공학석사 학위논문

Unsupervised Domain Adaptation via Joint Contrastive Learning

공동 대조적 학습을 이용한 비지도 도메인 적응 기법 연구

2021년 2월

서울대학교 대학원
전기·정보 공학부
박 창 화

공학석사 학위논문

Unsupervised Domain Adaptation via Joint Contrastive Learning

공동 대조적 학습을 이용한 비지도 도메인 적응 기법 연구

2021년 2월

서울대학교 대학원
전기·정보 공학부
박 창 화

Unsupervised Domain Adaptation via Joint Contrastive Learning

공동 대조적 학습을 이용한
비지도 도메인 적응 기법 연구

지도교수 윤 성 로
이 논문을 공학석사 학위논문으로 제출함

2021년 2월

서울대학교 대학원

전기 컴퓨터 공학부

박 창 화

박창화의 공학석사 학위 논문을 인준함

2021년 2월

위 원 장: 정 교 민 (인)
부위원장: 윤 성 로 (인)
위 원: 양 인 순 (인)

Abstract

Domain adaptation is introduced to exploit the label information of source domain when labels are not available for target domain. Previous methods minimized domain discrepancy in a latent space to enable transfer learning. These studies are based on the theoretical analysis that the target error is upper bounded by the sum of source error, the domain discrepancy, and the joint error of the ideal hypothesis. However, feature discriminability is sacrificed while enhancing the feature transferability by matching marginal distributions. In particular, the ideal joint hypothesis error in the target error upper bound, which was previously considered to be minute, has been found to be significant, impairing its theoretical guarantee.

In this paper, to manage the joint error, we propose an alternative upper bound on the target error that explicitly considers it. Based on the theoretical analysis, we suggest a joint optimization framework that combines the source and target domains. To minimize the joint error, we further introduce Joint Contrastive Learning (JCL) that finds class-level discriminative features. With a solid theoretical framework, JCL employs contrastive loss to maximize the mutual information between a feature and its label, which is equivalent to maximizing the Jensen-Shannon divergence between conditional distributions. Extensive experiments on domain adaptation datasets demonstrate that JCL outperforms existing state-of-the-art methods.

Keywords: adaptation models, deep learning, domain adaptation, transfer learning, contrastive learning

Student Number: 2019-25825

Contents

List of Tables

List of Figures

Chapter 1

Introduction

With the advancement in computational resources, deep neural networks have been successfully adopted in numerous applications and show impressive performance. However, collecting a large volume of labeled data to enable the deep neural networks is often expensive or even impractical. This hurdle became a serious bottleneck to the application of deep learning algorithms. To circumvent this problem, domain adaptation has been introduced [1]. In particular, domain adaptation utilizes labeled data from a source domain to classify the target domain.

The major characteristic of domain adaptation is the dataset shift [2], [3]. In other words, there is a discrepancy between the distributions of the source and target domains. It precludes a small target error when the classifier trained on the source domain is directly applied to the unlabeled target data. To theoretically analyze the target error, there have been approaches to bound it with the source error and the domain discrepancy. In particular, the target error is upper bounded by the sum of the source error, the domain discrepancy, and the error of ideal joint hypothesis [4], while the last term is often treated as constant in the literature. Based on the theoretical analysis, several studies [5]–[10] have endeavored to reduce the discrepancy between the marginal distributions of the domains in representation space.

Matching the source and target feature distributions has advanced domain adap-

tation performance as it enhances feature transferability. However, feature discriminability, which is influential in downstream tasks, is hindered. To illustrate, class-conditional distributions are disregarded in marginal distribution aligning methods. As a result, it is possible that decision boundaries traverse high-density regions of the target domain, rendering the learned classifier vulnerable to misclassification [11], [12].

Recent domain adaptation methods have explored two strategies to enhance feature discriminability. The first strategy is to match the first-order statistics of conditional distributions [13]–[15]. The second strategy is to use pairwise loss [16], [17] or triplet loss [18] to learn discriminative representations [14], [19], [20]. There are two key issues with these methods. First, aligning the source and target class centers can coarsely match the class-conditional distributions, but it can be far from fine alignment and discriminative features. Second, using pairwise loss or triplet loss cannot properly estimate and maximize the mutual information (MI) between a learned feature and its label. The MI between a feature and its label is equivalent to the Jensen-Shannon (JS) divergence between class-conditional distributions. Therefore, maximizing the MI theoretically guarantees the learning of class-wise discriminative representations. In this paper, we propose maximizing the MI to enhance feature discriminability.

In this dissertation, we first address the error of the ideal joint hypothesis, which has not been sufficiently investigated but can significantly affect the upper bound on the target error. To explicitly consider the joint hypothesis error, we propose an alternative upper bound on the target error from the perspective of joint optimization. The joint hypothesis error in the proposed upper bound is directly affected by the hypothesis. Therefore, it is straightforward to implement. Further, we propose a novel Joint Contrastive Learning (JCL) framework to unsupervised domain adaptation, which considers the union distribution of the source and target to minimize the proposed joint hypothesis error. To illustrate, labeled data from the source domain and unlabeled data from the target domain are unified and jointly optimized in this framework. In particular, we enlarge the JS divergence between different class-conditional distributions

of the combined dataset by maximizing the MI between a feature and its label using InfoNCE contrastive loss [21]. Several experiments are conducted and our proposed JCL achieves state-of-the-art results on benchmark datasets. Note that this dissertation is based on the following research [22]:

- Changhwa Park, Jonghyun Lee, Jaeyoon Yoo, Minhoe Hur, and Sungroh Yoon, "Joint Contrastive Learning for Unsupervised Domain Adaptation," Under review.

Chapter 2

Background

2.1 Domain Adaptation

Domain adaptation is exploiting labeled source data to improve the performance of task in the target domain which does not have the label information. We first introduce the setting of unsupervised domain adaptation with conventional notations. With the defined notations, we review a theoretical analysis for the target error upper bound [4].

2.1.1 Problem Setting and Notations

In an unsupervised domain adaptation setting, a set of labeled source data $\{(\boldsymbol{x}_s^i, y_s^i) \in (\mathcal{X} \times \mathcal{Y})\}_{i=1}^n$, sampled i.i.d. from the source domain distribution $\mathcal{D}_S$, and a set of unlabeled target data $\{\boldsymbol{x}_t^j \in \mathcal{X}\}_{j=1}^m$, sampled i.i.d. from the target domain distribution $\mathcal{D}_T$, are available. A domain is defined as a pair comprised of distribution $\mathcal{D}$ in the input space $\mathcal{X}$ and labeling function $f : \mathcal{X} \to \mathcal{Y}$. The output space $\mathcal{Y}$ is $[0, 1]$ in the theoretical analysis. To denote the source and target domains, we use $\langle \mathcal{D}_S, f_S \rangle$ and $\langle \mathcal{D}_T, f_T \rangle$, respectively. The objective of unsupervised domain adaptation is to learn a hypothesis function $h : \mathcal{X} \mapsto \mathcal{Y}$ that provides a good generalization in the target domain. Formally, the error of a hypothesis h with respect to a labeling function f_T under the target domain distribution $\mathcal{D}_T$ is defined as $\epsilon_T(h, f_T) \coloneqq \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}_T}[|h(\boldsymbol{x}) -$

$f_T(\boldsymbol{x})|]$. The introduced notations are summarized in Table 2.1

Table 2.1: Notations

Symbol	Definition
$\mathcal{X}$	An input space
$\mathcal{D}$	A distribution in the input space
f	A labeling function
$\langle \mathcal{D}, f \rangle$	A domain
h	A hypothesis function
$\epsilon(h, f)$	$\mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}}[\lvert h(\boldsymbol{x}) - f(\boldsymbol{x}) \rvert]$

2.1.2 Theoretical Background

We review the theoretical basis of domain adaptation [4]. The ideal joint hypothesis is defined as,

Definition 1 (Ben-David *et al.* [4])**.** Let $\mathcal{H}$ be a hypothesis space. The *ideal joint hypothesis* is the hypothesis which minimizes the combined error:

$$h^* \coloneqq \arg\min_{h \in \mathcal{H}} \epsilon_S(h, f_S) + \epsilon_T(h, f_T).$$

We denote the combined error of the ideal hypothesis by,

$$\lambda \coloneqq \epsilon_S(h^*, f_S) + \epsilon_T(h^*, f_T).$$

Ben-David *et al.* [4] proposed the following theorem for the upper bound on the target error, which is used as the theoretical background for numerous unsupervised domain adaptation methods.

Theorem 1 (Ben-David *et al.* [4])**.** *Let $\mathcal{H}$ be a hypothesis space, then the expected target error is upper bounded as,*

$$\epsilon_T(h, f_T) \leq \epsilon_S(h, f_S) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) + \lambda,$$

where $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T)$ is $\mathcal{H}\Delta\mathcal{H}$-distance between the source and target distributions. Formally,

$$d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) \coloneqq 2 \sup_{h,h' \in \mathcal{H}} |\Pr_{\boldsymbol{x}\sim\mathcal{D}_S}[h(\boldsymbol{x}) \neq h'(\boldsymbol{x})] - \Pr_{\boldsymbol{x}\sim\mathcal{D}_T}[h(\boldsymbol{x}) \neq h'(\boldsymbol{x})]|.$$

Theorem 1 shows that the target error is upper bounded by the sum of the source error, $\mathcal{H}\Delta\mathcal{H}$-distance between the domains, and the error of the ideal joint hypothesis, λ. The last term λ is often considered to be minute. Accordingly, many recent works on domain adaptation endeavored to allow a feature encoder $g : \mathcal{X} \mapsto \mathcal{Z}$ to learn such that the induced distributions of the domains in feature space $\mathcal{Z}$ have a minimal $\mathcal{H}\Delta\mathcal{H}$-distance, while also minimizing the source error [7], [23]–[25].

2.2 Approaches for Domain Adaptation

2.2.1 Marginal Distribution Alignment Based Approaches

As suggested by the theoretical analysis in [4], aligning the marginal distributions can result in reducing target error, and is common practice in domain adaptation [5]–[10]. Long *et al.* [9] utilized the Multiple Kernel variant of Maximum Mean Discrepancy (MK-MMD) to improve the transferability of feature representation from task-specific layers. They matched the mean-embeddings of the multi-layer representations across domains in a reproducing kernel Hilbert space. Long *et al.* [10] also used the MMD to measure the discrepancy in marginal distributions but in a different way. They proposed Joint Maximum Mean Discrepancy (JMMD) to reduce the discrepancy in the joint distributions of the multiple layers. On the other hand, there have been adversarial training based approaches, inspired by Generative Adversarial Networks (GANs) [26].

Ganin and Lempitsky [7] introduced a domain discriminator to convert the domain confusion into a minmax optimization. The domain discriminator is trained to distinguish a feature representation whether the feature originates from the source domain or the target domain. Conversely, a feature extractor is trained to deceive the domain discriminator, and, as a result, the model can learn domain-invariant features. Bousmalis *et al.* [27] hypothesized that modeling both private and shared components of the representations can improve the extraction of domain-invariant features and proposed Domain Separation Networks (DSN). Although matching marginal distributions has brought advances in performance, Shu *et al.* [11] and Zhao *et al.* [28] theoretically demonstrated that finding invariant representations is not sufficient to guarantee a small target error. Moreover, Chen *et al.* [29] empirically revealed an unexpected deterioration in discriminability while learning transferable representations. In particular, Chen *et al.* [29] compared the optimal joint error on the learned feature representation of Domain Adversarial Neural Network (DANN) [7], which attempted to align marginal distributions of the source and target domains, with that of pre-trained ResNet-50 [30]. They found that the optimal joint error on the feature space learned with DANN is much higher than that on the pre-trained feature space. This suggests that merely learning domain-invariant features is susceptible to substantial optimal joint error and loosely bounded target error. This shortcoming of marginal distribution alignment-based methods gave rise to conditional distribution matching approaches.

2.2.2 Conditional Distribution Matching Approaches

Researchers have attempted to learn class-level discriminative features using two main technologies: first-order statistics matching and Siamese-networks training [16]–[18]. Xie *et al.* [15] aligned the source and target centroids to learn semantic representations of the target data. Utilizing the MMD measurement, Kang *et al.* [13] proposed the minimization of intra-class discrepancy and maximizing the inter-class margin. Deng, Luo, and Zhu [14] employed pairwise margin loss to learn discriminative features and

minimized the distances between the first-order statistics to align the conditional distributions. However, there are two main differences between the proposed JCL approach and these methods. First, with the theoretical analysis that explicitly handles the joint hypothesis error, we propose to jointly optimize the source and target domains to have class-wise discriminative representations. Second, we theoretically guarantee learning discriminative features from the perspective of JS divergence by maximizing the MI between a feature and its label. Previous pairwise loss or triplet loss-based methods cannot properly bound and maximize the MI.

2.3 Contrastive Learning

Contrastive learning has been adopted for self-supervised learning and has led to significant performance enhancement. Oord, Li, and Vinyals [21] introduced *InfoNCE* loss to estimate and maximize the MI between a present context and a future signal. High-dimensional data is compressed into a condensed latent embedding to model conditional predictions and autoregressive models are utilized to predict many steps in the future. In the image domain, Hjelm *et al.* [31], Bachman, Hjelm, and Buchwalter [32], and Chen *et al.* [33] maximized the MI between features that originated from the same input with different augmentations. Hjelm *et al.* [31] introduced local Deep InfoMax (local DIM) which maximizes the MI between local features and global features to encode structure information and improve the quality of representation. Bachman, Hjelm, and Buchwalter [32] advanced local DIM with three key modifications: features across different augmented versions of each input are forced to be invariant, features from multiple scales are predicted, and a more powerful encoder is used. Chen *et al.* [33] utilized simple data augmentations, learnable nonlinear transformation, and large batch sizes to enhance the performance. He *et al.* [34] proposed a momentum encoder to accumulate a large number of negative examples and covered the underlying distribution effectively without having substantially large batch sizes. From the

information-theoretic perspective, Tschannen *et al.* [35] suggested that these methods could be subsumed under the same objective, *InfoMax* [36], and provided a different perspective on the success of these methods. As opposed to these methods, the proposed approach maximizes the MI between features from the same class to maximize the JS divergence between different class-conditional distributions.

Chapter 3

Method

3.1 An Alternative Upper Bound

In the previous studies, the ideal joint hypothesis error was assumed to be insignificant, and therefore, it was neglected. However, recent studies [28], [29] have suggested that this error can become substantial, and thus, it must be addressed adequately. To minimize the optimal joint error in Theorem 1, computing the ideal joint hypothesis is needed, but it is usually intractable. As an alternative, we aim to provide an upper bound on the target error, which explicitly incorporates the concept of joint error, and is free from the optimal hypothesis. A small ideal joint hypothesis error implies that there exists a joint hypothesis, which generalizes well on both the source and target domains. Intuitively, it is natural to consider jointly optimizing within the domains to minimize the joint error. From this point of view, we define a combined domain $\langle \mathcal{D}_U, f_U \rangle$ as below.

Definition 2. Let ϕ_S and ϕ_T be the density functions of the source and target distributions, respectively. Then, the distribution of the combined domain $\mathcal{D}_U$ is the mean distribution of the source and target distributions:

$$\phi_U(\boldsymbol{x}) \coloneqq \frac{1}{2}(\phi_S(\boldsymbol{x}) + \phi_T(\boldsymbol{x}))$$

$$f_U(\boldsymbol{x}) \coloneqq \frac{1}{2}(f_S(\boldsymbol{x}) + f_T(\boldsymbol{x}))$$

With the definition of the combined domain, the following theorem holds:

Theorem 2. *Let $\mathcal{H}$ be a hypothesis space, then the expected target error is upper bounded as,*

$$\epsilon_T(h, f_T) \leq \epsilon_S(h, f_S) + \frac{1}{4} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) + 2\epsilon_U(h, f_U).$$

Before continue to the proof of the theorem, We first introduce lemmas that are useful in proving the theorem.

Lemma 1. *Let $\mathcal{H}$ be a hypothesis space and $\mathcal{D}$ be any distribution over input space $\mathcal{X}$. Then $\forall h, h', h'' \in \mathcal{H}$, the following triangle inequality holds:*

$$\epsilon_{\mathcal{D}}(h, h') \leq \epsilon_{\mathcal{D}}(h, h'') + \epsilon_{\mathcal{D}}(h'', h').$$

Proof. From the definition of the error and the triangle inequality of norm, we have

$$\begin{aligned}
\epsilon_{\mathcal{D}}(h, h') &= \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}[|h(\boldsymbol{x}) - h'(\boldsymbol{x})|] \\
&= \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}[|h(\boldsymbol{x}) - h'(\boldsymbol{x}) + h''(\boldsymbol{x}) - h''(\boldsymbol{x})|] \\
&\leq \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}[|h(\boldsymbol{x}) - h''(\boldsymbol{x})| + |h''(\boldsymbol{x}) - h'(\boldsymbol{x})|] \\
&= \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}[|h(\boldsymbol{x}) - h''(\boldsymbol{x})|] + \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}[|h''(\boldsymbol{x}) - h'(\boldsymbol{x})|] \\
&= \epsilon_{\mathcal{D}}(h, h'') + \epsilon_{\mathcal{D}}(h'', h').
\end{aligned}$$

□

Lemma 2 (Ben-David *et al.* [4])**.** *For any hypothesis $h, h' \in \mathcal{H}$, the following inequality holds:*

$$|\epsilon_S(h, h') - \epsilon_T(h, h')| \leq \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T).$$

Proof. From the definition of the $\mathcal{H}\Delta\mathcal{H}$-distance, we have

$$\begin{aligned}
d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) &= 2 \sup_{h,h'\in\mathcal{H}} |\Pr_{\boldsymbol{x}\sim\mathcal{D}_S}[h(\boldsymbol{x}) \neq h'(\boldsymbol{x})] - \Pr_{\boldsymbol{x}\sim\mathcal{D}_T}[h(\boldsymbol{x}) \neq h'(\boldsymbol{x})]| \\
&= 2 \sup_{h,h'\in\mathcal{H}} |\epsilon_S(h, h') - \epsilon_T(h, h')| \\
&\geq 2|\epsilon_S(h, h') - \epsilon_T(h, h')|.
\end{aligned}$$

□

With the introduced lemmas, we can prove Theorem 2 as follows:

Proof. From Lemma 1, we have

$$\begin{aligned}\epsilon_T(h, f_T) &\leq \epsilon_T(h, f_U) + \epsilon_T(f_U, f_T) \\ &= \epsilon_S(h, f_U) + \epsilon_T(h, f_U) + \epsilon_T(f_U, f_T) - \epsilon_S(h, f_U) \\ &\leq \epsilon_S(h, f_U) + \epsilon_T(h, f_U) + \epsilon_T(f_U, f_T) + \epsilon_S(h, f_S) - \epsilon_S(f_U, f_S).\end{aligned}$$

First, using Lemma 2, the following inequality holds:

$$\begin{aligned}&\epsilon_T(f_U, f_T) - \epsilon_S(f_U, f_S) \\ &= \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_T}[|f_U(\boldsymbol{x}) - f_T(\boldsymbol{x})|] - \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_S}[|f_U(\boldsymbol{x}) - f_S(\boldsymbol{x})|] \\ &= \frac{1}{2}\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_T}[|f_S(\boldsymbol{x}) - f_T(\boldsymbol{x})|] - \frac{1}{2}\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_S}[|f_S(\boldsymbol{x}) - f_T(\boldsymbol{x})|] \\ &= \frac{1}{2}\epsilon_T(f_S, f_T) - \frac{1}{2}\epsilon_S(f_S, f_T) \\ &\leq \frac{1}{2}|\epsilon_T(f_S, f_T) - \epsilon_S(f_S, f_T)| \\ &\leq \frac{1}{4}d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T).\end{aligned}$$

Second, from the definition of $\mathcal{D}_U$ and f_U, we have

$$\begin{aligned}&\epsilon_S(h, f_U) + \epsilon_T(h, f_U) \\ &= \int \phi_S(\boldsymbol{x})|h(\boldsymbol{x}) - f_U(\boldsymbol{x})|\mathrm{d}\boldsymbol{x} + \int \phi_T(\boldsymbol{x})|h(\boldsymbol{x}) - f_U(\boldsymbol{x})|\mathrm{d}\boldsymbol{x} \\ &= 2\int \frac{1}{2}(\phi_S(\boldsymbol{x}) + \phi_T(\boldsymbol{x}))|h(\boldsymbol{x}) - f_U(\boldsymbol{x})|\mathrm{d}\boldsymbol{x} \\ &= 2\int \phi_U(\boldsymbol{x})|h(\boldsymbol{x}) - f_U(\boldsymbol{x})|\mathrm{d}\boldsymbol{x} \\ &= 2\epsilon_U(h, f_U).\end{aligned}$$

Combining the above two inequalities and an equality yields the proof. □

Comparison with Theorem 1

The main difference between Theorem 1 and Theorem 2 lies in λ in Theorem 1 and $2\epsilon_U(h, f_U)$ in Theorem 2. To illustrate, λ in Theorem 1 is composed of the ideal joint hypothesis, which is neither tractable nor manageable, and hence, it has been obliquely addressed [14], [19], [37], [38]. On the contrary, $2\epsilon_U(h, f_U)$, the alternative term in Theorem 2, is directly affected by the hypothesis h, and thus, it is straightforward to utilize. Differ from the previous studies that attempt to only alter λ from Theorem 1, Theorem 2 cannot be directly derived from Theorem 1 because the second term in Theorem 2 is smaller than that in Theorem 1.

The main idea here is that joint optimization in the source and target domains is demanded upon simply matching the marginal distributions of the domains. As the target labels are not provided, we must rely on the source labels. However, optimization of the source domain alone can result in poor generalization of the target domain. We therefore combine the source and target domains and propose their joint optimization. To estimate the joint hypothesis error, we resort to target pseudo-labels, and the following theorem holds:

Theorem 3. *Let $\mathcal{H}$ be a hypothesis space, and $f_{\hat{T}}$ be a target pseudo-labeling function. Accordingly, $f_{\hat{U}}$ is defined as, $f_{\hat{U}}(\boldsymbol{x}) \coloneqq \frac{1}{2}(f_S(\boldsymbol{x}) + f_{\hat{T}}(\boldsymbol{x}))$. Then the expected target error is upper bounded as,*

$$\epsilon_T(h, f_T) \leq \epsilon_S(h, f_S) + \frac{1}{4} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) + 2\epsilon_U(h, f_{\hat{U}}) + \epsilon_T(f_T, f_{\hat{T}}).$$

Proof. By Lemma 1, the following inequality holds.

$$\epsilon_T(h, f_T) \leq \epsilon_T(h, f_{\hat{T}}) + \epsilon_T(f_T, f_{\hat{T}}).$$

Meanwhile, using the same process in the proof of Theorem 2, we know that

$$\epsilon_T(h, f_{\hat{T}}) \leq \epsilon_S(h, f_S) + \frac{1}{4} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) + 2\epsilon_U(h, f_{\hat{U}}).$$

Combining the above two inequalities yields the proof. □

3.2 Joint Contrastive Learning

Theorems 2 and 3 suggest that the joint optimization to minimize the combined error of the joint hypothesis is required for better accuracy on the target domain. For the joint optimization, a typical classification framework using cross-entropy can be utilized; however, it is vulnerable to noisy labels [39], which are highly probable in the target pseudo-labels, and often result in poor margins [40]. As an alternative, discriminative feature learning can be used. Discriminative feature learning extracts the semantic features that differentiate dissimilar inputs and can benefit the classification. For instance, in unsupervised learning, which is similar to unsupervised domain adaptation because the target true labels are not available, discriminative feature learning has brought about considerable progress to downstream tasks [21], [33], [34]. In this respect, we utilize the notion of learning discriminative feature representation to minimize the joint hypothesis error.

3.2.1 Theoretical Guarantees

Formally, we aim to learn discriminative features on the intermediate representation space $\mathcal{Z}$ induced through the feature transformation g. We denote the induced distribution of the combined domain $\mathcal{D}_U$ over the representation space $\mathcal{Z}$ as $\mathcal{D}_U^{\mathcal{Z}}$, and its class-conditional distribution as $\mathcal{D}_{U|y}^{\mathcal{Z}}$, where y is a class label. We can then formalize our objective with JS divergence D_{JS} as follows:

$$\max_{\theta_g} D_{\mathrm{JS}}(\mathcal{D}_{U|0}^{\mathcal{Z}} \| \mathcal{D}_{U|1}^{\mathcal{Z}}), \tag{3.1}$$

where θ_g denotes the parameters of the feature encoder g. The values 0 and 1 are the class labels, and hence, the objective means maximizing the divergence between different class-conditional distributions. We first consider binary classification for the simplicity, and then we will generalize the theoretical analysis to multiclass classification problem.

Suppose that the label distribution of the combined domain is uniform, i.e., $P(y = 0) = P(y = 1)$. In practice, this can be achieved by reformulating a dataset to be class-wise uniform. Let Y be a uniform random variable that takes the value in $\{0, 1\}$ and let the distribution $\mathcal{D}^{\mathcal{Z}}_{U|Y}$ be the mixture of $\mathcal{D}^{\mathcal{Z}}_{U|0}$ and $\mathcal{D}^{\mathcal{Z}}_{U|1}$, according to Y. We denote the induced feature random variable with the distribution $\mathcal{D}^{\mathcal{Z}}_{U|Y}$ as $\boldsymbol{Z}_{U|Y}$. From the relation between JS divergence and MI, the following holds.

$$D_{\mathrm{JS}}(\mathcal{D}^{\mathcal{Z}}_{U|0} \| \mathcal{D}^{\mathcal{Z}}_{U|1}) = I(Y; \boldsymbol{Z}_{U|Y})$$

Therefore, we can transform our objective as follows:

$$\max_{\theta_g} I(Y; \boldsymbol{Z}_{U|Y}). \tag{3.2}$$

The MI between a label and a feature that is induced from the distribution conditioned on the label can be maximized using the following approach. We employ the InfoNCE loss proposed by Oord, Li, and Vinyals [21] to estimate and maximize the MI. InfoNCE is defined as,

$$\begin{aligned} I(X;Y) &\geq \mathbb{E}\left[\frac{1}{K}\sum_{i=1}^{K} \log \frac{e^{c(x_i,y_i)}}{\frac{1}{K}\sum_{j=1}^{K} e^{c(x_i,y_j)}}\right] \\ &\triangleq I_{\mathrm{NCE}}(X;Y), \end{aligned} \tag{3.3}$$

where the expectation is over K independent samples from the joint distribution $p(x, y)$ [41]. $c(x, y)$ is a *critic* function used to predict whether the inputs x and y were jointly drawn by yielding high values for the jointly drawn pairs and low values for the others [35].

The proposed JCL framework does not directly pair a feature and its label to maximize the MI between them. Instead, features from the same conditional distribution are paired, and we use I_{NCE} to maximize the MI between them. For a given Y, we sample two different data, $\boldsymbol{X}^{(1)}_{U|Y}$ and $\boldsymbol{X}^{(2)}_{U|Y}$, from the same conditional distribution, $\mathcal{D}^{\mathcal{X}}_{U|Y}$. Then, we obtain $\boldsymbol{Z}^{(1)}_{U|Y}$ and $\boldsymbol{Z}^{(2)}_{U|Y}$ from $\boldsymbol{X}^{(1)}_{U|Y}$ and $\boldsymbol{X}^{(2)}_{U|Y}$, respectively, through

the feature transformation, g. Therefore, Y, $\boldsymbol{X}_{U|Y}^{(1)}$, $\boldsymbol{X}_{U|Y}^{(2)}$, $\boldsymbol{Z}_{U|Y}^{(1)}$, and $\boldsymbol{Z}_{U|Y}^{(2)}$ satisfy the Markov relation:

$$\boldsymbol{Z}_{U|Y}^{(1)} \leftarrow \boldsymbol{X}_{U|Y}^{(1)} \leftarrow Y \rightarrow \boldsymbol{X}_{U|Y}^{(2)} \rightarrow \boldsymbol{Z}_{U|Y}^{(2)}, \tag{3.4}$$

and this is Markov equivalent to

$$\boldsymbol{Z}_{U|Y}^{(1)} \rightarrow \boldsymbol{X}_{U|Y}^{(1)} \rightarrow Y \rightarrow \boldsymbol{X}_{U|Y}^{(2)} \rightarrow \boldsymbol{Z}_{U|Y}^{(2)}. \tag{3.5}$$

By the data processing inequality, we know that

$$I(\boldsymbol{Z}_{U|Y}^{(1)}; \boldsymbol{Z}_{U|Y}^{(2)}) \leq I(Y; \boldsymbol{Z}_{U|Y}^{(1)}). \tag{3.6}$$

Meanwhile, we can observe that the following Markov relation holds.

$$Y \rightarrow (\boldsymbol{X}_{U|Y}^{(1)}, \boldsymbol{X}_{U|Y}^{(2)}) \rightarrow (\boldsymbol{Z}_{U|Y}^{(1)}, \boldsymbol{Z}_{U|Y}^{(2)}) \rightarrow \boldsymbol{Z}_{U|Y}^{(1)}. \tag{3.7}$$

Therefore, by the data processing inequality, we have

$$I(Y; \boldsymbol{Z}_{U|Y}^{(1)}) \leq I(Y; (\boldsymbol{Z}_{U|Y}^{(1)}, \boldsymbol{Z}_{U|Y}^{(2)})). \tag{3.8}$$

Combining Equation 3.6 and Equation 3.8 yields the following inequality.

$$I(\boldsymbol{Z}_{U|Y}^{(1)}; \boldsymbol{Z}_{U|Y}^{(2)}) \leq I(Y; \boldsymbol{Z}_{U|Y}^{(1)}, \boldsymbol{Z}_{U|Y}^{(2)}) \tag{3.9}$$

Therefore, $\max_{\theta_g} I(\boldsymbol{Z}_{U|Y}^{(1)}; \boldsymbol{Z}_{U|Y}^{(2)})$ can be seen as a lower bound for our objective $\max_{\theta_g} I(Y; \boldsymbol{Z}_{U|Y})$, and we optimize it with our InfoNCE loss $\mathcal{L}_{\mathrm{c}}$, as described below.

Comparison with InfoMax Objective.

Comparing our objective,

$$\max_{\theta_g} I(Y; \boldsymbol{Z}_{U|Y}), \tag{3.10}$$

with the InfoMax objective,

$$\max_{\theta_g} I(\boldsymbol{X}; g(\boldsymbol{X})), \tag{3.11}$$

[36] provides instructive insights. Recent progress on unsupervised representation learning [21], [31], [33], [42] can be subsumed under the same objective,

$$\max_{\theta_{g_1}, \theta_{g_2}} I(g_1(\boldsymbol{X}^{(1)}); g_2(\boldsymbol{X}^{(2)})), \tag{3.12}$$

where $\boldsymbol{X}^{(1)}$ and $\boldsymbol{X}^{(2)}$ are instances that originate from the same data [35]. Using the process similar to that derived above, it can be shown that the objective is a lower bound on the InfoMax objective. The main difference is that the InfoMax principle essentially aims to maximize the MI between data and its representation, whereas our objective focuses on maximizing the divergence between different class-conditional distributions in the feature space.

Comparison with Triplet Loss-based Methods.

The multi-class-K-pair loss [43], which is the generalized triplet loss [44], can be shown to be a special case of InfoNCE loss [35], and triplet loss is the same as in the $K = 2$ case. The drawback of using triplet loss to learn discriminative features is that it cannot tightly bound the MI when the MI is larger than $\log K$ because I_{NCE} is upper bounded by $\log K$. Pairwise margin loss also compares only two features, and hence, it is also expected to have a loose bound. Thus, triplet loss or pairwise loss-based domain adaptation methods [14], [19], [20] cannot guarantee class-level discriminative features from an information-theoretic perspective.

3.2.2 Generalization to Multiclass Classification

Here, we introduce how the proposed theoretical background can be generalized to a multiclass classification problem and explain why degenerate solutions can be avoided from an information-theoretic perspective. The generalized Jensen-Shannon (JS) divergence is defined as:

Definition 3 (Lin [45])**.** Let $\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_n$ be n probability distributions with weights $\pi_1, \pi_2, \cdots, \pi_n$, respectively, and let $Z_1, Z_2, \cdots, Z_n$ be random variables with distributions $\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_n$, respectively. Then, the generalized JS divergence is defined as:

$$D_{\mathrm{JS}}^{\pi}(\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_n) = H(Z) - \sum_{i=1}^{n} \pi_i H(Z_i),$$

where π is $(\pi_1, \pi_2, \cdots, \pi_n)$ and Z is a random variable with the mixture distribution of $\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_n$ with weights $\pi_1, \pi_2, \cdots, \pi_n$, respectively.

The generalized JS divergence measures the overall difference among a finite number of probability distributions. Notably, for a fixed π, the Bayes probability of error [46] is minimized if the generalized JS divergence is maximized [45]. With this divergence, we can generalize our objective to learn discriminative features in the representation space, $\mathcal{Z}$, as follows.

$$\max_{\theta_g} D_{\mathrm{JS}}^{\pi}(\mathcal{D}_{U|0}^{\mathcal{Z}}, \mathcal{D}_{U|1}^{\mathcal{Z}}, \cdots, \mathcal{D}_{U|C-1}^{\mathcal{Z}}), \tag{3.13}$$

where π denotes the marginal label distribution and C denotes the number of classes. From the definition of the generalized JS divergence, we know that

$$\begin{aligned}
&D_{\mathrm{JS}}^{\pi}(\mathcal{D}_{U|0}^{\mathcal{Z}}, \mathcal{D}_{U|1}^{\mathcal{Z}}, \cdots, \mathcal{D}_{U|C-1}^{\mathcal{Z}}) \\
&\quad = H(\boldsymbol{Z}_{U|Y}) - \sum_{y=1}^{n} \pi_y H(\boldsymbol{Z}_{U|y}) \\
&\quad = H(\boldsymbol{Z}_{U|Y}) - \sum_{y=1}^{n} P(Y=y) H(\boldsymbol{Z}_{U|Y}|Y=y) \\
&\quad = H(\boldsymbol{Z}_{U|Y}) - H(\boldsymbol{Z}_{U|Y}|Y) \\
&\quad = I(Y; \boldsymbol{Z}_{U|Y}).
\end{aligned} \tag{3.14}$$

Therefore, we can transform our objective as follows:

$$\max_{\theta_g} I(Y; \boldsymbol{Z}_{U|Y}). \tag{3.15}$$

With the same theoretical framework introduced in the main manuscript, we can optimize this objective with the InfoNCE loss.

Avoiding Degenerate Solutions

Algorithms that learn discriminative representations by alternating pseudo-labeling and updating the parameters of the network are susceptible to trivial solutions [47], referred to as degenerate solutions. For example, if the majority of samples are assigned to a few clusters, it is easy to discriminate between features, but this is unfavorable for downstream tasks. The proposed approach can avoid the tendency towards degenerate solutions since the method maximizes the MI between a feature and its label. From Equation 3.14, we can observe that maximizing the MI, $I(Y; \boldsymbol{Z}_{U|Y})$, trades off maximizing the entropy, $H(\boldsymbol{Z}_{U|Y})$, and minimizing the conditional entropy, $H(\boldsymbol{Z}_{U|Y}|Y)$. Only minimizing the conditional entropy can be vulnerable to the degenerate solutions, but the objective also includes the entropy maximization, which cannot be achieved in the degenerate solutions. Therefore, the objective naturally balances discriminative representation learning with dispersed features and avoids the degenerate solutions.

3.2.3 Training Procedure

In this section, we formulate the loss functions and architecture of the method based on the aforementioned theoretical frameworks. The overview of JCL is illustrated in Figure 3.1.

Momentum Contrast (MoCo) [34] is adopted as the proposed contrastive learning structure, with an encoder g_q with parameters θ_q and a momentum-updated encoder g_k with parameters θ_k for feature transformation $\mathcal{X} \mapsto \mathcal{Z}$. θ_k are updated by $\theta_k \leftarrow m\theta_k + (1-m)\theta_q$, where $m \in [0, 1)$ is a momentum coefficient. A fully connected (FC) layer projection head $l : \mathcal{Z} \mapsto \mathcal{W}$ is implemented to map the encoded representations

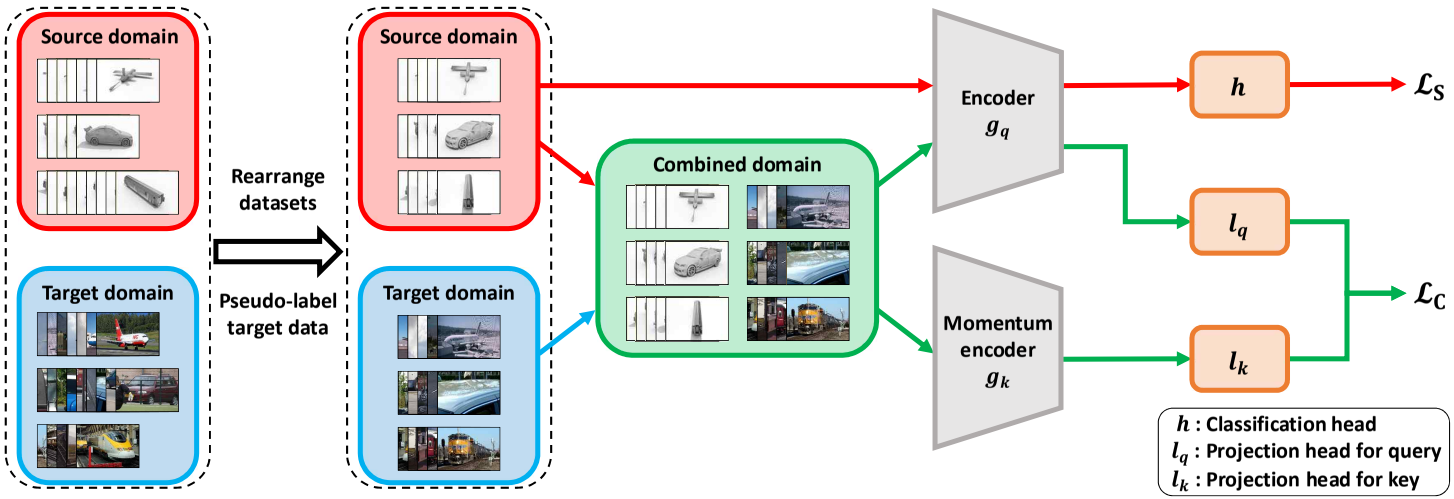


Figure 3.1: An overview of the JCL training process. To alleviate the problem of tradeoff between a small joint error and marginal distribution alignment when the label distributions are substantially different [28], we propose rearranging the datasets to obtain uniform label distributions. For contrastive learning, we adopt *Momentum Contrast* [34], which maintains a queue and a moving-averaged encoder on-the-fly to enable a large and consistent dictionary. After the encoders, a fully connected (FC) layer is applied for classification, and another FC layer is employed for contrastive loss.

to the space where the InfoNCE loss is applied. Empirical tests determine that it is beneficial to define InfoNCE loss in the projected space $\mathcal{W}$ rather than $\mathcal{Z}$, which is in agreement with the results of Chen *et al.* [33]. For the feature pairs in the InfoNCE loss, an encoded query $\boldsymbol{w}_q = l_q(g_q(\boldsymbol{x}))$ and a key $\boldsymbol{w}_k = l_k(g_k(\boldsymbol{x}))$ from the queue of encoded features are used, where l_q is a FC layer projection head for a query and l_k is a FC layer projection head for a key. We obtain the new keys on-the-fly by the momentum encoder and retain the queue of keys. For the critic function c, we employ a cosine similarity function $\mathrm{sim}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^\intercal \boldsymbol{v} / \|\boldsymbol{u}\| \|\boldsymbol{v}\|$ with a temperature hyper-parameter τ according to [48]. Our InfoNCE loss $\mathcal{L}_\mathrm{c}$ is then formulated as follows:

$$\mathcal{L}_\mathrm{c} = \mathbb{E}_{w_q \sim \mathcal{D}_U^W, w_k^+} \left[-\log \frac{\mathrm{e}^{\mathrm{sim}(w_q, w_k^+)/\tau}}{\sum_{w_k \in N_k \cup \{w_k^+\}} \mathrm{e}^{\mathrm{sim}(w_q, w_k)/\tau}} \right], \tag{3.16}$$

where w_k^+ is a feature that has the same label as w_q and N_k is a set of features that have different labels from w_q. For the classification task, we have another FC layer h as a classification head. To guarantee a small source error, we employ the broadly used cross-entropy loss,

$$\mathcal{L}_\mathrm{s} = \mathbb{E}_{(\boldsymbol{x}_s, y_s) \sim \mathcal{D}_S} \Big[-\log h(g_q(\boldsymbol{x}_s))_{y_s} \Big]. \tag{3.17}$$

Combining $\mathcal{L}_\mathrm{s}$ and $\mathcal{L}_\mathrm{c}$ with a hyper-parameter γ, the overall objective is formulated as follows:

$$\min_\theta \mathcal{L}_\mathrm{s} + \gamma \mathcal{L}_\mathrm{c}. \tag{3.18}$$

The labels of the target data are required to recognize whether or not the two samples of the combined domain have the same label. To facilitate this, we generate the pseudo-labels of the target data. In particular, we perform spherical K-means clustering of target data on the feature space $\mathcal{Z}$ and assign labels at the begining of each epoch. If the distance between a target sample and its assigned cluster center is larger than a constant d, then the target sample is excluded from the combined dataset.

Zhao *et al.* [28] showed that if the marginal label distributions of source and target domains are substantially different, a small joint error is not achievable while finding an invariant representation. To address this problem, we suggest the reformulation of datasets to provide uniform label distributions, in which the number of data per class is equalized by data rearrangement.

The pseudo code of JCL is provided in Algorithm 1.

Algorithm 1: Training procedure for JCL

Input : Labeled source data from $\mathcal{D}_S$ and unlabeled target data from $\mathcal{D}_T$.

Initialize encoders g_q and g_k; classification head h; projection heads l_q and l_k; and a queue of K keys;

while *iteration* $<$ *max_iteration* **do**

Cluster the target data using spherical K-means;

Split them into a certain dataset with pseudo-labels and an uncertain dataset;

Rearrange the source and certain target datasets to obtain uniform label distributions;

for $i \leftarrow 1$ **to** *iterations_per_epoch* **do**

Sample mini-batches of the source data $(\boldsymbol{x}_s, y_s)$, certain target data $(\boldsymbol{x}_{tc}, \hat{y}_{tc})$, and uncertain target data $(\boldsymbol{x}_{tu})$;

$\boldsymbol{x}_s^q = \text{pre-process}(\boldsymbol{x}_s)$, $\boldsymbol{x}_s^k = \text{pre-process}(\boldsymbol{x}_s)$;

$\boldsymbol{x}_{tc}^q = \text{pre-process}(\boldsymbol{x}_{tc})$, $\boldsymbol{x}_{tc}^k = \text{pre-process}(\boldsymbol{x}_{tc})$;

$\boldsymbol{x}_{tu} = \text{pre-process}(\boldsymbol{x}_{tu})$;

$\boldsymbol{z}_s^q = g_q(\boldsymbol{x}_s^q)$, $\boldsymbol{z}_s^k = g_k(\boldsymbol{x}_s^k)$;

Compute $\mathcal{L}_\text{s}$ on $(h(\boldsymbol{z}_s^q), y_s)$ using Equation (5);

$\boldsymbol{w}_s^q = l_q(\boldsymbol{z}_s^q)$, $\boldsymbol{w}_s^k = l_k(\boldsymbol{z}_s^k)$;

$\boldsymbol{w}_{tc}^q = l_q(g_q(\boldsymbol{x}_{tc}^q))$, $\boldsymbol{w}_{tc}^k = l_k(g_k(\boldsymbol{x}_{tc}^k))$;

Forward the uncertain target data to train the batch normalization layers, $g_q(\boldsymbol{x}_{tu})$;

Merge $\boldsymbol{w}_s^q$ and $\boldsymbol{w}_{tc}^q$ to obtain $\boldsymbol{w}_u^q$, and merge $\boldsymbol{w}_s^k$ and $\boldsymbol{w}_{tc}^k$ to obtain $\boldsymbol{w}_u^k$;

$\text{enqueue}(\text{queue}, \boldsymbol{w}_u^k),\ \text{dequeue}(\text{queue})$;

Compute $\mathcal{L}_\text{c}$ on $(\boldsymbol{w}_u^q, \text{queue})$ using Equation (4);

Update the query network parameters, θ_q with SGD;

Momentum update the key network parameters, θ_k;

end

end

Chapter 4

Experiments

4.1 Datasets and Baselines

ImageCLEF-DA[1] is a real-world dataset consisting of three domains: *Caltech-256* (**C**), *ImageNet ILSVRC 2012* (**I**), and *Pascal VOC 2012* (**P**). Each domain contains 600 images from 12 common classes. We evaluated all six possible transfer tasks among these three domains.

Office-Home [49] is a more challenging domain adaptation dataset than ImageCLEF-DA. It contains objects commonly found in office and home environments and has four different domains: artistic images (**Ar**), clip art (**Cl**), product images (**Pr**), and real-world images (**Rw**). There are around 15,500 images in 65 different categories in the dataset. We construct all twelve possible transfer tasks among the four domains of the dataset.

VisDA-2017 [50] is a dataset for the synthetic-to-real transfer task and has a high dataset shift. It includes 152,397 synthetic 2D renderings of 3D models and 55,388 real images across 12 classes. The gallery of VisDA-2017 dataset is provided in Figure 4.1

Baselines. We compare JCL with marginal distribution matching methods: Deep Adaptation Network (**DAN**) [9], Domain Adversarial Neural Network (**DANN**) [23],

[1] `https://www.imageclef.org/2014/adaptation`

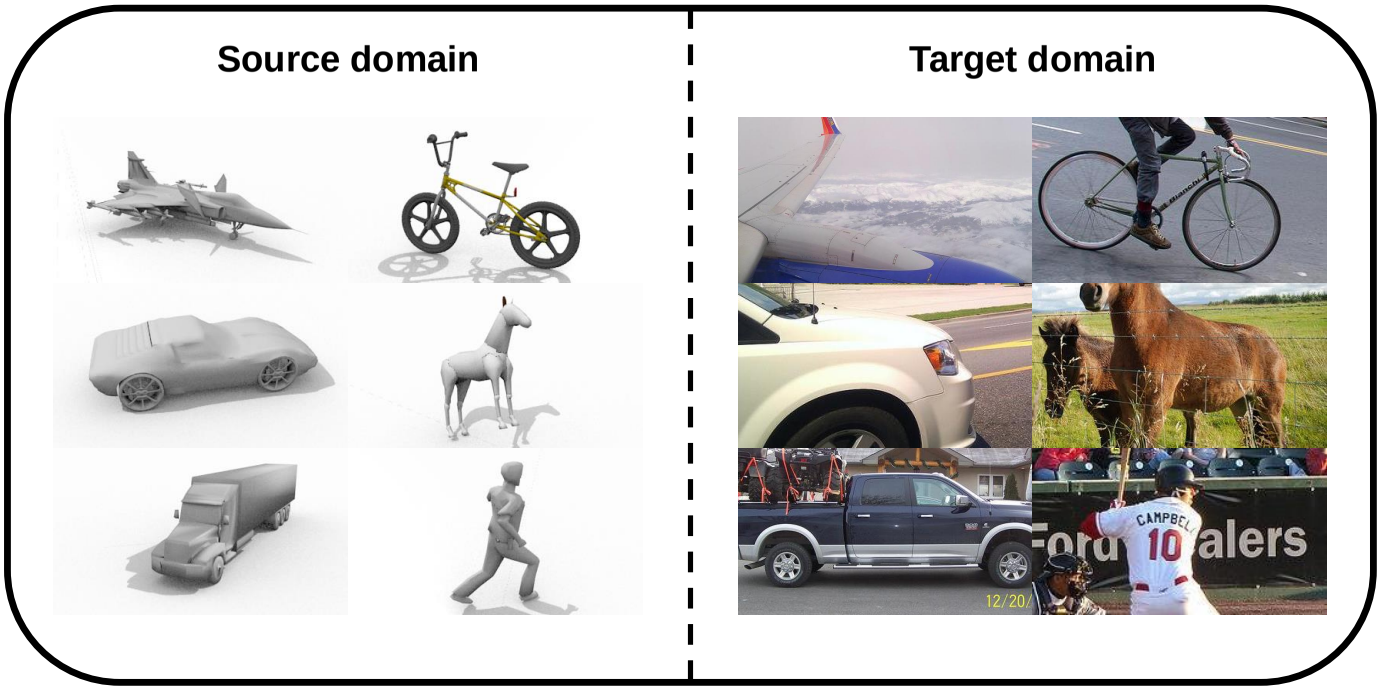


Figure 4.1: The gallery of VisDA-2017 dataset.

and Joint Adaptation Network (**JAN**) [10] and also with methods that endeavor to learn discriminative features: Multi-Adversarial Domain Adaptation (**MADA**) [25], Conditional Domain Adversarial Network (**CDAN**) [51], Adversarial Dropout Regularization (**ADR**) [52], Maximum Classifier Discrepancy (**MCD**) [53], Batch Spectral Penalization (**BSP**) [29], Cluster Alignment with a Teacher (**CAT**) [14], Contrastive Adaptation Network (**CAN**) [13], and Adversarial-Learned Loss for Domain Adaptation (**ALDA**) [54].

4.2 Implementation Details

We follow the standard experimental protocols for unsupervised domain adaptation [7], [10] and report the average accuracy over three independent runs. To select the hyper-parameters, we use the same protocol as the one described in [9]: we train a source classifier and a domain classifier on a validation set that consists of labeled source data and unlabeled target data, and then, we jointly evaluate the test errors of the classifiers. We tuned the weight hyper-parameter, γ, and distance threshold, d, for filtering the certain target data. The weight hyper-parameter, γ, was searched within $\{0.1, 0.5, 1.0, 2.0\}$ for ImageCLEF-DA and Office-Home datasets and $\{0.2, 0.3, 0.4, 0.5\}$ for VisDA-2017 dataset. The distance threshold hyper-parameter, d, was searched within $\{0.05, 0.1, 1.0\}$. The selected hyper-parameters for each task are listed in Table 4.1.

We adopt **ResNet-50** [30] for the ImageCLEF-DA and Office-Home datasets, and **ResNet-101** for the VisDA-2017 dataset as base networks. Batch normalization layers are specified to be domain-specific. We finetune from ImageNet [55] pre-trained models, with the exception of the last FC layer, which we replace with the task-specific FC layer. We also add another FC layer with an output dimension of 256 for contrastive learning. We utilize mini-batch SGD with momentum of 0.9 and follow the same learning rate schedule as [9], [10], [23]: the learning rate η_p is adjusted accord-

Table 4.1: Selected hyper-parameters for each task

Source	Target	γ	d
I	P	0.5	0.05
P	I	1.0	0.1
I	C	2.0	0.1
C	I	2.0	0.1
C	P	0.1	1.0
P	C	1.0	0.1
Ar	Cl	2.0	0.1
Ar	Pr	2.0	0.1
Ar	Rw	0.1	0.1
Cl	Ar	2.0	0.1
Cl	Pr	0.5	0.1
Cl	Rw	0.5	0.1
Pr	Ar	1.0	0.1
Pr	Cl	1.0	0.1
Pr	Rw	2.0	0.1
Rw	Ar	0.1	0.1
Rw	Cl	0.1	0.1
Rw	Pr	0.5	0.1
VisDA-2017 Training	VisDA-2017 Validation	0.3	0.1

ing to $\eta_p = \eta_0(1 + \alpha p)^{-\beta}$, where p is the training progress that increases from 0 to 1. The η_0 is the initial learning rate, which is set to 0.001 for the pre-trained layers and 0.01 for the added FC layers. The α and β are fixed to 10 and 0.75, respectively. The temperature parameter, τ, for the critic function was fixed to 0.05. For ImageCLEF-DA, Office-Home, and VisDA-2017 datasets, the queue size, considering the dataset sizes, was set to 4,096, 2,048, and 32,768, respectively, and the momentum coefficient, m, of the momentum encoder to 0.9, 0.9, and 0.99, respectively. For the metric measuring the distances in the feature space, $\mathcal{Z}$, cosine dissimilarity was applied. At the end of the encoders, we added L2 normalization layers. Unlike other contrastive learning methods, we did not utilize additional data augmentation for fair comparison with domain adaptation baselines; only random crop and horizontal flip were used. We empirically found that it is beneficial to forward pass the uncertain target data to train the batch normalization layers. The computing infrastructure used for running experiments is specified in Table 4.2.

Table 4.2: Computing infrastructure specifications

Item	Details
GPU	GeForce RTX 2080 Ti
CPU	Intel Core i9-10940X
RAM	128 GB
Operating system	Ubuntu 18.04
Libraries	python==3.8.5 pytorch==1.6.0

The total iterations for the ImageCLEF-DA, Office-Home, and VisDA-2017 experiments were 20,000, 10,000, and 50,000, respectively, and they took 4 h, 3 h, and 15 h, respectively, on an average.

4.3 Results

The results obtained using the ImageCLEF-DA dataset are reported in Table 4.3. The accuracies of the compared methods are directly reported from their original papers wherever available. For all six adaptation scenarios, our proposed method outperforms the other baseline methods and achieves state-of-the-art accuracy. In particular, the proposed method surpasses CAT by a substantial margin, validating the effectiveness of jointly learning discriminative features and the discussed information-theoretic guarantees. Moreover, the methods that consider conditional distributions achieve higher accuracies than those that focus on marginal distribution matching. These results suggest that learning discriminative features to minimize the joint hypothesis error is more crucial than general alignment.

The classification accuracies on the Office-Home dataset for unsupervised domain adaptation are shown in Table 4.4. For 9 out of 12 adaptation tasks, the proposed method surpasses the other compared methods by a large margin. In particular, JCL enhances the average accuracy of ALDA by 2.4%, achieving state-of-the-art performance.

In Table 4.5, the accuracy obtained for each class and the average accuracy over all twelve classes on the VisDA-2017 transfer task are reported. Among the twelve objects, "*truck*" is the most challenging object as the baselines show mediocre accuracies. Notably, the proposed method boosts the accuracy of the *truck* class by a significant margin, and, on average, it outperforms the other baseline methods. In particular, it advances the lowest accuracy among the twelve objects of CAN (59.9%) by 6.9%. These results can be attributed to the MI maximization between a feature and its label which trades-off maximizing entropy $H(\boldsymbol{z})$ and minimizing conditional entropy $H(\boldsymbol{z}|y)$, and thus avoids degenerate solutions [47].

We visualize the learned target representations of the VisDA-2017 task by t-SNE [57] in Figure 4.2 to compare our method with DANN in terms of feature discriminability. While aligning the marginal distributions of the source and target domains,

Table 4.3: Accuracy (%) on ImageCLEF-DA for unsupervised domain adaptation (ResNet-50)

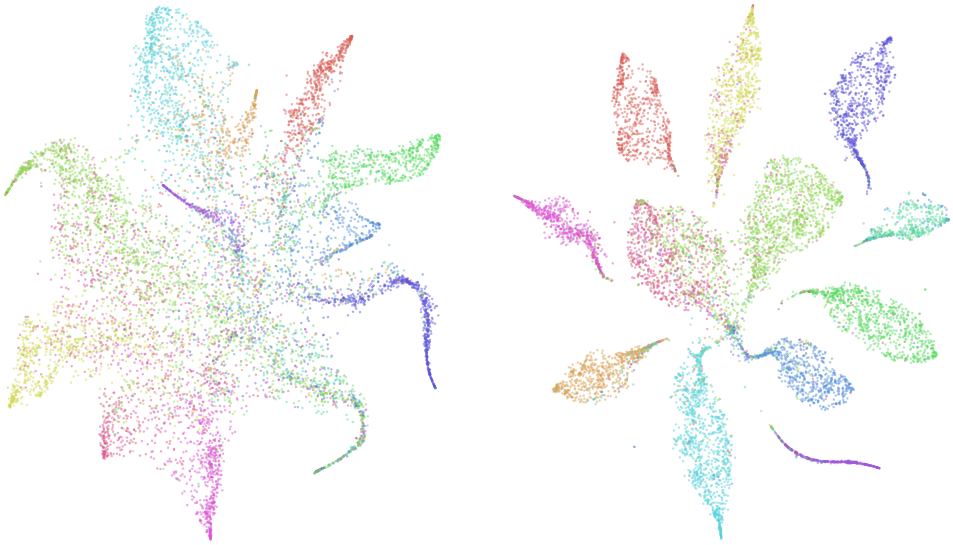
Method	I → P	P → I	I → C	C → I	C → P	P → C	Avg
ResNet-50 [30]	74.8 ± 0.3	83.9 ± 0.1	91.5 ± 0.3	78.0 ± 0.2	65.5 ± 0.3	91.2 ± 0.3	80.7
DANN [23]	75.0 ± 0.6	86.0 ± 0.3	96.2 ± 0.4	87.0 ± 0.5	74.3 ± 0.5	91.5 ± 0.6	85.0
DAN [9]	74.5 ± 0.4	82.2 ± 0.2	92.8 ± 0.2	86.3 ± 0.4	69.2 ± 0.4	89.8 ± 0.4	82.5
JAN [10]	76.8 ± 0.4	88.0 ± 0.2	94.7 ± 0.2	89.5 ± 0.3	74.2 ± 0.3	91.7 ± 0.3	85.8
MADA [25]	75.0 ± 0.3	87.9 ± 0.2	96.0 ± 0.3	88.8 ± 0.3	75.2 ± 0.2	92.2 ± 0.3	85.8
CDAN+E [51]	77.7 ± 0.3	90.7 ± 0.2	**97.7** ± 0.3	91.3 ± 0.3	74.2 ± 0.2	94.3 ± 0.3	87.7
CAT [14]	77.2 ± 0.2	91.0 ± 0.3	95.5 ± 0.3	91.3 ± 0.3	75.3 ± 0.6	93.6 ± 0.5	87.3
JCL	**78.1** ± 0.3	**93.4** ± 0.2	**97.7** ± 0.2	**93.5** ± 0.4	**78.1** ± 0.7	**97.7** ± 0.4	**89.8**

Table 4.4: Accuracy (%) on Office-Home for unsupervised domain adaptation (ResNet-50)

Method	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
ResNet-50 [30]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DANN [23]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
DAN [9]	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
JAN [10]	45.9	61.2	68.9	50.4	59.7	61.0	45.8	43.4	70.3	63.9	52.4	76.8	58.3
CDAN+E [51]	50.7	70.6	76.0	57.6	70.0	70.0	57.4	50.9	77.3	70.9	56.7	81.6	65.8
BSP [29]	52.0	68.6	76.1	58.0	70.3	70.2	58.6	50.2	77.6	**72.2**	**59.3**	81.9	66.3
ALDA [54]	**53.7**	70.1	76.4	60.2	72.6	71.5	56.8	51.9	77.1	70.2	56.3	82.1	66.6
JCL	**53.7**	**77.3**	**77.0**	**65.1**	**73.7**	**75.6**	**64.9**	**52.1**	**80.3**	68.7	55.9	**83.2**	**69.0**

Table 4.5: Accuracy (%) on VisDA-2017 for unsupervised domain adaptation (ResNet-101)

Method	airplane	bicycle	bus	car	horse	knife	motorcycle	person	plant	skateboard	train	truck	Average
ResNet-101 [30]	72.3	6.1	63.4	**91.7**	52.7	7.9	80.1	5.6	90.1	18.5	78.1	25.9	49.4
DANN [23]	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	54.6	82.8	7.8	57.4
DAN [9]	68.1	15.4	76.5	87.0	71.1	48.9	82.3	51.5	88.7	33.2	**88.9**	42.2	62.8
JAN [10]	75.7	18.7	82.3	86.3	70.2	56.9	80.5	53.8	92.5	32.2	84.5	54.5	65.7
MCD [53]	87.0	60.9	83.7	64.0	88.9	79.6	84.7	76.9	88.6	40.3	83.0	25.8	71.9
ADR [52]	87.8	79.5	83.7	65.3	92.3	61.8	88.9	73.2	87.8	60.0	85.5	32.3	74.8
SE [56]	95.9	87.4	**85.2**	58.6	96.2	95.7	90.6	80.0	94.8	90.8	88.4	47.9	84.3
BSP [29]	92.4	61.0	81.0	57.5	89.0	80.6	90.1	77.0	84.2	77.9	82.1	38.4	75.9
CAN [13]	**97.0**	87.2	82.5	74.3	**97.8**	**96.2**	**90.8**	80.7	**96.6**	**96.3**	87.5	59.9	87.2
ALDA [54]	93.8	74.1	82.4	69.4	90.6	87.2	89.0	67.6	93.4	76.1	87.7	22.2	77.8
JCL	**97.0** $\pm$ 0.1	**91.3** $\pm$ 0.5	84.5 $\pm$ 1.7	66.8 $\pm$ 2.1	96.1 $\pm$ 0.3	95.6 $\pm$ 0.6	89.8 $\pm$ 0.8	**81.5** $\pm$ 0.7	94.7 $\pm$ 0.2	95.6 $\pm$ 0.9	86.1 $\pm$ 0.3	**71.8** $\pm$ 0.4	**87.6** $\pm$ 0.2



(a) t-SNE of DANN. (b) t-SNE of JCL.

Figure 4.2: Visualization for different methods (best viewed in color).

the features are not well discriminated with DANN. On the contrary, the target features learned using our method are clearly discriminated, demonstrating that our objective to maximize the JS divergence between conditional distributions is achieved.

4.4 Ablation Studies

To investigate the effectiveness of our method in minimizing the joint hypothesis error by learning discriminative representations, we conduct the same pilot analysis as Chen *et al.* [29]; we train a linear classifier on the representations learned using DANN and our method. The linear classifier is trained on both source and target data using the labels. The average error rate of the linear classifier corresponds to half of the ideal joint hypothesis error. The results are shown in Figure 4.3. We can observe that the ideal joint hypothesis error of the representation learned using our method is significantly lower than that learned using DANN. This implies that the proposed method is effective in achieving our objective to enhance feature discriminability.

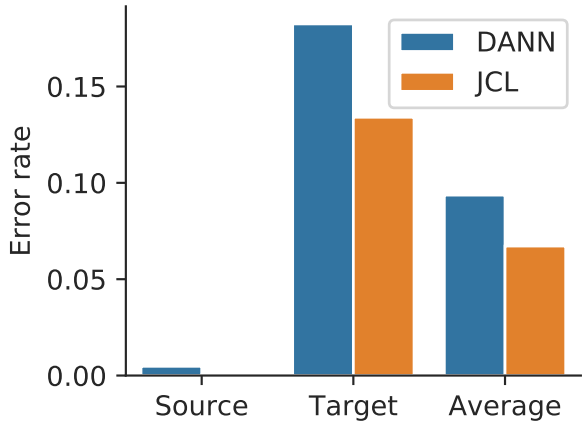
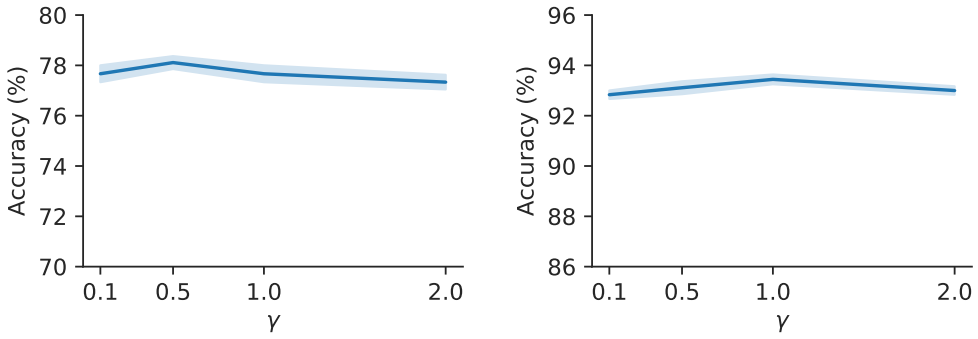


Figure 4.3: Classification error rate on the learned representations.

We investigate the sensitivity of JCL to the weight hyper-parameter γ, and the results are shown in Figure 4.4. We could observe that JCL is not sensitive to the



(a) $\mathbf{I} \rightarrow \mathbf{P}$.

(b) $\mathbf{P} \rightarrow \mathbf{I}$.

Figure 4.4: The accuracy sensitivity of JCL to γ. The results for other tasks are similar.

change in the value of γ.

Chapter 5

Conclusion

In this study, we suggest an alternative upper bound on the target error to explicitly manage the joint hypothesis error. The proposed upper bound with the joint hypothesis error provides a new perspective on the target error that the joint optimization on the both domains is demanded. Further, a novel approach to domain adaptation, JCL, is proposed to minimize the joint error. The proposed approach differs from previous domain adaptation methods that consider conditional distributions, as it can maximize the JS divergence between class-conditional distributions with information-theoretic guarantees. The effectiveness of the proposed method is validated with several experiments.

In chapter 2, we introduced the problem setting of unsupervised domain adaptation and explained theoretical background that brings the upper bound on the target error. We reviewed previous studies that aligned marginal distributions to minimize the upper bound. To improve the target accuracy further, there have been conditional distributions matching methods, and we described these works. Finally, contrastive learning, which is closely related to this work, was also summarized.

In chapter 3, we suggested an alternative upper bound on the target error that explicitly address the joint hypothesis error. Compared with the previous theoretical analysis, it enables managing the joint hypothesis error which can affect the target error

rate severely. Based on this theoretical framework, we introduced Joint Contrastive Learning scheme, which theoretically guarantees maximizing the JS divergence between class-conditional distributions to minimize the joint error.

In chapter 4, we demonstrated the experimental results on several domain adaptation datasets, including ImageCLEF-DA, Office-Home, and VisDA-2017. For most of the adaptation scenarios, our proposed method outperformed the other baseline methods and showed its effectiveness. Moreover, the proposed method brought lower classification error rate on the learned representation compared to the baseline method DANN, elucidating the reduced joint error as intended.

The proposed method advances domain adaptation performance, but this study has potential limitations. First, the target error of the learned model depends on the quality of the pseudo target pseudo-labels. Although we can expect the target pseudo-labels are generally correct since the source and target domains are similar, the target pseudo-labels can be completely erroneous when the dataset shift is substantial. In the worst case, the model will learn totally mistaken feature representations and the performance of the model may collapse. Second, uncertain target samples that are far from the closest cluster center are abandoned and unused, whereas certain target samples are fixed to one-hot pseudo-labels. This one-hot encoding procedure prevents the model from exploiting the uncertainty information of each sample. If the model can take the advantage of uncertainty information of each sample and utilize both the uncertain and certain samples, the accuracy of the learned model can be improved further.

From the limitations of this study, we suggest the following topics to be addressed in the future. First, research on how to improve the quality of target pseudo-labels can be conducted. Advancements in clustering the source and target samples together and assigning the correct target pseudo-labels can benefit domain adaptation methods using target pseudo-labels. Second, how to give soft pseudo-labels to target samples that can extract the manifold information of source and target can be studied. Third, with the soft pseudo-labels, how can positive or negative pairs be defined for discriminative

learning is also an interesting research direction.

Bibliography

[1] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[2] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset shift in machine learning*. The MIT Press, 2009.

[3] A. Gretton, A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Schölkopf, "Covariate shift by kernel mean matching," *Dataset shift in machine learning*, vol. 3, no. 4, p. 5, 2009.

[4] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[5] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," *arXiv preprint arXiv:1412.3474*, 2014.

[6] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *European conference on computer vision*, Springer, 2016, pp. 443–450.

[7] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 37, Lille, France: PMLR, 2015, pp. 1180–1189.

[8] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7167–7176.

[9] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 37, Lille, France: PMLR, 2015, pp. 97–105.

[10] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, JMLR. org, 2017, pp. 2208–2217.

[11] R. Shu, H. Bui, H. Narui, and S. Ermon, "A DIRT-t approach to unsupervised domain adaptation," in *International Conference on Learning Representations*, 2018.

[12] J. Yoo, C. Park, Y. Hong, and S. Yoon, "Learning condensed and aligned features for unsupervised domain adaptation using label propagation," *arXiv preprint arXiv:1903.04860*, 2019.

[13] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4893–4902.

[14] Z. Deng, Y. Luo, and J. Zhu, "Cluster alignment with a teacher for unsupervised domain adaptation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9944–9953.

[15] S. Xie, Z. Zheng, L. Chen, and C. Chen, "Learning semantic representations for unsupervised domain adaptation," in *International Conference on Machine Learning*, 2018, pp. 5423–5432.

[16] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a" siamese" time delay neural network," in *Advances in neural information processing systems*, 1994, pp. 737–744.

[17] Y. Luo, J. Zhu, M. Li, Y. Ren, and B. Zhang, "Smooth neighbors on teacher graphs for semi-supervised learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8896–8905.

[18] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[19] D.-D. Chen, Y. Wang, J. Yi, Z. Chen, and Z.-H. Zhou, "Joint semantic domain alignment and target classifier learning for unsupervised domain adaptation," *arXiv preprint arXiv:1906.04053*, 2019.

[20] S. Dai, Y. Cheng, Y. Zhang, Z. Gan, J. Liu, and L. Carin, "Contrastively smoothed class alignment for unsupervised domain adaptation," *arXiv preprint arXiv:1909.05288*, 2019.

[21] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[22] C. Park, J. Lee, J. Yoo, M. Hur, and S. Yoon, "Joint contrastive learning for unsupervised domain adaptation," *arXiv preprint arXiv:2006.10297*, 2020.

[23] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. March, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016.

[24] H. Zhao, S. Zhang, G. Wu, J. M. F. Moura, J. P. Costeira, and G. J. Gordon, "Adversarial multiple source domain adaptation," in *Advances in Neural Information Processing Systems 31*, Curran Associates, Inc., 2018, pp. 8559–8570.

[25] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[26] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[27] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29, Curran Associates, Inc., 2016, pp. 343–351. [Online]. Available: https://proceedings.neurips.cc/paper/2016/file/45fbc6d3e05ebd93369ce542e Paper.pdf.

[28] H. Zhao, R. T. D. Combes, K. Zhang, and G. Gordon, "On learning invariant representations for domain adaptation," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 97, Long Beach, California, USA: PMLR, 2019, pp. 7523–7532.

[29] X. Chen, S. Wang, M. Long, and J. Wang, "Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation," in *International Conference on Machine Learning*, 2019, pp. 1081–1090.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[31] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *International Conference on Learning Representations*, 2019.

[32] P. Bachman, R. D. Hjelm, and W. Buchwalter, "Learning representations by maximizing mutual information across views," in *Advances in Neural Information Processing Systems*, 2019, pp. 15 509–15 519.

[33] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *arXiv preprint arXiv:2002.05709*, 2020.

[34] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," *arXiv preprint arXiv:1911.05722*, 2019.

[35] M. Tschannen, J. Djolonga, P. K. Rubenstein, S. Gelly, and M. Lucic, "On mutual information maximization for representation learning," in *International Conference on Learning Representations*, 2020.

[36] R. Linsker, "Self-organization in a perceptual network," *Computer*, vol. 21, no. 3, pp. 105–117, 1988.

[37] C. Chen, W. Xie, W. Huang, Y. Rong, X. Ding, Y. Huang, T. Xu, and J. Huang, "Progressive feature alignment for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 627–636.

[38] K. Saito, Y. Ushiku, and T. Harada, "Asymmetric tri-training for unsupervised domain adaptation," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, JMLR. org, 2017, pp. 2988–2997.

[39] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Advances in Neural Information Processing Systems 31*, Curran Associates, Inc., 2018, pp. 8778–8788.

[40] G. Elsayed, D. Krishnan, H. Mobahi, K. Regan, and S. Bengio, "Large margin deep networks for classification," in *Advances in Neural Information Processing Systems 31*, Curran Associates, Inc., 2018, pp. 842–852.

[41] B. Poole, S. Ozair, A. Van Den Oord, A. Alemi, and G. Tucker, "On variational bounds of mutual information," in *Proceedings of the 36th International Con-*

ference on Machine Learning, ser. Proceedings of Machine Learning Research, vol. 97, Long Beach, California, USA: PMLR, 2019, pp. 5171–5180.

[42] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," *arXiv preprint arXiv:1906.05849*, 2019.

[43] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Advances in neural information processing systems*, 2016, pp. 1857–1865.

[44] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Journal of Machine Learning Research*, vol. 10, no. Feb, pp. 207–244, 2009.

[45] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information theory*, vol. 37, no. 1, pp. 145–151, 1991.

[46] M. Hellman and J. Raviv, "Probability of error, equivocation, and the chernoff bound," *IEEE Transactions on Information Theory*, vol. 16, no. 4, pp. 368–372, 1970.

[47] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 132–149.

[48] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3733–3742.

[49] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *(IEEE) Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[50] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, and K. Saenko, *Visda: The visual domain adaptation challenge*, 2017. eprint: `arXiv:1710.06924`.

[51] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Advances in Neural Information Processing Systems*, 2018, pp. 1640–1650.

[52] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Adversarial dropout regularization," in *International Conference on Learning Representations*, 2018.

[53] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3723–3732.

[54] M. Chen, S. Zhao, H. Liu, and D. Cai, "Adversarial-learned loss for domain adaptation.," in *AAAI*, 2020, pp. 3521–3528.

[55] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.

[56] G. French, M. Mackiewicz, and M. Fisher, "Self-ensembling for visual domain adaptation," in *International Conference on Learning Representations*, 2018.

[57] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

초 록

도메인 적응 기법은 타겟 도메인의 라벨 정보가 없는 상황에서 비슷한 도메인인 소스 도메인의 라벨 정보를 활용하기 위해 개발되었다. 기존의 방법론들은 잠재 공간에서 도메인들 사이의 분포 차이를 줄임으로써 전이 학습이 가능하게 하였다. 이러한 기법들은 소스 도메인의 에러율, 도메인 간 분포 차이, 그리고 양 도메인에서 이상적인 분류기의 에러율의 합이 타겟 도메인의 에러율의 상계가 된다는 이론을 바탕으로 한다. 그러나 도메인들 사이의 분포 차이를 줄이는 방법들은 동시에 잠재 공간에서 서로 다른 라벨을 갖는 데이터들 사이의 구별성을 감소시켰다. 특히, 작을 것이라 생각되던 양 도메인에서 이상적인 분류기의 에러율이 큰 것으로 나타났다.

본 논문에서는 기존의 이론에서는 다루지 않은 양 도메인에서 분류기의 에러율을 조절할 수 있게하기 위해 새로운 이론을 제시한다. 이 이론적 배경을 바탕으로 소스 도메인과 타겟 도메인을 함께 학습하는 공동 대조적 방법을 소개한다. 본 공동 대조적 학습 방법에서는 각 라벨별로 구분되는 잠재 공간을 학습하기 위해 각 데이터의 특징과 라벨 사이의 상호 정보량을 최대화한다. 이 각 데이터의 특징과 라벨 사이의 상호 정보량은 각 라벨 분포 사이의 젠센-샤논 거리와 같으므로 이를 최대화하는 것은 곧 라벨들이 잘 구별되는 잠재 공간을 학습하는 것이다. 마지막으로 공동 대조적 학습 방법을 여러 데이터 셋에 적용하여 기존 방법론들과 비교하였다.

주요어: 적응 모델, 딥러닝, 도메인 적응, 이전 학습, 대조적 학습

학번: 2019-25825