(\frac{1}{\tau})$ after $\tau$ iterations, if the learning rate is considered

to be small enough, that is, $\eta \leqslant \frac{1}{\beta}$.

Now, 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 [218, 220]. One necessary condition requires the learning rate ($\eta$) 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 $\eta_t = \frac{a}{b+t}$, for constants $a > 0$ and $b > 0$. Also, in practice, the sequence $|\theta^t_{d2jle}|$ is bounded and the Markov chain is ergodic which, along with the condition on learning rate, establish the convergence of stochastic approximation procedure. Together with the condition on the variational learning (7.43), this ensures the convergence of the proposed D2JLE model.

## 7.5 Experimental Results and Discussions

In this section, the performance of the proposed D2JLE model is extensively studied and the corresponding results are reported. In order to evaluate the efficacy of the proposed architecture, several existing algorithms are considered, which include regularized generalized canonical correlation analysis (RGCCA) [187], multiset canonical correlation analysis (MCCA) [96], graph-regularized MCCA (GMCCA) [25], graph-regularized kernel MCCA (GMKCCA) [25], large-scale generalized canonical correlation analysis (LasCCA) [53], distributed algorithm for canonical correlation analysis (DisCCA) [53], multi-view discriminant analysis (MvDA) [92], MvDA with view-consistency (MvDA-VC) [93], multi-view common component discriminant analysis (MvCCDA) [217], multimodal deep Boltzmann machine (MDBM) [180], deep adversarial canonical correlation analysis (DACCA) [44], deep canonical correlation analysis with view generation (DCCA-VG) [193], towards deep and discriminative canonical correlation analysis (TDDCCA) [37], multimodal deep learning framework with cross-weights (MDL-CW) [157], multimodal graph neural network (MMGNN) [54], multi-view linear discriminant analysis network (MvLDAN) [79], multi-view graph restricted Boltzmann machine (mgRBM) [221], tensor canonical correlation analysis (TCCA) [213], multi-view generative adversarial network (MVGAN) [209], fGraphDBN [24], SPDNet [83], and CRBM [152]. The performance of the proposed approach as well as existing methods is demonstrated in terms of both training-testing and 10-fold cross-validation (CV). In case of training-testing, overall classification accuracy is considered, while 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 for 10-fold CV.

### 7.5.1 Description of Data Sets

In order to evaluate the performance of different algorithms, four benchmark databases, namely, Digits [63], Caltech [114], NUS-WIDE-OBJECT (NW-OBJECT) [27], and Animals with Attributes (AwA) [208], three HEp-2 cell image databases, and one real-life Virus data set [106] are considered. All the databases, studied in this chapter, are image based. For example, the Digits data set consists of handwritten numerals, Caltech and NW-OBJECT

databases contain images of various types of objects, and AwA data set is comprised of images of several kind of animals. Different staining patterns are identified from the three HEp-2 cell image databases, which include MIVIA (ICPR 2012 HEp-2 cell classification contest data set) [51], ICPR (ICPR 2014 HEp-2 cell classification contest data set) [75], and SNP [206]. The Virus data set contains negative stain transmission electron microscopy (TEM) images, which are primarily used for the detection of virus particles. A brief description of the NW-OBJECT data set is presented in Chapter 4, whereas the description of the Digits database is illustrated in Chapter 5. While a brief description of the AwA data set is provided in Chapter 6, the description of the HEp-2 cell image databases are summarized in Chapter 3. The Virus data set includes 15 different virus classes, each of which is represented by 100 TEM images.

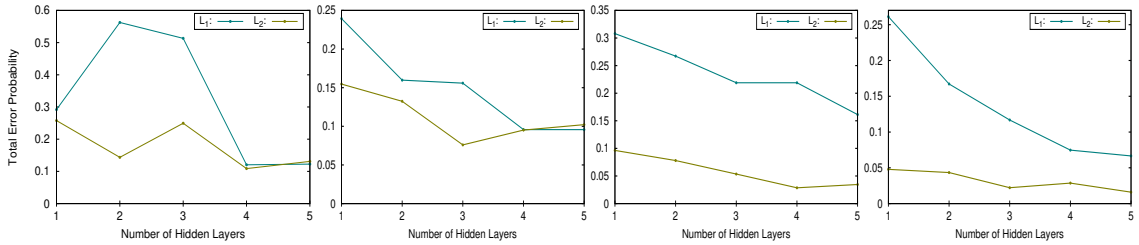
In case of benchmark databases, the pre-extracted feature sets are considered along with the corresponding images as input to the proposed D2JLE model as well as existing algorithms. For the HEp-2 cell image databases and the Virus data set, completed local binary pattern [61] at scales $S_1$, $S_2$, $S_3$, and $S_4$, and rotation-invariant co-occurrence of adjacent local binary pattern [145] at scales $S_1$, $S_2$, and $S_4$, are employed to provide the input feature sets along with the corresponding images for the evaluation of the proposed model and existing approaches. It is to be noted here that while the Digits and Caltech data sets exhibit large variance in the dimensionality of the input feature sets, the NW-OBJECT and ICPR data set have large number of samples with small dimensionality of the input feature sets. The MIVIA and SNP databases are represented by small number of samples having small number of classes, whereas the Virus data set is represented by small number of samples having large number of classes. On the other hand, AwA is a large scale database having large number of samples and classes with large dimension of each of the feature sets. Each data set is randomly partitioned into two sets for training-testing and ten separate folds for 10-fold CV. In both the cases, the samples are equally distributed with respect to given classes. Detailed description of all the data sets is reported in Appendix A.

### 7.5.2 Model Architecture and Implementation Details

In existing literature, the architecture of a deep framework is heuristically determined for all the databases under consideration. Hence, it does not take into account the diversities present in the nature of the problem as well as the complexities associated with the given data sets. However, in the proposed method, an upper bound on the error probability (7.38) is estimated in terms of the architecture of the model, which enables the framework to select an optimal architecture for the analysis of the given multi-view data. In the current study, greedy layer-wise pretraining [163] is performed to initialize the model parameters sensibly.

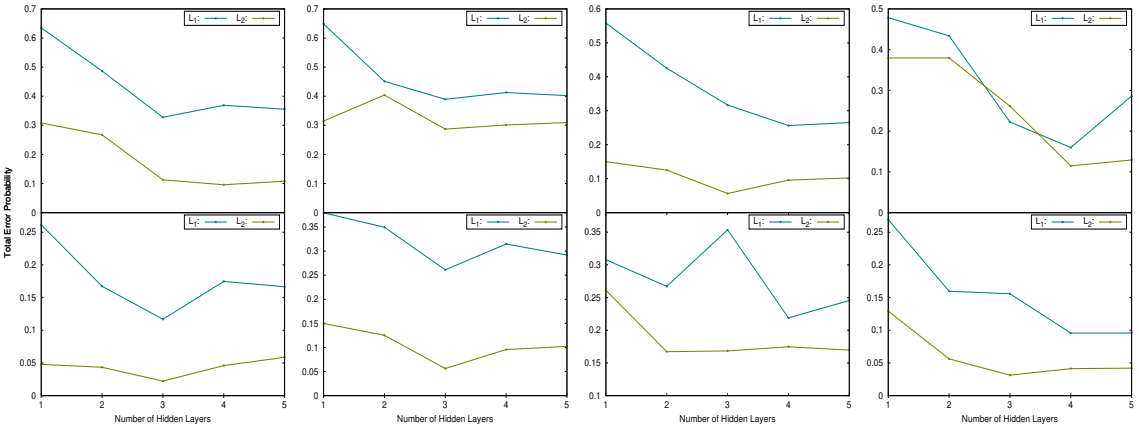
In order to determine the optimal number of layers in the proposed D2JLE model, extensive experiments are carried out on benchmark, HEp-2, and Virus data sets. During the pretraining, the number of modality-specific hidden layers ($L_1$) is varied from 1 to 5 for each of the data sets, keeping the number of joint hidden layers ($L_2$) fixed at 1 and the corresponding values of error bound are noted. Then, $L_1$ is fixed at the value for which the error bound has achieved the minimum value, while $L_2$ is varied from 1 to 5 and the variation in the error bound is observed. The value of $L_2$ for which error bound attains the minimum value is considered for the analysis of the particular data set. The variations of

error bound with respect to $L_1$ and $L_2$, corresponding to the benchmark, HEp-2, and Virus databases, are presented in Figure 7.2 and Figure 7.3, respectively, for both training-testing and 10-fold CV. The optimal values of $L_1$ and $L_2$, obtained from the corresponding error plots, are tabulated in Table 7.1. It establishes the fact that different number of layers is required by the proposed deep architecture to address the challenges offered by each of the data sets for various experimental set-up.




(a) Digits (b) Caltech (c) NW-OBJECT (d) AwA

Figure 7.2: Variation of error bound with respect to the architecture of the proposed D2JLE model on benchmark data.




(a) MIVIA (b) SNP (c) ICPR (d) Virus

Figure 7.3: Variation of error bound with respect to the architecture of the proposed D2JLE framework (top row: training-testing and bottom row: 10-fold CV on HEp-2 and Virus data).

The hidden nodes of the architecture are represented by the corresponding probability values and the parameters are updated based on the mini-batches formed from the given set of training samples. Each of the $L_1$ layers consists of 25 hidden nodes, whereas the $L_2$ joint layers have 10 hidden nodes each. The number of epochs, and the values of momentum and weight decay are considered to be 100, 0.5, and 0.0005, respectively. The value of learning rate is initialized at 0.01 and then, gradually decreased with the increase in number of epochs. For the estimation of data-independent expectations, 100 Gibbs steps and 20 separate Markov chains are considered.

Table 7.1: Optimal Number of Layers for D2JLE Model Based on Estimated Error Bound

| Different Metrices | Different Data Sets | | Number of Layers $(L_1, L_2)$ |
|---|---|---|---|
| Training-Testing | Benchmark | Digits | 4,4 |
| | | Caltech | 5,3 |
| | | NW-OBJECT | 5,4 |
| | | AwA | 5,5 |
| | HEp-2 | MIVIA | 3,4 |
| | | SNP | 3,3 |
| | | ICPR | 4,3 |
| | Virus | | 4,4 |
| 10-fold CV | HEp-2 | MIVIA | 3,3 |
| | | SNP | 3,3 |
| | | ICPR | 4,2 |
| | Virus | | 4,3 |

### 7.5.3 Effectiveness of Proposed Model for Image Analysis

In the current study, a novel deep learning model, termed as D2AIA, is developed for proper characterization and classification of an input image into different categories. In the proposed D2AIA model, the theory of LE [16] is judiciously incorporated into the framework of DDBM in such a way that the underlying low-dimensional image manifold is encapsulated efficiently using the LE algorithm and the corresponding locality preserving properties as well as the supervised information of sample categories are propagated properly through the deep layers of DDBM architecture. In this section, the performance of the proposed D2AIA model is studied on four benchmark, three HEp-2, and one Virus databases, considering only the image modality. The efficacy of the proposed D2AIA model is compared with that of the several state-of-the-art approaches, which are primarily considered to identify the low-dimensional embedding from the input high-dimensional pixel space. Firstly, the input image is converted into one-dimensional representation using row concatenation and then the representation is provided as input to the DDBM model. The corresponding model is referred to as Concat+DDBM. Similarly, PCA [195], LDA [14], and CNN [21] are applied to the input images for obtaining the low-dimensional representations, which are then applied to the input of DDBM and the corresponding approaches are termed as PCA+DDBM, LDA+DDBM, and CNN+DDBM, respectively. It is to be noted here that the architecture of the DDBM model remains the same for the existing approaches as well as the proposed model.

Considering the results reported in Table 7.2, it can be observed that although the CNN+DDBM approach achieves satisfactory results on all the benchmark databases, the proposed D2AIA model attains highest classification accuracy with respect to the existing image analysis based methods on all four benchmark databases considered. In case of HEp-2 and Virus data sets, the results are presented in Table 7.3, which demonstrate that the proposed model outperforms four existing approaches for image analysis on all the three HEp-2 cell image databases as well as the real-life Virus database. From the p-values obtained through two statistical significance tests, it can be observed that the proposed

Table 7.2: Classification Accuracy of Proposed Model for Image Analysis on Benchmark Data

| Data Sets | Concat + DDBM | PCA + DDBM | LDA + DDBM | CNN + DDBM | D2AIA |
|---|---|---|---|---|---|
| Digits | 34.60 | 71.20 | 84.70 | 89.00 | **91.20** |
| Caltech | 40.68 | 68.44 | 73.26 | 78.45 | **82.64** |
| NW-OBJECT | 37.95 | 43.64 | 37.72 | 48.27 | **53.94** |
| AwA | 42.22 | 61.92 | 67.08 | 80.30 | **85.95** |

Table 7.3: Classification Accuracy of Proposed Model for Image Analysis on HEp-2 and Virus Data Sets

| Data Sets | Different Metrics | | Concat + DDBM | PCA + DDBM | LDA + DDBM | CNN + DDBM | D2AIA |
|---|---|---|---|---|---|---|---|
| MIVIA | Train-Test | | 24.52 | 45.10 | 51.77 | 53.13 | **70.84** |
| | 10-fold CV | Mean | 41.77 | 47.19 | 52.81 | 59.60 | **72.27** |
| | | Median | 43.05 | 48.72 | 52.21 | 58.99 | **73.58** |
| | | StdDev | 5.94 | 8.83 | 5.00 | **3.28** | 4.47 |
| | | Paired-$t$:p | 2.53E-10 | 1.85E-05 | 2.13E-06 | 6.81E-05 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | - |
| SNP | Train-Test | | 49.63 | 61.47 | 59.98 | 65.74 | **70.86** |
| | 10-fold CV | Mean | 54.29 | 60.22 | 63.38 | 66.08 | **71.42** |
| | | Median | 55.34 | 61.63 | 62.33 | 65.55 | **72.17** |
| | | StdDev | 5.18 | 4.41 | 4.81 | 4.09 | **3.17** |
| | | Paired-$t$:p | 3.86E-06 | 5.77E-05 | 1.17E-03 | 1.17E-03 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.53E-03 | 3.46E-03 | - |
| ICPR | Train-Test | | 39.79 | 58.83 | 64.88 | 66.57 | **90.23** |
| | 10-fold CV | Mean | 50.39 | 51.78 | 52.54 | 55.97 | **84.08** |
| | | Median | 51.53 | 52.31 | 54.99 | 54.98 | **83.94** |
| | | StdDev | 9.59 | 6.16 | 5.83 | 3.79 | **3.74** |
| | | Paired-$t$:p | 2.69E-06 | 1.71E-07 | 1.15E-07 | 6.81E-09 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | - |
| Virus | Train-Test | | 57.87 | 59.60 | 70.93 | 60.67 | **83.07** |
| | 10-fold CV | Mean | 64.00 | 70.07 | 77.40 | 78.33 | **83.60** |
| | | Median | 64.00 | 70.67 | 77.33 | 77.67 | **83.67** |
| | | StdDev | 3.30 | 3.18 | 2.62 | 4.85 | **2.22** |
| | | Paired-$t$:p | 3.18E-08 | 5.32E-07 | 5.31E-04 | 4.11E-03 | - |
| | | Wilcoxon:p | 2.52E-03 | 2.52E-03 | 3.42E-03 | 8.27E-03 | - |

D2AIA model attains significantly better p-values for all the 32 cases. Hence, it is evident that the topological properties of the input images are not only extracted efficiently but also propagated properly through the deep layers of the proposed D2AIA model, which in turn, allows the model to appropriately classify the input images into different categories.

### 7.5.4 Comparative Performance Analysis

Finally, the performance of the proposed D2JLE model is studied on four benchmark, three HEp-2 cell, and Virus data sets with reference to several existing approaches, which include consensus principle based methods, complementary principle based approaches, both consensus and complementary principles based methods, and spatial proximity based approaches. The corresponding results are reported in Table 7.4, Table 7.5, Table 7.6, Table 7.7, Table 7.8, Table 7.9, Table 7.10, and Table 7.11. It is to be mentioned here that, in case of proposed framework, the class labels of input samples are predicted from the architecture itself. Hence, no classifier is employed in the proposed method for classification purpose.

Table 7.4: Performance Analysis of Consensus Principle Based Approaches on Benchmark Data

| Data Sets | RGCCA | MCCA | GMCCA | GMKCCA | LasCCA | DisCCA | DCCA-VG | MDL-CW | D2JLE |
|---|---|---|---|---|---|---|---|---|---|
| Digits | 90.30 | 87.00 | 11.20 | 6.60 | 10.20 | 5.60 | 89.00 | 86.40 | **97.50** |
| Caltech | 33.71 | 41.83 | 4.82 | 7.48 | 4.06 | 3.17 | 49.68 | 54.25 | **93.79** |
| NW-OBJECT | 18.91 | 30.34 | 4.56 | 6.43 | 7.40 | 10.93 | 19.23 | 18.20 | **58.32** |
| AwA | 28.32 | 38.83 | 41.62 | 51.36 | 60.19 | 62.69 | 45.00 | 60.25 | **90.52** |

Table 7.5: Performance Analysis of Consensus Principle Based Approaches on HEp-2 and Virus Data Sets

| Data Sets | Different Metrics | | RGCCA | MCCA | GMCCA | GMKCCA | LasCCA | DisCCA | DCCA-VG | MDL-CW | D2JLE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| MIVIA | Train-Test | | 56.95 | 55.86 | 56.81 | 57.08 | 59.67 | 58.99 | 65.26 | 63.08 | **73.02** |
| | 10-fold CV | Mean | 67.69 | 70.68 | 70.27 | 69.25 | 69.05 | 68.57 | 69.18 | 68.84 | **95.40** |
| | | Median | 67.41 | 70.14 | 71.16 | 69.80 | 69.46 | 68.78 | 69.12 | 68.44 | **95.52** |
| | | StdDev | 2.88 | 2.40 | 2.97 | 2.40 | 2.32 | 2.26 | **1.52** | 2.08 | **1.52** |
| | | Paired-$t$:p | 1.46E-11 | 4.88E-10 | 2.20E-10 | 3.95E-11 | 1.33E-12 | 3.00E-12 | 9.72E-13 | 3.02E-13 | - |
| | | Wilcoxon:p | 2.52E-03 | 2.53E-03 | 2.52E-03 | 2.52E-03 | 2.52E-03 | 2.52E-03 | 2.50E-03 | 2.52E-03 | - |
| SNP | Train-Test | | 60.19 | 57.63 | 51.97 | 57.84 | 55.71 | 53.47 | 62.97 | 61.26 | **72.79** |
| | 10-fold CV | Mean | 67.10 | 70.66 | 70.77 | 70.00 | 68.47 | 67.76 | 73.01 | 71.09 | **89.19** |
| | | Median | 69.40 | 72.68 | 70.77 | 69.95 | 70.49 | 70.49 | 73.50 | 71.86 | **89.67** |
| | | StdDev | 8.18 | 6.33 | 5.08 | 4.96 | 7.84 | 9.02 | 3.60 | **3.55** | 4.17 |
| | | Paired-$t$:p | 8.22E-06 | 7.03E-06 | 1.47E-06 | 1.80E-06 | 7.36E-06 | 1.93E-05 | 6.01E-07 | 1.82E-07 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.52E-03 | 2.53E-03 | 2.53E-03 | 2.52E-03 | - |
| ICPR | Train-Test | | 82.28 | 81.31 | 81.47 | 81.00 | 81.82 | 81.48 | 83.87 | 83.28 | **94.04** |
| | 10-fold CV | Mean | 82.65 | 80.99 | 71.54 | 72.95 | 76.26 | 74.45 | 81.26 | 77.02 | **91.16** |
| | | Median | 82.91 | 80.45 | 69.90 | 73.79 | 77.17 | 74.92 | 80.56 | 76.67 | **91.44** |
| | | StdDev | 3.32 | 4.67 | 5.97 | 5.60 | 5.80 | 7.42 | 3.54 | 3.71 | **2.92** |
| | | Paired-$t$:p | 1.28E-05 | 5.99E-08 | 1.80E-08 | 4.74E-08 | 2.81E-06 | 1.34E-05 | 4.18E-10 | 9.51E-11 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.52E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | - |
| Virus | Train-Test | | 75.60 | 75.67 | 73.47 | 71.20 | 67.67 | 76.40 | 79.67 | 78.67 | **88.93** |
| | 10-fold CV | Mean | 73.93 | 69.80 | 71.60 | 70.47 | 68.60 | 67.07 | 78.53 | 76.47 | **83.77** |
| | | Median | 74.33 | 69.33 | 71.00 | 70.33 | 70.33 | 70.00 | 78.33 | 77.00 | **84.33** |
| | | StdDev | 3.76 | 3.19 | 3.28 | 2.98 | 5.26 | 5.99 | 3.09 | 2.93 | **1.81** |
| | | Paired-$t$:p | 2.02E-06 | 6.01E-09 | 2.70E-07 | 9.07E-10 | 4.79E-06 | 3.61E-06 | 3.11E-04 | 2.10E-05 | - |
| | | Wilcoxon:p | 2.50E-03 | 2.50E-03 | 2.47E-03 | 2.52E-03 | 2.50E-03 | 2.53E-03 | 3.84E-03 | 2.49E-03 | - |

#### 7.5.4.1 Performance of Consensus Principle Based Methods

In this section, several state-of-the-art consensus principle based methods, namely, RGCCA [187], MCCA [96], GMCCA [25], GMKCCA [25], LasCCA [53], DisCCA [53], DCCA-VG [193], and MDL-CW [157], are considered for performance evaluation of the proposed D2JLE model. Out of these methods, RGCCA, MCCA, GMCCA, GMKCCA, LasCCA, and DisCCA are classical approaches, whereas DCCA-VG and MDL-CW are deep learning models. The experimental set-up for the existing consensus principle based approaches is discussed in details in Section 6.5.4.1 of Chapter 6. From the results reported in Table 7.4, it can be observed that although RGCCA and MCCA achieve considerable classification accuracy on Digits data set, they fail to attain similar results on rest of the benchmark databases. However, the proposed model exhibits significantly better performance with respect to the existing consensus principle based methods on all the four benchmark databases. The results reported in Table 7.5 corresponding to the HEp-2 and Virus data sets demonstrate that the proposed method outperforms all the eight multiset consensus principle based methods on three HEp-2 cell and Virus data sets for both training-testing and 10-fold CV. The statistical significance test reveals that the proposed architecture achieves significantly better p-values for all the 64 cases.

Table 7.6: Performance Analysis of Complementary Principle Based Methods on Benchmark Data

| Data | MvDA | MvDA-VC | MvCCDA | MvLDAN | MVGAN | D2JLE |
|---|---|---|---|---|---|---|
| Digits | 92.40 | 93.50 | 92.80 | 90.70 | 82.70 | **97.50** |
| Caltech | 76.30 | 75.29 | 76.68 | 79.47 | 77.31 | **93.79** |
| NW-OBJECT | 29.03 | 28.62 | 37.33 | 36.27 | 27.61 | **58.32** |
| AwA | 55.31 | 65.82 | 59.64 | 61.76 | 58.47 | **90.52** |

#### 7.5.4.2 Performance of Complementary Principle Based Approaches

Here, the performance of the proposed model is analyzed with reference to several complementary principle based approaches, namely, MvDA [92], MvDA-VC [93], MvCCDA [217], MvLDAN [79], and MVGAN [209]. Out of these methods, MvDA, MvDA-VC, and MvCCDA are classical approaches, whereas MvLDAN and MVGAN are deep learning models. The experimental set-up for the existing complementary principle based approaches is discussed in details in Section 6.5.4.2 of Chapter 6. The results corresponding to Table 7.6 demonstrate that the complementary principle based methods achieve considerable accuracy on all the four benchmark databases. However, the highest classification accuracy is attained by the proposed model in all the cases. From the results reported in Table 7.7, it can be observed that the proposed model performs considerably better than all the existing approaches on all the three HEp-2 cell and Virus data sets for both training-testing and 10-fold CV. The p-values obtained from the two statistical significance tests indicate that the proposed model attains significantly better p-values for all the 40 cases.

Table 7.7: Performance Analysis of Complementary Principle Based Approaches on HEp-2 and Virus Data Sets

| Data | Different Metrics | | MvDA | MvDA-VC | MvCCDA | MvLDAN | MVGAN | D2JLE |
|---|---|---|---|---|---|---|---|---|
| MIVIA | Train-Test | | 63.62 | 64.99 | 64.31 | 63.22 | 63.90 | **73.02** |
| | 10-fold CV | Mean | 78.93 | 77.96 | 80.57 | 84.18 | 85.64 | **95.40** |
| | | Median | 78.78 | 78.44 | 80.14 | 83.78 | 85.10 | **95.52** |
| | | StdDev | 3.74 | 2.68 | 2.28 | 2.59 | 3.30 | **1.52** |
| | | Paired-$t$:p | 5.55E-07 | 1.66E-09 | 2.17E-08 | 1.05E-08 | 4.14E-07 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.52E-03 | 2.52E-03 | 2.52E-03 | 2.52E-03 | - |
| SNP | Train-Test | | 62.22 | 62.22 | 63.07 | 63.61 | 61.47 | **72.79** |
| | 10-fold CV | Mean | 71.53 | 70.22 | 79.07 | 74.28 | 72.81 | **89.19** |
| | | Median | 73.50 | 70.77 | 82.40 | 75.85 | 71.67 | **89.67** |
| | | StdDev | 6.17 | 5.33 | 6.74 | 9.44 | 5.69 | **4.17** |
| | | Paired-$t$:p | 2.55E-06 | 2.05E-06 | 3.69E-04 | 5.97E-04 | 3.20E-05 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | - |
| ICPR | Train-Test | | 82.13 | 81.70 | 83.37 | 83.51 | 83.97 | **94.04** |
| | 10-fold CV | Mean | 70.76 | 71.23 | 71.06 | 74.80 | 72.34 | **91.16** |
| | | Median | 70.19 | 70.04 | 70.85 | 75.48 | 72.33 | **91.44** |
| | | StdDev | 4.86 | 4.70 | 5.08 | **2.78** | 3.81 | 2.92 |
| | | Paired-$t$:p | 1.86E-10 | 1.30E-10 | 2.72E-10 | 2.73E-07 | 2.06E-08 | - |
| | | Wilcoxon:p | 2.52E-03 | 2.52E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | - |
| Virus | Train-Test | | 79.47 | 80.13 | 79.47 | 80.60 | 80.53 | **88.93** |
| | 10-fold CV | Mean | 75.37 | 77.40 | 76.00 | 79.27 | 78.53 | **83.77** |
| | | Median | 75.67 | 77.33 | 75.67 | 80.00 | 79.67 | **84.33** |
| | | StdDev | 3.33 | 3.42 | 3.64 | 2.82 | 3.17 | **1.81** |
| | | Paired-$t$:p | 1.07E-05 | 4.46E-05 | 1.98E-05 | 1.71E-04 | 5.17E-04 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.47E-03 | 2.52E-03 | 2.52E-03 | 3.82E-03 | - |

#### 7.5.4.3 Performance of Consensus and Complementary Principles Based Methods

The proficiency of the proposed framework is compared with that of several state-of-the-art methods, which are based on both the consensus and complementary principles. These methods include MDBM [180], mgRBM [221], DACCA [44], TCCA [213], TDDCCA [37], and MMGNN [54]. 0 Out of these methods, mgRBM is a classical approaches, whereas MDBM, DACCA, TCCA, TDDCCA, and MMGNN are deep learning models. The shared subspace is represented with 2048, 50, 80, 50, $64|V|B$, and 50 features by MDBM, mgRBM, DACCA, TDDCCA, TCCA, and MMGNN, respectively, where $|V|$ and $B$ denote the number of input views and total number of batches, considered for a particular data set. The architecture for each of these models follows the same as described in the corresponding papers. From the results reported in Table 7.8, it is evident that significant improvement in classification accuracy is achieved by the proposed architecture in comparison to the existing multi-view approaches those are based on both consensus and complementary principles, for all the benchmark databases, except for TCCA on Digits database. In case

Table 7.8: Comparative Performance Analysis of Both Correlation and Complementary Based Approaches on Benchmark Databases

| Data Sets | MDBM | mgRBM | DACCA | TCCA | TDDCCA | MMGNN | D2JLE |
|---|---|---|---|---|---|---|---|
| Digits | 10.00 | 88.60 | 84.60 | **97.80** | 85.90 | 88.90 | 97.50 |
| Caltech | 2.53 | 78.75 | 73.89 | 87.58 | 74.52 | 74.27 | **93.79** |
| NW-OBJECT | 26.07 | 32.16 | 38.42 | 33.61 | 17.80 | 37.87 | **58.32** |
| AwA | 56.86 | 68.57 | 47.60 | 63.46 | 62.49 | 63.01 | **90.52** |

Table 7.9: Comparative Performance Analysis of Both Correlation and Complementary Based Approaches on HEp-2 and Virus Data Sets

| Data | Different Metrics | | MDBM | mgRBM | DACCA | TCCA | TDDCCA | MMGNN | D2JLE |
|---|---|---|---|---|---|---|---|---|---|
| MIVIA | Train-Test | | 61.17 | 65.67 | 65.12 | 67.22 | 64.80 | 64.99 | **73.02** |
| | 10-fold CV | Mean | 85.78 | 87.62 | 86.60 | 90.34 | 88.64 | 89.66 | **95.40** |
| | | Median | 85.03 | 87.76 | 86.73 | 90.82 | 88.78 | 89.80 | **95.52** |
| | | StdDev | 3.40 | 2.77 | 2.57 | 2.05 | 2.10 | 2.07 | **1.52** |
| | | Paired-$t$:p | 1.03E-06 | 9.56E-07 | 7.08E-07 | 5.93E-06 | 6.11E-07 | 9.31E-06 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.52E-03 | 2.50E-03 | 2.50E-03 | 2.53E-03 | - |
| SNP | Train-Test | | 62.93 | 65.28 | 67.32 | 70.40 | 69.07 | 68.38 | **72.79** |
| | 10-fold CV | Mean | 65.46 | 67.36 | 68.36 | 78.14 | 73.31 | 71.31 | **89.19** |
| | | Median | 68.58 | 70.49 | 70.77 | 82.13 | 73.22 | 73.22 | **89.67** |
| | | StdDev | 6.83 | 7.53 | 5.21 | 7.65 | 5.86 | 6.26 | **4.17** |
| | | Paired-$t$:p | 5.08E-07 | 1.51E-05 | 1.92E-06 | 5.26E-04 | 7.50E-06 | 8.77E-06 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | - |
| ICPR | Train-Test | | 81.94 | 83.00 | 83.13 | 85.51 | 84.38 | 83.89 | **94.04** |
| | 10-fold CV | Mean | 72.40 | 79.72 | 76.75 | 81.89 | 81.09 | 80.68 | **91.16** |
| | | Median | 71.94 | 79.13 | 75.61 | 81.51 | 80.48 | 79.46 | **91.44** |
| | | StdDev | 4.40 | 4.23 | 3.88 | 4.01 | 3.55 | 4.11 | **2.92** |
| | | Paired-$t$:p | 7.03E-09 | 3.40E-09 | 3.68E-09 | 2.10E-08 | 4.60E-10 | 3.74E-09 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | 2.53E-03 | - |
| Virus | Train-Test | | 73.93 | 81.00 | 81.40 | 82.13 | 81.73 | 81.60 | **88.93** |
| | 10-fold CV | Mean | 71.60 | 74.53 | 77.80 | 81.20 | 79.20 | 79.20 | **83.77** |
| | | Median | 69.67 | 74.00 | 77.33 | 80.33 | 78.67 | 79.00 | **84.33** |
| | | StdDev | 4.81 | 3.72 | 4.36 | 2.66 | 3.03 | 3.66 | **1.81** |
| | | Paired-$t$:p | 1.30E-05 | 6.25E-06 | 4.24E-04 | 5.19E-04 | 5.18E-05 | 7.97E-04 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.50E-03 | 3.74E-03 | 3.79E-03 | 2.52E-03 | 3.46E-03 | - |

of HEp-2 and Virus data sets, the results are presented in Table 7.9, which signify that the proposed model outperforms six existing methods for all three HEp-2 cell image data sets and the real-life Virus database considered. Statistical significance analysis reveals that the proposed model achieves significantly better p-values for all the 48 cases.

Table 7.10: Performance Analysis of Spatial Proximity Based Approaches on Benchmark Databases

| Data Sets | fGraphDBN | SPDNet | CRBM | D2JLE |
|---|---|---|---|---|
| Digits | 96.90 | **98.20** | 92.60 | 97.50 |
| Caltech | **94.68** | 89.23 | 90.37 | 93.79 |
| NW-OBJECT | 50.45 | 53.51 | 55.75 | **58.32** |
| AwA | 85.01 | 88.04 | 87.62 | **90.52** |

Table 7.11: Performance Analysis of Spatial Proximity Based Approaches on HEp-2 and Virus Data Sets

| Data | Different Metrics | | fGraphDBN | SPDNet | CRBM | D2JLE |
|---|---|---|---|---|---|---|
| MIVIA | Train-Test | | 66.35 | 68.99 | 65.12 | **73.02** |
| | 10-fold CV | Mean | 89.49 | 92.14 | 90.26 | **95.40** |
| | | Median | 89.12 | 91.91 | 89.62 | **95.52** |
| | | StdDev | 2.07 | **1.01** | 1.88 | 1.52 |
| | | Paired-$t$:p | 6.10E-07 | 1.80E-06 | 3.20E-06 | - |
| | | Wilcoxon:p | 2.52E-03 | 2.53E-03 | 2.52E-03 | - |
| SNP | Train-Test | | 68.47 | 71.86 | **72.97** | 72.79 |
| | 10-fold CV | Mean | 76.60 | 80.18 | 81.04 | **89.19** |
| | | Median | 78.58 | 79.95 | 82.32 | **89.67** |
| | | StdDev | 6.60 | 4.63 | 6.76 | **4.17** |
| | | Paired-$t$:p | 5.34E-05 | 3.88E-04 | 4.48E-03 | - |
| | | Wilcoxon:p | 2.53E-03 | 3.44E-03 | 6.26E-03 | - |
| ICPR | Train-Test | | 88.23 | 89.38 | 85.12 | **94.04** |
| | 10-fold CV | Mean | 80.86 | 81.27 | 82.70 | **91.16** |
| | | Median | 80.85 | 80.56 | 82.25 | **91.44** |
| | | StdDev | 4.41 | 4.74 | 4.09 | **2.92** |
| | | Paired-$t$:p | 4.25E-09 | 5.06E-08 | 3.35E-06 | - |
| | | Wilcoxon:p | 2.53E-03 | 2.52E-03 | 2.52E-03 | - |
| Virus | Train-Test | | 81.00 | 81.53 | 81.87 | **88.93** |
| | 10-fold CV | Mean | 76.13 | 79.93 | 80.53 | **83.77** |
| | | Median | 75.33 | 79.67 | 80.00 | **84.33** |
| | | StdDev | 3.68 | 3.30 | 3.12 | **1.81** |
| | | Paired-$t$:p | 1.97E-05 | 4.20E-04 | 3.01E-03 | - |
| | | Wilcoxon:p | 2.53E-03 | 3.40E-03 | 6.23E-03 | - |

#### 7.5.4.4 Performance of Spatial Proximity Based Approaches

Finally, various state-of-the-art spatial proximity based approaches, namely, fGraphDBN [24], SPDNet [83], and CRBM [152], are considered for the performance evaluation of the proposed model, and the corresponding results are reported in Table 7.10 and Table 7.11. Out of these methods, only CRBM is a classical approach, whereas fGraphDBN and SPDNet are deep learning models. The shared subspace is represented with 50, 100,

and 5000 features by CRBM, fGraphDBN, and SPDNet, respectively. The architecture for each of these models follows the same as described in the corresponding papers. Considering the results reported in Table 7.10, it can be observed that considerable classification accuracy is achieved by the spatial proximity based approaches on all the benchmark databases. In fact, the fGraphDBN and SPDNet models attain highest classification accuracy on Caltech and Digits data sets, respectively. However, the proposed D2JLE model performs significantly better than the state-of-the-art approaches on rest of the benchmark databases. In case of HEp-2 data sets, the results are presented in Table 7.11, which demonstrate that although similar classification accuracy is achieved by the existing methods, the highest classification accuracy is attained by the proposed model on all three HEp-2 cell data sets, expect for CRBM on SNP database. Statistical significance analysis reveals that the proposed model achieves significantly better p-values for all the 18 cases. For the Virus data set, the results are reported in Table 7.9 which signify that the proposed model outperforms three existing methods on the real-life Virus database considered. From the p-values obtained through two statistical significance tests, it can be observed that the proposed D2JLE model attains significantly better p-values for all the 6 cases.

The better performance of the proposed D2JLE model over state-of-the-art approaches is achieved due to the following reasons:

- the proposed model is developed based on the framework of MDDBM, and so, the joint subspace of the model efficiently encapsulates the latent non-linear data distribution over the space of multimodal inputs;
- the discriminative ability of the model is enhanced by incorporating the supervised information of sample categories at each layer of the framework;
- the intrinsic topological properties of the image modalities are precisely captured and propagated through the deep layers of the model using the objective function of LE;
- the geometric structures corresponding to the multiple image modalities are consolidate properly in the joint subspace through approximate joint diagonalization of graph Laplacians; and
- the relevance of each of the given input modality is evaluated based on which all the imperative information is integrated in the joint subspace.

In effect, the proposed model provides significantly better performance as compared to existing approaches.

## 7.6 Conclusion

The primary contributions of this chapter is five-fold, namely, (a) developing D2AIA model for efficient characterization and classification of a given set of input images into different categories; (b) introducing D2JLE model by judiciously integrating the theory of LE and simultaneous diagonalization of Laplacians with the learning objective of MDDBM framework; (c) determining the data specific architecture of the proposed model based on the estimated upper bound of the total error probability of the network; (d) consolidating the theory of the proposed framework with convergence analysis; and finally, (e) illustrating

the efficacy of the D2JLE model on different domains of applications, namely, object detection, staining pattern recognition from HEp-2 cell images, and virus particle identification from negative stain transmission electron microscopy images.

This chapter presents a novel geometrically motivated deep predictive model, termed as D2JLE, which can process multiple image and non-image modalities simultaneously. It considers the theory of LE in order to recognize and represent the geometric structures of the image manifold, embedded in the high-dimensional pixel space. The corresponding topological properties of a given image modality are propagated precisely through the deep layers of the framework. The intrinsic geometric structures of multiple image modalities are appropriately consolidated in the joint subspace of the D2JLE model by computing an approximate common eigenbasis of multiple Laplacians simultaneously. The imperative information corresponding to both image and non-image modalities are incorporated in the joint subspace through the learning objective of MDDBM framework. A relevance measure is proposed in the current study for each input modality based on the discrimination criterion of the corresponding embedding and subsequently, the joint subspace is learned from the weighted combination of the individual subspaces. An upper bound on the error probability of the proposed model is estimated in terms of the model architecture, which allows determination of the optimal architecture of the model for each database considered. The proposed D2JLE model is further consolidated with convergence analysis. The proficiency of the proposed model is established on four benchmark databases, three HEp-2 cell image data sets, and the real-life Virus database considering both training-testing and 10-fold CV.

# Chapter 8

# Conclusion and Future Directions

A concise summary of the primary contributions of the research work, discussed in the corresponding chapters, is reported in this chapter. The future scope of work, along with the possible extensions and applications, are also provided in this chapter.

## 8.1 Major Contributions

The thesis focuses on devising a set of novel models to address the classification problem of multi-view data. The significant challenges associated with multi-view classification include (i) selection of relevant views for proper characterization of the given classes, (ii) learning discriminative joint subspace for efficient encapsulation of the latent non-linear data distribution, (iii) extracting correlated structures across different views (iv) capturing cross-view dependency between each pair of given input views and (v) consolidating image and non-image information while preserving the topological properties of the image views. All the aforementioned issues have been addressed in this thesis. The principle attributes of the proposed models are summarized next.

Chapter 3 introduces a novel class-pair specific descriptive selection method, which identifies a set of relevant descriptors for representing the intrinsic properties of a particular pair of classes, and then the final feature set for multiple classes is formed from the relevant descriptors of all possible pairs of classes. To evaluate the efficacy of a descriptor in discriminating observations from a given class-pair, a new framework, termed as Rough-Bayesian model, is developed by judiciously integrating the merits of rough sets and Bayes decision theory. During the computation of the relevance of each given descriptors, the proposed method takes care of the presence of both noisy features in a descriptors and noisy observations in an input class. Finally, support vector machine is used to predict the class labels of the given observations. The performance of the proposed method is studied with reference to several state-of-the-art approaches on a set of real-life human epithelial type 2 (HEp-2) cell image databases. Significant improvement in classification performance of HEp-2 cell images is noted if class-pair specific descriptors are considered, instead of selecting a uniform set of descriptors for multiple classes.

The results reported in Chapter 3 demonstrate that given the set deep features as input, the proposed method provides outstanding performance in comparison to the hand-crafted

features for all data sets considered. This is due to the fact that the deep architectures takes into account the nature and complexity of the given classification problem while extracting features at various layers. Hence, in Chapter 4, a novel deep framework, termed as multimodal discriminative deep Boltzmann machine (MDDBM), is proposed which provides better classification performance through end-to-end learning. The learning objective of the proposed architecture includes the merits of deep Boltzmann machines. It enables the network to figure out activations for the architecture nodes in each layer, which constitute a plausible explanation of how the observed data vectors would have been generated. 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 classifier. The proficiency of the proposed architecture is demonstrated with reference to several classical as well as deep learning models on various multi-view benchmark and real-life cancer data sets.

Learning the joint subspace from the modality-specific subspaces becomes difficult if the individual views correspond to completely different sources. However, if the proposed model is learned in such a way that the modality-specific subspaces are highly correlated, then the inherent characteristics of the views can be efficiently modeled by the joint subspace. In this regard, a novel architecture, termed as discriminative deep canonical correlation analysis (D2CCA), is proposed in Chapter 5, by incorporating the theory of canonical correlation analysis into the learning objective of MDDBM, discussed in Chapter 4. Hence, the joint subspace can efficiently capture the non-linear correlated structures across different views. Considering supervised information into the objective function enhances the discriminative ability of the joint subspace, which in turn, entitles the D2CCA model to serve as the feature extractor as well as classifier. 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 different domains of applications, namely, object recognition, document classification, multilingual categorization, and cancer subtype identification.

The D2CCA framework, discussed in Chapter 5, is developed primarily based on the consensus principle. However, the complementary knowledge among different views also contain useful information, which may essentially facilitate accurate classification of given observations into different categories. In this regard, a novel deep learning model, termed as discriminative deep generalized dependency analysis (D2GDA), is proposed in Chapter 6 to address view-consistency and view discrepancy simultaneously. Based on the concept of Hilbert-Schmidt independence criterion, a loss function is proposed to efficiently capture the cross-view dependency across several views. A view-pair specific constraint is incorporated in the loss function to extract the relevant cross-view information in terms of consensus and/or complementary knowledge from the given input pairs of views. Incorporating the loss function, corresponding to the proposed cross-view dependency analysis, into the learning objective of MDDBM architecture, introduced in Chapter 4, enables the D2GDA model to encapsulate the latent probability distribution of the given multimodal data as well as predict class labels of the given observations. The error analysis, generalization ability, and convergence analysis establish the efficacy of the proposed model. The comparative performance analysis demonstrates the proficiency of the proposed model on several multi-view benchmark and real-life cancer data sets.

Combining information from multiple modalities is particularly challenging when the

input modalities involve both image and non-image information. It is primarily due to the fact that as opposed to the non-image counterparts, the image modalities embody neighborhood information which needs to be encapsulated properly for efficient representation of the given input data. In this regard, a novel geometrically motivated deep predictive model, termed as discriminative deep joint Laplacian embedding (D2JLE), is introduced in Chapter 7, which can process multiple image and non-image modalities simultaneously. It considers the theory of Laplacian eigenmap in order to recognize and represent the geometric structures of the image manifold, embedded in the high-dimensional pixel space. The intrinsic structures of multiple image modalities are appropriately consolidated in the joint subspace of the D2JLE model by computing an approximate common eigenbasis of multiple Laplacians simultaneously. The imperative information corresponding to both image and non-image modalities are incorporated in the joint subspace through the learning objective of MDDBM framework. A relevance measure is proposed in the current study for each input modality and subsequently, the joint subspace is learned from the weighted combination of the individual subspaces. The error analysis and convergence analysis establish the efficacy of the proposed model. The proficiency of the proposed model is established on four benchmark databases, three HEp-2 cell image data sets, and the real-life Virus database with reference to several state-of-the-art spatial proximity based approaches.

In brief, the concept of analyzing multiple image and non-image views simultaneously proposed in this thesis is unique.

## 8.2 Future Directions

There are various key characteristics of the research presented in this thesis that can be extended further for the progress of multi-view data analysis. Some improvements and future directions are reported next with which the research can be continued.

- **Incomplete views**: The multi-view predictive models, proposed in this thesis, considers that all the views are available for the analysis of the given set of observations. However, in real-world applications, it may so happen that some of the observations have missing values in certain views due to the presence of noise or measurement errors. Multi-view deep model can be developed to process input data with incomplete views. Given the available views as input, the deep model can disentangle the factors of variations and comprehend the hidden attributes, so that the missing values of the corresponding observations can be inferred.

- **Updation in Database**: The proposed predictive models in the thesis are developed based on the assumption that all the views are available for analysis of the given multi-view data. However, a large amount of data is being incorporated in the existing databases on a regular basis. Either a set of new observations is added or even an entire new view is generated for better analysis of the given observations. For instance, the databases in The Cancer Genome Atlas (TCGA) (https://cancergenome.nih.gov/) have been updated, both in terms of observations and views, for more than 20 times over the last 5 years. Hence, the learning of the models needs to be modified in such a way that it can adapt to the changes in the databases robustly.

- **Forward-Forward Learning**: It is a greedy multi-layer learning procedure, developed based on the concept of noise contrastive estimation [62] and Boltzmann machines [74]. The key idea of the learning procedure is to substitute the forward and backward passes of backpropagation algorithm by two forward passes which can operate in the same way as each other, but on different data and with opposite objectives. While the positive pass operates on real data, the negative pass operates on data generated from model distribution. The advantage of the forward-forward learning procedure over backpropagation algorithm is that it can be applied to cases where the detailed description of the forward computation is not known beforehand. The proposed multi-view deep models can be learned using this procedure to predict the class-label of unknown observations.

- **Tensor Based Learning**: In multi-view data analysis, tensor is the extension of matrix factorization. In tensor-based learning models, canonical polyadic decomposition is generally performed on the parameters of the model. As a consequence, the model can process multilinear arrays, exploiting the structural information along every data dimension. Additionally, the number of parameters required to be trained in the model decreases significantly. Hence, incorporating tensor based learning in the proposed multi-view predictive models can result into efficient and fast learning algorithm.

# Appendix A

# Description of Data Sets

The appendix presents a brief description of the HEp-2 cell image data sets, benchmark databases, real-life cancer data sets, and the Virus database considered in this thesis for performance evaluation of the proposed models as well as state-of-the-art approaches. Overall, three HEp-2 cell image data sets, namely, MIVIA database (ICPR 2012 HEp-2 cell classification contest data set) [51], ICPR image database (ICPR 2014 HEp-2 cell classification contest data set) [75], and SNP HEp-2 database [206]; seven benchmark databases, namely, Digits [63], Caltech [114], CiteSeer [161], Cora [161], NUS-WIDE-OBJECT (NW-OBJECT) [27], Reuters [7], and Animals with Attributes (AwA) [208]; five omics data sets [194], namely, cervical carcinoma (CESC), colorectal carcinoma (CRC), kidney carcinoma (KIDNEY), lower grade glioma (LGG), and lung carcinoma (LUNG); and the real-life Virus database [106]; are employed in the current work.

## A.1 HEp-2 Cell Image Databases

This section presents a brief description of the three HEp-2 cell image data sets.

1. **MIVIA:** The MIVIA or ICPR 2012 HEp-2 cell classification contest data set is comprised of indirect immunofluorescence (IIF) images. It is the outcome of a research project jointly conducted by the Mivia Lab of the University of Salerno and the University Campus Biomedico of Rome. IIF is considered a powerful, sensitive, and comprehensive test for antinuclear autoantibodies (ANA) analysis. IIF slides are examined at the fluorescence microscope and their diagnosis requires the description of staining pattern. Among the many staining patterns which can be observed, six of them are relevant to the diagnostic purposes:

    - *Homogeneous*: diffuse staining of the interphase nuclei and staining of the chromatin of mitotic cells;
    - *Fine Speckled*: fine granular nuclear staining of interphase cell nuclei;
    - *Coarse Speckled*: coarse granular nuclear staining of interphase cell nuclei;
    - *Nucleolar*: large coarse speckled staining within the nucleus, less than six in number per cell;

- *Cytoplasmatic*: fine fluorescent fibres running the length of the cell; and
- *Centromere*: several discrete speckles ( 40-60) distributed throughout the interphase nuclei and characteristically found in the condensed nuclear chromatin.

Specialists take HEp-2 images with an acquisition unit consisting of the fluorescence microscope (40-fold magnification) coupled with a 50W mercury vapour lamp and with a digital camera. The camera has a CCD with squared pixel of equal side to 6.45 $\mu$m. The images have a resolution of 1388 $\times$ 1038 pixels, a color depth of 24 bits and they are stored in bitmap format. It contains 1455 cells obtained from 28 images: 721 training cell images and 734 test cell images. Specialists manually segment and annotate each cell at a workstation monitor since at the fluorescence microscope is not possible to observe one cell at a time, and report data on the observable staining pattern.

2. **SNP:** The SNP HEp-2 Cell database (SNP) was obtained at Sullivan Nicolaides Pathology laboratory, Australia. The database has five patterns; centromere, coarse speckled, fine speckled, homogeneous and nucleolar. The 18-well slide of HEp-2000 IIF assay from Immuno Concepts N.A. Ltd. with screening dilution 1:80 was used to prepare 40 specimens. Each specimen image was captured using a monochrome high dynamic range cooled microscopy camera, which was fitted on a microscope with a plan-Apochromat 20$\times$/0.8 objective lenses and an LED illumination source. DAPI image channel was used to automatically extract the cell image masks. There are 1,884 cell images extracted from 40 specimen images. The specimen images are divided into training and testing sets with 20 images each (4 images for each pattern). In total there are 869 and 937 cell images extracted for training and testing.

3. **ICPR:** The data set has been collected at Sullivan Nicolaides Pathology laboratory, Australia. It utilizes 419 patient positive sera, which were prepared on the 18-well slide of HEp-2 IIF assay from Immuno Concepts N.A. Ltd. with screening dilution 1:80. The specimens were then automatically photographed using a monochrome high dynamic range cooled microscopy camera which was fitted on a microscope with a plan-Apochromat 20$\times$/0.8 objective lens and an LED illumination source. Approximately 100-200 cell images were extracted from each patient serum. In total there were 68,429 cell images extracted; 13,596 images used for training, made publicly available, and 54,833 for testing, privately maintained by the organizers. The set of 13,596 HEp-2 cell images is partitioned into 6797 training and 6799 test cell images. The cell images belong to the following six staining patterns, which are given by

   (a) *Homogeneous*: uniform diffuse fluorescence covering the entire nucleoplasm sometimes accentuated in the nuclear periphery;

   (b) *Speckled*: these patterns have two sub-categories;

      - Coarse Speckled: densely distributed, variously sized speckles, generally associated with larger speckles, throughout nucleoplasm of interphase cells; nucleoli are negative;
      - Fine Speckled: fine speckled staining in a uniform distribution, sometimes very dense so that an almost homogeneous pattern is attained; nucleoli may be positive or negative;

(c) *Nucleolar*: brightly clustered large granules corresponding to decoration of the fibrillar centers of the nucleoli as well as the coiled bodies;

(d) *Centromere*: rather uniform discrete speckles located throughout the entire nucleus;

(e) *Golgi*: staining of a polar organelle adjacent to and partially surrounding the nucleus, composed of irregular large granules; nuclei and nucleoli are negative;

(f) *Nuclear Membrane*: smooth homogeneous ring-like fluorescence of the nuclear membrane in interphase cells

## A.2 Benchmark Databases

A detailed description of the benchmark databases is presented in this section.

### A.2.1 Digits

The data has been downloaded from https://archive.ics.uci.edu/ml/datasets/Multiple+ Features. The set consists of features of handwritten numerals ('0'-'9') extracted from a collection of Dutch utility maps. For each class, 200 patterns have been digitized in binary images which make a total of 2000 images in the database. These images of the digits are represented in terms of the following six feature sets:

1. mfeat-pix: 240 pixel averages in 2 x 3 windows.
2. mfeat-fou: 76 Fourier coefficients of the character shapes.
3. mfeat-fac: 216 profile correlations.
4. mfeat-zer: 47 Zernike moments.
5. mfeat-kar: 64 Karhunen-Love coefficients.
6. mfeat-mor: 6 morphological features.

It is to be mentioned here that for this database, the view mfeat-mor, consisting of 6 features, is not taken into consideration for the CCA based methods, since it does not met the minimum feature criteria with respect to the output number of features for the computation of canonical variables. Similar arguments hold for discriminant analysis based methods. However, deep learning based methods do not impose such conditions on input dimension of the modalities. Thus, in case of deep learning based methods, all six modalities are considered to extract features from the set.

### A.2.2 Caltech

This data has been downloaded from http://www.vision.caltech.edu/Image_Datasets/ Caltech101. Caltech-101 consists of pictures of objects belonging to 101 categories. There are 40 to 800 images per category. Most categories have about 50 images. Collected in September 2003 by Fei-Fei Li, Marco Andreetto, and Marc Aurelio Ranzato. The size of each image is roughly 300 x 200 pixels. The 2386 images of the database are represented in terms of the following six feature sets:

1. 48 Gabor features
2. 40 Wavelet moments
3. 254 CENTRIST features
4. 1984 HOG features
5. 512 GIST features
6. 928 local binary patterns

Caltech-20 is a subset of Caltech-101, which contains only 20 classes. In the current research work, Caltech-20 is used to analyze the performance of the proposed model as well as existing approaches.

### A.2.3 CiteSeer

The CiteSeer database is obtained from http://networkrepository.com. The set is generated by sampling scientific documents from CiteSeer digital library. The publications are classified into one of the six classes, namely, Agents, AI, DB, IR, ML, and HCI. There are 3312 papers in the data set. The papers are selected in a way such that in the final set every paper cites or is cited by atleast one other paper. After stemming and removing the stopwords, a vocabulary of size 3703 unique words is obtained. All the words with document frequency less than 10 are removed. Each publication in the database is described by a 0 or 1 valued word vector indicating the absence or presence of the corresponding word in the document.

### A.2.4 Cora

This is a standard benchmark dataset of research articles, downloaded from http:// networkrepository.com. The Cora data set consists of machine learning papers. These papers are classified into one of the seven topics, which include Case_Based, Genetic_Algorithms, Neural_Networks, Probabilistic_Methods, Reinforcement_Learning, Rule_Learning, and Theory. The articles are selected in a way such that in the final set every paper cites or is cited by atleast one other paper. There are 2708 papers in the data set. After stemming and removing the stopwords, a vocabulary of size 1433 unique words is obtained. All the words with document frequency less than 10 are removed. Each article in the database is described by a 0 or 1 valued word vector indicating the absence or presence of the corresponding word in the document.

### A.2.5 NW-OBJECT

The NW-OBJECT data set, which is formally referred to as NUS-WIDE-OBJECT, is a subset of NUS-WIDE database and has been downloaded from https://lms.comp.nus.edu.sg/wp-content/uploads/2019/research/nuswide/NUS-WIDE.html. This database, created by Lab for Media Search in National University of Singapore, is intended for object recognition task. It consists of 30000 images, categorized into 31 different classes. The 30000 images of the database are represented in terms of the following five feature sets:

1. 64 color histogram features
2. 144 color correlogram features
3. 75 edge direction histogram features
4. 128 texture wavelet features
5. 225 block-wise color moment features

### A.2.6 Reuters

This multilingual data has been obtain from http://archive.ics.uci.edu/ml/machine-learning-databases/00259. The collection contains feature characteristics of documents originally written in English language and the corresponding translations in French, German, Spanish, and Italian languages over a common set of 6 categories. This collection can be used for multilingual categorization and multiview learning research. Documents have been translated, preprocessed, and are made available as feature characteristics in a "bag of words" format. 18758 documents are partitioned into 6 categories which include CCAT, C15, ECAT, E21, GCAT and M11, and represented in terms of the following five feature sets:

1. EN-EN : Original English documents with vocabulary size 21531
2. FR-EN : French documents translated to English with vocabulary size 24893
3. GR-EN : German documents translated to English with vocabulary size 34279
4. IT-EN : Italian documents translated to English with vocabulary size 15506
5. SP-EN : Spanish documents translated to English with vocabulary size 11547

### A.2.7 AwA

This image based dataset has been obtained from https://cvml.ista.ac.at/AwA. The image data was collected from public sources, such as Flickr, in 2016. It provides a platform to benchmark transfer-learning algorithms, in particular attribute base classification. It consists of 30475 images of 50 animals classes with six pre-extracted feature representations for each image. The animals classes are aligned with Osherson's classical class/attribute matrix, thereby providing 85 numeric attribute values for each class. Using the shared attributes, it is possible to transfer information between different classes. The six pre-trained feature sets are given by

1. 2000 color histogram features
2. 2000 local self-similarity features
3. 252 pyramidHOG features
4. 2000 SIFT features
5. 2000 colorSIFT features
6. 2000 SURF features.

## A.3 Omics Data Sets

This section presents a brief description of the five multimodal omics data sets of The Cancer Genome Atlas (TCGA) [194] used in this work. All the data sets have been downloaded from the Genomic Data Commons Data Portal (http://cancergenome.nih.gov).

### A.3.1 Cervical Carcinoma (CESC)

This cancer accounts for 528,000 new cases and 266,000 deaths worldwide each year, more than any other gynecological tumor [50]. By comprehensive integrated analysis, TCGA research network has identified three subtypes in CESC [143]. The CESC data set consists of 104 samples: 32 samples of keratin-low squamous subgroup, 46 samples of keratin-high squamous subgroup, and 26 samples of adenocarcinoma-rich subgroup.

### A.3.2 Colorectal Carcinoma (CRC)

It is the third most commonly diagnosed cancer in both men and women and account for nine percent of all cancer deaths [130]. The colon and rectum are parts of the digestive system and cancer forms in the colon and/or the rectum. There are 261 samples in the CRC data set. Depending on the site of origin, the samples of CRC are divided into two subtypes, namely, colon carcinoma and rectum carcinoma, having 192 and 69 samples, respectively.

### A.3.3 Kidney Carcinoma (KIDNEY)

The kidney cancer data set has two histological subtypes, namely, renal clear cell carcinoma (KIRC) and renal papillary cell carcinoma (KIRP). These subtypes were included in the 2004 World Health Organization (WHO) classification of adult renal tumors [151]. The KIDNEY data consists of 305 samples of kidney cancer with 95 samples of KIRC and 210 samples of KIRP.

### A.3.4 Lower Grade Glioma (LGG)

According to World Health Organization, lower-grade glioma is grades II and III, which is made up of diffuse low-grade and intermediate-grade gliomas. As LGG has highly variable clinical behavior, it is very difficult to predict LGG based on histologic class [**?**]. Some are indolent; others quickly progress to glioblastoma. In the current research work, 374 LGG samples are used to analyze the performance of each algorithm. The first subtype exhibits IDH mutation with no 1p/19q codeletion and has 180 samples. The second subtype has 129 samples that exhibit both IDH mutation and 1p/19q codeletion. The wild-type IDH subtype is the third subtype, which has 65 samples.

### A.3.5 Lung Carcinoma (LUNG)

There are two subtypes, namely, lung squamous cell carcinoma (LUSC) and lung adenocarcinoma (LUAD) are present in the current research work, based on the same primary site of origin. According to the 2015 WHO lung cancer classification [**?**], these had been

the two major subtypes. The total number of samples in LUSC and LUAD are 234 and 312, respectively.

These subtypes are clinically relevant and provide a roadmap for patient stratification and trials of targeted therapies. While CESC and CRC databases have four modalities, namely, DNA methylation (mDNA), protein expression (Protein), microRNA sequence (microRNA), and gene expression (RNA), KIDNEY, LGG, and LUNG data sets have five modalities, which are mDNA, Protein, microRNA, RNA, and copy number segmentation (CNS).

**Data Platforms and Preprocessing**: For the DNA methylation modality, methylation $\beta$-values from Illumina Human Methylation 450 platform is used. The Human Methylation 450 gives methylation $\beta$- values of approximately 450,000 CpG sites. Additionally, CpG locations with missing gene information were filtered out from the study. The top 2,000 most variable CpG sites are used in the current research problem. For the protein modality of all the data sets, reverse phase protein array data from the MDA_RPPA_Core platform is used. The number of proteins is different for each sample. Only a set of common proteins which is present in all the samples is considered to construct the protein expression data set. The miRNA sequence data is log transformed. The expression values of this modality are not available for most of the samples in the data sets. To avoid considering features with too many missing values, for all the omic modalities, those features for which the corresponding omic expression value is not present for more than 5% of the total number of samples are excluded. For the remaining features, missing values are replaced using 0. For the RNA modality of all the data sets, RNA-sequence data from the Illumina HiSeq platform is used which contains normalized RPKM (reads per kilobase of exon per million) counts for 20,531 genes. The data is then log transformed and 2,000 most variable genes based on their expression profile across the samples are considered. For KIDNEY, LGG, and LUNG data sets, CNS data from Affymetrix SNP Array 6.0 platform is used. The raw copy number segmented data is processed using the CNregions function of iCluster+ [134] R-package to reduce the redundant copy number regions. The CNregions function has an epsilon parameter which denotes the maximum Euclidean distance between adjacent probes tolerated for defining a non-redundant region. The number of non-redundant copy number regions extracted for a data set depends on the value of the epsilon parameter and is proportional to the number of samples in the data set. It is recommended in [134] to choose a value of epsilon such that the reduced dimension is less than 10,000. The default value of 0.005 is considered for the epsilon parameter of the CNregions function for the data sets.

## A.4 Virus Cell Image Data Set

The samples of the real-life Virus database are imaged using negative stain transmission electron microscope (TEM). The set includes 15 different virus classes, each of which is represented by 100 TEM images. Different viruses exhibit different structural properties. However, the diameter or cross-section remains almost constant within a particular virus class. Usually, the diameter varies from 25nm to 270nm depending on the morphology of the viruses. The virus particles, which are presented in the data set, include Dengue,

Cowpox, Ebola, Influenza, Adenovirus, Astrovirus, Rotavirus, Norovirus, Crimean-Congo Haemorrhagic Fever, Lassa, Marburg, Orf, Papilloma, Rift Valley, and WestNile.

# Appendix B

# Basics of Rough Sets

The appendix presents a brief description of the theory of rough sets. An approximation space or information system is a pair $< \mathbb{U}, \mathbb{A} >$ [149], where $\mathbb{U} = \{x_1, \cdots, x_i, \cdots, x_n\}$ be a non-empty set, the universe of discourse, and $\mathbb{A}$ is a family of attributes, also called knowledge in the universe. $V$ is the value domain of $\mathbb{A}$ and $\hat{f}$ is an information function $\hat{f} : \mathbb{U} \times \mathbb{A} \rightarrow V$. Any subset $\mathbb{P}$ of knowledge $\mathbb{A}$ defines an equivalence or indiscernibility relation $IND(\mathbb{P})$ on $\mathbb{U}$

$$IND(\mathbb{P}) = \{(x_i, x_j) \in \mathbb{U} \times \mathbb{U} | \forall a \in \mathbb{P}, \hat{f}(x_i, a) = \hat{f}(x_j, a)\}.$$

If $(x_i, x_j) \in IND(\mathbb{P})$, then $x_i$ and $x_j$ are indiscernible by attributes from $\mathbb{P}$. The partition of $\mathbb{U}$ generated by $IND(\mathbb{P})$ is denoted as

$$\mathbb{U}/IND(\mathbb{P}) \equiv \mathbb{U}/\mathbb{P} = \{[x_i]_{\mathbb{P}} : x_i \in \mathbb{U}\}; \tag{B.1}$$

where $[x_i]_{\mathbb{P}}$ is the equivalence class containing $x_i$. The elements in $[x_i]_{\mathbb{P}}$ are indiscernible or equivalent with respect to knowledge $\mathbb{P}$. Equivalence classes, also termed as information granules, are used to characterize arbitrary subsets of $\mathbb{U}$. The equivalence classes of $IND(\mathbb{P})$ and the empty set $\varnothing$ are the elementary sets in the approximation space $< \mathbb{U}, \mathbb{A} >$.

Given an arbitrary set $X \subseteq \mathbb{U}$, in general, it may not be possible to describe $X$ precisely in $< \mathbb{U}, \mathbb{A} >$. One may characterize $X$ by a pair of lower and upper approximations, defined as follows [149]:

$$\underline{\mathbb{P}}(X) = \bigcup \{[x_i]_{\mathbb{P}} \mid [x_i]_{\mathbb{P}} \subseteq X\} \text{ and} \tag{B.2}$$

$$\overline{\mathbb{P}}(X) = \bigcup \{[x_i]_{\mathbb{P}} \mid [x_i]_{\mathbb{P}} \cap X \neq \varnothing\}. \tag{B.3}$$

Hence, the lower approximation $\underline{\mathbb{P}}(X)$ is the union of all the elementary sets which are subsets of $X$, and the upper approximation $\overline{\mathbb{P}}(X)$ is the union of all the elementary sets which have a non-empty intersection with $X$. The tuple $< \underline{\mathbb{P}}(X), \overline{\mathbb{P}}(X) >$ is the representation of an ordinary set $X$ in the approximation space $< \mathbb{U}, \mathbb{A} >$ or simply called the rough set of $X$. The lower (respectively, upper) approximation $\underline{\mathbb{P}}(X)$ (respectively, $\overline{\mathbb{P}}(X)$) is interpreted as the collection of those elements of $\mathbb{U}$ that definitely (respectively, possi-

bly) belong to $X$. A set $X$ is said to be definable or exact in $< \mathbb{U}, \mathbb{A} >$ iff $\underline{\mathbb{P}}(X) = \overline{\mathbb{P}}(X)$. Otherwise $X$ is indefinable and termed as a rough set. $B_{\mathbb{P}}(X) = \overline{\mathbb{P}}(X) \backslash \underline{\mathbb{P}}(X)$ is called a boundary set.

An information system $< \mathbb{U}, \mathbb{A} >$ is called a decision table if the set $\mathbb{A} = \mathbb{C} \cup \mathbb{D}$, where $\mathbb{C}$ and $\mathbb{D}$ are condition and decision attribute sets, respectively. The dependency between $\mathbb{C}$ and $\mathbb{D}$ can be defined as [149]

$$\gamma_{\mathbb{C}}(\mathbb{D}) = \frac{|POS_{\mathbb{C}}(\mathbb{D})|}{|\mathbb{U}|} \tag{B.4}$$

where $POS_{\mathbb{C}}(\mathbb{D}) = \cup \underline{\mathbb{C}} X_i$ is called positive region of $\mathbb{D}$ with respect to $\mathbb{C}$, $X_i$ is the $i$-th equivalence class induced by $\mathbb{D}$ and $|\cdot|$ denotes the cardinality of a set.

# Appendix C

# Basics of Support Vector Machine

The appendix presents a brief description of the theory of support vector machine which has been considered for classification of the proposed method in Chapter 3 as well as numerous existing approaches. The support vector machine (SVM) [197] is developed based on supervised statistical learning theory, which is considered in the proposed method for classification purpose. Given the feature representation corresponding to the training data, the SVM attempts to obtain a hyperplane or decision boundary in the specified feature space that has the largest distance to the nearest training data points, since in general it is assumed that the larger the margin the lower the generalization error of the classifier. In other words, the SVM maximizes the difference between the hyperplane and support vectors as well as minimizes the classification error between the given classes.

Let us consider, the training data points are denoted as $x_j \in \Re^m,\ \forall j$ and $y_j \in \{+1, -1\}$ accounts for the class to which the sample $x_j$ belongs, considering a two-class problem. The key idea of SVM is determine $w \in \Re^m$ and $b \in \Re^m$ such that the prediction given by $\text{sign}(w^T\phi(x) + b)$ is correct for most of the training data points, where $T$ signifies the transpose operator. So, the objective of SVM classifier is to minimize the following function:

$$\min_{w,b,\zeta} \frac{1}{2} w^T w + \mathcal{C} \sum_{j=1}^{N} \zeta_j \quad \text{subject to} \quad y_j(w^T\phi(x_j) + b) \geqslant 1 - \zeta_j \text{ and } \zeta_j \geqslant 0,\ \forall j. \tag{C.1}$$

Intuitively, weâĂŹre trying to maximize the margin by minimizing $\|w\|^2 = w^T w$, while incurring a penalty when a data point is misclassified or within the margin boundary. Ideally, the value $y_j(w^T\phi(x_j) + b)$ would be $\geqslant 1$ for all the given data points, which indicates a perfect prediction. However, in real-life problems, the input data points are not always perfectly separable with a hyperplane, so some of the points are allowed to be at a distance from the corresponding correct margin boundary. The penalty term $\mathcal{C}$ accounts for the trade-off between the margin width and rate of misclassification. In case of SVM with linear kernels, $\phi(\cdot)$ corresponds to the identity function and the classifier considers one-vs-rest strategy for multi-class classification.

# List of Related Publications

## International Journal Papers

J1. **Debamita Kumar** and Pradipta Maji. Discriminative Deep Generalized Dependency Analysis for Multi-View Data. ***IEEE Transactions on Artificial Intelligence***, pages 1-12, 2023. DOI:10.1109/TAI.2023.3306739.

J2. **Debamita Kumar** and Pradipta Maji. Discriminative Deep Canonical Correlation Analysis for Multi-View Data. ***IEEE Transactions on Neural Networks and Learning Systems***, pages 1-13, 2023. DOI:10.1109/TNNLS.2023.3277633.

J3. **Debamita Kumar** and Pradipta Maji. Rough-Bayesian Approach to Select Class-Pair Specific Descriptors for HEp-2 Cell Staining Pattern Recognition. ***Pattern Recognition***, 117:107982, 2021.

J4. **Debamita Kumar** and Pradipta Maji. Selection of Relevant Texture Descriptors for Recognition of HEp-2 Cell Staining Patterns. ***International Journal of Machine Learning and Cybernetics***, 11(9):2127–2147, 2020.

# References


[1] A. H. Abdulnabi, B. Shuai, Z. Zuo, L.-P. Chau, and G. Wang. Multimodal Recurrent Neural Networks With Information Transfer Layers for Indoor Scene Labeling. *IEEE Transactions on Multimedia*, 20(7):1656–1671, 2018.

[2] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski. A Learning Algorithm for Boltzmann Machines. *Cognitive Science*, 9(1):147–169, 1985.

[3] A. S. Al-Waisy, R. Qahwaji, S. S. Ipson, and S. Al-Fahdawi. A Multimodal Deep Learning Framework Using Local Feature Representations For Face Recognition. *Machince Vision and Applications*, 29:35–54, 2018.

[4] I. Aleksander and H. Morton. The Logic of Neural Cognition. In *Advanced Neural Computers*, pages 97–102. Elsevier, 1990.

[5] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy. Deep Variational Information Bottleneck. In *Proceedings of the International Conference on Learning Representations*, 2017.

[6] M. R. Amer, T. Shields, B. Siddiquie, A. Tamrakar, A. Divakaran, and S. Chai. Deep Multimodal Fusion: A Hybrid Approach. *International Journal of Computer Vision*, 126(2-4):440–456, 2018.

[7] M. R. Amini, N. Usunier, and C. Goutte. Learning from Multiple Partially Observed Views - an Application to Multilingual Text Categorization. *Advances in Neural Information Processing Systems*, 22:28–36, 2009.

[8] J. R. Anderson and C. Peterson. A Mean Field Theory Learning Algorithm for Neural Networks. *Complex Systems*, 1:995–1019, 1987.

[9] G. Andrew, R. Arora, J. Bilmes, and K. Livescu. Deep Canonical Correlation Analysis. In *Proceedings of International Conference on Machine Learning*, pages 1247–1255, 2013.

[10] F. R. Bach and M. I. Jordan. Kernel Independent Component Analysis. *Journal of Machine Learning Research*, 3(July):1–48, 2002.

[11] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan. Multiple Kernel Learning, Conic Duality, and the SMO Algorithm. In *Proceedings of the Twenty-First International Conference on Machine Learning*, page 6, 2004.

[12] M. Behmanesh, P. Adibi, J. Chanussot, C. Jutten, and S. M. S. Ehsani. Geometric Multimodal Learning Based on Local Signal Expansion for Joint Diagonalization. *IEEE Transactions on Signal Processing*, 69:1271–1286, 2021.

[13] M. Belgiu and L. Drăguţ. Random Forest in Remote Sensing: A Review of Applications and Future Directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, 11:24–31, 2016.

[14] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[15] M. Belkin and P. Niyogi. Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering. *Advances in Neural Information Processing Systems*, 14, 2001.

[16] M. Belkin and P. Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 15(6):1373–1396, 2003.

[17] S. K. Berberian. *Introduction to Hilbert Space*, volume 287. American Mathematical Society, 1999.

[18] C. M. Bishop and N. M. Nasrabadi. *Pattern Recognition and Machine Learning*, volume 4. Springer, 2006.

[19] A. Blum and T. Mitchell. Combining Labeled and Unlabeled Data With Co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pages 92–100, 1998.

[20] A. Blum and T. Mitchell. Combining Labeled and Unlabeled Data with Co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pages 92–100, 1998.

[21] D. Cascio, V. Taormina, and G. Raso. Deep CNN for IIF Images Classification in Autoimmune Diagnostics. *Applied Sciences*, 9(8):1618, 2019.

[22] S. Di. Cataldo, A. Bottino, I. Islam, T. F. Vieira, and E. Ficarra. Subclass Discriminant Analysis of Morphological and Textural Features for HEp-2 Staining Pattern Classification. *Pattern Recognition*, 47(7):2389–2399, 2014.

[23] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan. Multi-view Clustering Via Canonical Correlation Analysis. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 129–136, 2009.

[24] D. Chen, J. Lv, and Z. Yi. Graph Regularized Restricted Boltzmann Machine. *IEEE Transactions on Neural Networks and Learning Systems*, 29(6):2651–2659, 2017.

[25] J. Chen, G. Wang, and G. B. Giannakis. Graph Multiview Canonical Correlation Analysis. *IEEE Transactions on Signal Processing*, 67(11):2826–2838, 2019.

[26] N. Chen, J. Zhu, and E. Xing. Predictive Subspace Learning for Multi-view Data: A Large Margin Approach. In *Advances in Neural Information Processing Systems*, volume 23, 2010.

[27] 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 *Proceedings of the ACM Conference on Image and Video Retrieval*, pages 1–9, 2009.

[28] R. R. Coifman and S. Lafon. Diffusion Maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006.

[29] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker. Geometric Diffusions as a Tool for Harmonic Analysis and Structure Definition of Data: Diffusion Maps. *Proceedings of the National Academy of Sciences*, 102(21):7426–7431, 2005.

[30] H. D. Couture, R. Kwitt, J. S. Marron, M. Troester, C. M. Perou, and M. Niethammer. Deep Multi-View Learning via Task-Optimal CCA. *arXiv preprint arXiv:1907.07739*, 2019.

[31] T. M. Cover and J. A. Thomas. Elements of Information Theory. *New York: John Wiley & Sons*, 68:69–73, 1991.

[32] K. Cukier. Data, data everywhere, 2010.

[33] S. Dasgupta, M. Littman, and D. McAllester. PAC Generalization Bounds for Co-training. *Advances in Neural Information Processing Systems*, 14, 2001.

[34] D. L. Davies and D. W. Bouldin. A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1:224–227, 1979.

[35] T. Diethe, D. R. Hardoon, and J. Shawe-Taylor. Multiview Fisher Discriminant Analysis. In *Proceedings of the Neural Information Processing Systems Workshop on Learning from Multiple Sources*, 2008.

[36] B. Dolhansky and C. C. Ferrer. Eye In-Painting With Exemplar Generative Adversarial Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7902–7911, 2018.

[37] M. Dorfer and G. Widmer. Towards Deep and Discriminative Canonical Correlation Analysis. In *Proceedings of the ICML Workshop on Multi-View Representaiton Learning*, pages 1–5, 2016.

[38] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*, volume 3. Wiley New York, 1973.

[39] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley, 1973.

[40] J. C. Dunn. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact, Well-Separated Clusters. *Journal of Cybernetics*, 3:32–57, 1974.

[41] D. Eynard, K. Glashoff, M. M. Bronstein, and A. M. Bronstein. Multimodal Diffusion Geometry by Joint Diagonalization of Laplacians. *arXiv preprint arXiv:1209.2295*, 2012.

[42] D. Eynard, A. Kovnatsky, M. M. Bronstein, K. Glashoff, and A. M. Bronstein. Multimodal Manifold Analysis by Simultaneous Diagonalization of Laplacians. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(12):2505–2517, 2015.

[43] S. Fan, X. Wang, C. Shi, E. Lu, K. Lin, and B. Wang. One2Multi Graph Autoencoder for Multi-View Graph Clustering. In *Proceedings of the Web Conference 2020*, pages 3070–3076, 2020.

[44] W. Fan, Y. Ma, H. Xu, X. Liu, J. Wang, Q. Li, and J. Tang. Deep Adversarial Canonical Correlation Analysis. In *Proceedings of the International Conference on Data Mining*, pages 352–360, 2020.

[45] M. Faraki, M. T. Harandi, A. Wiliem, and B. C. Lovell. Fisher Tensors for Classifying Human Epithelial Cells. *Pattern Recognition*, 47(7):2348–2359, 2014.

[46] A. Fathi and A. R. Naghsh-Nilchi. Noise Tolerant Local Binary Pattern Operator for Efficient Texture Analysis. *Pattern Recognition Letters*, 33(9):1093 – 1100, 2012.

[47] D. J. Felleman and D. C. Van Essen. Distributed Hierarchical Processing in the Primate Cerebral Cortex. *Cerebral Cortex*, 1(1):1–47, 1991.

[48] C.-M. Feng, Y. Xu, J.-X. Liu, Y.-L. Gao, and C.-H. Zheng. Supervised Discriminative Sparse PCA for Com-Characteristic Gene Selection and Tumor Classification on Multiview Biological Data. *IEEE Transactions on Neural Networks and Learning Systems*, 30(10):2926–2937, 2019.

[49] F. Feng, X. Wang, and R. Li. Cross-Modal Retrieval With Correspondence Autoencoder. In *Proceedings of the ACM International Conference on Multimedia*, pages 7–16, 2014.

[50] J. Ferlay, I. Soerjomataram, R. Dikshit, S. Eser, C. Mathers, M. Rebelo, D. M. Parkin, D. Forman, and F. Bray. Cancer Incidence and Mortality Worldwide: Sources, Methods and Major Patterns. *International Journal of Cancer*, 136(5):E359–E386, 2015.

[51] P. Foggia, G. Percannella, P. Soda, and M. Vento. Benchmarking HEp-2 Cells Classification Methods. *IEEE Transactions on Medical Imaging*, 32(10):1878–1889, 2013.

[52] Y. Freund and R. E. Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[53] X. Fu, K. Huang, E. E. Papalexakis, H. Song, P. P. Talukdar, N. D. Sidiropoulos, C. Faloutsos, and T. Mitchell. Efficient and Distributed Algorithms for Large-Scale Generalized Canonical Correlations Analysis. In *Proceedings of the IEEE 16th International Conference on Data Mining*, pages 871–876, 2016.

[54] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12746–12756, 2020.

[55] Z. Gao, L. Wang, L. Zhou, and J. Zhang. HEp-2 Cell Image Classification with Deep Convolutional Neural Networks. *IEEE Journal of Biomedical and Health Informatics*, 21(2):416–428, 2016.

[56] A. Globerson and T. Jaakkola. Approximate Inference Using Conditional Entropy Decompositions. In *Artificial Intelligence and Statistics*, pages 131–138. PMLR, 2007.

[57] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[58] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 27, 2014.

[59] A. Gretton, R. Herbrich, A. Smola, O. Bousquet, and B. Schölkopf. Kernel Methods for Measuring Independence. *Journal of Machine Learning Research*, 6:2075–2129, 2005.

[60] Yimo Guo, Guoying Zhao, and Matti PietikÃď inen. Discriminative Features for Texture Description. *Pattern Recognition*, 45(10):3834–3843, 2012.

[61] Z. Guo, L. Zhang, and D. Zhang. A Completed Modeling of Local Binary Pattern Operator for Texture Classification. *IEEE Transactions on Image Processing*, 19(6):1657–1663, 2010.

[62] M. Gutmann and A. Hyvärinen. Noise-Contrastive Estimation: A New Estimation Principle for Unnormalized Statistical Models. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

[63] M. GÃűnen. Bayesian Supervised Dimensionality Reduction. *IEEE Transactions on Cybernetics*, 43(6):2179–2189, 2013.

[64] W. Hamilton, P. Bajaj, M. Zitnik, D. Jurafsky, and J. Leskovec. Embedding Logical Queries on Knowledge Graphs. *Advances in Neural Information Processing Systems*, 31, 2018.

[65] 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 Transactions on Cybernetics*, 48(11):3171–3183, 2018.

[66] J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011. ISBN: 978-0123814791.

[67] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical Correlation Analysis: An Overview With Application to Learning Methods. *Neural Computation*, 16(12):2639–2664, 2004.

[68] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang. Face Recognition Using Laplacianfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):328–340, 2005.

[69] G. E. Hinton. Training Products of Experts by Minimizing Contrastive Divergence. *Neural Computation*, 14(8):1771–1800, 2002.

[70] G. E. Hinton. Boltzmann Machine. *Scholarpedia*, 2(5):1668, 2007.

[71] G. E. Hinton, S. Osindero, and Y-W. Teh. A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, 18(7):1527–1554, 2006.

[72] G. E. Hinton and R. R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786):504–507, 2006.

[73] G. E. Hinton and T. J. Sejnowski. Optimal Perceptual Inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 448. Citeseer, 1983.

[74] G. E. Hinton and T. J. Sejnowski. Learning and Relearning in Boltzmann Machines. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, 1(282-317):2, 1986.

[75] P. Hobson, B. C. Lovell, G. Percannella, A. Saggese, M. Vento, and A. Wiliem. HEp-2 Staining Pattern Recognition at Cell and Specimen Levels: Datasets, Algorithms and Results. *Pattern Recognition Letters*, 82:12–22, 2016.

[76] C. Hong, J. Yu, J. Wan, D. Tao, and M. Wang. Multimodal Deep Autoencoder for Human Pose Recovery. *IEEE Transactions on Image Processing*, 24(12):5659–5670, 2015.

[77] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 2012.

[78] H. Hotelling. Relations Between Two Sets of Variates. *Biometrika*, 28(3/4):321–377, 1936.

[79] P. Hu, D. Peng, Y. Sang, and Y. Xiang. Multi-View Linear Discriminant Analysis Network. *IEEE Transactions on Image Processing*, 28(11):5352–5365, 2019.

[80] Y. Hua, J. Guo, and H. Zhao. Deep Belief Networks and Deep Learning. In *Proceedings of the International Conference on Intelligent Computing and Internet of Things*, pages 1–4. IEEE, 2015.

[81] C. Huang, H. Ai, Y. Li, and S. Lao. High-Performance Rotation Invariant Multiview Face Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):671–686, 2007.

[82] W. Huang, T. Zhang, Y. Rong, and J. Huang. Adaptive Sampling Towards Fast Graph Representation Learning. *Advances in Neural Information Processing Systems*, 31, 2018.

[83] Z. Huang and L. Van Gool. A Riemannian Network for SPD Matrix Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, pages 2036–2042, 2017.

[84] Z. Huang, J. T. Zhou, H. Zhu, C. Zhang, J. Lv, and X. Peng. Deep Spectral Representation Learning From Multi-View Data. *IEEE Transactions on Image Processing*, 30:5352–5362, 2021.

[85] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-Image Translation With Conditional Adversarial Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1125–1134, 2017.

[86] A. K. Jain, R. P. W. Duin, and J. Mao. Statistical Pattern Recognition: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37, 2000.

[87] J.-H. Jang, J. Choi, H. W. Roh, S. J. Son, C. H. Hong, E. Y. Kim, T. Y. Kim, and D. Yoon. Deep Learning Approach for Imputation of Missing Values in Actigraphy Data: Algorithm Development Study. *JMIR mHealth and uHealth*, 8(7):e16113, 2020.

[88] X. Jia, L. Shen, X. Zhou, and S. Yu. Deep Convolutional Neural Network Based HEp-2 Cell Classification. In *Proceedings of the 23rd International Conference on Pattern Recognition (ICPR)*, pages 77–80. IEEE, 2016.

[89] S. Jiang, J. Hu, C. L. Magee, and J. Luo. Deep Learning for Technical Document Classification. *IEEE Transactions on Engineering Management*, 2022.

[90] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant. *Applied Logistic Regression*, volume 398. John Wiley & Sons, 2013.

[91] S. M. Kakade and D. P. Foster. Multi-view Regression Via Canonical Correlation Analysis. In *Learning Theory*, pages 82–96, 2007.

[92] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen. Multi-View Discriminant Analysis. In *Proceedings of the European Conference on Computer Vision*, pages 808–821, 2012.

[93] M. Kan, S. Shan, H. Zhang, S. Lao, and X. Chen. Multi-View Discriminant Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1):188–194, 2016.

[94] Y. Karklin and M. S. Lewicki. Learning Higher-Order Structures in Natural Images. *Network: Computation in Neural Systems*, 14(3):483–499, 2003.

[95] A. Karpathy and L. Fei-Fei. Deep Visual-Semantic Alignments for Generating Image Descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015.

[96] J. R. Kettenring. Canonical Analysis of Several Sets of Variables. *Biometrika*, 58(3):433–451, 1971.

[97] M. R. Khan and J. E. Blumenstock. Multi-GCN: Graph Convolutional Networks for Multi-View Networks, With Applications to Global Poverty. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[98] C. Kingsford and S. L. Salzberg. What Are Decision Trees? *Nature Biotechnology*, 26(9):1011–1013, 2008.

[99] X. Kong, K. Li, J. Cao, Q. Yang, and L. Wenyin. HEp-2 Cell Pattern Classification with Discriminative Dictionary Learning. *Pattern Recognition*, 47(7):2379–2388, 2014.

[100] A. Kumar, P. Rai, and H. Daume III. Co-regularized Spectral Clustering with Multiple Kernels. 2010.

[101] A. Kumar, P. Rai, and H. Daume III. Co-regularized Multi-view Spectral Clustering. In *Advances in Neural Information Processing Systems*, volume 24, 2011.

[102] D. Kumar and P. Maji. Selection of Relevant Texture Descriptors for Recognition of HEp-2 Cell Staining Patterns. *International Journal of Machine Learning and Cybernetics*, 11:2127–2147, 2020.

[103] D. Kumar and P. Maji. Rough-Bayesian Approach to Select Class-Pair Specific Descriptors for HEp-2 Cell Staining Pattern Recognition. *Pattern Recognition*, 117:107982–108000, 2021.

[104] D. Kumar and P. Maji. Discriminative Deep Canonical Correlation Analysis for Multi-View Data. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2023.

[105] D. Kumar and P. Maji. Discriminative Deep Generalized Dependency Analysis for Multi-View Data. *IEEE Transactions on Artificial Intelligence*, pages 1–12, 2023.

[106] G. Kylberg, M. Uppstrom, G. Borgefors, and I.-M Sintorn. Segmentation of Virus Particle Candidates in Transmission Electron Microscopy Images. *Journal of Microscopy*, 245:140–147, 2012.

[107] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the Kernel Matrix with Semidefinite Programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

[108] H. Larochelle, M. Mandel, R. Pascanu, and Y. Bengio. Learning Algorithms for the Classification Restricted Boltzmann Machine. *Journal of Machine Learning Research*, 13(Mar):643–669, 2012.

[109] Y. LeCun, Y. Bengio, and G. Hinton. Deep Learning. *Nature*, 521(7553):436–444, 2015.

[110] G. Lee, S. Doyle, J. Monaco, A. Madabhushi, M. D. Feldman, S. R. Master, and J. E. Tomaszewski. A Knowledge Representation Framework for Integration, Classification of Multi-scale Imaging and Non-imaging Data: Preliminary Results in Predicting Prostate Cancer Recurrence by Fusing Mass Spectrometry and Histology. In *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 77–80, 2009.

[111] T. S. Lee, D. Mumford, R. Romero, and V. A. F. Lamme. The Role of the Primary Visual Cortex in Higher Level Vision. *Vision Research*, 38(15-16):2429–2454, 1998.

[112] H. Lei, T. Han, F. Zhou, Z. Yu, J. Qin, A. Elazab, and B. Lei. A Deeply Supervised Residual Network for HEp-2 Cell Classification via Cross-Modal Transfer Learning. *Pattern Recognition*, 79:290–302, 2018.

[113] A. Levin and A. Shashua. Principal Component Analysis over Continuous Subspaces and Intersection of Half-Spaces. In *Proceeding of the European Conference on Computer Vision Copenhagen*, pages 635–650. Springer, 2002.

[114] F. Li, R. Fergus, and P. Perona. One-Shot Learning of Object Categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, 2006.

[115] S. Li, J. Kawale, and Y. Fu. Deep Collaborative Filtering via Marginalized Denoising Autoencoder. In *Proceedings of the International on Conference on Information and Knowledge Management*, pages 811–820, 2015.

[116] S. Z. Li, X. W. Hou, H. J. Zhang, and Q. S. Cheng. Learning Spatially Localized, Parts-Based Representation. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 207–212. IEEE, 2001.

[117] Y. Li and L. Shen. HEp-Net: A Smaller and Better Deep-learning Network for HEp-2 Cell Classification. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 7(3):266–272, 2019.

[118] Y. Li, L. Shen, and S. Yu. HEp-2 Specimen Image Segmentation and Classification Using Very Deep Fully Convolutional Network. *IEEE Transactions on Medical Imaging*, 36(7):1561–1572, 2017.

[119] Y. Li, M. Yang, and Z. Zhang. A Survey of Multi-View Representation Learning. *IEEE Transactions on Knowledge and Data Engineering*, 31(10):1863–1883, 2018.

[120] S. Liao, M. W. K. Law, and A. C. S. Chung. Dominant Local Binary Patterns for Texture Classification. *IEEE Transactions on Image Processing*, 18(5):1107–1118, 2009.

[121] L. Liu, S. Lao, P. W. Fieguth, Y. Guo, X. Wang, and M. Pietikainen. Median Robust Extended Local Binary Pattern for Texture Classification. *IEEE Transactions on Image Processing*, 25(3):1368–1381, 2016.

[122] Q. Liu, J. Xu, R. Jiang, and W. H. Wong. Density Estimation Using Deep Generative Neural Networks. *Proceedings of the National Academy of Sciences*, 118(15):e2101344118, 2021.

[123] Y. Liu, S. Zhou, and Q. Chen. Discriminative Deep Belief Networks for Visual Data Classification. *Pattern Recognition*, 44(10-11):2287–2296, 2011.

[124] D. Lopez-Paz, S. Sra, A. Smola, Z. Ghahramani, and B. Schölkopf. Randomized Nonlinear Component Analysis. In *Proceedings of International Conference on Machine Learning*, pages 1359–1367, 2014.

[125] Y. Luo, D. Tao, K. Ramamohanarao, C. Xu, and Y. Wen. Tensor Canonical Correlation Analysis for Multi-View Dimension Reduction. *IEEE Transactions on Knowledge and Data Engineering*, 27(11):3111–3124, 2015.

[126] P. Maji. A Rough Hypercuboid Approach for Feature Selection in Approximation Spaces. *IEEE Transactions on Knowledge and Data Engineering*, 26(1):16–29, 2014.

[127] P. Maji and S. Paul. Rough Set Based Maximum Relevance-Maximum Significance Criterion and Gene Selection from Microarray Data. *International Journal of Approximate Reasoning*, 52(3):408–426, 2011.

[128] S. Manivannan, W. Li, S. Akbar, R. Wang, J. Zhang, and S. J. McKenna. An Automated Pattern Recognition System for Classifying Indirect Immunofluorescence Images of HEp-2 Cells and Specimens. *Pattern Recognition*, 51:12–26, 2016.

[129] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille. Deep Captioning with Multimodal Recurrent Neural Networks (m-rnn). *ArXiv Preprint ArXiv:1412.6632*, 2014.

[130] I. Mármol, C. Sánchez de Diego, A. P. Dieste, E. Cerrada, and M. J. R. Yoldi. Colorectal Carcinoma: A General Overview and Future Perspectives in Colorectal Cancer. *International Journal of Molecular Sciences*, 18(1):197, 2017.

[131] R. Memisevic, L. Sigal, and D. J. Fleet. Shared Kernel Information Embedding for Discriminative Inference. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):778–790, 2012.

[132] X-L. Meng and W. H. Wong. Simulating Ratios of Normalizing Constants via a Simple Identity: A Theoretical Exploration. *Statistica Sinica*, pages 831–860, 1996.

[133] P. L. Meroni and P. H. Schur. ANA Screening: An Old Test with New Recommendations. *Autoimmunity Reviews*, 69(8):1420–1422, 2010.

[134] Q. Mo and R. Shen. iClusterPlus: Integrative Clustering of Multiple Genomic Data Sets. 2013.

[135] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo Samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006.

[136] J. R. Movellan. Contrastive Hebbian Learning in the Continuous Hopfield Model. In *Connectionist Models*, pages 10–17. 1991.

[137] D. Mumford. On the Computational Architecture of the Neocortex: II The Role of Cortico-Cortical Loops. *Biological Cybernetics*, 66(3):241–251, 1992.

[138] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[139] I. Muslea, S. Minton, and C. A. Knoblock. Active Learning with Multiple Views. *Journal of Artificial Intelligence Research*, 27(1):203–233, 2006.

[140] R. M. Neal. *Probabilistic Inference Using Markov Chain Monte Carlo Methods*. Department of Computer Science, University of Toronto Toronto, ON, Canada, 1993.

[141] R. M. Neal. Annealed Importance Sampling. *Statistics and Computing*, 11:125–139, 2001.

[142] 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*, pages 355–368. 1998.

[143] Cancer Genome Atlas Research Network. Integrated Genomic and Molecular Characterization of Cervical Cancer. *Nature*, 543(7645):378, 2017.

[144] K. Nigam and R. Ghani. Analyzing the Effectiveness and Applicability of Co-training. In *Proceedings of the Ninth International Conference on Information and Knowledge Management*, pages 86–93, 2000.

[145] R. Nosaka and K. Fukui. HEp-2 Cell Classification using Rotation Invariant Co-Occurrence Among Local Binary Patterns. *Pattern Recognition*, 47(7):2428–2436, 2014.

[146] R. Nosaka, Y. Ohkawa, and K. Fukui. Feature Extraction Based on Co-Occurrence of Adjacent Local Binary Patterns. In *Proceedings of the Advances in Image and Video Technology*, pages 82–91, 2012.

[147] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[148] T. Ojala, M. PietikÃďinen, and D. Harwood. A Comparative Study of Texture Measures with Classification Based on Feature Distributions. *Pattern Recognition*, 29(1):51–59, 1996.

[149] Z. Pawlak. *Rough Sets: Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, Dordrecht and Boston and London, 1991. ISBN: 0792314727.

[150] M. Peng, N. K. Gupta, and A. F. Armitage. An Investigation Into the Improvement of Local Minima of the Hopfield Network. *Neural Networks*, 9(7):1241–1253, 1996.

[151] S. R. Prasad, P. A. Humphrey, J. R. Catena, V. R. Narra, J. R. Srigley, A. D. Cortez, N. C. Dalrymple, and K. N. Chintapalli. Common and Uncommon Histologic Subtypes of Renal Cell Carcinoma: Imaging Spectrum with Pathologic Correlation. *Radiographics*, 26(6):1795–1806, 2006.

[152] D. A. Puente and I. M. Eremin. Convolutional Restricted Boltzmann Machine Aided Monte Carlo: An Application to Ising and Kitaev Models. *Physical Review B*, 102(19):195148–1–195148–12, 2020.

[153] X. Qi, R. Xiao, C. G. Li, Y. Qiao, J. Guo, and X. Tang. Pairwise Rotation Invariant Co-Occurrence Local Binary Pattern. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11):2199–2213, 2014.

[154] Y. Qian, J. Liang, W. Pedrycz, and C. Dang. Positive Approximation: An Accelerator for Attribute Reduction in Rough Set Theory. *Artificial Intelligence*, 174:597–618, 2010.

[155] N. Quadrianto and C. H. Lampert. Learning Multi-view Neighborhood Preserving Projections. In *Proceedings of the International Conference on Machine Learning*, pages 425–432, 2011.

[156] N. Rappoport and R. Shamir. Multi-Omic and Multi-View Clustering Algorithms: Review and Cancer Benchmark. *Nucleic Acids Research*, 46(20):10546–10562, 2018.

[157] S. Rastegar, M. Soleymani, H. R. Rabiee, and S. M. Shojaee. Mdl-cw: A Multimodal Deep Learning Framework With Cross Weights. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2601–2609, 2016.

[158] M. Reed and B. Simon. *Methods of Modern Mathematical Physics: Functional Analysis*. Academic Press, 1980.

[159] M. Reuter, F.-E. Wolter, and N. Peinecke. LaplaceâĂŞBeltrami Spectra as âĂŸShape-DNAâĂŹ of Surfaces and Solids. *Computer-Aided Design*, 38(4):342–366, 2006.

[160] L. F. Rodrigues, M. C. Naldi, and J. F. Mari. Comparing Convolutional Neural Networks and Preprocessing Techniques for HEp-2 Cell Classification in Immunofluorescence Images. *Computers in Biology and Medicine*, 116(103542), 2020.

[161] R. Rossi and N. Ahmed. The Network Data Repository With Interactive Graph Analytics and Visualization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[162] S. T. Roweis and L. K. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500):2323–2326, 2000.

[163] R. Salakhutdinov and G. Hinton. Deep Boltzmann Machines. In *Proceedings of the Artificial Intelligence and Statistics*, pages 448–455, 2009.

[164] R. Salakhutdinov and I. Murray. On the Quantitative Analysis of Deep Belief Networks. In *Proceedings of the International Conference on Machine Learning*, pages 872–879, 2008.

[165] M. Salzmann, C. H. Ek, R. Urtasun, and T. Darrell. Factorized Orthogonal Latent Spaces. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9, pages 701–708, 2010.

[166] A. Sano, W. Chen, D. Lopez-Martinez, S. Taylor, and R. W. Picard. Multimodal Ambulatory Sleep Detection Using LSTM Recurrent Neural Networks. *IEEE Journal of Biomedical and Health Informatics*, 23(4):1607–1617, 2019.

[167] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008.

[168] O. Schwartz, T. J. Sejnowski, and P. Dayan. Soft Mixer Assignment in a Hierarchical Generative Model of Natural Scene Statistics. *Neural Computation*, 18(11):2680–2718, 2006.

[169] S. Y. Sekeh, B. Oselio, and A. O. Hero. Learning to Bound the Multi-Class Bayes Error. *IEEE Transactions on Signal Processing*, 68:3793–3807, 2020.

[170] T. Shakunaga and K. Shigenari. Decomposed Eigenface for Face Recognition Under Various Lighting Conditions. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 864–871. IEEE, 2001.

[171] A. Sharma, A. Kumar, H. Daume, and D. W. Jacobs. Generalized Multiview Analysis: A Discriminative Latent Space. In *Proceedings of the Computer Vision and Pattern Recognition*, pages 2160–2167. IEEE, 2012.

[172] L. Shen, X. Jia, and Y. Li. Deep Cross Residual Network for HEp-2 Cell Staining Pattern Classification. *Pattern Recognition*, 82:68–78, 2018.

[173] A. Shon, K. Grochow, A. Hertzmann, and R. Rao. Learning Shared Latent Structure for Image Synthesis and Robotic Imitation. In *Advances in Neural Information Processing Systems*, volume 18, 2005.

[174] S. H. Silva and P. Najafirad. Opportunities and Challenges in Deep Learning Adversarial Robustness: A Survey. *arXiv preprint arXiv:2007.00753*, 2020.

[175] V. Sindhwani, P. Niyogi, and M. Belkin. A Co-regularization Approach to Semi-supervised Learning with Multiple Views. In *Proceedings of the Workshop on Learning with Multiple Views at 22nd International Conference on Machine Learning*, pages 1135–1142, 2005.

[176] P. Smolensky. Information Processing in Dynamical Systems: Foundations of Harmony Theory. Technical report, Colorado Univ at Boulder Dept of Computer Science, 1986.

[177] P. Soda and G. Iannello. Aggregation of Classifiers for Staining Pattern Recognition in Antinuclear Autoantibodies Analysis. *IEEE Transactions on Information Technology in Biomedicine*, 13(3):322–329, 2009.

[178] K. Somandepalli, N. Kumar, R. Travadi, and S. Narayanan. Multimodal Representation Learning Using Deep Multiset Canonical Correlation. *arXiv preprint arXiv:1904.01775*, 2019.

[179] N. Srivastava and R. Salakhutdinov. Multimodal Learning with Deep Boltzmann Machines. In *Advances in Neural Information Processing Systems*, pages 2222–2230, 2012.

[180] N. Srivastava and R. Salakhutdinov. Multimodal Learning with Deep Boltzmann Machines. *Journal of Machine Learning Research*, 15:2949–2980, 2014.

[181] R. Stoklasa, T. Majtner, and D. Svoboda. Efficient k-NN Based HEp-2 Cells Classifier. *Pattern Recognition*, 47(7):2409–2418, 2014.

[182] S. Sun, L. Mao, Z. Dong, and L. Wu. *Multiview Machine Learning*. Springer, 2019.

[183] I. Sutskever, J. Martens, and G. E. Hinton. Generating Text with Recurrent Neural Networks. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 1017–1024, 2011.

[184] A. F. Syafiandini, I. Wasito, S. Yazid, A. Fitriawan, and M. Amien. Multimodal Deep Boltzmann Machines for Feature Selection on Gene Expression Data. In *Proceedings of the International Conference on Advanced Computer Science and Information Systems*, pages 407–412, 2016.

[185] X. Tan and B. Triggs. Enhanced Local Texture Feature Sets for Face Recognition Under Difficult Lighting Conditions. *IEEE Transactions on Image Processing*, 19(6):1635–1650, 2010.

[186] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323, 2000.

[187] A. Tenenhaus and M. Tenenhaus. Regularized Generalized Canonical Correlation Analysis. *Psychometrika*, 76(2):257–284, 2011.

[188] I. Theodorakopoulos, D. Kastaniotis, G. Economou, and S. Fotopoulos. HEp-2 Cells Classification Via Sparse Representation of Textural Features Fused into Dissimilarity Space. *Pattern Recognition*, 47(7):2367–2378, 2014.

[189] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Academic Press, Inc., USA, 4th edition, 2008. ISBN: 9781597492720.

[190] T. Tieleman. Training Restricted Boltzmann Machines Using Approximations to the Likelihood Gradient. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1064–1071, 2008.

[191] P. Tiwari, S. Viswanath, G. Lee, and A. Madabhushi. Multi-modal Data Fusion Schemes for Integrated Classification of Imaging and Non-imaging Biomedical Data. In *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 165–168, 2011.

[192] M. Toğaçar, B. Ergen, and Z. Cömert. Detection of Lung Cancer on Chest CT Images Using Minimum Redundancy Maximum Relevance Feature Selection Method with Convolutional Neural Networks. *Biocybernetics and Biomedical Engineering*, 40(1):23–39, 2020.

[193] K. G. Toker and S. E. Yüksel. Deep Canonical Correlation Analysis for Hyperspectral Image Classification. In *Proceedings of the Remote Sensing of the Ocean, Sea Ice, Coastal Waters, and Large Water Regions*, volume 11150, page 1115009. International Society for Optics and Photonics, 2019.

[194] K. Tomczak, P. Czerwińska, and M. Wiznerowicz. The Cancer Genome Atlas (TCGA): An Immeasurable Source of Knowledge. *Contemporary Oncology*, 19(1A):A68, 2015.

[195] M. A. Turk and A. P. Pentland. Face Recognition Using Eigenfaces. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 586–587. IEEE Computer Society, 1991.

[196] B. Vallet and B. LÃľvy. Spectral Geometry Processing with Manifold Harmonics. *Computer Graphics Forum*, 27(2):251–260, 2008.

[197] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 1995. ISBN: 0-387-94559-8.

[198] S. Viswanath and A. Madabhushi. Consensus Embedding: Theory, Algorithms and Application to Segmentation and Classification of Biomedical Data. *BMC Bioinformatics*, 13(26), 2012.

[199] C. Vununu, S.-H. Lee, and K.-R. Kwon. A Deep Feature Extraction Method for HEp-2 Cell Image Classification. *Electronics*, 8(1):20, 2019.

[200] M. J. Wainwright, T. S. Jaakkola, and A. S. Willsky. A New Class of Upper Bounds on the Log Partition Function. *IEEE Transactions on Information Theory*, 51(7):2313–2335, 2005.

[201] M. J. Wainwright and M. I. Jordan. Log-Determinant Relaxation for Approximate Inference in Discrete Markov Random Fields. *IEEE Transactions on Signal Processing*, 54(6):2099–2109, 2006.

[202] W. Wang, R. Arora, K. Livescu, and J. Bilmes. On Deep Multi-View Representation Learning. In *Proceedings of International Conference on Machine Learning*, pages 1083–1092, 2015.

[203] M. Wardetzky. Convergence of the Cotangent Formula: An Overview. *Discrete Differential Geometry*, pages 275–286, 2008.

[204] M. Welling and G. E. Hinton. A New Learning Algorithm for Mean Field Boltzmann Machines. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 351–357, 2002.

[205] A. Wiliem, C. Sanderson, Y. Wong, P. Hobsone, R. F. Minchin, and B. C. Lovell. Automatic Classification of Human Epithelial Type 2 Cell Indirect Immunofluorescence Images Using Cell Pyramid Matching. *Pattern Recognition*, 47(7):2315–2324, 2014.

[206] A. Wiliem, Y. Wong, C. Sanderson, P. Hobson, S. Chen, and B. C. Lovell. Classification of Human Epithelial Type 2 Cell Indirect Immunofluoresence Images Via Codebook Based Descriptors. In *Proceedings of the IEEE Workshop on Applications of Computer Vision*, pages 95–102, 2013.

[207] H. Wold. Path Models with Latent Variables: The NIPALS Approach. In H. M. Blalock, A. Aganbegian, F. M. Borodkin, R. Boudon, and V. Capecchi, editors, *Quantitative Sociology*, pages 307–357. Academic Press, 1975.

[208] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata. Zero-Shot Learning âĂŤ A Comprehensive Evaluation of the Good, the Bad and the Ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(9):2251–2265, 2019.

[209] Q. Xuan, Z. Chen, Y. Liu, H. Huang, G. Bao, and D. Zhang. Multiview Generative Adversarial Network and its Application in Pearl Classification. *IEEE Transactions on Industrial Electronics*, 66(10):8244–8252, 2018.

[210] F. Xue, X. Wu, S. Cai, and J. Wang. Learning Multi-View Camera Relocalization With Graph Neural Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11372–11381, 2020.

[211] J. Xue, H. Zhang, and K. Dana. Deep Texture Manifold for Ground Terrain Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2018.

[212] X. Yan, S. Hu, Y. Mao, Y. Ye, and H. Yu. Deep Multi-View Learning Methods: A Review. *Neurocomputing*, 448:106–129, 2021.

[213] X. Yang, W. Liu, and W. Liu. Tensor Canonical Correlation Analysis Networks for Multi-View Remote Sensing Scene Recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(6):2948–2961, 2020.

[214] X.-S. Yang. *Introduction to Algorithms for Data Mining and Machine Learning*. Academic press, 2019.

[215] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Understanding Belief Propagation and Its Generalizations. *Exploring Artificial Intelligence in the New Millennium*, 8(236-239):0018–9448, 2003.

[216] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Constructing Free-Energy Approximations and Generalized Belief Propagation Algorithms. *IEEE Transactions on Information Theory*, 51(7):2282–2312, 2005.

[217] 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 Recognition*, 92:37–51, 2019.

[218] L. Younes. On the Convergence of Markovian Stochastic Algorithms with Rapidly Decreasing Ergodicity Rates. *Stochastics: An International Journal of Probability and Stochastic Processes*, 65(3-4):177–228, 1999.

[219] S. Yu, B. Krishnapuram, R. Rosales, and R. B. Rao. Bayesian Co-training. *The Journal of Machine Learning Research*, pages 2649–2680, 2011.

[220] A. L. Yuille. The Convergence of Contrastive Divergences. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 1593–1600, 2005.

[221] N. Zhang and S. Sun. Multiview Graph Restricted Boltzmann Machines. *IEEE Transactions on Cybernetics*, 52(11):12414–12428, 2022.

[222] Z. Zhang, J. Geiger, J. Pohjalainen, A. E.-D. Mousa, W. Jin, and B. Schuller. Deep Learning for Environmentally Robust Speech Recognition: An Overview of Recent Developments. *ACM Transactions on Intelligent Systems and Technology*, 9(5):1–28, 2018.

[223] Z. Zhang, Q. Zhu, G.-S. Xie, Y. Chen, Z. Li, and S. Wang. Discriminative Margin-Sensitive Autoencoder for Collective Multi-View Disease Analysis. *Neural Networks*, 123:94–107, 2020.

[224] H. Zhao, Z. Ding, and Y. Fu. Multi-View Clustering via Deep Matrix Factorization. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 2921–2927, 2017.

[225] J. Zhao, X. Xie, X. Xu, and S. Sun. Multi-View Learning Overview: Recent Progress and New Challenges. *Information Fusion*, 38:43–54, 2017.

[226] Z.-Q. Zhao, P. Zheng, S. t. Xu, and X. Wu. Object Detection with Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11):3212–3232, 2019.

[227] Y. Zhou, N. Hu, and C. J. Spanos. Veto-Consensus Multiple Kernel Learning. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2407–2414, 2016.

[228] W. Ziarko. Variable Precision Rough Set Model. *Journal of Computer and System Sciences*, 46:39–59, 1993.

[229] V. A. Zimmer, B. Glocker, N. Hahner, E. Eixarch, G. Sanroma, E. GratacÃşs, D. Rueckert, M. ÃĄ. GonzÃąlez Ballester, and G. Piella. Learning and Combining Image Neighborhoods Using Random Forests for Neonatal Brain Disease Classification. *Medical Image Analysis*, 42:189–199, 2017.

[230] V. A. Zimmer, K. Lekadir, C. Hoogendoorn, A. F. Frangi, and G. Piella. A Framework for Optimal Kernel-Based Manifold Embedding of Medical Image Data. *Computerized Medical Imaging and Graphics*, 41:93–107, 2015.