Multi-view Adversarially Learned Inference for Cross-domain Joint Distribution Matching

Changying Du
360 Search Lab
Beijing 100015, China
ducyatict@gmail.com

Changde Du*
Institute of Automation, CAS
Beijing 100190, China
duchangde2016@ia.ac.cn

Xingyu Xie
Nanjing Univ. of Aeronautics and Astronautics, Nanjing, China
nuaaxing@gmail.com

Chen Zhang
360 Search Lab
Beijing 100015, China
zhangchen1@360.cn

Hao Wang†
360 Search Lab
Beijing 100015, China
cashenry@126.com

ABSTRACT

Many important data mining problems can be modeled as learning a (bidirectional) multidimensional mapping between two data domains. Based on the generative adversarial networks (GANs), particularly conditional ones, cross-domain joint distribution matching is an increasingly popular kind of methods addressing such problems. Though significant advances have been achieved, there are still two main disadvantages of existing models, i.e., the requirement of large amount of paired training samples and the notorious instability of training. In this paper, we propose a multi-view adversarially learned inference (ALI) model, termed as MALI, to address these issues. Unlike the common practice of learning direct domain mappings, our model relies on shared latent representations of both domains and can generate arbitrary number of paired faking samples, benefiting from which usually very few paired samples (together with sufficient unpaired ones) is enough for learning good mappings. Extending the vanilla ALI model, we design novel discriminators to judge the quality of generated samples (both paired and unpaired), and provide theoretical analysis of our new formulation. Experiments on image translation, image-to-attribute and attribute-to-image generation tasks demonstrate that our semi-supervised learning framework yields significant performance improvements over existing ones. Results on cross-modality retrieval show that our latent space based method can achieve competitive similarity search performance in relative fast speed, compared to those methods that compute similarities in the high-dimensional data space.

KEYWORDS

Adversarially Learned Inference, Generative Adversarial Networks, Joint Distribution Matching, Multi-view Learning, Conditional Generation, Cross-modality Retrieval

ACM Reference Format:
Changying Du, Changde Du, Xingyu Xie, Chen Zhang, and Hao Wang. 2018. Multi-view Adversarially Learned Inference for Cross-domain Joint Distribution Matching. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3219819.3219957

1 INTRODUCTION

Many important data mining problems can be modeled as learning a (bidirectional) multidimensional mapping between two data domains (views), e.g., (1) in image translation/editing [8, 16, 21, 30, 32], the goal is to transform an input image into an output one with desired properties; (2) in image annotating/attribute predicting [4, 5, 28, 29], one aims to extract meaningful semantic labels as many as possible from a given image; (3) in attribute-conditional image generation [5], the task is to map discrete word space into pixel space; and (4) in cross-modality retrieval one has to transform the queries or/and the items into the same modality or a shared latent space. Traditionally, such problems are modeled with a large number of paired data samples from both domains, and the sufficiency of paired data usually is critical to the domain mapping performance.

Based on the generative adversarial networks (GANs) [6], a powerful leaning paradigm that can match true data distribution by transforming random noise vectors, recent studies have shown that purely unsupervised joint distribution matching models, such as CycleGAN [9, 32], DualGAN [30], and CoGAN [17] can yield promising results, without any paired data samples reflecting the relationship of both domains. While these achievements are clearly significant, they typically assume the interested data domains have strong structural

*He is also with the University of Chinese Academy of Sciences (CAS).
†Corresponding author

KDD '18, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219957

similarity (e.g., the edges-to-shoes image translation problem [8]), so that the cycle-consistency principle and the problem-specific (weight-sharing) generative networks can be used. So far, developing general-purpose unsupervised domain mapping framework still remains difficult, because there exist infinitely many possible mapping functions that satisfy the requirement to map a sample from one domain to another. A more practical choice is to consider semi-supervised learning to alleviate this nonidentifiability issue, where we can utilize the few paired data samples and the usually abundant unpaired samples in each domain simultaneously. Two state-of-the-art methods of this kind are the triangle GAN ($\triangle$-GAN) [5] and the ALICE model [14], which can be considered as combinations of the conditional GAN [8, 20] and the Adversarially Learned Inference (ALI)/bidirectional GAN (BiGAN) [2, 3]. However, these models only learns bidirectional direct mappings between domains, without considering a shared low-dimensional latent space for both domains. We argue that such a latent space can be beneficial in many ways, e.g., 1) paired faking samples generated from a common latent space may be utilized for improving the semi-supervised model; 2) when cycle-consistency principle is not applicable due to one-to-many mapping relation between domains, e.g., in the attribute-conditional image generation problem, a latent space can be utilized for constructing a smaller cycle; 3) in cross-domain (modality) retrieval we can perform efficient similarity computation in this low-dimensional space; 4) a joint latent space is more feasible for multi-view classification; and 5) some applications may be interested in generating new paired samples rather than mapping one into another;

In this paper, we propose a Multi-view Adversarially Learned Inference model, termed as MALI, to address these issues. Unlike the common practice of learning direct domain mappings, our model relies on shared latent representations of both domains, and can generate arbitrary number of paired faking samples, benefiting from which usually very few paired samples (together with sufficient unpaired ones) is enough for learning good mappings. Extending the vanilla ALI model [3], we design novel discriminators to judge the quality of generated samples (both paired and unpaired), and provide theoretical analysis of our new formulation. Experiments on image translation, image-to-attribute and attribute-to-image generation tasks demonstrate that our semi-supervised learning framework yields significant performance improvements over existing ones. Results on cross-modality retrieval show that our latent space based method can achieve competitive similarity search performance in relative fast speed, compared to those methods that compute similarity in the high-dimensional data space.

2 PRELIMINARIES

2.1 Generative Adversarial Networks

GANs [6] consist of a generator and a discriminator that compete in a two-player minimax game. The generator is learned to map samples from an arbitrary latent distribution to data, while the discriminator tries to distinguish between real and generated samples. The objective can be written as

$$\min_{\theta}\max_{\omega}\mathcal{L}_{\text{GAN}} = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}))}\big[\log D_\omega(\mathbf{x})\big] + \mathbb{E}_{\tilde{\mathbf{x}}\sim p_\theta(\mathbf{x}|\mathbf{z}),\ \mathbf{z}\sim p(\mathbf{z})}\big[\log\big(1-D_\omega(\tilde{\mathbf{x}})\big)\big]$$

where $p(\mathbf{x})$ denotes the empirical distribution of the data, $p(\mathbf{z})$ is usually specified as a simple distribution (e.g., the standard normal), $p_\theta(\mathbf{x}|\mathbf{z})$ is the generator parameterized by θ, and $D_\omega(\cdot)$ is the discriminator parameterized by ω.

2.2 Variational Autoencoder (VAE)

As a powerful unsupervised representation learning framework, VAE [10, 23] defines a latent variable generative model by $p(\mathbf{Z}) = \prod_{i=1}^N p(\mathbf{z}_i)$, $p_\theta(\mathbf{X}|\mathbf{Z}) = \prod_{i=1}^N p_\theta(\mathbf{x}_i|\mathbf{z}_i)$, where θ denotes that the likelihood is parameterized by a decoder Deep Neural Network (DNN), and $\mathbf{Z}\in\mathbb{R}^{K\times N}$ and $\mathbf{X}\in\mathbb{R}^{D\times N}$ denote the latents and the observed data, respectively. Specifying $p_\phi(\mathbf{Z}|\mathbf{X}) = \prod_{i=1}^N p_\phi(\mathbf{z}_i|\mathbf{x}_i)$ as the DNN-parameterized inference model (encoder), VAE seeks to maximize the evidence lower bound (ELBO) w.r.t. ϕ and θ.

It is noted that VAEs for image generation often yields blurry images due to its maximum-likelihood based formulation. While GANs can generate more realistic samples, they are more difficult to train due to its well known instability and mode collapse problems. Fortunately, it is shown that GANs and VAEs can be combined together to get more stable and compelling models [7, 11, 19, 24].

2.3 Adversarially Learned Inference (ALI)

ALI/BiGAN [2, 3] aims to learn a bidirectional model that can produce high-quality samples for both the latent and data spaces. Different from VAE, it casts the learning of such a bidirectional model in a GAN-like adversarial framework. Specifically, a discriminator is trained to distinguish between two joint distributions: that of the real data sample and its inferred latent code, and that of the real latent code and its generated data sample. The minimax objective of ALI can be written as

$$\min_{\theta,\phi}\max_{\omega}\mathcal{L}_{\text{ALI}} = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}),\ \tilde{\mathbf{z}}\sim p_\phi(\mathbf{z}|\mathbf{x})}\big[\log D_\omega(\mathbf{x},\tilde{\mathbf{z}}))\big] + \mathbb{E}_{\tilde{\mathbf{x}}\sim p_\theta(\mathbf{x}|\mathbf{z}),\ \mathbf{z}\sim p(\mathbf{z})}\big[\log(1-D_\omega(\tilde{\mathbf{x}},\mathbf{z}))\big].$$

where $p_\phi(\mathbf{z}|\mathbf{x})$ is the generator for latent code, and $D_\omega(\mathbf{x},\mathbf{z})$ is the joint sample discriminator.

ALI was designed for single view (domain) data originally, but recent works applied it to the cross-domain learning scenario [5, 14]. Though encouraging results are reported, these models cast aside the latent code space built in ALI.

3 MULTI-VIEW ALI

For multi-view data, here we only consider two views, and extensions to multiple views is straightforward. Assume $\mathbf{x}$ denotes one view and $\mathbf{y}$ denotes the other view of the same instance. In practical applications, the available data usually appear in three forms, i.e., the paired data $(\mathbf{x},\mathbf{y})$ with both views, the unpaired data only with view $\mathbf{x}$, and the unpaired data only with view $\mathbf{y}$. We use $p(\mathbf{x},\mathbf{y})$, $p(\mathbf{x})$ and $p(\mathbf{y})$ to

denote the empirical distributions of these three kinds of data, respectively. Since the paired data usually is expensive to obtain, a good multi-view learning model should not only exploit the empirical joint distribution $p(\mathbf{x},\mathbf{y})$ but also benefit from the empirical marginals $p(\mathbf{x})$ and $p(\mathbf{y})$ as more as possible, which is known as semi-supervised cross-domain (view) joint distribution matching.

3.1 The Basic Idea

Our key assumption is that the two views $\mathbf{x}$ and $\mathbf{y}$ of a joint data sample $(\mathbf{x},\mathbf{y}) \sim p(\mathbf{x},\mathbf{y})$ can be generated via the decoding conditional distributions $p_{\theta_x}(\mathbf{x}|\mathbf{z})$ and $p_{\theta_y}(\mathbf{y}|\mathbf{z})$ respectively, where the latent code $\mathbf{z}$ is shared for both views and the subscripts θ_x and θ_y indicate that the conditional distributions are parameterized by deep neural networks (DNN). Given the prior $\mathbf{z} \sim p(\mathbf{z})$, we may infer the latent via Bayes rule, however the posterior $p_\theta(\mathbf{z}|\mathbf{x},\mathbf{y})$ is intractable to compute generally. Instead, we introduce two encoding conditional distributions $p_{\phi_x}(\mathbf{z}|\mathbf{x})$ and $p_{\phi_y}(\mathbf{z}|\mathbf{y})$ parameterized by DNNs ϕ_x and ϕ_y, and force them to be close to each other for a given data pair $(\mathbf{x},\mathbf{y})$. Note that, the encoders are also generators for the latent codes.

Let $\tilde{\mathbf{x}} \sim p_{\theta_x}(\mathbf{x}|\mathbf{z})$ and $\tilde{\mathbf{y}} \sim p_{\theta_y}(\mathbf{y}|\mathbf{z})$ be a pair of generated samples, $\tilde{\mathbf{z}}_x \sim p_{\phi_x}(\mathbf{z}|\mathbf{x})$ and $\tilde{\mathbf{z}}_y \sim p_{\phi_y}(\mathbf{z}|\mathbf{y})$ be the random latent codes inferred from different data views, $\hat{\mathbf{y}}_x \sim p_{\theta_y}(\mathbf{y}|\tilde{\mathbf{z}}_x)$ and $\hat{\mathbf{x}}_y \sim p_{\theta_x}(\mathbf{x}|\tilde{\mathbf{z}}_y)$ be the missing-view estimators obtained through the available views. Then the basic minimax objective of our multi-view ALI (MALI) model can be written as

$$\begin{aligned}
&\min_{\Theta} \max_{\omega_x,\omega_y,\varpi,\zeta} \mathcal{L}^0_{\text{MALI}} \\
&= \mathbb{E}_{\mathbf{z},\tilde{\mathbf{x}},\tilde{\mathbf{y}}}\big[\log\big(D_{\omega_x}(\tilde{\mathbf{x}},\mathbf{z})\cdot D_{\omega_y}(\tilde{\mathbf{y}},\mathbf{z})\cdot(1-D_\zeta(\tilde{\mathbf{x}},\tilde{\mathbf{y}}))\big)\big] \\
&\quad + \mathbb{E}_{\mathbf{x},\tilde{\mathbf{z}}_x,\hat{\mathbf{y}}_x}\big[\log\big((1-D_{\omega_x}(\mathbf{x},\tilde{\mathbf{z}}_x))\cdot D_\varpi(\mathbf{x},\hat{\mathbf{y}}_x)\big)\big] \\
&\quad + \mathbb{E}_{\mathbf{y},\tilde{\mathbf{z}}_y,\hat{\mathbf{x}}_y}\big[\log\big((1-D_{\omega_y}(\mathbf{y},\tilde{\mathbf{z}}_y))\cdot(1-D_\varpi(\hat{\mathbf{x}}_y,\mathbf{y}))\big)\big] \\
&\quad + \mathbb{E}_{\mathbf{x},\mathbf{y},\tilde{\mathbf{z}}_x,\tilde{\mathbf{z}}_y}\big[\alpha\|\tilde{\mathbf{z}}_x-\tilde{\mathbf{z}}_y\|_2^2 + \log(D_\zeta(\mathbf{x},\mathbf{y}))\big], \qquad (1)
\end{aligned}$$

where we use $\Theta = (\theta_x,\theta_y,\phi_x,\phi_y)$ to denote all the parameters of generators (two encoders and two decoders). Intuitively, four discriminators are designed to distinguish four kinds of joint samples, respectively:

- $D_{\omega_x}(\mathbf{x},\mathbf{z})$ distinguishes samples drawn from
$$p_{\theta_x}(\mathbf{x},\mathbf{z}) = p_{\theta_x}(\mathbf{x}|\mathbf{z})p(\mathbf{z})$$
from samples drawn from
$$p_{\phi_x}(\mathbf{x},\mathbf{z}) = p_{\phi_x}(\mathbf{z}|\mathbf{x})p(\mathbf{x}),$$
which in turn helps generators to produce high-quality samples for both the latent and data view $\mathbf{x}$;
- $D_{\omega_y}(\mathbf{y},\mathbf{z})$ distinguishes samples drawn from
$$p_{\theta_y}(\mathbf{y},\mathbf{z}) = p_{\theta_y}(\mathbf{y}|\mathbf{z})p(\mathbf{z})$$
from samples drawn from
$$p_{\phi_y}(\mathbf{y},\mathbf{z}) = p_{\phi_y}(\mathbf{z}|\mathbf{y})p(\mathbf{y}),$$
which in turn helps generators to produce high-quality samples for both the latent and data view $\mathbf{y}$;
- $D_\zeta(\mathbf{x},\mathbf{y})$ distinguishes real paired samples drawn from $p(\mathbf{x},\mathbf{y})$ from faking paired samples drawn from $p_\theta(\mathbf{x},\mathbf{y})$, which in turn helps generators to produce high-quality faking paired samples and thus provides supervised information for the learning of decoders;
- $D_\varpi(\mathbf{x},\mathbf{y})$ distinguishes samples drawn from $p_x(\mathbf{x},\mathbf{y})$ from samples drawn from $p_y(\mathbf{x},\mathbf{y})$, which in turn helps generators to produce high-quality samples for both data views.

where

$$\begin{aligned}
p_\theta(\mathbf{x},\mathbf{y}) &= \int p_\theta(\mathbf{x},\mathbf{y},\mathbf{z})d\mathbf{z} = \int p_{\theta_x}(\mathbf{x}|\mathbf{z})p_{\theta_y}(\mathbf{y}|\mathbf{z})p(\mathbf{z})d\mathbf{z},\\
p_y(\mathbf{x},\mathbf{y}) &= \int p_y(\mathbf{x},\mathbf{y},\mathbf{z})d\mathbf{z} = \int p_{\theta_y}(\mathbf{y}|\mathbf{z})p_{\phi_x}(\mathbf{z}|\mathbf{x})p(\mathbf{x})d\mathbf{z},\\
p_x(\mathbf{x},\mathbf{y}) &= \int p_x(\mathbf{x},\mathbf{y},\mathbf{z})d\mathbf{z} = \int p_{\theta_x}(\mathbf{x}|\mathbf{z})p_{\phi_y}(\mathbf{z}|\mathbf{y})p(\mathbf{y})d\mathbf{z}.
\end{aligned}$$

Note that, the squared loss term in $\mathcal{L}^0_{\text{MALI}}$ forces the latent codes corresponding to two different views of the same data instance to be similar, which is critical (due to our shared latent space assumption) for injecting the supervised information for cross-domain encoder learning under our multi-view ALI framework. The regularization parameter $\alpha > 0$ balances this term and other adversarial loss terms.

To analyze the equilibrium condition of the minimax game in (1), we first consider the optimal discriminators for any given generator Θ. These optimal discriminators then allow reformulation of objective (1).

Proposition 1. For any fixed generator Θ, the optimal discriminators of problem (1) is given by

$$\begin{aligned}
D^*_{\omega_x}(\mathbf{x},\mathbf{z}) &= \frac{p_{\theta_x}(\mathbf{x},\mathbf{z})}{p_{\theta_x}(\mathbf{x},\mathbf{z})+p_{\phi_x}(\mathbf{x},\mathbf{z})},\\
D^*_{\omega_y}(\mathbf{y},\mathbf{z}) &= \frac{p_{\theta_y}(\mathbf{y},\mathbf{z})}{p_{\theta_y}(\mathbf{y},\mathbf{z})+p_{\phi_y}(\mathbf{y},\mathbf{z})},\\
D^*_\zeta(\mathbf{x},\mathbf{y}) &= \frac{p(\mathbf{x},\mathbf{y})}{p(\mathbf{x},\mathbf{y})+p_\theta(\mathbf{x},\mathbf{y}))}\\
D^*_\varpi(\mathbf{x},\mathbf{y}) &= \frac{p_y(\mathbf{x},\mathbf{y})}{p_y(\mathbf{x},\mathbf{y})+p_x(\mathbf{x},\mathbf{y})}.
\end{aligned}$$

Proof. Given generator Θ, the training criterion for the discriminators is to maximize

$$\begin{aligned}
&\mathcal{L}^0_{\text{MALI}}(\omega_x,\omega_y,\zeta,\varpi)\\
&= \mathbb{E}_{\mathbf{z},\tilde{\mathbf{x}}}\big[\log D_{\omega_x}(\tilde{\mathbf{x}},\mathbf{z})\big] + \mathbb{E}_{\mathbf{x},\tilde{\mathbf{z}}_x}\big[\log(1-D_{\omega_x}(\mathbf{x},\tilde{\mathbf{z}}_x))\big]\\
&\quad + \mathbb{E}_{\mathbf{z},\tilde{\mathbf{y}}}\big[\log D_{\omega_y}(\tilde{\mathbf{y}},\mathbf{z})\big] + \mathbb{E}_{\mathbf{y},\tilde{\mathbf{z}}_y}\big[\log(1-D_{\omega_y}(\mathbf{y},\tilde{\mathbf{z}}_y))\big]\\
&\quad + \mathbb{E}_{\mathbf{x},\mathbf{y}}\big[\log D_\zeta(\mathbf{x},\mathbf{y})\big] + \mathbb{E}_{\tilde{\mathbf{x}},\tilde{\mathbf{y}}}\big[\log(1-D_\zeta(\tilde{\mathbf{x}},\tilde{\mathbf{y}}))\big]\\
&\quad + \mathbb{E}_{\mathbf{x},\hat{\mathbf{y}}_x}\big[\log D_\varpi(\mathbf{x},\hat{\mathbf{y}}_x)\big] + \mathbb{E}_{\mathbf{y},\hat{\mathbf{x}}_y}\big[\log(1-D_\varpi(\hat{\mathbf{x}}_y,\mathbf{y}))\big]\\
&= \mathbb{E}_{p_{\theta_x}(\mathbf{x},\mathbf{z})}\big[\log D_{\omega_x}(\mathbf{x},\mathbf{z})\big] + \mathbb{E}_{p_{\phi_x}(\mathbf{x},\mathbf{z})}\big[\log(1-D_{\omega_x}(\mathbf{x},\mathbf{z}))\big]\\
&\quad + \mathbb{E}_{p_{\theta_y}(\mathbf{y},\mathbf{z})}\big[\log D_{\omega_y}(\mathbf{y},\mathbf{z})\big] + \mathbb{E}_{p_{\phi_y}(\mathbf{y},\mathbf{z})}\big[\log(1-D_{\omega_y}(\mathbf{y},\mathbf{z}))\big]\\
&\quad + \mathbb{E}_{p(\mathbf{x},\mathbf{y})}\big[\log D_\zeta(\mathbf{x},\mathbf{y})\big] + \mathbb{E}_{p_\theta(\mathbf{x},\mathbf{y})}\big[\log(1-D_\zeta(\mathbf{x},\mathbf{y}))\big]\\
&\quad + \mathbb{E}_{p_y(\mathbf{x},\mathbf{y})}\big[\log D_\varpi(\mathbf{x},\mathbf{y})\big] + \mathbb{E}_{p_x(\mathbf{x},\mathbf{y})}\big[\log(1-D_\varpi(\mathbf{x},\mathbf{y}))\big].
\end{aligned}$$

Following [6], we utilize the fact that for any $(a, b) \in \mathbb{R}^2 \backslash \{0, 0\}$, the problem $\max_t a \log(t) + b \log(1-t)$ attains its maximum in [0, 1] at $t = \frac{a}{a+b}$. This concludes the proof. □

Theorem 1. The training criterion $\mathcal{L}^0_{\text{MALI}}$ of problem (1) achieves its optimal value $-4 \log 4$, if and only if the generators and discriminators satisfy: (i) $p(\mathbf{x}, \mathbf{y}) = p_\theta(\mathbf{x}, \mathbf{y})$; (ii) $p_y(\mathbf{x}, \mathbf{y}) = p_x(\mathbf{x}, \mathbf{y})$; (iii) $p_{\phi_x}(\mathbf{x}, \mathbf{z}) = p_{\theta_x}(\mathbf{x}, \mathbf{z})$; (iv) $p_{\phi_y}(\mathbf{y}, \mathbf{z}) = p_{\theta_y}(\mathbf{y}, \mathbf{z})$; and (v) $p_{\phi_x}(\mathbf{z}|\mathbf{x}) = p_{\phi_y}(\mathbf{z}|\mathbf{y})$ and $H(\mathbf{z}|\mathbf{x}) = H(\mathbf{z}|\mathbf{y}) = 0$ for any $(\mathbf{x}, \mathbf{y}) \sim p(\mathbf{x}, \mathbf{y})$, where $H(\cdot)$ denotes the entropy of a random variable.

Proof. Given the optimal discriminators specified in Proposition 1, the minimax game in (1) can be reformulated as

$$\begin{aligned}
&\min_{\Theta} \mathcal{L}^0_{\text{MALI}}(\Theta) \\
&= 2 \cdot JSD(p_{\theta_x}(\mathbf{x}, \mathbf{z}) \| p_{\phi_x}(\mathbf{x}, \mathbf{z})) + 2 \cdot JSD(p(\mathbf{x}, \mathbf{y}) \| p_\theta(\mathbf{x}, \mathbf{y})) \\
&\quad + 2 \cdot JSD(p_{\theta_y}(\mathbf{y}, \mathbf{z}) \| p_{\phi_y}(\mathbf{y}, \mathbf{z})) + \mathbb{E}_{\mathbf{x}, \mathbf{y}, \tilde{\mathbf{z}}_x, \tilde{\mathbf{z}}_y} \left[\alpha \| \tilde{\mathbf{z}}_x - \tilde{\mathbf{z}}_y \|_2^2 \right] \\
&\quad + 2 \cdot JSD(p_y(\mathbf{x}, \mathbf{y}) \| p_x(\mathbf{x}, \mathbf{y})) - 4 \log 4,
\end{aligned}$$

where JSD denotes the Jensen-Shannon divergence (JSD) between two distributions, which is always non-negative and zero only when they are equal. The squared loss term also is non-negative and zero only when the encoding conditionals satisfy that $p_{\phi_x}(\mathbf{z}|\mathbf{x}) = p_{\phi_y}(\mathbf{z}|\mathbf{y})$, and $H(\mathbf{z}|\mathbf{x}) = H(\mathbf{z}|\mathbf{y}) = 0$ for any $(\mathbf{x}, \mathbf{y}) \sim p(\mathbf{x}, \mathbf{y})$, i.e., the latent codes are obtained via deterministic transformations of the input data. Thus $-4 \log 4$ is the global minimum. □

Note that, the paired data generating distribution $p_\theta(\mathbf{x}, \mathbf{y})$ is expected to match the empirical joint distribution $p(\mathbf{x}, \mathbf{y})$ as long as the decoding conditionals $p_{\theta_x}(\mathbf{x}|\mathbf{z})$ and $p_{\theta_y}(\mathbf{y}|\mathbf{z})$ are optimized. At the same time, these conditionals are also amenable to form good matching of the marginals $p(\mathbf{x})$ and $p(\mathbf{y})$ through $\int p_{\theta_x}(\mathbf{x}, \mathbf{z}) d\mathbf{z}$ and $\int p_{\theta_y}(\mathbf{y}, \mathbf{z}) d\mathbf{z}$, respectively.

According to Theorem 1, we should set the encoding mappings (from each data view to the latent space) to be deterministic. However, in real world there exist many examples of one-to-many mapping relation between domains, e.g., the attribute-conditional image generation problem. For such cases, we choose random encoding mapping for the data domain (e.g., attributes) where a sample is expected to correspond to many samples in the other domain (e.g., images). Specifically, the randomness is introduced by sampling the inferred latent codes from a Gaussian conditional distribution, the mean and covariance of which both are computed via deterministic (encoding) transformations of the input data. We found this strategy works well in practice, since the squared loss term in $\mathcal{L}^0_{\text{MALI}}$ cannot be optimized to zero generally.

3.2 Exploiting Cycle-consistencies

Theorem 1 shows that the optimal generators are determined by both the paired data and the unpaired data, which forms a semi-supervised learning mechanism. However, we note that the supervised signal for learning the encoding conditionals only come from the empirical joint distribution $p(\mathbf{x}, \mathbf{y})$, which may be too weak to learn an effective model when the paired data is small. Moreover, too large α for the squared loss term in $\mathcal{L}^0_{\text{MALI}}$ may lead to instability of the adversarial game. The reason is that the encoders may ignore to match the prior $p(\mathbf{z})$ on latent codes. To this end, we propose to employ the abundant faking paired samples $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \sim p_\theta(\mathbf{x}, \mathbf{y})$ to regularize the encoders. Assume $\hat{\mathbf{z}}_x \sim p_{\phi_x}(\mathbf{z}|\tilde{\mathbf{x}})$ and $\hat{\mathbf{z}}_y \sim p_{\phi_y}(\mathbf{z}|\tilde{\mathbf{y}})$ denote the latent code reconstructions via $\mathbf{z} \to \tilde{\mathbf{x}} \to \hat{\mathbf{z}}_x$ and $\mathbf{z} \to \tilde{\mathbf{y}} \to \hat{\mathbf{z}}_y$, respectively. Then our idea can be formulated as

$$\begin{aligned}
\min_{\Theta} \max_{\omega_x, \omega_y, \varpi, \zeta} \mathcal{L}^1_{\text{MALI}} &= \mathcal{L}^0_{\text{MALI}} + \\
&\alpha \cdot \mathbb{E}_{\mathbf{z}, \tilde{\mathbf{x}}, \tilde{\mathbf{y}}, \hat{\mathbf{z}}_x, \hat{\mathbf{z}}_y} \left[\|\hat{\mathbf{z}}_x - \mathbf{z}\|_2^2 + \|\hat{\mathbf{z}}_y - \mathbf{z}\|_2^2 + \|\hat{\mathbf{z}}_x - \hat{\mathbf{z}}_y\|_2^2 \right].
\end{aligned}$$

Such a treatment can also be interpreted as applying the cycle-consistency principle (usually used on observable data domains [9, 32]) in the latent space.

Similarly, we can also construct auto-encoding cycles for the unpaired data $\mathbf{x}$ and $\mathbf{y}$ to improve the discriminative power of the latent codes, which was shown beneficial for model stability [14]. Suppose $\hat{\mathbf{x}} \sim p_{\theta_x}(\mathbf{x}|\tilde{\mathbf{z}}_x)$ and $\hat{\mathbf{y}} \sim p_{\theta_y}(\mathbf{y}|\tilde{\mathbf{z}}_y)$ denote the data view reconstructions via $\mathbf{x} \to \tilde{\mathbf{z}}_x \to \hat{\mathbf{x}}$ and $\mathbf{y} \to \tilde{\mathbf{z}}_y \to \hat{\mathbf{y}}$, respectively. Then the objective of our model can be naively reformulated as

$$\mathcal{L}^1_{\text{MALI}} + \mathbb{E}_{\mathbf{x}, \hat{\mathbf{x}}} \left[\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2 \right] + \mathbb{E}_{\mathbf{y}, \hat{\mathbf{y}}} \left[\|\hat{\mathbf{y}} - \mathbf{y}\|_2^2 \right].$$

The shortcoming of this model is that squared reconstruction loss is computed in the data space, which often leads the generated samples to be blurry. To remedy, we follow [14] to derive an adversarial implementation. Taking data view $\mathbf{x}$ as example, we use a discriminator η_x to distinguish between two joints, the joint of $\mathbf{x}$ and itself, and the joint of $\mathbf{x}$ and its reconstruction. Thus our model becomes

$$\begin{aligned}
\min_{\Theta} \max_{\omega_x, \omega_y, \eta_x, \eta_y, \varpi, \zeta} \mathcal{L}^2_{\text{MALI}} &= \mathcal{L}^1_{\text{MALI}} \\
&+ \mathbb{E}_{\mathbf{x}, \hat{\mathbf{x}}} \left[\log \left(D_{\eta_x}(\mathbf{x}, \mathbf{x}) \cdot (1 - D_{\eta_x}(\mathbf{x}, \hat{\mathbf{x}})) \right) \right] \\
&+ \mathbb{E}_{\mathbf{y}, \hat{\mathbf{y}}} \left[\log \left(D_{\eta_y}(\mathbf{y}, \mathbf{y}) \cdot (1 - D_{\eta_y}(\mathbf{y}, \hat{\mathbf{y}})) \right) \right].
\end{aligned}$$

Note that, the auto-encoding cycle-consistency we used is different from that analyzed in [14], where larger cycles involve both data domains are considered. When cycle-consistency principle is not applicable due to one-to-many mapping relation between domains, consistency under our smaller cycle still can be used.

3.3 Incorporating Conditional GANs

Finally, we can use conditional GAN to boost the learning of domain correspondence as in [5, 14]. Conditional GAN is a more direct way for injecting supervised information than the squared loss term in $\mathcal{L}^0_{\text{MALI}}$. Recall that $\hat{\mathbf{y}}_x \sim p_{\theta_y}(\mathbf{y}|\tilde{\mathbf{z}}_x)$ and $\hat{\mathbf{x}}_y \sim p_{\theta_x}(\mathbf{x}|\tilde{\mathbf{z}}_y)$ are the missing-view estimators obtained from the available views. For view $\mathbf{x}$, we use a discriminator χ_x to distinguish between two joints, the joint of $\mathbf{x}$ and itself, and the joint of $\mathbf{x}$ and its cross-domain estimator $\hat{\mathbf{x}}_y$. Similar strategy is adopted for view $\mathbf{y}$. Therefore, our overall model

can be formulated as

$$\min_{\Theta} \max_{\Omega} \mathcal{L}^*_{\text{MALI}} = \mathcal{L}^2_{\text{MALI}} + \mathbb{E}_{\mathbf{x},\mathbf{y},\hat{\mathbf{x}}_y}\big[\log\big(D_{\chi_x}(\mathbf{x},\mathbf{x})\cdot(1-D_{\chi_x}(\mathbf{x},\hat{\mathbf{x}}_y))\big)\big] + \mathbb{E}_{\mathbf{x},\mathbf{y},\hat{\mathbf{y}}_x}\big[\log\big(D_{\chi_y}(\mathbf{y},\mathbf{y})\cdot(1-D_{\chi_x}(\mathbf{y},\hat{\mathbf{y}}_x))\big)\big],$$

where $\Omega = (\omega_x, \omega_y, \eta_x, \eta_y, \chi_x, \chi_y, \varpi, \zeta)$ denotes all the parameters of discriminators.

4 RELATED WORK

The original ALI/BiGAN model [2, 3] has already been extended to address the joint distribution matching problem [1, 5, 14]. The △-GAN model [5] can be considered as a combination of ALI and the conditional GAN [8, 20]. The ALICE model [14] adopted a similar idea, but with conditional entropy regularization, which is basically equivalent to the cycle-consistency principle. Both of them cast aside the latent code space built in ALI, and thus cannot benefit from it in the five ways stated before. Another recent work for semi-supervised joint distribution matching is Triple GAN [15], which consists of two conditional GANs. In fact, when paired data is sufficient enough, conditional GAN itself works well for this task. Models of this kind typically condition the image generation on class labels [20], attributes [21], texts [22, 31] and images [8, 13]. Many unsupervised learning methods have also been developed for this task, e.g., the DiscoGAN [9] uses two generators to model the bidirectional mapping between domains, and another two discriminators to decide whether a generated sample is fake or not in each individual domain. Similar work includes CycleGAN [32], DualGAN [30] and DTN [27]. Additional weight-sharing constraints are introduced in CoGAN [17] and UNIT [16] to exploit the structural similarity of both data domains.

5 EXPERIMENTS

We present results on four tasks: (1) image-to-image translation on MNIST [12]; (2) image-to-attribute prediction (also known as multi-label classification) on CelebA; (3) attribute-to-image generation on CelebA [18]; and (4) cross-modality retrieval on the ImageNet-EEG dataset [25]. For all experiments, we set the regularization parameter α to 10, without introducing additional regularization for other terms.

5.1 Evaluation Metrics

For multi-label classification experiments, we use two popular evaluation metrics following [5], i.e., the Precision (P) and the normalized Discounted Cumulative Gain (nDCG). Given the ground truth label vector $\mathbf{y} \in \{0,1\}^L$ and the prediction $\hat{\mathbf{y}} \in [0,1]^L$, Precision at k is defined as $\text{P@}k = \frac{1}{k}\sum_{i\in \text{Top}_k(\hat{\mathbf{y}})} y_i$ where $\text{Top}_k(\hat{\mathbf{y}})$ denotes the indices set that corresponds to the top k scoring labels predicted by $\hat{\mathbf{y}}$. That is, Precision at k counts the fraction of correct predictions in the top k scoring labels.

DCG is the total gain accumulated at a particular rank k, which is defined as

$$\text{DCG@}k = \sum_{i\in \text{Top}_k(\hat{\mathbf{y}})} \frac{y_i}{\log(i+1)}$$

Normalizing DCG at rank k by the value of the ideal ranking gives

$$\text{nDCG@}k = \frac{\text{DCG@}k}{\sum_{i=1}^{\min(k,\|\mathbf{y}\|_0)} \frac{1}{\log(i+1)}}$$

For cross-modality retrieval experiments, we use the following popular metrics: (1) Precision at S: the percentage of true relevant instances among top S retrieved ones; and (2) Recall at S: the percentage of true relevant instances that are retrieved as the top S similar ones; (3) mean average precision (mAP), which computes the area under the entire precision-recall curve and evaluates the overall retrieval performance of different hashing algorithms. Note that, in evaluating these metrics for a given query image, we use a fixed small set (of size 100) of Euclidean nearest neighbors to the query as its ground-truth relevant images.

5.2 Image-to-image translation

We first assess our model on the MNIST-to-MNIST-transpose dataset [5], where the two image domains are the MNIST images and their corresponding transposed ones. For fair comparison, we used the same DNN architecture as in [5].

The goal is to learn bidirectional mappings between these two domains with a fraction of samples randomly chosen from the training set being paired, and then predict on the testing set. In total, the training set and the testing set have 50000 samples each domain and 10000 samples each domain, respectively. As can be seen from Figure 1, the proposed MALI model can faithfully reconstruct the input images in the transposed space (even when only 100 paired samples are available), while Δ-GAN often generates confusing images that are easy to be classified incorrectly by humans. For supporting quantitative evaluation, we follow [5] to train a classifier on the MNIST dataset. The classification accuracy of this classifier on the test set approaches 99.4%, and is, therefore, trustworthy as an evaluation metric. Given an input MNIST image $\mathbf{x}$, we first encode this image into $\tilde{\mathbf{z}}_x$ using the learned encoder and then generate a transposed image $\hat{\mathbf{y}}_x$ using the learned generator. To utilize the pre-trained classifier, we then manually transpose it back to normal digit $\hat{\mathbf{y}}_x^\top$, and finally send this new image to the classifier. Results are summarized in Table 1, which are averages over 5 runs with different random splits of the training data. MALI achieves significantly better performance than Δ-GAN, Triple GAN and DiscoGAN.

5.3 Multi-label classification

We then apply our method to the attribute predicting (multi-label classification) problem of face images from the CelebA dataset. This dataset consists of 202,599 images annotated with 40 binary attributes. We treat the images and the attributes as two different data domains (views). Following [5],

Figure 1: Image translation on the MNIST-to-MNIST-transpose dataset. Δ-GAN-100 and MALI-100 denote models learned with 100 paired samples, and Δ-GAN-all and MALI-all denote models learned with all training samples being paired.

Table 1: Classification accuracy (%) on the MNIST-to-MNIST-transpose dataset with varying number of paired samples

Algorithm	100 paired	1000 paired	All paired
DiscoGAN	-	-	15.00 ± 0.20
Triple GAN	63.79 ± 0.85	84.93 ± 1.63	86.70 ± 1.52
△-GAN	83.20 ± 1.88	88.98 ± 1.50	93.34 ± 1.46
MALI	**97.46 ± 0.43**	**98.05 ± 0.21**	**98.69 ± 0.18**

we scale and crop the images to 64 × 64 pixels, and split the dataset into training (162770 samples), testing (19962 samples), and validation (19867 samples) sets. To set the semi-supervised learning scenario, only a fraction of samples randomly chosen from the training set are treated as paired data, and other samples in the training set do not disclose their domain pairing information to the cross-domain joint distribution matching algorithms. Table 2 provides the quantitative comparison results of P@k and nDCG@k on CelebA, where three different values (3, 5, and 10) of k and three different proportions (1%, 10%, and 100%) of paired data are considered. The results of Triple GAN and Δ-GAN are cited from [5], while results of MALI are the averages of 5 independent trials. For fair comparison, we use the same CNN network architectures as in [5] to encode/decode an image. As can be seen, our method outperforms the competitors by a large margin in all cases. We also note that our method achieves significantly higher classification accuracy (0.894) than that (0.86) of ALICE [14]. We attribute this to our shared low-dimensional latent space for both domains, where abundant faking paired samples $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \sim p_\theta(\mathbf{x}, \mathbf{y})$ are exploited to boost the encoders. Note that ALICE cannot use its cycle consistency on the image domain since uncertainty in image domain is desirable, while our MALI utilizes the auto-encoding constraints for both domains rather than the whole-cycle consistency. Figure 2 shows some example results of our MALI model for qualitative examination, from which we can see the learned attribute predictor indeed works well.

Table 2: Results (%) of P@k and nDCG@k for attribute predicting (multi-label classification) on CelebA. Three different values (3, 5, and 10) of k and three different proportions (1%, 10%, and 100%) of paired data are considered. For each case, we list P before nDCG with a separator '/'.

Algorithm	1% paired	10% paired	All paired
$k = 3$			
Triple GAN	55.30 / 56.87	76.09 / 75.44	83.54 / 84.74
△-GAN	62.62 / 62.72	76.04 / 76.27	84.81 / 86.85
MALI	**83.50 / 86.01**	**91.64 / 93.24**	**92.54 / 93.98**
$k = 5$			
Triple GAN	49.35 / 52.73	73.68 / 74.55	80.89 / 78.58
△-GAN	59.55 / 60.53	74.06 / 75.49	80.39 / 79.41
MALI	**76.91 / 80.80**	**85.33 / 88.49**	**87.33 / 90.02**
$k = 10$			
Triple GAN	40.97 / 50.74	62.13 / 73.56	70.12 / 79.37
△-GAN	53.21 / 58.39	63.68 / 75.22	70.37 / 81.47
MALI	**72.69 / 76.48**	**80.61 / 84.14**	**83.50 / 86.40**

5.4 Attribute-to-image generation

In order to qualitatively evaluate the learned image generator and the attribute predictor (multi-label classifier) on CelebA, given an input face image, we first use the classifier to predict attributes, and then use the image generator to produce images based on the predicted attributes. Figure 2 shows example results, where we used random attribute-encoding mapping to generate diverse images with the same attributes. Both the learned attribute predictor and the image generator provides good results.

We further show another set of image editing experiment in Figure 3. For each sub-figure, we first randomly choose a set of binary attribute vectors to generate the images in the 1st row, and then we turn on/off some bit(s) of the chosen binary attribute vectors to generate the images in the 2nd row. To gain some perceptual intuition on such kind of image editing, here we directly use the mean of the latent Gaussian conditional distribution for decoding easy-to-compare images. It is interesting to see that MALI has great flexibility to adjust the generated images by changing certain input attributes. For instance, by switching on the pale skin attribute, one can modify the skin color of the generated face images.

5.5 Cross-modality retrieval

The ImageNet-EEG dataset [25] involves six subjects who were shown images of objects while Electroencephalogram (EEG) data was recorded. As visual stimuli, 50 images from 40 different ImageNet classes for a total of 2,000 images were employed. Each image class was presented in batches of

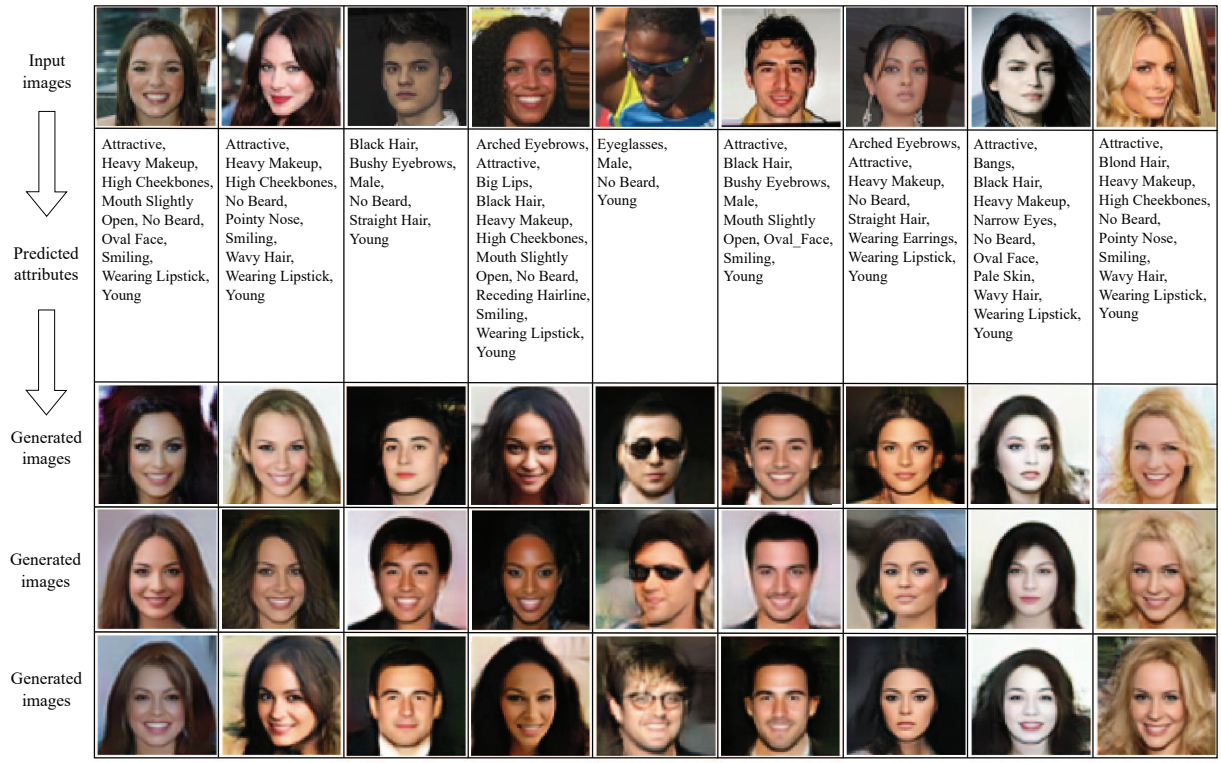


Figure 2: Image-to-attribute-to-image results of the proposed MALI model on CelebA.

25 seconds, followed by a 10 second black screen to "clear" the visual pathway. The total duration of each experiment was 1,400 seconds (23 minutes and 20 seconds). After the EEG data acquisition, there are 11,466 EEG sequences (536 recordings were discarded because they were too short or too altered to be included in the experiment). The EEG sequence for each image has a shape of [500, 128] (500 time points and 128-channel electrodes). From each recorded EEG sequence, the first 50 time points were ignored due to the possibility of interferences with previous displayed image. The following 450 time points were used for our experiments.

To simulate semi-supervised learning scenario, we use a sliding window with shape of [200, 128] to build the paired and unpaired EEG datasets. Specifically, we obtain the paired EEG data ([11466, 200, 128]) by sampling the first 200 time points of all EEG sequences, and we obtain the unpaired EEG data by applying the sliding window with ten different offsets ($\{20, 40, \cdots, 200\}$) to the EEG sequences. Furthermore, we selected all images, belonging to the given stimuli categories, from the ImageNet database as unpaired image data. Finally, we utilize the pre-trained Inception-v3 model [26] and Principle Components Analysis (PCA) to extract the 2048-dim feature representations for image and EEG data, respectively. The details of the data used in our experiments were summarized in Table 3.

In the experiments, we consider the single-layer fully connected networks as the type of generative, inference and discriminative networks. In particular, the architecture of the generators '**z**-**x**' and '**z**-**y**' has a shape of '500-2048', where we didn't use any activation function. For fair comparison, we set the architecture of the transformations between $\mathbf{x}$ and $\mathbf{y}$ as '2048-500-2048' for ALICE and $\triangle$-GAN. We used the Adam optimizer with learning rate 1×10^{-4} in the training of all models. For each image class, we randomly select 5 EEG samples from the paired dataset as queries, which are not available during model training. For each selected EEG query, we find its S nearest neighbors in the 500-dim latent space of MALI based on Euclidean distance, and return the matching images of these neighbors. Note that, the considered competitors can only map the EEG query into the 2048-dim CNN feature space for similarity search, which is more time-consuming. We use Precison100@S and Recall100@S to evaluate the retrieval performance, where 100 is the number of ground truth neighbors.

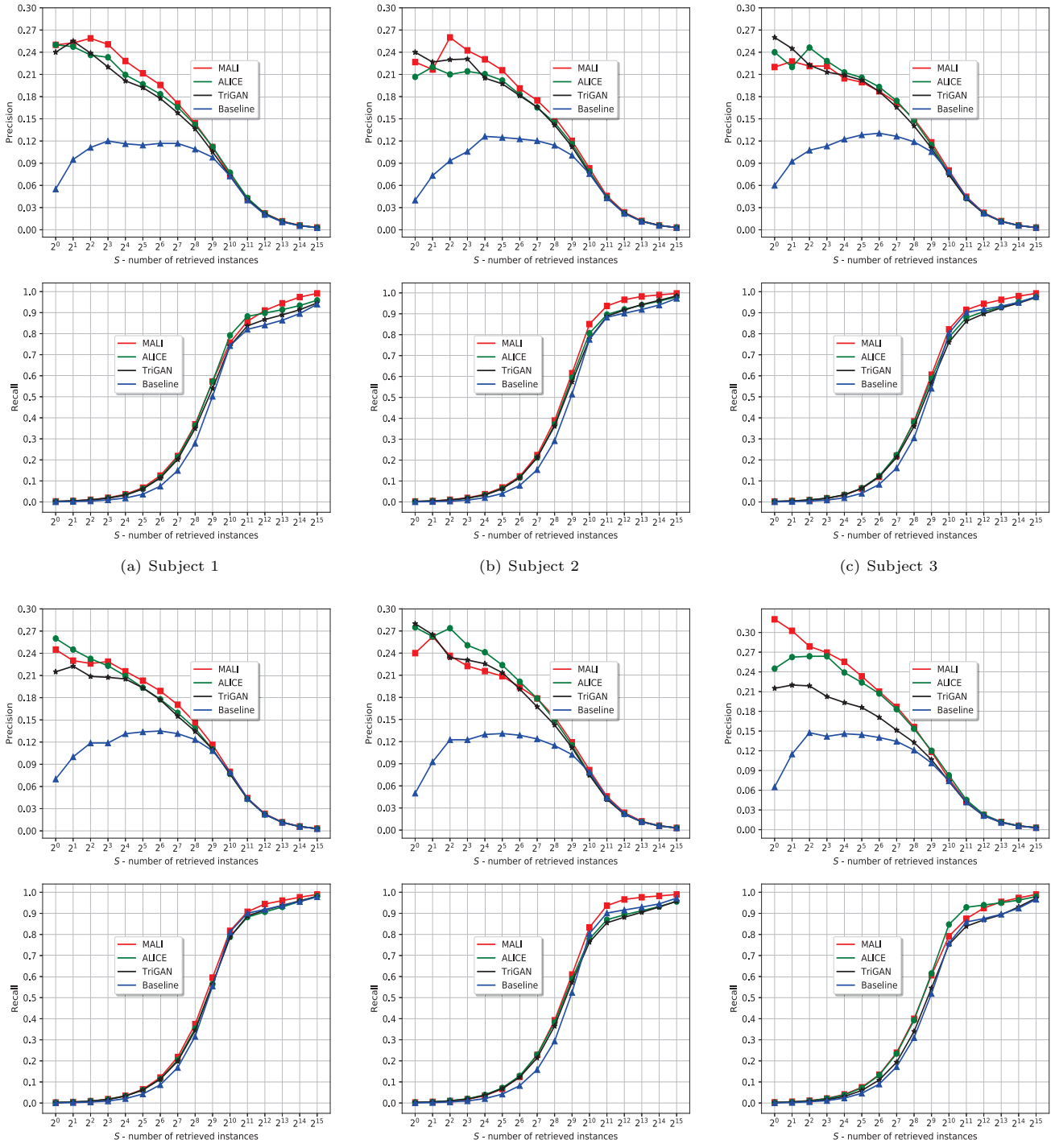
Table 4 displays the mAP performance of different cross-modal retrieval methods on the ImageNet-EEG dataset and the averaged precision and recall over all queries are shown in Figure 4. It is clear that three joint distribution matching methods (MALI, $\triangle$-GAN and ALICE) consistently outperform the baseline method: given an EEG query, we first find its nearest neighbor in the paired EEG dataset, and then perform image retrieval based on the corresponding image representation of that nearest neighbor EEG instance. We also see that our MALI beats state-of-the-art semi-supervised

Figure 3: Image editing results of the proposed MALI model on CelebA.

Table 3: The details of the ImageNet-EEG dataset used in our cross-modal image retrieval experiments.

Subjects	#Classes	#Instances	#Paired	#Query	#Unpaired-Image	#Unpaired-EEG
Subject 1	40	1985	1785	200	61341	19850
Subject 2	30	1497	1347	150	61341	14970
Subject 3	40	1996	1796	200	61341	19960
Subject 4	40	1996	1796	200	61341	19960
Subject 5	40	1996	1796	200	61341	19960
Subject 6	40	1996	1796	200	61341	19960

cross-domain joint distribution matching algorithms △-GAN and ALICE on most subjects. We ascribe this to the fact that our model relies on shared latent representations of both domains, and can generate arbitrary number of paired faking samples, benefiting from which usually very few paired samples (together with sufficient unpaired ones) is enough for learning good domain mappings. Another important feature of the proposed method that provides advantages over competitors is its ability to conduct cross-modal retrieval in latent space. Previous GAN-based semi-supervised methods



(a) Subject 1 (b) Subject 2 (c) Subject 3

(d) Subject 4 (e) Subject 5 (f) Subject 6

Figure 4: Comparisons of the precision and recall results on ImageNet-EEG dataset. We evaluate the performance with Precison100@S and Recall100@S, where 100 is the number of ground truth neighbors and S increases up to 2^{15}. These results show that our latent space based method can achieve competitive similarity search performance in relative fast speed, compared to those methods that compute similarities in the high-dimensional data space.

(e.g., △-GAN and ALICE) can only perform cross-modal retrieval in original high dimensional space, which is inefficient. By introducing shared latent variables, the computational complexity of our MALI model in cross-modal retrieval can be reduced effectively.

Table 4: mAP results (%) on ImageNet-EEG data.

Subjects	Baseline	△-GAN	ALICE	MALI
Subject 1	12.25	15.29	15.98	**16.38**
Subject 2	12.52	16.01	15.60	**17.05**
Subject 3	13.08	16.19	**16.80**	16.63
Subject 4	13.52	15.07	15.30	**16.39**
Subject 5	13.24	16.01	**17.22**	17.08
Subject 6	14.17	16.65	17.58	**17.71**
Average	13.13	15.87	16.41	**16.87**

6 CONCLUSION

We have proposed a multi-view ALI model for semi-supervised cross-domain joint distribution matching. Unlike the common practice of learning direct domain mappings, our model relies on shared latent representations of both domains and can generate arbitrary number of paired faking samples, benefiting from which usually very few paired samples (together with sufficient unpaired ones) is enough for learning good mappings. Extending the vanilla ALI model, we designed novel discriminators to judge the quality of generated samples (both paired and unpaired), and provided theoretical analysis of our new formulation. Experiments on image translation, image-to-attribute and attribute-to-image generation tasks demonstrate that our new framework yields significant performance improvements over existing ones. Results on cross-modality retrieval show that our latent space based method can achieve competitive similarity search performance in relative fast speed, compared to those methods that compute similarities in the high-dimensional data space. Future works include extending our model to handle more than two data domains and applying our framework to multi-view classification with missing view imputation, etc.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (No. 61602449, 61672501, 61603372).

REFERENCES

[1] Mickaël Chen and Ludovic Denoyer. 2017. Multi-view generative adversarial networks. In *ECML/PKDD*. 175–188.
[2] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. 2017. Adversarial feature learning. In *ICLR*.
[3] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martin Arjovsky, Olivier Mastropietro, and Aaron Courville. 2017. Adversarially learned inference. In *ICLR*.
[4] Linan Feng and Bir Bhanu. 2016. Semantic concept co-occurrence patterns for image annotation and retrieval. *IEEE transactions on PAMI* 38, 4 (2016), 785–799.
[5] Zhe Gan, Liqun Chen, Weiyao Wang, Yunchen Pu, Yizhe Zhang, Hao Liu, Chunyuan Li, and Lawrence Carin. 2017. Triangle Generative Adversarial Networks. In *NIPS*.
[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NIPS*. 2672–2680.
[7] Zhiting Hu, Zichao Yang, Ruslan Salakhutdinov, and Eric P Xing. 2018. On unifying deep generative models. In *ICLR*.
[8] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *CVPR*.
[9] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jungkwon Lee, and Jiwon Kim. 2017. Learning to discover cross-domain relations with generative adversarial networks. In *ICML*.
[10] Diederik P Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. In *ICLR*.
[11] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. 2016. Autoencoding beyond pixels using a learned similarity metric. In *ICML*. 1558–1566.
[12] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.
[13] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, et al. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*.
[14] Chunyuan Li, Hao Liu, Changyou Chen, Yuchen Pu, Liqun Chen, Ricardo Henao, and Lawrence Carin. 2017. Alice: Towards understanding adversarial learning for joint distribution matching. In *NIPS*. 5501–5509.
[15] Chongxuan Li, Kun Xu, Jun Zhu, and Bo Zhang. 2017. Triple Generative Adversarial Nets. In *NIPS*.
[16] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. 2017. Unsupervised Image-to-Image Translation Networks. In *NIPS*.
[17] Ming-Yu Liu and Oncel Tuzel. 2016. Coupled generative adversarial networks. In *NIPS*. 469–477.
[18] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep learning face attributes in the wild. In *ICCV*. 3730–3738.
[19] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. 2017. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks. In *ICML*.
[20] Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014).
[21] Guim Perarnau, Joost van de Weijer, Bogdan Raducanu, and Jose M Álvarez. 2016. Invertible Conditional GANs for image editing. *arXiv preprint arXiv:1611.06355* (2016).
[22] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. Generative adversarial text to image synthesis. In *ICML*.
[23] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In *ICML*.
[24] Mihaela Rosca, Balaji Lakshminarayanan, David Warde-Farley, and Shakir Mohamed. 2017. Variational Approaches for Auto-Encoding Generative Adversarial Networks. *arXiv preprint arXiv:1706.04987* (2017).
[25] Concetto Spampinato, Simone Palazzo, Isaak Kavasidis, Daniela Giordano, Nasim Souly, and Mubarak Shah. 2017. Deep learning human mind for automated visual classification. In *CVPR*.
[26] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. In *CVPR*. 2818–2826.
[27] Yaniv Taigman, Adam Polyak, and Lior Wolf. 2017. Unsupervised cross-domain image generation. In *ICLR*.
[28] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. 2016. Cnn-rnn: A unified framework for multi-label image classification. In *CVPR*. 2285–2294.
[29] Jiajun Wu, Yinan Yu, Chang Huang, and Kai Yu. 2015. Deep multiple instance learning for image classification and auto-annotation. In *CVPR*. 3460–3469.
[30] Zili Yi, Hao Zhang, Tan Ping, and Gong Minglun. 2017. DualGAN: Unsupervised Dual Learning for Image-to-Image Translation. In *ICCV*.
[31] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaolei Huang, Xiaogang Wang, and Dimitris Metaxas. 2017. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*.
[32] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*.