Adversarial Incomplete Multi-view Clustering

Cai Xu, Ziyu Guan*, Wei Zhao, Hongchang Wu, Yunfei Niu and **Beilei Ling**
State Key Lab of ISN, School of Computer Science and Technology, Xidian University
{cxu_3@stu., zyguan@, ywzhao@mail., hcwu@stu., yfniu@stu., blling@stu.}xidian.edu.cn

Abstract

Multi-view clustering aims to leverage information from multiple views to improve clustering. Most previous works assumed that each view has complete data. However, in real-world datasets, it is often the case that a view may contain some missing data, resulting in the incomplete multi-view clustering problem. Previous methods for this problem have at least one of the following drawbacks: (1) employing shallow models, which cannot well handle the dependence and discrepancy among different views; (2) ignoring the hidden information of the missing data; (3) dedicated to the two-view case. To eliminate all these drawbacks, in this work we present an Adversarial Incomplete Multi-view Clustering (AIMC) method. Unlike most existing methods which only learn a new representation with existing views, AIMC seeks the common latent space of multi-view data and performs missing data inference simultaneously. In particular, the element-wise reconstruction and the generative adversarial network (GAN) are integrated to infer the missing data. They aim to capture overall structure and get a deeper semantic understanding respectively. Moreover, an aligned clustering loss is designed to obtain a better clustering structure. Experiments conducted on three datasets show that AIMC performs well and outperforms baseline methods.

1 Introduction

Many real-world datasets contain multiple kinds of features. Such datasets are called multi-view data. For example, photos shared on an online social network have both visual features and tag descriptions. Different views exhibit consistency and complementary properties of the same data, leading to extensive research on multi-view learning. Synthesizing multi-view features leads to a more comprehensive description of the data, which could benefit many tasks such as classification [Xu *et al.*, 2018a] and clustering [De Sa *et al.*, 2010]. This work is concerned with multi-view clustering, which integrates multiple views to help identify essential grouping structure in an unsupervised manner.

Most of the previous studies on multi-view clustering always assume that all of the views are complete. However, in real-life cases some views could be missing for some data instances. For instance, in Twitter, tweets can contain text, images and videos, but only a part of them contain all the three views. As a result, the lack of partial views makes most multi-view clustering methods inevitably degenerate or even fail. In this paper, we are focused on this Incomplete Multi-view Clustering (IMC) problem.

The prevalent solutions to the IMC problem mainly depend on Non-negative Matrix Factorization (NMF) [Li *et al.*, 2014; Zhao *et al.*, 2016; Xu *et al.*, 2018b]. NMF based methods learn a common latent space for complete instances and private latent representations for incomplete instances. Nevertheless, they cannot be easily extended to handle more than two incomplete views. Weighted NMF based approaches [Shao *et al.*, 2015; Hu and Chen, 2018; 2019] first fill in the missing values by average feature values or matrix completion methods, and then handle the problem with the help of weighted NMF by giving filled data lower weights than original data. However, such a simple padding scheme is invalid when the missing ratio is large. Meanwhile, these methods cannot fully capture the hidden information of the missing data for consensus representation learning because the missing data is not well recovered.

Recently, some IMC methods considering missing data inference have been proposed. In [Wen *et al.*, 2019], Wen *et al.* proposed to learn the missing views of instances jointly with the NMF framework. However, this factorization model corresponds to a shallow projection of the data and may not well handle the dependence and discrepancy among different views. Wang *et al.* proposed a deep IMC method [Wang *et al.*, 2018] based on Generative Adversarial Network (GAN) [Goodfellow *et al.*, 2014]. It uses one view to generate the missing data of the other view by Cycle GAN [Zhu *et al.*, 2017]. This method can only handle two-view data because the cycle generation framework is dedicated to the two-view case.

To eliminate the above limitations and drawbacks, we propose a new IMC framework, named Adversarial Incomplete Multi-view Clustering (AIMC). As shown in Figure 1, the encoders learn the aligned subspaces $\mathbf{z}^{(v)}$'s of multiple views,

*Corresponding author

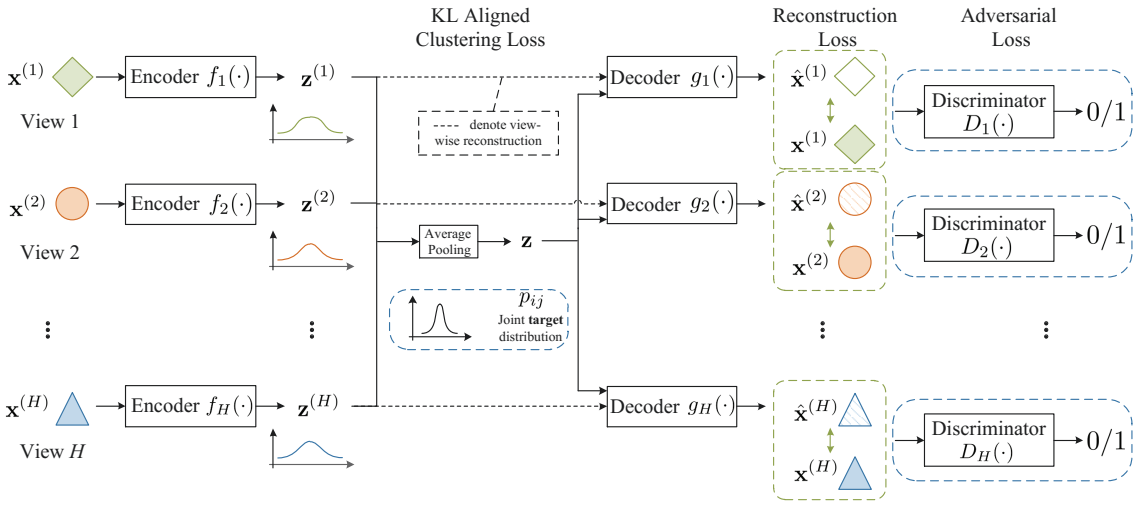


Figure 1: Illustration of AIMC. The encoders learn the aligned subspaces $\mathbf{z}^{(v)}$'s of multiple views. The common latent subspace $\mathbf{z}$ is obtained via average pooling. The decoders use an element-wise reconstruction loss to capture good high-level features and inference of missing data. The additional discriminators are used to distinguish the original data $\mathbf{x}^{(v)}$'s and reconstructed data $\hat{\mathbf{x}}^{(v)}$'s. The decoders will get a deeper semantic understanding when the discriminators cannot distinguish them.

where we require the same dimension of the subspaces to represent the same high-level feature. We use average pooling to learn common representation $\mathbf{z}$, which synthesizes the consistent and complementary information from all the views. Then for each view v, both $\mathbf{z}^{(v)}$ and $\mathbf{z}$ are used separately to reconstruct $\mathbf{x}^{(v)}$ via the same decoder g_v. The motivation of this design is twofold: (1) We need to align the subspaces of $\mathbf{z}^{(v)}$'s through $\mathbf{z}$ by forcing them to be similar. The naive idea of minimizing the distance between each $\mathbf{z}^{(v)}$ and $\mathbf{z}$ would force one view to exhibit complementary high-level features in other views. By putting them in the same role for reconstructing $\mathbf{x}^{(v)}$, we implicitly force them to be similar. (2) When view v is missing, we can still employ $\mathbf{z}$ for a good inference of $\mathbf{x}^{(v)}$ using consistent information among views. To obtain good high-level features and inference of missing data, we train the framework by three losses: an element-wise reconstruction loss, an adversarial loss and a Kullback-Leibler (KL) aligned clustering loss. The reconstruction loss is standard for high-level subspace learning, but ignores the correlations between data dimensions. Hence, we train additional discriminators D_v to distinguish the original data and reconstructed data. The decoders will get a deeper semantic understanding when the discriminators cannot distinguish them [Pathak *et al.*, 2016], making decoders good generators for missing data inference. The KL aligned clustering loss is imposed on complete data only, which tries to make the latent distributions for all the views consistent and compact cluster-wise. After training, we infer missing data and apply AIMC once more on the generated complete data to obtain the updated common representation $\mathbf{z}$ for clustering.

The major contribution of this work is a novel method for the IMC problem with the consideration of more accurately inferring missing information by the adversarial encoder-decoder pipeline. Meanwhile, the encoders learn high-level common representation via multi-layer non-linear transformations. Another contribution is that we extend the KL aligned clustering loss [Xie *et al.*, 2016] to the multi-view case to simultaneously learn aligned high-level subspaces and capture better clustering structures for all the views. Finally, we empirically evaluate AIMC on three real world datasets and show its superiority over state-of-art baseline methods.

2 Related Work

In the section, we briefly review two lines of related work, incomplete multi-view clustering and unsupervised multi-view deep learning.

2.1 Incomplete Multi-view Clustering (IMC)

Li *et al.* [Li *et al.*, 2014] present the first NMF based IMC method, PVC, which learns common representations for complete instances and private latent representations for incomplete instances with the same basis matrices. Then these common and private representations are used together for clustering. Inspired by this work, [Zhao *et al.*, 2016] integrated PVC and manifold learning to learn the global structure of multi-view data. Nevertheless, these methods can only deal with two-view data, limiting their application scope. Weighted NMF based approaches, such as [Shao *et al.*, 2015; Hu and Chen, 2018; 2019], were proposed to deal with more than two views by filling in the missing data and giving them lower weights. However, they still failed to capture the hidden distribution of missing data. Wen *et al.* [Wen *et al.*, 2019] added an error matrix to compensate the missing data. The original incomplete data matrix and the error matrix were combined to form a completed data matrix in a matrix factorization architecture. The common latent representation and error matrix (i.e. missing data) were jointly optimized. Although this work tried to capture the hidden distribution of missing data, it was still based on matrix factorization which

corresponds to a shallow projection. Such a simple model cannot well handle complex relationships between low-level features of multi-view data.

2.2 Unsupervised Multi-view Deep Learning

Inspired by deep learning [Hinton and Salakhutdinov, 2006], different deep models were proposed recently for multi-view learning. There are three main categories. The first category is based on Canonical Correlation Analysis (CCA) [Hotelling, 1936], which finds linear projections of two views that are maximally correlated. Andrew *et al.* [Andrew *et al.*, 2013] proposed deep extension of CCA (Deep CCA) that simultaneously learned two deep nonlinear mappings of two views. The second one is based on autoencoder. Ngiam *et al.* [Ngiam *et al.*, 2011] explored extracting shared representations by training a two-view deep autoencoder which aimed to best reconstruct the two-view input. Shahroudy *et al.* [Shahroudy *et al.*, 2018] introduced autoencoder-based network to analyze the multi-view (RGB+Deep features) videos. Wang *et al.* [Wang *et al.*, 2015] found that CCA-based approaches tended to promote autoencoder-based approaches and proposed a deep model that combined CCA and autoencoder. Recently, there exist several approaches utlizing GAN [Goodfellow *et al.*, 2014] for multi-view learning such as CycleGAN [Zhu *et al.*, 2017] and StarGAN [Choi *et al.*, 2018]. However, those methods mainly focused on cross-domain data generation. Wang [Wang *et al.*, 2018] proposed Consistent GAN for the two-view IMC problem. It uses one view to generate the missing data of the other view, and then performs clustering on the generated complete data. However, it is dedicate to the two-view case, while our AIMC can be applied to IMC problems with an arbitrary number of views.

3 The Method

In this section, we present Adversarial Incomplete Multi-view Clustering (AIMC) in detail, together with its implementation.

3.1 Notations and Problem Statement

In the incomplete multi-view clustering setting, an instance is characterized by multiple views and may have complete or partial views as shown in Figure 2. Suppose we are given a dataset with H views, N complete instances, and $\widetilde{N}$ incomplete instances. We use $\mathbf{x}_n^{(v)}/\widetilde{\mathbf{x}}_{\widetilde{n}}^{(v)} \in \mathbb{R}^{d_v}(v = 1, \ldots, H)$ to denote the feature vector for the v-th view of the n-th/$\widetilde{n}$-th instance in the set of complete/incomplete instances, respectively, where d_v is the dimensionality of the v-th view. An indicator matrix $\mathbf{M} \in \mathbb{R}^{H \times \widetilde{N}}$ for incomplete instances is defined as:

$$\mathbf{M}_{v\widetilde{n}} = \begin{cases} 1 \text{ if the } \widetilde{n}\text{-th instance has the } v\text{-th view} \\ 0 \text{ otherwise} \end{cases} \quad (1)$$

where each column of $\mathbf{M}$ encodes the view status (present/absent) for the corresponding instance. Since there are missing views, for every $\widetilde{n}$ we have $\sum_{v=1}^{H} \mathbf{M}_{v\widetilde{n}} < H$.

Our goal is to group all the $N + \widetilde{N}$ instances into K clusters.

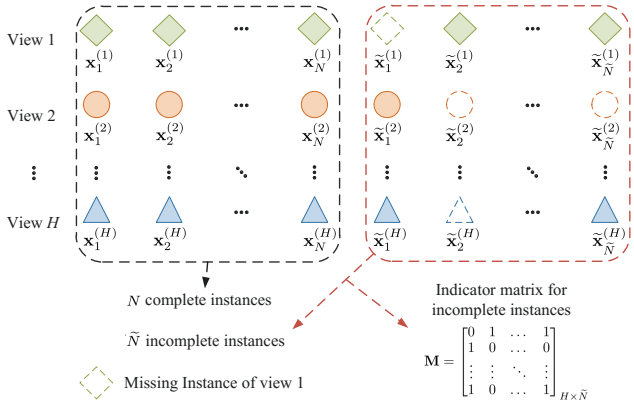


Figure 2: Notations for incomplete multi-view data.

3.2 Adversarial Encoder-Decoder Pipeline

As shown in Figure 1, the overall architecture is an encoder-decoder pipeline which consists of H encoders $\{f_v\}_{v=1}^{H}$, H decoders $\{g_v\}_{v=1}^{H}$ and H discriminators $\{D_v\}_{v=1}^{H}$. The encoders aim to obtain high-level latent representations and the common latent representation. The decoders take these representations to reconstruct the original data and also generate missing views of incomplete instances. The discriminators are added to assist in training this model for better inferring missing data. After obtaining the predicted missing data, the encoders use the generated complete data to calculate the common representation, which can capture the hidden information of the missing data. Details regarding each component will be elaborated as below.

Encoder

The v-th encoder f_v is dedicated to learning aligned subspace $\mathbf{z}^{(v)} \in \mathbb{R}^c$ for all the input data of the v-th view. The same latent space dimensionality c is shared among the H views. The latent representations $\{\mathbf{z}^{(v)}\}_{v=1}^{H}$ need to be aligned. To this end, we add the average pooling layer, which averages all the latent representations for different views, to get the common representation $\mathbf{z}$. One feature in the common subspace will be activated when it is activated in at least one view. The activation intensity depends on the number of activated views. I.e., consistent features tend to have higher intensity than complementary features. To achieve better subspace alignment and inference of missing data, we put each $\mathbf{z}^{(v)}$ and $\mathbf{z}$ in the same role (i.e. input to the decoder) for reconstructing the corresponding $\mathbf{x}^{(v)}$.

Decoder

We now discuss the second half of our pipeline, the decoders $\{g_v\}_{v=1}^{H}$, which reconstruct the original data and generate the missing views of incomplete instances using the common latent representations. $\hat{\mathbf{x}}^{(v)}$, the output of g_v, is the reconstruction of the original data $\mathbf{x}^{(v)}$ (or the generated missing data). The traditional element-wise mean squared error is utilized to evaluate reconstruction. Nevertheless, this objective does not explicitly try to capture the correlations between data dimensions which could be important for describing the underlying distribution of a view. To capture such correlations

for better missing data inference, we employ the discriminators $\{D_v\}_{v=1}^{H}$. For each view, the original data $\mathbf{x}^{(v)}$ and the reconstructed data $\hat{\mathbf{x}}^{(v)}$ with their labels (True and False respectively) are used to train the discriminator D_v. Each pair (g_v, D_v) forms a adversarial relationship, leading to a deeper semantic understanding of the data, and consequently better generation of missing data. This framework explicitly supplements the missing data to learn a more complete common representation for incomplete multi-view data.

3.3 Loss Function

We train our AIMC model by the original complete instances $\{\mathbf{x}_n^{(v)}\}$ and incomplete instances $\{\widetilde{\mathbf{x}}_{\widetilde{n}}^{(v)}\}$. The goal is to learn common representation for multi-view data and to precisely reconstruct the original data simultaneously. We now describe different components of our loss function.

Reconstruction Loss

We use L_2 distance as our element-wise reconstruction loss function, $\mathcal{L}_{\mathcal{R}}$:

$$\mathcal{L}_{\mathcal{R}} = \sum_{v=1}^{H}\left(\sum_{n=1}^{N}\left(\left\|\mathbf{x}_n^{(v)} - g_v(\mathbf{z}_n)\right\|_2 + \left\|\mathbf{x}_n^{(v)} - g_v(\mathbf{z}_n^{(v)})\right\|_2\right) + \sum_{\mathbf{M}_{v\widetilde{n}}=1}\sum_{\widetilde{n}=1}^{\widetilde{N}}\left(\left\|\widetilde{\mathbf{x}}_{\widetilde{n}}^{(v)} - g_v(\widetilde{\mathbf{z}}_{\widetilde{n}})\right\|_2 + \left\|\widetilde{\mathbf{x}}_{\widetilde{n}}^{(v)} - g_v(\widetilde{\mathbf{z}}_{\widetilde{n}}^{(v)})\right\|_2\right)\right) \tag{2}$$

where $\mathbf{z}_n^{(v)}/\widetilde{\mathbf{z}}_{\widetilde{n}}^{(v)}$ is the latent representation of the v-th view and the n-th/$\widetilde{n}$-th instance of complete/incomplete instances. $\mathbf{z}_n/\widetilde{\mathbf{z}}_{\widetilde{n}}$ is the common latent representation of the complete/incomplete instances. These latent vectors are calculated as in Eq. (3).

$$\mathbf{z}_n^{(v)} = f_v(\mathbf{x}_n^{(v)}), \mathbf{z}_n = \frac{1}{H}\sum_{v=1}^{H}\mathbf{z}_n^{(v)} \tag{3a}$$

$$\widetilde{\mathbf{z}}_{\widetilde{n}}^{(v)} = f_v(\widetilde{\mathbf{x}}_{\widetilde{n}}^{(v)}), \widetilde{\mathbf{z}}_{\widetilde{n}} = \frac{1}{\sum_{v=1}^{H}\mathbf{M}_{v\widetilde{n}}}\sum_{v=1}^{H}\mathbf{M}_{v\widetilde{n}}\widetilde{\mathbf{z}}_{\widetilde{n}}^{(v)} \tag{3b}$$

The first term of $\mathcal{L}_{\mathcal{R}}$ is the reconstruction criterion utilizing the common latent representation. It aims to learn the bidirectional mapping between the original data space and the common embedding space. The second term denotes view-specific reconstruction, which implicitly forces the latent representation of each view close to the common representation by feeding them into the same decoder network. The last two terms are designed in the same notion for the incomplete part. However, the L_2 loss focuses on each data dimension separately while ignores the correlations between data dimensions. In the next, we detail the adversarial loss in our model to alleviate this problem.

Adversarial Loss

The adversarial loss is based on idea of GAN [Goodfellow *et al.*, 2014]. Traditional GAN consist of two parts, generator G and discriminator D. G maps the data $\mathbf{w}$ from noise distribution $P_{\mathbf{w}}$ to data distribution P_{data}. The training procedure is a two-player game where D tries to distinguish the ground truth data and the output of G, while G tries to confuse D by generating data as "real" as possible. The objective for GAN can be formulated as follows:

$$\min_G \max_D E_{\mathbf{x}\sim P_{data}}[\log D(\mathbf{x})] + E_{\mathbf{w}\sim P_{\mathbf{w}}}[\log(1 - D(G(\mathbf{w})))] \tag{4}$$

In our model, the problem of multi-view reconstruction is modeled in the adversarial setting to learn the conditional distributions $P_{data^{(v)}}(\mathbf{x}^{(v)}|\star)$, $P_{data^{(v)}}(\widetilde{\mathbf{x}}^{(v)}|\star)$, where $\star$ can be common latent representations $\mathbf{z}$, $\widetilde{\mathbf{z}}$, or view-specific latent representations $\mathbf{z}^{(v)}$, $\widetilde{\mathbf{z}}^{(v)}$. Specifically, we adapt the traditional GAN framework to multi-view reconstruction by treating decoders as generators, i.e., $g \triangleq G$. The adversarial loss for reconstructing complete data $\mathbf{x}^{(v)}$ by common latent representation $\mathbf{z}$ is

$$\min_{\{g_v\}_{v=1}^{H}} \max_{\{D_v\}_{v=1}^{H}} \sum_{v=1}^{H}\left(E_{\mathbf{x}^{(v)}\sim P_{data^{(v)}}}[\log D_v(\mathbf{x}^{(v)})] + E_{\mathbf{z}\sim P_{\mathbf{z}}}[\log(1 - D_v(g_v(\mathbf{z})))]\right) \tag{5}$$

Similarly, we consider the adversarial loss for reconstructing complete data $\mathbf{x}^{(v)}$ by view-specific latent representation $\mathbf{z}^{(v)}$. For incomplete data $\widetilde{\mathbf{x}}^{(v)}$, the adversarial loss is designed in the same manner. We omit the details due to the forms are similar to Eq. (5). Hence the overall adversarial loss for our model, $\mathcal{L}_{\mathcal{A}}$, is the sum of the above four terms.

The decoders are trained to reconstruct the original data ($\mathbf{x}^{(v)}$ and $\widetilde{\mathbf{x}}^{(v)}$) through the L_2 reconstruction loss $\mathcal{L}_{\mathcal{R}}$ and the adversarial loss $\mathcal{L}_{\mathcal{A}}$. The reconstruction loss tries to reproduce original data as accurately as possible, while the adversarial loss tries to capture the underlying data distribution. Combining these two loss functions would lead to robust encoders and decoders for high-level representation projection and missing data inference.

KL Aligned Clustering Loss

According to the aforementioned analysis, we use average pooling to get the common representation of multiple views. In order to further polish the latent representations for multi-view data, we propose the KL aligned clustering loss on complete instances, which minimizes KL divergence between the clustering distributions calculated by $\mathbf{z}^{(v)}$'s and emphasized target distribution calculated by $\mathbf{z}$. Following [Xie *et al.*, 2016], we model the probability of a complete data instance n being assigned to cluster k using the Student's t-distribution [Maaten and Hinton, 2008], producing a distribution for each view:

$$q_{nk}^{(v)} = \frac{(1 + \|\mathbf{z}_n^{(v)} - \mu_k^{(v)}\|_2^2/\delta)^{-\frac{\delta+1}{2}}}{\sum_{k'}(1 + \|\mathbf{z}_n^{(v)} - \mu_{k'}^{(v)}\|_2^2/\delta)^{-\frac{\delta+1}{2}}} \tag{6}$$

where δ is the number of degrees of freedom for the Student's t-distribution. Following Xie *et al.*, we set δ to 1. $\mu_k^{(v)}$ is the clustering centroid of the k-th cluster in the latent space for the v-th view. The centroids are initialized by K-means and updated by Stochastic Gradient Descent (SGD) [Xie *et al.*, 2016]. $q_{nk}^{(v)}$ can be interpreted as the probability of assigning

instance n to cluster k (i.e., a soft assignment) for the v-th view.

Similarly, we calculate the soft assignment distribution q_{nk} for the common latent representation $\mathbf{z}_n$. In order to improve cluster compactness, we put more emphasis on data points assigned with high confidence by calculate the emphasized target distribution p_{nk} as follows:

$$p_{nk} = \frac{q_{nk}^2/f_k}{\sum_{k'} q_{nk'}^2/f_{k'}} \tag{7}$$

where $f_k = \sum_n q_{nk}$ are soft cluster frequencies. It first squares q_{nk} and then normalizes it by frequency per cluster. Hence, high probabilities will be emphasized. We define the objective function as a KL divergence between the soft assignments $q_{nk}^{(v)}$ and the emphasized target distribution p_{nk} as follows:

$$\mathcal{L}_{\mathcal{D}} = \sum_v KL(P\|Q^{(v)}) = \sum_{v,n,k} p_{nk} \log \frac{p_{nk}}{q_{nk}^{(v)}} \tag{8}$$

The distribution gap of different views will be reduced gradually by optimizing $\mathcal{L}_{\mathcal{D}}$, leading to more compatible latent representations across different views. Meanwhile, the clusters are iteratively refined, which helps obtain a better clustering structure of the common representation $\mathbf{z}$.

Joint Loss

By synthesizing the above objectives, the overall optimization problem of AIMC is formulated as

$$\min_{\{f_v,g_v,\mu_k^{(v)}\}_{v=1}^H} \max_{\{D_v\}_{v=1}^H} \mathcal{L} = \mathcal{L}_{\mathcal{R}} + \alpha\mathcal{L}_{\mathcal{A}} + \beta\mathcal{L}_{\mathcal{D}} \tag{9}$$

where $\alpha, \beta > 0$ are hyper-parameters.

3.4 Implementation

The whole process of using AIMC for clustering is summarized as below.

Step 1: Training AIMC model. We use the original complete instances $\{\mathbf{x}_n^{(v)}\}$ and incomplete instances $\{\widetilde{\mathbf{x}}_{\widetilde{n}}^{(v)}\}$ to train AIMC by optimizing Eq. (9). We use the adaptive moment (Adam) optimizer to train our model and set the learning rate to 0.0001. Our model is implemented by PyTorch and run on Ubuntu Linux 16.04.

Step 2: Generating missing data. In this step, we put the common representation $\widetilde{\mathbf{z}}_{\widetilde{n}}$ calculated through Eq.(3) into the trained decoder network $\{g_v\}_{v=1}^H$ to generate the missing views of incomplete instances:

$$\hat{\mathbf{x}}_{\widetilde{n}}^{(v)} = g_v(\widetilde{\mathbf{z}}_{\widetilde{n}}) \tag{10}$$

Step 3: Calculating the common representation using the generated complete data. We combine the original incomplete instances $\{\widetilde{\mathbf{x}}_{\widetilde{n}}^{(v)}\}$ and the generated missing data $\{\hat{\mathbf{x}}_{\widetilde{n}}^{(v)}\}$ to compute the common representations $\{\hat{\mathbf{z}}_{\widetilde{n}}^{(v)}\}$ for the incomplete part:

$$\hat{\mathbf{z}}_{\widetilde{n}}^{(v)} = \frac{1}{H}\sum_{v=1}^{H}\Big(\mathbf{M}_{v\widetilde{n}} f_v(\widetilde{\mathbf{x}}_{\widetilde{n}}^{(v)}) + (1-\mathbf{M}_{v\widetilde{n}}) f_v(\hat{\mathbf{x}}_{\widetilde{n}}^{(v)})\Big) \tag{11}$$

In this way, the model can capture the hidden information of the missing data. Finally, we apply K-means on the learned common representations $\{\{\mathbf{z}_n\}_{n=1}^N, \{\hat{\mathbf{z}}_{\widetilde{n}}\}_{\widetilde{n}=1}^{\widetilde{N}}\}$ to quantitatively test AIMC's performance on data clustering.

4 Experiments

We evaluate the clustering performance of AIMC on three datasets. Important statistics are summarized in Table 1 and a brief introduction of the datasets is presented below.

Reuters [Amini *et al.*, 2009] consists of 111740 documents written in 5 languages of 6 categories represented as TFIDF vectors. We utilize documents written in English, French and German as three views. For each category, we randomly choose 500 documents. Totally 3000 documents are used. **BDGP** [Cai *et al.*, 2012] contains 2500 instances about drosophila embryos of 5 categories. We utilize 1000D lateral visual vector, 500D dorsal visual vector and 79D texture feature vector as three views. We use all instances in our experiment. **Youtube** [Omid *et al.*, 2013] contains 92457 instances from 31 categories, each described by 13 feature types. We sample 500 instances from each category. We select 512D vision feature, 2000D audio feature and 1000D text feature as three views.

We compare AIMC with the following baseline algorithms. Due to the missing data, we cannot directly perform the algorithms that are only applicable on complete data. Following [Hu and Chen, 2018], we first fill the missing data with the average feature values for each view, and then perform these algorithms. **Best Single View (BSV)** clusters on each view, and reports the best result. **Concat** concatenates feature vectors of different views to apply K-means. **Deep Multi-view Semi-NMF (DMSNMF)** [Zhao *et al.*, 2017] is an deep method for complete multi-view data clustering. **Doubly Aligned Incomplete Multi-view Clustering (DAIMC)** [Hu and Chen, 2018] is a weighted NMF based IMC method. **Unified Embedding Alignment Framework (UEAF)** [Wen *et al.*, 2019] is the state-of-the-art IMC method with missing data inference.

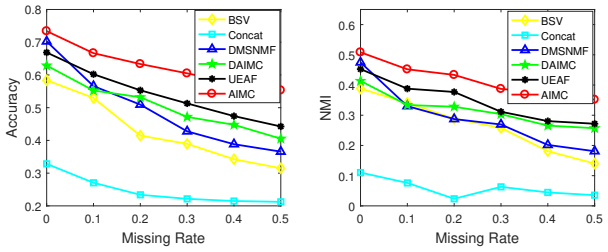
As in [Hu and Chen, 2018], we randomly select $\widetilde{N}$ instances as incomplete data and randomly remove some views from each of them. The Missing Rate (MR) $\frac{\widetilde{N}}{N+\widetilde{N}}$ is from 0 to 0.5. The clustering *Accuracy* and *Normalized Mutual Information* (NMI) are used to evaluate clustering performance [Shao *et al.*, 2015]. All of the hyper-parameters of these methods are selected through grid-search.

4.1 Results

Figure 3, 4, 5 show the clustering performance of AIMC and baseline methods. First, the performance of IMC methods drop more slowly than others with the increase of missing

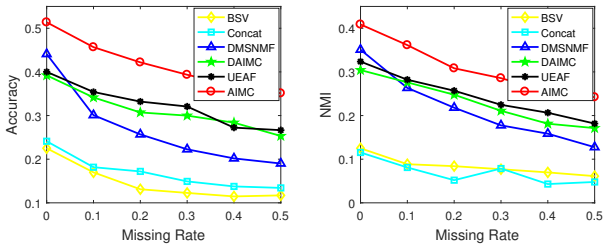
Dataset	Size	# of categories	Dimensionality
Reuters	3000	6	21531/24893/34279
BDGP	2500	5	79/500/1000
Youtube	15500	31	512/1000/2000

Table 1: Dataset summary.



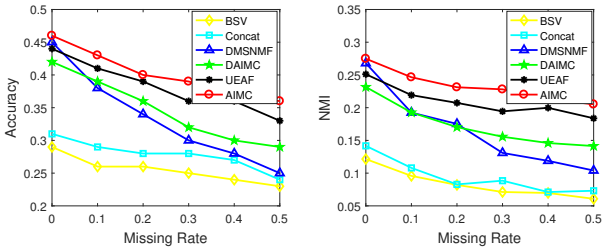
(a) Accuracy (b) NMI

Figure 3: Mean Accuracy and NMI on BDGP dataset



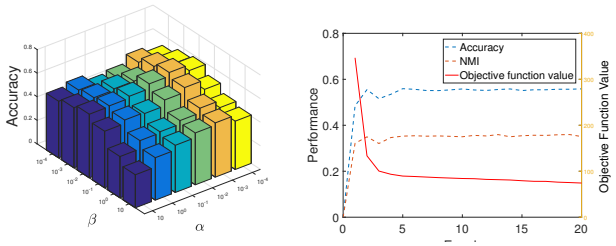
(a) Accuracy (b) NMI

Figure 4: Mean Accuracy and NMI on Youtube dataset



(a) Accuracy (b) NMI

Figure 5: Mean Accuracy and NMI on Reuters dataset



(a) Parameter analysis (b) Convergence analysis

Figure 6: Parameter and convergence study on the BDGP dataset

rate. We can see that IMC methods are more effective for the IMC problem. Second, when the missing rate is 0, deep methods (AIMC, DMSNMF) outperform shallow ones. This is intuitive since the deep model could better handle the dependence and discrepancy among different views by hierarchical modeling. Thirdly, AIMC and UEAF consistently outperforms DAIMC on the two datasets, which indicates that capturing the hidden distribution of the missing data can greatly facilitate the IMC. We use t-test with significance level 0.05 to test the significance of performance difference. Results show that AIMC significantly outperforms all the baseline methods.

4.2 Analysis

In this subsection, we will analyze AIMC from two perspectives, i.e., parameter setting and convergence analysis.

The AIMC method contains two hyper-parameters: α and β. Here we explore their impact to performance on the BDGP dataset. We set missing rate as 0.2, and report the accuracy by varying α and β in the set $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$. Fig 6(a) shows the results. We find a general pattern: the performance curves first go up and then go down when increasing the hyper-parameters. This indicates the discriminators and KL aligned clustering term are useful for IMC. Based on the results, we set $\alpha = 0.001$, $\beta = 0.01$ in other experiments. Note that we follow the experimental methodology employed by previous works [Li *et al.*, 2014; Zhao *et al.*, 2016] that set hyper-parameters by looking at performance on the BDGP test set. The results for BDGP could be overly optimistic. However, the hyper-parameters of AIMC and all the baselines are tuned in exactly the same way, so it is a fair comparison. Moreover, AIMC was run with the same hyper-parameter setting on the three datasets and achieved good performance on all of them, which means AIMC is not very sensitive with respect to the hyper-parameters.

Fig 6(b) shows the curves of loss value, Accuracy and NMI against the number of epochs for AIMC. We set the missing rate as 0.5 on BDGP dataset. We find at the beginning the loss value drops and the performance increases rapidly. The optimization procedure of AIMC typically converges in around 5 epochs.

5 Conclusion

In this paper, we proposed a Adversarial Incomplete Multi-view Clustering (AIMC) method for IMC with an arbitrary number of views. AIMC tries to seek a common high-level representation for incomplete multi-view data. It also tries to capture hidden information of the missing data by missing data inference via the element-wise reconstruction and the GAN. Experimental results on three real-world datasets confirmed the effectiveness of AIMC compared to state-of-the-art incomplete multi-view clustering methods.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grant Nos. 61672409, 61522206, 61876144, 61876145), the Major Basic Research Project of Shaanxi Province (Grant No. 2017ZDJC-31), Shaanxi Province Science Fund for Distinguished Young Scholars (Grant No. 2018JC-016), and the Fundamental Research Funds for the Central Universities (Grant Nos. JB190301, JB190305)

References

[Amini *et al.*, 2009] Massih Amini, Nicolas Usunier, and Cyril Goutte. Learning from multiple partially observed views-an application to multilingual text categorization. In *NIPS*, pages 28–36, 2009.

[Andrew *et al.*, 2013] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *ICML*, pages 1247–1255, 2013.

[Cai *et al.*, 2012] Xiao Cai, Hua Wang, Heng Huang, and Chris Ding. Joint stage recognition and anatomical annotation of drosophila gene expression patterns. *Bioinformatics*, 28:i16–i24, 2012.

[Choi *et al.*, 2018] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, pages 8789–8797, 2018.

[De Sa *et al.*, 2010] Virginia R De Sa, Patrick W Gallagher, Joshua M Lewis, and Vicente L Malave. Multi-view kernel construction. *Machine learning*, 79(1-2):47–71, 2010.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[Hinton and Salakhutdinov, 2006] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[Hotelling, 1936] Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

[Hu and Chen, 2018] Menglei Hu and Songcan Chen. Doubly aligned incomplete multi-view clustering. In *IJCAI*, pages 2262–2268, 2018.

[Hu and Chen, 2019] Menglei Hu and Songcan Chen. One-pass incomplete multi-view clustering. In *AAAI*, 2019.

[Li *et al.*, 2014] Shaoyuan Li, Zhihua Zhou, and Jiang Yuan. Partial multi-view clustering. In *AAAI*, pages 1968–1974, 2014.

[Maaten and Hinton, 2008] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9:2579–2605, 2008.

[Ngiam *et al.*, 2011] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *ICML*, pages 689–696, 2011.

[Omid *et al.*, 2013] Madani Omid, Georg Manfred, and A. Ross David. On using nearly-independent feature families for high precision and confidence. *Machine Learning*, 92:457–477, 2013. published online 30 May 2013.

[Pathak *et al.*, 2016] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, pages 2536–2544, 2016.

[Shahroudy *et al.*, 2018] Amir Shahroudy, Tian-Tsong Ng, Yihong Gong, and Gang Wang. Deep multimodal feature analysis for action recognition in rgb+ d videos. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(5):1045–1058, 2018.

[Shao *et al.*, 2015] Weixiang Shao, Lifang He, and S Yu Philip. Multiple incomplete views clustering via weighted nonnegative matrix factorization with $l_{2,1}$ regularization. In *ECML PKDD*, pages 318–334, 2015.

[Wang *et al.*, 2015] Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. On deep multi-view representation learning. In *ICML*, pages 1083–1092, 2015.

[Wang *et al.*, 2018] Qianqian Wang, Zhengming Ding, Zhiqiang Tao, Quanxue Gao, and Yun Fu. Partial multi-view clustering via consistent gan. In *ICDM*, pages 1290–1295, 2018.

[Wen *et al.*, 2019] Jie Wen, Zheng Zhang, Yong Xu, Bob Zhang, Fei Lunke, and Hong Liu. Unified embedding alignment with missing views inferring for incomplete multi-view clustering. In *AAAI*, 2019.

[Xie *et al.*, 2016] Jun-Yuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, pages 478–487, 2016.

[Xu *et al.*, 2018a] Cai Xu, Ziyu Guan, Wei Zhao, Yunfei Niu, Quan Wang, and Zhiheng Wang. Deep multi-view concept learning. In *IJCAI*, pages 2898–2904, 2018.

[Xu *et al.*, 2018b] Nan Xu, Yanqing Guo, Xin Zheng, Qianyu Wang, and Xiangyang Luo. Partial multi-view subspace clustering. In *MM*, pages 1794–1801, 2018.

[Zhao *et al.*, 2016] Handong Zhao, Hongfu Liu, and Yun Fu. Incomplete multi-modal visual data grouping. In *IJCAI*, pages 2392–2398, 2016.

[Zhao *et al.*, 2017] Handong Zhao, Zhengming Ding, and Yun Fu. Multi-view clustering via deep matrix factorization. In *AAAI*, pages 2921–2927, 2017.

[Zhu *et al.*, 2017] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint*, 2017.