This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Incomplete Multi-View Multi-Label Learning via Disentangled Representation and Label Semantic Embedding

Xu Yan[1], Jun Yin[1,*], Jie Wen[2]
[1]Shanghai Maritime University, [2]Harbin Institute of Technology, Shenzhen
yanxu@stu.shmtu.edu.cn, junyin@shmtu.edu.cn, jiewen_pr@126.com

Abstract

In incomplete multi-view multi-label learning scenarios, it is crucial to use the incomplete multi-view data to extract consistent and specific representations from different data sources and to fully exploit the missing label information. However, most previous approaches ignore the separation problem between view-shared and specific information. To address this problem, in this paper, we propose a method that can separate view-consistent features from view-specific features under the Variational Autoencoder (VAE) framework. Specifically, we first introduce cross-view reconstruction to capture view-consistent features and extract shared information from different views through unsupervised pre-training. Subsequently, we develop a disentangling module to learn specific features by minimizing the variational upper bound of mutual information between consistent and specific features. Finally, we utilize prior label relevance information derived from training data to guide the learning of the distribution of label semantic embeddings, aggregating relevant semantic embeddings and maintaining the label relevance topology in the semantic space. In extensive experiments, our model outperforms existing state-of-the-art algorithms on several real-world datasets, which fully validates its strong adaptability to missing views and labels.

1. Introduction

In recent years, with the increasing diversity and complexity of data, multi-view learning has gained widespread attention as an effective data processing technique. In many real-world applications, such as image categorization, video analysis, and text mining, data is often presented in the form of multiple views, which makes multi-view learning methods an effective tool for solving complex problems [28, 35, 36]. In the past few years, many multi-view learning methods have been proposed based on the assumption of view completeness [4, 30, 38]. However, it is not uncommon to have missing views when dealing with multi-view data, which poses a challenge to the learning effectiveness of the model [19, 39].

Meanwhile, multi-label learning is an equally important research area. Unlike traditional single-label classification tasks, multi-label learning aims to deal with situations where instances may belong to more than one category at the same time. The key to multi-label learning is how to deal with the correlation between labels, which is helpful in improving the classification performance [12, 42]. Multi-label datasets are often incomplete due to various uncertainties involved in manual labeling. So research on incomplete multi-label learning is particularly important.

Traditional multi-label classification relies on features extracted from a single view for classification, which leads to insufficient feature information. Using multi-view data for multi-label classification can effectively alleviate this problem. Currently, there are many models dealing with multi-view multi-label classification (MVMLC) tasks that show promising performance [33, 37, 41]. These models typically assume that the data is complete, and currently, only a few methods are capable of handling both missing views and missing labels simultaneously.

To deal with the Double Incomplete Multi-View Multi-Label Classification problem (DIMVMLC), several deep learning models based on autoencoders have been proposed [22, 23, 31]. However, there are still only a few methods for disentangling consistent features and specific features. SIMM [33] identifies consistent features by minimizing confusing adversarial loss and multi-label loss, uses orthogonal constraints to eliminate shared information, and extracts specific information from raw information. However, SIMM cannot be applied to the missing data case, and the forced orthogonality constraint eliminates linear correlation, and there may still be a nonlinear relationship between shared and specific features. This relationship may be difficult to capture by simple orthogonality constraints, affecting the expressiveness of the features. MTD [23] simultaneously learns consistent and specific features, sepa-

*Corresponding author

rating them using multi-view cross-channel contrastive loss. Optimizing both tasks simultaneously may lead to a complex optimization process where the model needs to trade off between two objectives, increasing the difficulty of convergence.

To address the above complex double incomplete multi-view multi-label learning problem, we propose a **D**isentangled **R**epresentation and **L**abel **S**emantic Embedding based method (DRLS), as shown in Fig. 1. Our approach consists of three main parts: (1) Learning multi-view consistent features by minimizing the KL divergence between the approximate posterior of each view's consistent features and the multi-view joint posterior. (2) Learning the disentangled specific features by minimizing the variational upper bound of mutual information between the consistent features and the specific features. (3) We model the distribution of label semantic embeddings by leveraging prior label relevance information from the training data and utilizing graph neural networks. Our model is built upon the Variational Autoencoder (VAE) framework. Numerous experiments demonstrate the excellent performance of DRLS. Finally, the contributions of this paper are summarized as follows:

- We propose a novel approach based on disentangled representation and label semantic embedding for solving the DIMVMLC problem. It can effectively utilize the consistent and specific information of multi-view data, and separate them through a disentangled representation learning module for effective multi-label classification tasks.
- We introduce a pre-training strategy that learns view-consistent representations via pre-training and then minimizes the variational upper bound of mutual information between consistent and specific features to separate them effectively.
- We design a graph neural network-based label semantic embedding learning module that simultaneously aggregates embeddings of related labels and maintains the label relevance topology in the semantic space.
- Our approach achieves leading performance on five publicly available datasets and can be applied to both complete and arbitrarily missing data.

2. Related works

Many MVMLC methods have been proposed. For example, CDMM [40] builds independent classifiers based on neural networks constructed with nonlinear kernel mapping to ensure that each view learns consistent labeling results. The diversity across views is addressed by integrating the Hilbert-Schmidt independence criterion [7] with an exploration of the variations among different views. MMFA [3] initially extracts both common and private features from different views, then computes the similarity between the original features and the common features. The attention weight matrix is derived by multiplying this similarity with the synergistic feature matrix, which is subsequently used for multi-label prediction.

For DIMVMLC, iMvWL [27] uses Nonnegative Matrix Factorization (NMF) to learn the shared subspace of multiple views and introduces label information and local label correlation matrices to improve the discriminative power of the subspace. NAIML [20] exploits both consistency among multiple views and the complex global and local structure among labels to cope with the lack of available labels. Both of these methods are shallow machine learning approaches, and current deep neural network-based methods show better performance in this area, such as DDINet [31], DICNet [22], and SIP [21]. DDINet is the first to use a deep neural network to deal with DIMVMLC. DICNet leverages contrastive loss to learn shared features across views, aligns cross-view instances, and improves feature discrimination through a multi-view fusion module, effectively handling missing views and labels. The SIP model utilizes the principles of the information bottleneck theory to reduce the non-shared information while preserving the relevant features, and captures the associations between labels through prototype instances in the latent space.

3. Method

3.1. Problem definition

Given a dataset $\{x^{(v)}\}_{v=1}^{m}$ consisting of n samples, each with m views, where $x^{(v)} \in \mathbb{R}^{d_v}$. Here d_v denotes the feature dimension of the vth view. The label vector is $y \in \{0,1\}^k$, where k denotes the number of categories. If $y_i = 0$, the sample does not have the i-th label; if $y_i = 1$, it does. To handle missing data, we define the set $\mathcal{W}\,(|\mathcal{W}| \leq m)$ to denote the set of observable views of a sample, and the observable multi-view data can be defined as $\{x^{(v)}\}_{v\in\mathcal{W}}$, abbreviated as $\{x\}$. We also define a set of observable labels for a sample $\mathcal{U}\,(|\mathcal{U}| \leq k)$. Our goal is to train a model to predict the labels of unseen samples. Our loss and objective functions are defined in the case of a single sample.

3.2. View-consistent feature

In traditional multi-view learning, consistent features are often learned separately without cross-view interaction [33], making it hard to fully capture the common semantic information across views. In this subsection, we first introduce how to learn multi-view consistent features through cross-view reconstruction.

First, we define $q_{\phi_v}(c \mid x^{(v)})$ with learnable parameters ϕ_v to approximate the posterior $p(c \mid x^{(v)})$, where c represents the consistent feature. The consistent features of each view should contain the public information of all views, inspired by [2], we learn the consistent features by constraining the approximate posterior of each view to be similar to

the multi-view joint posterior:

$$\min \frac{1}{|\mathcal{W}|} \sum_{v \in \mathcal{W}} \left[D_{KL}(q_{\phi_v}(c \mid x^{(v)}) \| p(c \mid \{x\})) \right] \tag{1}$$

where $D_{KL}(\cdot\|\cdot)$ represents the KL divergence, $|\mathcal{W}|$ denotes the number of available views, and $p(c \mid \{x\})$ denotes the multi-view joint posterior.

The Evidence Lower Bound (ELBO) is derived from Eq. (1) by further applying Bayes' theorem to decompose the multi-view joint posterior distribution $p(c \mid \{x\})$. Due to space constraints, the exact derivation is given in the supplementary material Sec. 6.1.

The final result is as follows:

$$\begin{aligned} &\log p(\{x\}) - \frac{1}{|\mathcal{W}|} \sum_{v \in \mathcal{W}} \left[D_{KL}(q_{\phi_v}(c|x^{(v)}) \| p(c|\{x\})) \right] \\ &= \frac{1}{|\mathcal{W}|} \sum_{v \in \mathcal{W}} \Big[\mathbb{E}_{q_{\phi_v}(c|x^{(v)})}[\log p(\{x\}|c)] \\ &\qquad - D_{KL}(q_{\phi_v}(c \mid x^{(v)}) \| p(c)) \Big] \end{aligned} \tag{2}$$

This establishes the Evidence Lower Bound (ELBO). Thus, by maximizing this lower bound, we are effectively solving the minimization problem in Eq. (1).

By assuming each view is conditionally independent from the others given c, we can efficiently decompose the data likelihood as $p(\{x\}|c) = \prod_{i \in \mathcal{W}} p\left(x^{(i)}|c\right) \approx \prod_{i \in \mathcal{W}} p_{\theta_i}\left(x^{(i)}|c\right)$, where $p_{\theta_i}\left(x^{(i)}|c\right)$ with learnable parameters θ_i approximates $p(x^{(i)}|c)$. The objective in Eq. (2) involves extracting consistent features from different views. To facilitate subsequent analysis, we introduce an identifier for the consistent features of each view to distinguish them, i.e., $\{c^{(v)}\}_{v=1}^{m}$. Introducing this identifier does not alter the model architecture. Additionally, we introduce a trade-off parameter α [10], which controls the trade-off between reconstruction quality and regularization, resulting in the following lower bound:

$$\begin{aligned} &\mathcal{L}_{consist} = \\ &\frac{1}{|\mathcal{W}|} \sum_{v \in \mathcal{W}} \Big[\mathbb{E}_{q_{\phi_v}(c^{(v)}|x^{(v)})} \Big[\sum_{i \in \mathcal{W}} \log p_{\theta_i}(x^{(i)}|c^{(v)}) \Big] \\ &\qquad - \alpha D_{KL}(q_{\phi_v}(c^{(v)} \mid x^{(v)}) \| p(c)) \Big] \end{aligned} \tag{3}$$

here $q_{\phi_v}(c^{(v)} \mid x^{(v)})$ can be regarded as an encoder, and we assume that $q_{\phi_v}(c^{(v)} \mid x^{(v)}) \sim \mathcal{N}(\mu_{\phi_v}(x^{(v)}), \sigma^2_{\phi_v}(x^{(v)})I)$, where $\mu_{\phi_v}(x^{(v)})$ and $\sigma^2_{\phi_v}(x^{(v)})$ denote the mean and variance encoder of the vth view, and I denotes the identity matrix. Due to the non-differentiable nature of the sampling process, we introduce the reparameterization trick [16] to obtain the features:

$$c^{(v)} = \mu_{\phi_v}(x^{(v)}) + \sigma_{\phi_v}(x^{(v)}) \odot \delta \tag{4}$$

where $\delta \sim \mathcal{N}(0, I)$, $\odot$ denotes the Hadamard product, and $c^{(v)} \in \mathbb{R}^{d_e}$, where d_e represents the feature dimension. The distribution $p_{\theta_i}(x^{(i)}|c^{(v)})$ in Eq. (3) is modeled as an MLP decoder, assuming $p_{\theta_i}(x^{(i)}|c^{(v)}) \sim \mathcal{N}(\mu_{\theta_i}(c^{(v)}), I)$, yielding the reconstructed feature $\hat{x}^{(i,v)} = \mu_{\theta_i}(c^{(v)})$.

A brief analysis of Eq. (3) is as follows: The first term represents both self- and cross-view reconstruction. It allows $c^{(v)}$ from each view to reconstruct the original data of different views, which helps $c^{(v)}$ capture shared information. The second term, the KL divergence, aligns the posterior $q_{\phi_v}(c^{(v)} \mid x^{(v)})$ of each view with the shared prior $p(c) \sim \mathcal{N}(0, I)$. These two terms simultaneously encourage $c^{(v)}$ to encode the shared information between views.

Finally, in order to obtain a fused consistent feature shared by multiple views for the classification task, we compute the multi-view joint posterior $p(c|\{x\})$ using the Product-of-Expert (PoE) approach [11]. This method is flexible and effective in dealing with the missing views by directly removing the corresponding parameter when a view is missing. According to the setting in [32], we compute $p(c|\{x\})$ as follows:

$$\begin{aligned} p(c|\{x\}) &\propto p(c) \prod_{v \in \mathcal{W}} p(c \mid x^{(v)}) \\ &:= p(c) \prod_{v \in \mathcal{W}} q_{\phi_v}(c^{(v)} \mid x^{(v)}) \end{aligned} \tag{5}$$

The same reparameterization trick is used to obtain the fused multi-view consistent feature $\bar{c} \in \mathbb{R}^{d_e}$, which is sampled from $p(c|\{x\})$.

3.3. Disentangled view-specific feature

In this section, we introduce view-specific features $\{s^{(v)}\}_{v=1}^{m}$ that encode information specific to each view beyond the consistent features. We consider minimizing the mutual information $I(c^{(v)}; s^{(v)})$ between the consistent and specific features for each view, which allows us to maximally disentangle the specific features from the consistent features. A natural idea is to do the learning of consistent features first, followed by disentangling the specific features from them [15]. Thus, we first pre-train with $\mathcal{L}_{consist}$ for a certain number of epochs, depending on the dataset size. In the second stage, we decrease the learning rate for the encoders of consistent features $\{q_{\phi_v}(c^{(v)} \mid x^{(v)})\}_{v=1}^{m}$ to stabilize the pre-trained features during further disentanglement. Our disentangling objective function is as follows:

$$\min \frac{1}{|\mathcal{W}|} \sum_{v \in \mathcal{W}} I(c^{(v)}; s^{(v)}) \tag{6}$$

We rewrite $I(c^{(v)}; s^{(v)})$ as follows, more details are provided in the supplementary material Sec. 6.2:

$$I(c^{(v)}; s^{(v)}) = I(x^{(v)}; c^{(v)}) + I(x^{(v)}; s^{(v)}) - I(x^{(v)}; c^{(v)}, s^{(v)}) \tag{7}$$

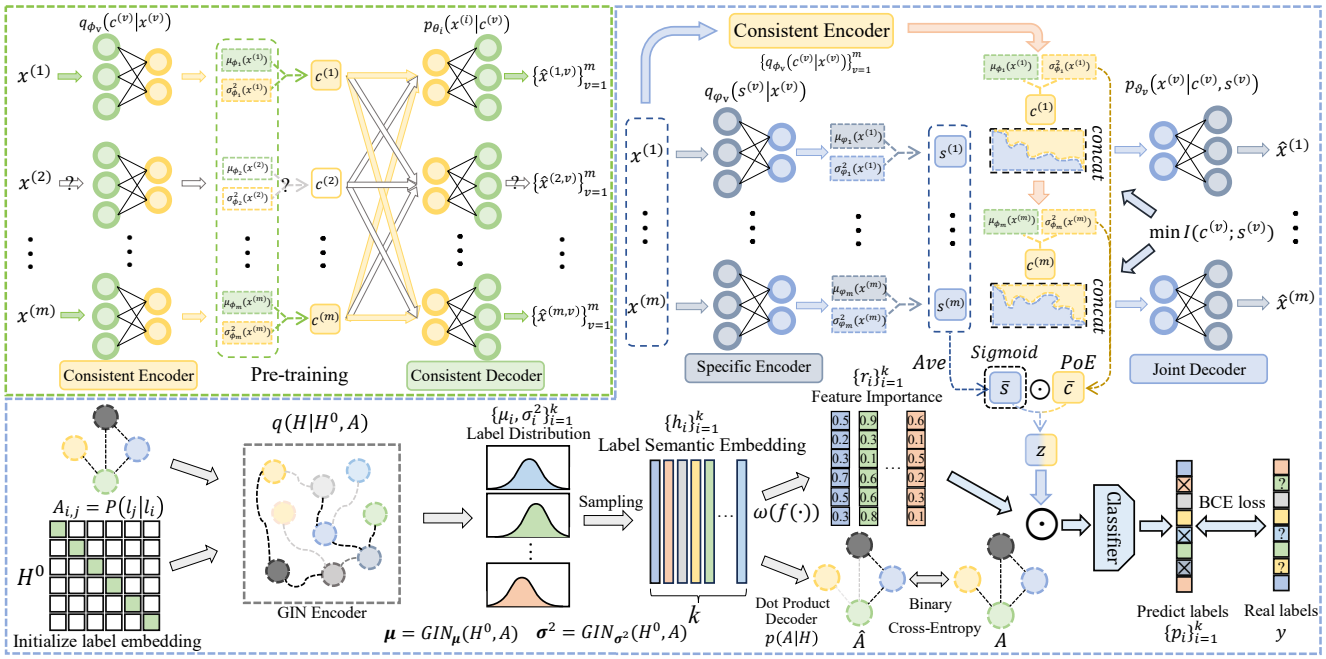


Figure 1. The main framework of DRLS. The upper-left corner of the figure represents the view-consistent feature learning module, the upper-right corner shows the disentangled view-specific feature learning module, and the lower part depicts the label semantic embedding learning module.

Clearly, directly optimizing Eq. (7) is challenging. Therefore, we introduce the variational upper bounds for $I(x^{(v)};c^{(v)})$ and $I(x^{(v)};s^{(v)})$, as well as the variational lower bound for $I(x^{(v)};c^{(v)},s^{(v)})$ [14], resulting in:

$$\begin{aligned} I(c^{(v)};s^{(v)}) \leq\ & D_{KL}(q_{\phi_v}(c^{(v)}|x^{(v)})\|p(c^{(v)})) \\ & + D_{KL}(q_{\varphi_v}(s^{(v)}|x^{(v)})\|p(s^{(v)})) \\ & -\mathbb{E}_{q_{\phi_v}(c^{(v)}|x^{(v)})q_{\varphi_v}(s^{(v)}|x^{(v)})}[\log p_{\vartheta_v}(x^{(v)}|c^{(v)},s^{(v)})] \end{aligned} \tag{8}$$

We observe that minimizing this variational upper bound on mutual information trains a decoder $p_{\vartheta_v}(x^{(v)}|c^{(v)},s^{(v)})$ and a specific feature encoder $q_{\varphi_v}(s^{(v)}|x^{(v)})$. Here, we concatenate $c^{(v)}$ and $s^{(v)}$ as input to the MLP decoder $p_{\vartheta_v}(x^{(v)}|c^{(v)},s^{(v)})$ to produce the reconstructed feature $\hat{x}^{(v)}$. Similar to Sec. 3.2, $q_{\varphi_v}(s^{(v)}|x^{(v)}) \sim \mathcal{N}(\mu_{\varphi_v}(x^{(v)}),\sigma^2_{\varphi_v}(x^{(v)})I)$ parameterizes the distribution of the specific features. The feature $s^{(v)} \in \mathbb{R}^{d_e}$ is derived using the reparameterization trick as shown in Eq. (4). We assume that $p(c^{(v)})$ and $p(s^{(v)})$ follow standard Gaussian distributions. Finally, we obtain the disentanglement loss:

$$\begin{aligned} \mathcal{L}_{disent} = \frac{1}{|\mathcal{W}|}\sum_{v\in\mathcal{W}} \big[& D_{KL}(q_{\phi_v}(c^{(v)}|x^{(v)})\|p(c^{(v)})) \\ & + D_{KL}(q_{\varphi_v}(s^{(v)}|x^{(v)})\|p(s^{(v)})) \\ & -\mathbb{E}_{q_{\phi_v}(c^{(v)}|x^{(v)})q_{\varphi_v}(s^{(v)}|x^{(v)})}[\log p_{\vartheta_v}(x^{(v)}|c^{(v)},s^{(v)})]\big] \end{aligned} \tag{9}$$

Comparing Eq. (3) and Eq. (9), the consistent objective function requires each consistent feature $c^{(v)}$ to reconstruct the original features across views, whereas in the disentanglement loss, we use the concatenation of consistent and specific features exclusively for self-view reconstruction. Since we pre-train a set of encoders $\{q_{\phi_v}(c^{(v)}|x^{(v)})\}_{v=1}^m$, in the second stage, $c^{(v)}$ and $s^{(v)}$ are concatenated to reconstruct $x^{(v)}$. As $c^{(v)}$ already contains the information shared between $x^{(v)}$ and other views, $s^{(v)}$ only needs to encode the complementary information that is not captured by $c^{(v)}$. This pre-training strategy enables more efficient disentanglement. We use a simple and effective method to obtain the fused multi-view specific feature $\bar{s} \in \mathbb{R}^{d_e}$, for subsequent classification tasks:

$$\bar{s} = \frac{1}{|\mathcal{W}|}\sum_{v\in\mathcal{W}} s^{(v)} \tag{10}$$

Here, we do not use PoE to fuse the specific features as in Sec. 3.2, because it is more suitable for merging shared information. PoE is sensitive to consensus among different views [18]. Specifically, if one view's probability distribution is overly sharp and biased, the joint distribution will have a biased mean prediction [1, 18, 26]. Given that specific features from different views often have differences, we choose a simple average fusion method to mitigate extreme variations and retain the overall trend.

Finally, we fuse the multi-view consistent and specific information. Following the method in [23], we use specific features to enhance the consistent features:

$$z = \omega(\bar{s}) \odot \bar{c} \tag{11}$$

$z \in \mathbb{R}^{d_e}$ represents the final fused representation, $\omega(\cdot)$ is the sigmoid activation function, and $\odot$ denotes the Hadamard product.

3.4. Label semantic embedding

To fully utilize the incomplete supervisory information, we introduce label semantic embedding learning guided by prior information. Here, we construct our model within the framework of the Variational Graph Autoencoder (VGAE) [17], which employs variational inference to model the latent distribution of label embeddings during the learning process.

First, we define a relationship graph among labels, where the graph $G = (V, E)$ represents the label relationship graph, with V denoting the nodes (categories) and E representing the edges connecting the nodes. We model the adjacency matrix $A \in \mathbb{R}^{k\times k}$ of graph G in terms of conditional probabilities, specifically $P(l_j|l_i)$, which denotes the probability that label l_j occurs when label l_i occurs, where l_i represents the ith label. This is computed based on the label matrix $Y \in \mathbb{R}^{n_{train}\times k}$ from the training set. Since $P(l_j|l_i)$ is not equal to $P(l_i|l_j)$, the adjacency matrix A is asymmetric, which more accurately reflects real-world scenarios. The specific calculation method is as follows:

$$A_{ij} = \begin{cases} P(l_j|l_i) = \frac{C_{ij}}{d_i}, & \text{if } i \neq j \\ 0, & \text{if } i = j \end{cases} \tag{12}$$

where $C = Y^\top Y$, $C \in \mathbb{R}^{k\times k}$ is the label co-occurrence matrix, C_{ij} is the co-occurrence count of labels l_i and l_j, $d_i = \sum_{a=1}^{n_{train}} Y_{ai} = Y_{:i}^\top Y_{:i}$ for $i = 1, 2, \ldots, k$, which represents the total occurrence count of label l_i. It should be noted that we set the diagonal of the adjacency matrix A to 0.

Graph Isomorphism Network (GIN) provides stronger feature aggregation capabilities through learnable weights and information from self-nodes [34]. Therefore, we use GIN to model the posterior distribution of label semantic embeddings $H \in \mathbb{R}^{k\times d_e}$:

$$\begin{aligned} q(H|H^0, A) &= \prod_{i=1}^{k} q(h_i|H^0, A), \\ \text{with } q(h_i|H^0, A) &\sim \mathcal{N}(\mu_i, \sigma_i^2 I) \end{aligned} \tag{13}$$

here, $h_i \in \mathbb{R}^{d_e}$ is the i-th row of H, representing the semantic embedding of the i-th label, and $H^0 \in \mathbb{R}^{k\times k}$ is an identity matrix, which means that the embedding for label i is initialized as a one-hot vector with the i-th position set to 1. This approach ensures that the initial distance between different labels is large, which helps to better distinguish the semantic differences between labels during the subsequent learning process. $\boldsymbol{\mu} = \mathrm{GIN}_{\boldsymbol{\mu}}(H^0, A)$, $\boldsymbol{\mu} \in \mathbb{R}^{k\times d_e}$ is the mean matrix, where the i-th row corresponds to the mean vector μ_i for the i-th label; similarly, $\boldsymbol{\sigma}^2 = \mathrm{GIN}_{\boldsymbol{\sigma}^2}(H^0, A)$. Here, we use a three-layer GIN, where one layer can be defined as $H^{(t+1)} = f^{(t+1)}[(1 + \epsilon^{(t+1)})H^{(t)} + AH^{(t)}]$, t represents the current number of network layers, $\epsilon^{(t+1)}$ represents a learnable parameter, and $f^{(t+1)}$ represents a two-layer fully connected network. $H^{(t+1)} \in \mathbb{R}^{k\times d_{t+1}}$ represents the output of the intermediate layer.

Subsequently, we also use the reparameterization trick to sample the label semantic embeddings, with a formula similar to Eq. (4), i.e., $h_i = \mu_i + \sigma_i \odot \delta, \delta \sim \mathcal{N}(0, I)$. Finally, we obtain the label semantic embedding matrix H.

We use a simple and efficient inner product between variables as a generative model:

$$\begin{aligned} p(A|H) &= \prod_{i=1}^{k}\prod_{j=1}^{k} p(A_{ij}|h_i, h_j), \\ \text{with } p(A_{ij}|h_i, h_j) &= \omega(h_i^\top h_j) \end{aligned} \tag{14}$$

where $\omega(\cdot)$ denotes the sigmoid activation function. By using the label correlation matrix as the adjacency matrix of a graph, we enable GIN to aggregate semantically related embeddings to nearby positions in the label semantic space. However, it cannot guarantee that unrelated labels are effectively pushed apart. Therefore, we introduce the generative model from VGAE to reconstruct the adjacency matrix A, ensuring that the label semantic embeddings effectively preserve the topology of label correlations in the semantic space. Finally, our variational lower bound is:

$$\mathcal{L}_{le} = \mathbb{E}_{q(H|H^0,A)}[\log p(A|H)] - D_{KL}(q(H|H^0, A)\|p(H)) \tag{15}$$

We further adopt a Gaussian prior $p(H) = \prod_i p(h_i) = \prod_i \mathcal{N}(h_i|0, I)$. We use binary cross-entropy to compute the first term of $\mathcal{L}_{le}$ as the reconstruction loss and add an identity matrix to A before computing the cross-entropy loss to avoid enforcing $\omega(h_i^\top h_i) \approx 0$ due to its zero diagonal.

3.5. Overall loss function

Finally, we utilize label semantic embeddings for feature selection, choosing discriminative and specific features for each label. First, the learned label semantic embedding h_i is mapped to a feature importance vector [9] via a fully connected layer $f(\cdot)$, specifically $r_i = \omega(f(h_i))$, where $r_i \in \mathbb{R}^{d_e}$ and $\omega(\cdot)$ is the sigmoid activation function. Subsequently, we use r_i to select the features and input them into a simple neural network to obtain predictions:

$$p_i = \omega\left(f_{cls}\left(z \odot r_i\right)\right) \tag{16}$$

where $\odot$ denotes the Hadamard product, f_{cls} is a fully connected network, and p_i is the predicted probability of the sample being labeled as the i-th class. We use the commonly used binary cross-entropy function to calculate the

Table 1. Statistics for five multi-view multi-label datasets, where m denotes the number of views, n denotes the number of samples, k denotes the number of categories, and n/k denotes the average number of labels per sample.

Dataset	m	n	k	n/k
Corel5k	6	4999	260	3.396
Pascal07	6	9963	20	1.465
Espgame	6	20770	268	4.686
IAPRTC12	6	19627	291	5.719
Mirflickr	6	25000	38	4.716

multi-label classification loss:

$$\mathcal{L}_{BCE} = -\frac{1}{|\mathcal{U}|}\sum_{i\in\mathcal{U}}\left(y_i \log p_i + (1-y_i)\log(1-p_i)\right) \tag{17}$$

where y_i denotes the ground truth and we exclude missing labels from loss computation to filter out the impact of missing labels. Our final loss function is:

$$\mathcal{L} = \mathcal{L}_{BCE} + \beta\mathcal{L}_{disent} - \gamma\mathcal{L}_{le} \tag{18}$$

where β and γ are trade-off parameters used to balance the different loss terms.

4. Experiments

4.1. Datasets

We conduct experiments based on the settings from [21, 23, 31] to evaluate the performance of our proposed model on five datasets. They are Corel5k [5], Pascal07 [6], Espgame [29], IAPRTC12 [8], and Mirflickr [13]. In these five datasets, we utilize six different types of features, which are DenseHue, DenseSift, GIST, HSV, RGB, and LAB, with dimensions of 100, 1000, 512, 4096, 4096, and 4096, respectively. More detailed information can be found in Tab. 1.

4.2. Compared methods

In our experiments, we select eight methods for comparison with our model: DM2L [24], CDMM [40], iMvWL [27], NAIML [20], DDINet [31], DICNet [22], MTD [23], and SIP [21]. It is important to note that iMvWL, NAIML, DDINet, DICNet, MTD, and SIP are incomplete multi-view multi-label methods. CDMM can only handle complete multi-view multi-label data; therefore, we follow the settings from [21] and use the mean of available instances from the corresponding view to fill in the unavailable instances for missing views. To handle incomplete labels, we treat the missing labels as negative labels. DM2L is a single-view incomplete multi-label method, and we record the best result of each view in our experiments.

4.3. Implementation details

For the datasets, in order to simulate real-world data missingness, we generate incomplete datasets following the settings in [27] and [21]. For each view, we randomly exclude 50% of the instances, ensuring at least one view per sample. Additionally, we randomly remove 50% of both positive and negative labels. We fill the missing features and labels with zeros. We use 70% of the samples for training and the remaining 30% for validation and testing. To ensure the stability of the results, we repeat the experiment 10 times and report the mean and variance of the results. Additional details are in the supplementary material Sec. 7.

4.4. Evaluation metrics

Consistent with some previous studies [21, 23, 31], in our experiments, we evaluate the performance of the model using six commonly used multi-label classification metrics, namely, Average Precision (AP), Area Under the Receiver Operating Characteristic Curve (AUC), Ranking Loss (RL), Hamming Loss (HL), OneError (OE), and Coverage (Cov). To better visualize the results, we use 1-RL, 1-HL, 1-OE, and 1-Cov to record the final results, where larger values indicate better performance.

4.5. Experimental results

In Tab. 2, we present the experimental results of nine methods on five public multi-view multi-label datasets, which include six evaluation metrics. The results in Tab. 2 are obtained under experimental settings with a 50% view missing rate, a 50% label missing rate, and 70% of training samples. The results in Tab. 2 show that our method outperforms the other eight competitors on each metric in most datasets. For example, on the AP metric of the IAPRTC12 dataset, our method is 2.4% higher than the second-ranked MTD. Specifically, Ave.R represents the average ranking of each method across the six metrics, and DRLS ranks first on all five datasets, which fully demonstrates the effectiveness of our method in addressing the DIMVMLC problem.

In addition, further analysis of the experimental results reveals that CDMM, DDINet, DICNet, MTD, and SIP outperform the other three methods on all five datasets. These are deep learning methods, indicating the great potential of deep learning in addressing the DIMVMLC problem.

Besides solving the DIMVMLC problem, we also further validated the performance of our method in addressing the MVMLC problem. In Tab. 3, we present the experimental results of four methods on five datasets, with experiments conducted under the settings of 70% training samples, full views, and full labels. We select three representative evaluation metrics. From the results in Tab. 3, it is evident that our model still demonstrates competitive performance under this setup, indicating that our method can effectively address the MVMLC problem.

Table 2. Experimental results of nine methods on five datasets with 50% missing views, 50% missing labels, and 70% training samples. Ave.R denotes the average ranking of each method on the six metrics, and the numbers at the bottom right are the standard deviations.

Dataset	Metric	DM2L	CDMM	iMvWL	NAIML	DDINet	DICNet	MTD	SIP	DRLS
Corel5k	AP	$0.262_{0.005}$	$0.354_{0.004}$	$0.283_{0.008}$	$0.309_{0.004}$	$0.360_{0.009}$	$0.378_{0.004}$	$0.413_{0.007}$	$0.416_{0.015}$	$0.433_{0.008}$
	1-HL	$0.987_{0.000}$	$0.987_{0.000}$	$0.978_{0.000}$	$0.987_{0.000}$	$0.987_{0.000}$	$0.987_{0.000}$	$0.988_{0.000}$	$0.988_{0.000}$	$0.988_{0.000}$
	1-RL	$0.843_{0.002}$	$0.884_{0.003}$	$0.865_{0.005}$	$0.878_{0.002}$	$0.865_{0.005}$	$0.877_{0.004}$	$0.892_{0.004}$	$0.910_{0.003}$	$0.916_{0.002}$
	AUC	$0.845_{0.002}$	$0.888_{0.003}$	$0.868_{0.005}$	$0.881_{0.002}$	$0.868_{0.005}$	$0.881_{0.003}$	$0.895_{0.004}$	$0.912_{0.003}$	$0.918_{0.002}$
	1-OE	$0.295_{0.014}$	$0.410_{0.007}$	$0.311_{0.015}$	$0.350_{0.009}$	$0.437_{0.012}$	$0.464_{0.012}$	$0.491_{0.010}$	$0.492_{0.018}$	$0.509_{0.019}$
	1-Cov	$0.647_{0.005}$	$0.723_{0.007}$	$0.702_{0.008}$	$0.725_{0.005}$	$0.689_{0.012}$	$0.714_{0.010}$	$0.748_{0.009}$	$0.786_{0.007}$	$0.804_{0.006}$
	Ave.R	8.167	4.833	7.667	5.333	6.000	4.833	2.667	1.833	1.000
Pascal07	AP	$0.471_{0.008}$	$0.508_{0.005}$	$0.437_{0.018}$	$0.488_{0.003}$	$0.532_{0.010}$	$0.502_{0.007}$	$0.550_{0.004}$	$0.550_{0.009}$	$0.567_{0.008}$
	1-HL	$0.928_{0.001}$	$0.931_{0.001}$	$0.882_{0.004}$	$0.928_{0.001}$	$0.932_{0.001}$	$0.930_{0.001}$	$0.932_{0.001}$	$0.931_{0.002}$	$0.934_{0.001}$
	1-RL	$0.761_{0.005}$	$0.812_{0.004}$	$0.736_{0.015}$	$0.783_{0.001}$	$0.808_{0.005}$	$0.781_{0.007}$	$0.830_{0.003}$	$0.825_{0.006}$	$0.843_{0.004}$
	AUC	$0.779_{0.004}$	$0.838_{0.003}$	$0.767_{0.015}$	$0.811_{0.001}$	$0.829_{0.004}$	$0.805_{0.006}$	$0.849_{0.004}$	$0.845_{0.005}$	$0.864_{0.003}$
	1-OE	$0.420_{0.011}$	$0.419_{0.008}$	$0.362_{0.023}$	$0.421_{0.006}$	$0.448_{0.015}$	$0.426_{0.013}$	$0.457_{0.008}$	$0.463_{0.012}$	$0.477_{0.011}$
	1-Cov	$0.692_{0.004}$	$0.759_{0.003}$	$0.677_{0.015}$	$0.727_{0.002}$	$0.757_{0.005}$	$0.728_{0.007}$	$0.783_{0.004}$	$0.777_{0.005}$	$0.798_{0.004}$
	Ave.R	7.667	4.833	9.000	6.500	4.167	6.167	2.167	2.833	1.000
Espgame	AP	$0.212_{0.002}$	$0.289_{0.003}$	$0.244_{0.005}$	$0.246_{0.002}$	$0.286_{0.004}$	$0.299_{0.004}$	$0.306_{0.003}$	$0.310_{0.004}$	$0.326_{0.005}$
	1-HL	$0.982_{0.000}$	$0.983_{0.000}$	$0.972_{0.000}$	$0.983_{0.000}$	$0.983_{0.000}$	$0.983_{0.000}$	$0.983_{0.000}$	$0.983_{0.000}$	$0.983_{0.000}$
	1-RL	$0.781_{0.001}$	$0.832_{0.001}$	$0.808_{0.002}$	$0.818_{0.002}$	$0.815_{0.003}$	$0.833_{0.003}$	$0.837_{0.001}$	$0.849_{0.002}$	$0.858_{0.002}$
	AUC	$0.785_{0.001}$	$0.836_{0.001}$	$0.813_{0.002}$	$0.824_{0.002}$	$0.819_{0.003}$	$0.837_{0.002}$	$0.842_{0.001}$	$0.853_{0.002}$	$0.862_{0.002}$
	1-OE	$0.294_{0.006}$	$0.396_{0.005}$	$0.343_{0.013}$	$0.339_{0.003}$	$0.427_{0.010}$	$0.437_{0.010}$	$0.448_{0.006}$	$0.451_{0.012}$	$0.473_{0.001}$
	1-Cov	$0.488_{0.003}$	$0.574_{0.004}$	$0.548_{0.004}$	$0.571_{0.003}$	$0.553_{0.005}$	$0.598_{0.006}$	$0.601_{0.004}$	$0.628_{0.004}$	$0.652_{0.003}$
	Ave.R	8.833	4.500	8.000	5.667	5.500	3.500	2.667	1.833	1.000
IAPRTC12	AP	$0.234_{0.003}$	$0.305_{0.004}$	$0.237_{0.003}$	$0.261_{0.001}$	$0.302_{0.005}$	$0.327_{0.005}$	$0.332_{0.002}$	$0.331_{0.007}$	$0.356_{0.006}$
	1-HL	$0.980_{0.000}$	$0.981_{0.000}$	$0.969_{0.000}$	$0.980_{0.000}$	$0.980_{0.000}$	$0.980_{0.000}$	$0.981_{0.000}$	$0.981_{0.000}$	$0.981_{0.000}$
	1-RL	$0.823_{0.002}$	$0.862_{0.002}$	$0.833_{0.002}$	$0.848_{0.001}$	$0.853_{0.002}$	$0.872_{0.002}$	$0.875_{0.001}$	$0.887_{0.004}$	$0.896_{0.003}$
	AUC	$0.825_{0.001}$	$0.864_{0.002}$	$0.835_{0.001}$	$0.850_{0.001}$	$0.855_{0.003}$	$0.873_{0.001}$	$0.876_{0.001}$	$0.888_{0.003}$	$0.898_{0.002}$
	1-OE	$0.340_{0.006}$	$0.432_{0.008}$	$0.352_{0.008}$	$0.390_{0.005}$	$0.435_{0.009}$	$0.465_{0.013}$	$0.471_{0.006}$	$0.466_{0.001}$	$0.490_{0.012}$
	1-Cov	$0.529_{0.004}$	$0.597_{0.004}$	$0.564_{0.005}$	$0.592_{0.004}$	$0.594_{0.007}$	$0.648_{0.005}$	$0.649_{0.002}$	$0.679_{0.008}$	$0.707_{0.007}$
	Ave.R	8.333	4.500	8.167	6.667	5.667	4.167	2.333	2.167	1.000
Mirflickr	AP	$0.514_{0.006}$	$0.570_{0.002}$	$0.490_{0.012}$	$0.551_{0.002}$	$0.588_{0.003}$	$0.586_{0.005}$	$0.608_{0.004}$	$0.615_{0.004}$	$0.630_{0.005}$
	1-HL	$0.878_{0.001}$	$0.886_{0.001}$	$0.839_{0.002}$	$0.882_{0.001}$	$0.888_{0.001}$	$0.888_{0.001}$	$0.891_{0.001}$	$0.891_{0.001}$	$0.895_{0.001}$
	1-RL	$0.831_{0.003}$	$0.856_{0.001}$	$0.803_{0.008}$	$0.844_{0.001}$	$0.865_{0.002}$	$0.861_{0.004}$	$0.875_{0.001}$	$0.878_{0.002}$	$0.885_{0.002}$
	AUC	$0.828_{0.003}$	$0.846_{0.001}$	$0.787_{0.012}$	$0.837_{0.001}$	$0.853_{0.002}$	$0.848_{0.004}$	$0.861_{0.002}$	$0.864_{0.002}$	$0.872_{0.003}$
	1-OE	$0.510_{0.008}$	$0.631_{0.004}$	$0.511_{0.022}$	$0.585_{0.003}$	$0.636_{0.008}$	$0.642_{0.006}$	$0.656_{0.004}$	$0.664_{0.006}$	$0.686_{0.009}$
	1-Cov	$0.604_{0.005}$	$0.640_{0.001}$	$0.572_{0.013}$	$0.631_{0.002}$	$0.654_{0.003}$	$0.646_{0.006}$	$0.677_{0.002}$	$0.676_{0.004}$	$0.692_{0.003}$
	Ave.R	8.167	6.000	8.833	7.000	4.167	4.667	2.667	2.167	1.000

4.6. Parameter analysis

In our DRLS model, there are three hyperparameters that need to be tuned: α, β, and γ. To evaluate the sensitivity of our model to these hyperparameters, we conduct experiments on the Corel5k and Pascal07 datasets, with experiments carried out under conditions where 50% of the views are missing, 50% of the labels are missing, and 70% of the training samples are used.

Figs. 2a and 2b show the AP values of our model under different combinations of β and γ, while Figs. 2c and 2d show the AP and AUC values for different values of α. To ensure the validity of the experiments, the other unrelated hyperparameters are fixed. Based on Figs. 2a and 2b, for the Corel5k dataset, β and γ are taken in the range of $[1e-3, 1e-1]$ and $[1e-5, 1e-3]$, respectively, and for the Pascal07 dataset, β and γ are taken in the range of $[1e-3, 1e-1]$ and $[1e-5, 1e-3]$, respectively. Under these conditions, our model demonstrates good performance. Regarding the hyperparameter α, from Figs. 2c and 2d, it can be seen that the variation in AP and AUC values is minimal for both datasets, indicating that our model is not sensitive to α. We set α within the range of $[1e-1, 3e-1]$.

4.7. Ablation study

Our model consists of multiple modules and objective functions. To further analyze the contribution of each module and objective function, we conduct a series of ablation experiments on the Corel5k and Pascal07 datasets under a set-

Table 3. Experimental results of four methods on five datasets with full views, full labels, and 70% training samples.

Dataset	Metric	DICNet	MTD	SIP	DRLS
Corel5k	AP	0.504	0.486	0.520	0.551
	1-RL	0.925	0.921	0.937	0.947
	AUC	0.928	0.924	0.939	0.949
Pascal07	AP	0.609	0.652	0.645	0.655
	1-RL	0.860	0.890	0.887	0.889
	AUC	0.875	0.898	0.899	0.901
Espgame	AP	0.388	0.397	0.356	0.399
	1-RL	0.872	0.880	0.875	0.893
	AUC	0.875	0.883	0.878	0.896
IAPRTC12	AP	0.428	0.441	0.366	0.447
	1-RL	0.910	0.916	0.904	0.926
	AUC	0.911	0.916	0.906	0.927
Mirflickr	AP	0.661	0.682	0.676	0.698
	1-RL	0.897	0.906	0.907	0.915
	AUC	0.879	0.888	0.890	0.898

Table 4. Ablation results on two datasets with 50% missing views, 50% missing labels, and 70% training samples. 'w/o' denotes 'without'. 'A' denotes the consistent feature extraction module, 'B' denotes the disentangled specific feature extraction module, and DRLS denotes the full model. $\mathcal{L}_{con.kl}$ represents the KL divergence term of $\mathcal{L}_{consist}$.

Method	Corel5k			Pascal07		
	AP	1-RL	AUC	AP	1-RL	AUC
A	0.416	0.909	0.912	0.559	0.836	0.858
A+B	0.431	0.913	0.916	0.563	0.840	0.863
DRLS	0.433	0.916	0.918	0.567	0.843	0.864
DRLS w/o $\mathcal{L}_{con.kl}$	0.415	0.908	0.911	0.541	0.823	0.846
DRLS w/o $\mathcal{L}_{disent}$	0.414	0.913	0.916	0.557	0.834	0.857
DRLS w/o $\mathcal{L}_{le}$	0.430	0.913	0.915	0.565	0.842	0.863

ting of 50% missing views and 50% missing labels.

The results of the ablation experiments are presented in Tab. 4. Specifically, we first test the performance of the model using only the consistent feature extraction module. Then, we add the disentangled specific feature extraction module, and finally, we incorporate the label semantic embedding learning module. Through these three experimental setups, we can observe that each module contributes to the model's performance. Additionally, by observing the results in Tab. 4, we can see that the disentangled specific feature extraction module plays a key role in improving model performance. Furthermore, the label semantic embedding learning module also helps to enhance the model's classification ability.

In addition, we also validate the impact of different objective functions on model performance. We find that removing $\mathcal{L}_{con.kl}$ or $\mathcal{L}_{disent}$ leads to a significant decline in model performance. Among them, $\mathcal{L}_{con.kl}$ is the objective

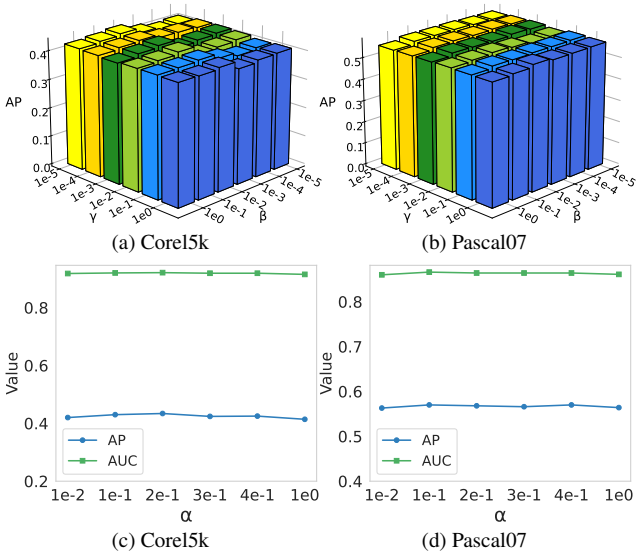


Figure 2. The AP values under different combinations of β and γ on the Corel5k (a) and Pascal07 (b) datasets; the AP and AUC values for different values of α on the Corel5k (c) and Pascal07 (d) datasets.

function used in the pre-training stage. The experimental results show that $\mathcal{L}_{con.kl}$ plays a regularization role, reducing redundant information in the shared features. The disentangling loss $\mathcal{L}_{disent}$ is responsible for separating the consistent features from the specific features, which improves the quality of the specific features.

5. Conclusion

In this paper, we propose a novel incomplete multi-view multi-label learning framework. Based on the assumptions of consistency and complementarity of multi-view data, we establish a method using Disentangled Representation and Label Semantic embedding (DRLS). Firstly, view-consistent features are learned using cross-view reconstruction, and the difficulty of disentangling specific features is reduced by unsupervised pre-training. Next, we achieve disentangled learning of specific features by minimizing the variational upper bound of mutual information between consistent and specific features. Meanwhile, we guide the learning of label semantic embeddings with the help of a priori label relevance to aggregate relevant semantic embeddings and maintain the topology of label relevance in the semantic space. Finally, we conduct a large number of experiments to demonstrate the effectiveness of our method in addressing the issues of missing views and labels.

Acknowledgment. This work is supported by National Natural Science Foundation of China (62476166) and Shanghai Pujiang Program (22PJD029).

References

[1] Ana Lawry Aguila and Andre Altmann. A tutorial on multi-view autoencoders using the multi-view-ae library. *arXiv preprint arXiv:2403.07456*, 2024. 4

[2] Luigi Antelmi, Nicholas Ayache, Philippe Robert, and Marco Lorenzi. Sparse multi-channel variational autoencoder for the joint analysis of heterogeneous data. In *International Conference on Machine Learning*, pages 302–311. PMLR, 2019. 2

[3] Yusheng Cheng, Qingyan Li, Yibin Wang, and Weijie Zheng. Multi-view multi-label learning with view feature attention allocation. *Neurocomputing*, 501:857–874, 2022. 2

[4] Bassoma Diallo, Jie Hu, Tianrui Li, Ghufran Ahmad Khan, Xinyan Liang, and Hongjun Wang. Auto-attention mechanism for multi-view deep embedding clustering. *Pattern Recognition*, 143:109764, 2023. 1

[5] Pinar Duygulu, Kobus Barnard, Joao FG de Freitas, and David A Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Computer Vision—ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part IV 7*, pages 97–112. Springer, 2002. 6

[6] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338, 2010. 6

[7] Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pages 63–77. Springer, 2005. 2

[8] Michael Grubinger, Paul Clough, Henning Müller, and Thomas Deselaers. The iapr tc-12 benchmark: A new evaluation resource for visual information systems. In *International workshop ontoImage*, 2006. 6

[9] Jun-Yi Hang and Min-Ling Zhang. Collaborative learning of label semantics and deep label-specific features for multi-label classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9860–9871, 2021. 5

[10] Irina Higgins, Loic Matthey, Arka Pal, Christopher P Burgess, Xavier Glorot, Matthew M Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR (Poster)*, 3, 2017. 3

[11] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8): 1771–1800, 2002. 3

[12] Sheng-Jun Huang and Zhi-Hua Zhou. Multi-label learning by exploiting label correlations locally. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 949–955, 2012. 1

[13] Mark J Huiskes and Michael S Lew. The mir flickr retrieval evaluation. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pages 39–43, 2008. 6

[14] HyeongJoo Hwang, Geon-Hyeong Kim, Seunghoon Hong, and Kee-Eung Kim. Variational interaction information maximization for cross-domain disentanglement. *Advances in Neural Information Processing Systems*, 33:22479–22491, 2020. 4, 1

[15] Guanzhou Ke, Bo Wang, Xiaoli Wang, and Shengfeng He. Rethinking multi-view representation learning via distilled disentangling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26774–26783, 2024. 3

[16] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[17] Thomas N Kipf and Max Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016. 5

[18] Ana Lawry Aguila, James Chapman, and Andre Altmann. Multi-modal variational autoencoders for normative modelling across multiple imaging modalities. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 425–434. Springer, 2023. 4

[19] Haobin Li, Yunfan Li, Mouxing Yang, Peng Hu, Dezhong Peng, and Xi Peng. Incomplete multi-view clustering via prototype-based imputation. *arXiv preprint arXiv:2301.11045*, 2023. 1

[20] Xiang Li and Songcan Chen. A concise yet effective model for non-aligned incomplete multi-view and missing multi-label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):5918–5932, 2021. 2, 6

[21] Chengliang Liu, Gehui Xu, Jie Wen, Yabo Liu, Chao Huang, and Yong Xu. Partial multi-view multi-label classification via semantic invariance learning and prototype modeling. In *Forty-first International Conference on Machine Learning*. 2, 6

[22] Chengliang Liu, Jie Wen, Xiaoling Luo, Chao Huang, Zhihao Wu, and Yong Xu. Dicnet: Deep instance-level contrastive network for double incomplete multi-view multi-label classification. In *Proceedings of the AAAI conference on artificial intelligence*, pages 8807–8815, 2023. 1, 2, 6

[23] Chengliang Liu, Jie Wen, Yabo Liu, Chao Huang, Zhihao Wu, Xiaoling Luo, and Yong Xu. Masked two-channel decoupling framework for incomplete multi-view weak multi-label learning. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 4, 6

[24] Zhongchen Ma and Songcan Chen. Expand globally, shrink locally: Discriminant multi-label learning with missing labels. *Pattern Recognition*, 111:107675, 2021. 6

[25] William McGill. Multivariate information transmission. *Transactions of the IRE Professional Group on Information Theory*, 4(4):93–111, 1954. 1

[26] Yuge Shi, Brooks Paige, Philip Torr, et al. Variational mixture-of-experts autoencoders for multi-modal deep generative models. *Advances in neural information processing systems*, 32, 2019. 4

[27] Qiaoyu Tan, Guoxian Yu, Carlotta Domeniconi, Jun Wang, and Zili Zhang. Incomplete multi-view weak-label learning. In *Ijcai*, pages 2703–2709, 2018. 2, 6

[28] Chang Tang, Zhenglai Li, Jun Wang, Xinwang Liu, Wei Zhang, and En Zhu. Unified one-step multi-view spectral clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(6):6449–6460, 2022. 1

[29] Luis Von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326, 2004. 6

[30] Xu Wang, Dezhong Peng, Peng Hu, and Yongsheng Sang. Adversarial correlated autoencoder for unsupervised multi-view representation learning. *Knowledge-Based Systems*, 168:109–120, 2019. 1

[31] Jie Wen, Chengliang Liu, Shijie Deng, Yicheng Liu, Lunke Fei, Ke Yan, and Yong Xu. Deep double incomplete multi-view multi-label learning with incomplete labels and missing views. *IEEE transactions on neural networks and learning systems*, 2023. 1, 2, 6

[32] Mike Wu and Noah Goodman. Multimodal generative models for scalable weakly-supervised learning. *Advances in neural information processing systems*, 31, 2018. 3

[33] Xuan Wu, Qing-Guo Chen, Yao Hu, Dengbao Wang, Xiaodong Chang, Xiaobo Wang, and Min-Ling Zhang. Multi-view multi-label learning with view-specific information extraction. In *IJCAI*, pages 3884–3890, 2019. 1, 2

[34] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018. 5

[35] Xiaoqiang Yan, Shizhe Hu, Yiqiao Mao, Yangdong Ye, and Hui Yu. Deep multi-view learning methods: A review. *Neurocomputing*, 448:106–129, 2021. 1

[36] Jun Yin and Shiliang Sun. Incomplete multi-view clustering with reconstructed views. *IEEE Transactions on Knowledge and Data Engineering*, 35(3):2671–2682, 2021. 1

[37] Jun Yin and Wentao Zhang. Multi-view multi-label learning with double orders manifold preserving. *Applied Intelligence*, 53(12):14703–14716, 2023. 1

[38] Jun Yin, Shiliang Sun, Lai Wei, and Pei Wang. Discriminatively fuzzy multi-view k-means clustering with local structure preserving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 16478–16485, 2024. 1

[39] Honglin Yuan, Shiyun Lai, Xingfeng Li, Jian Dai, Yuan Sun, and Zhenwen Ren. Robust prototype completion for incomplete multi-view clustering. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 10402–10411, 2024. 1

[40] Dawei Zhao, Qingwei Gao, Yixiang Lu, Dong Sun, and Yusheng Cheng. Consistency and diversity neural network multi-view multi-label learning. *Knowledge-Based Systems*, 218:106841, 2021. 2, 6

[41] Dawei Zhao, Qingwei Gao, Yixiang Lu, Dong Sun, and Yusheng Cheng. Consistency and diversity neural network multi-view multi-label learning. *Knowledge-Based Systems*, 218:106841, 2021. 1

[42] Yue Zhu, James T Kwok, and Zhi-Hua Zhou. Multi-label learning with global and local label correlation. *IEEE Transactions on Knowledge and Data Engineering*, 30(6):1081–1094, 2017. 1