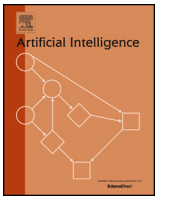
Artificial Intelligence 346 (2025) 104363

Contents lists available at ScienceDirect

Artificial Intelligence

journal homepage: www.elsevier.com/locate/artint



NT-FAN: A simple yet effective noise-tolerant few-shot adaptation network

Wenjing Yang [a,1], Haoang Chi [b,1], Yibing Zhan [c], Bowen Hu [a], Xiaoguang Ren [b], Dapeng Tao [d], Long Lan [a,*]

[a] College of Computer Science and Technology, National University of Defense Technology, Changsha, 410073, Hunan, PR China
[b] Academy of Military Sciences, Beijing, 100071, PR China
[c] JD Explore Academy, Beijing, 100176, PR China
[d] Yunnan University, Kunming, 650091, Yunnan, PR China

ARTICLE INFO

Keywords:
Representation learning
Weak-supervised learning
Few-shot learning
Transfer learning

ABSTRACT

Few-shot domain adaptation (FDA) aims to train a target model with *clean* labeled data from the source domain and *few* labeled data from the target domain. Given a limited annotation budget, source data may contain many noisy labels, which can detrimentally impact the performance of models in real-world applications. This problem setting is denoted as *wildly few-shot domain adaptation* (WFDA), simultaneously taking care of label noise and data shortage. While previous studies have achieved some success, they typically rely on multiple adaptation models to collaboratively filter noisy labels, resulting in substantial computational overhead. To address WFDA more simply and elegantly, we offer a theoretical analysis of this problem and propose a comprehensive upper bound for the excess risk on the target domain. Our theoretical result reveals that correct domain-invariant representations can be obtained even in the presence of source noise and limited target data without incurring additional costs. In response, we propose a simple yet effective WFDA method, referred to as *noise-tolerant few-shot adaptation network* (NT-FAN). Experiments demonstrate that our method significantly outperforms all the state-of-the-art competitors while maintaining a more *lightweight* architecture. Notably, NT-FAN consistently exhibits robust performance when dealing with more realistic and intractable source noise (e.g., instance-dependent label noise) and severe source noise (e.g., a 40% noise rate) in the source domain.

1. Introduction

Representation learning [45,46,62,74,81,90,92,110] aims to automatically acquire effective low-dimensional representations from high-dimensional raw data through machine learning algorithms. It has seen significant success in a variety of downstream tasks such as image classification [36,30,110], objective recognition [67], neural machine translation [4], etc. This success can be attributed to advancements in deep learning and the availability of an increasing amount of labeled data. These algorithms rely on abundant

* Corresponding author.
E-mail addresses: wenjing.yang@nudt.edu.cn (W. Yang), haoangchi618@gmail.com (H. Chi), zhanyibing@jd.com (Y. Zhan), hbw_14@nudt.edu.cn (B. Hu), rxg_nudt@126.com (X. Ren), dptao@ynu.edu.cn (D. Tao), long.lan@nudt.edu.cn (L. Lan).
[1] Equal contribution.

https://doi.org/10.1016/j.artint.2025.104363
Received 19 September 2023; Received in revised form 5 December 2024; Accepted 19 May 2025

Available online 22 May 2025

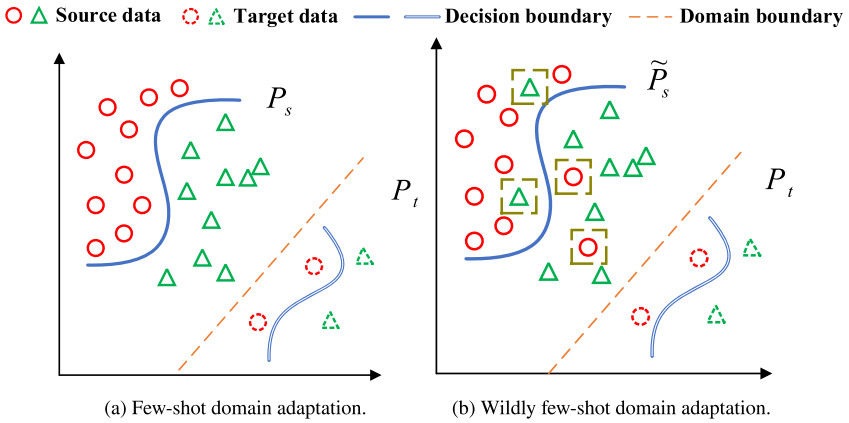


Fig. 1. Problem setting of wildly few-shot domain adaptation (WFDA). (a) As the basis of WFDA, to address data shortage, FDA is to learn a target model with few labeled target data and many labeled source data, where there exists covariate shift between them (i.e., domain boundary). (b) In the wild, perfectly annotated data is hard to acquire, and thus WFDA is to learn a target model with few labeled target data and noisy source data (i.e., data in a dotted box). Therefore, WFDA simultaneously takes care of label noise in the source domain and data shortage in the target domain.

data to estimate its probability distribution through maximum likelihood estimation while retaining a strong generalization ability. However, collecting ample labeled data for specific tasks can be challenging in real scenarios, especially in niche or sensitive fields like medicine [66]. This situation makes learning effective data representations with limited labeled data critically important. Such a problem is referred to as *few-shot learning* (FSL) [20,65,76,113]. Currently, FSL necessitates a large volume of labeled data from a source domain, closely resembling the target domain, to effectively pre-train an encoder. More generally, there may exist covariate shift in FSL [10,47,63], which is commonly known as *few-shot domain adaptation* (FDA) [15,19,56,83,111]. FDA and its associated methods have achieved great success in data shortage cases [56].

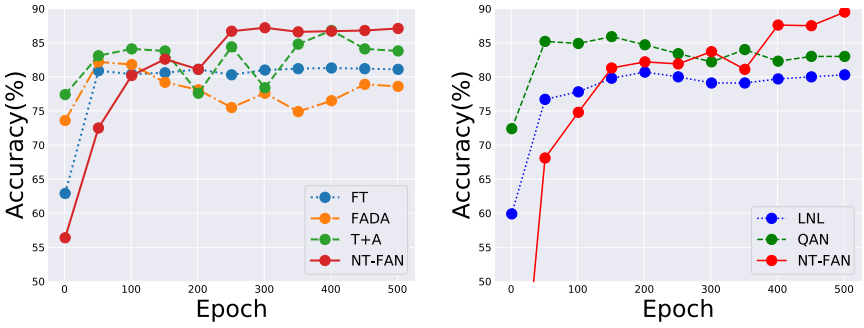
As mentioned above, data shortage is common in highly specialized or sensitive areas. Correctly annotating a large-scale dataset takes significant time and expert effort. Given limited budgets, it is usually unrealistic to acquire perfectly clean big data [38,73,86]. Even popular benchmark ImageNet [69] has at least 6% incorrect labels in the validation set [61]. This gives rise to a new setting, *wildly few-shot domain adaptation* (WFDA) (Fig. 1) [3,14], aiming to learn an accurate target model with a large number of noisy data on the source domain (i.e., $\tilde{P}_s$) and few labeled data on the target domain (i.e., P_t). However, existing FDA methods assume the source data (i.e., P_s) is correctly labeled, making them inapplicable for WFDA (Fig. 2). An alternative approach involves two-step methods [26,31,40]. In detail, they initially employ label-noise learning algorithms to correct the noisy labels on the source domain, followed by applying naive FDA algorithms. Unfortunately, recent label-noise learning algorithms cannot correct the noisy labels, even with a modest noise rate, like 10% [43,101]. As indicated by [46], the negative effect of residual noisy labels can accumulate during the adaptation procedure, causing sustained performance degradation.

Recent studies have increasingly turned their attention to domain adaptation in noisy environments, such as unsupervised domain adaptation in the wild [46] and few-shot domain adaptation in the wild [14]. A common strategy in these methods is to simultaneously maintain multiple models that, while possessing the same structure, have distinct parameters. These models collaboratively filter noisy data and facilitate domain knowledge transfer. However, this approach has two primary limitations: 1) it struggles to handle more realistic and intricate label noise, such as instance-dependent label noise [9]); 2) the training of multiple models significantly elevates computational costs, a concern especially pertinent for large-scale datasets. For irregular label noise, these methods may fail to select trustworthy data, thus resulting in poor adaptation performance and very limited information on the target domain (Fig. 2).

The above issues require an in-depth understanding of WFDA from another perspective. We theoretically analyze the excess risk of WFDA on the target domain. It uncovers how source noise, insufficient target data, and domain discrepancy fundamentally contribute to errors on the target domain (Theorem 2). From our theoretical results, we derive the following insights. 1) Learning the ground-truth conditional distribution of noisy source domain (i.e., $p_s(y|x)$) is possible through a corrected loss function, for instance, via the noise transition matrix [68]; 2) The primary contributor to error is the insufficient target data, which can be significantly mitigated through data augmentation techniques; 3) Domain discrepancy can be tackled using domain adversarial training.

The key to domain adaptation is to find useful *domain-invariant representations* (DIR) [17,21]. A good performance on the source domain implies a good performance on the target domain. However, the challenge is posed by the label noise on the source domain. Fortunately, a recent work [41] found that if an architecture "suits" one task, training with noisy labels can induce useful hidden representations. Inspired by this discovery, we can use one specific hidden layer of the deep network to yield nearly clean representations of source data. As these hidden representations are not significantly affected by the source noise, they can be directly used to perform domain adversary, mapping source and target data to the same distribution in embedding space robustly. Besides, we adopt a novel data augmentation strategy [56] that pairs data into 4 groups based on whether their categories and domains are the same, and therefore, classical domain discriminator is extended to group discriminator. The pipeline of our method is shown in Fig. 3.

To verify the effectiveness of our proposed method, we perform experiments on seven important domain adaptation benchmarks. Our approach not only significantly outperforms the strong baselines, but also is more lightweight, reducing nearly half the parameters and computational overhead compared with previous methods. In addition, our approach consistently performs well in the presence



(a) $M \rightarrow U$, 20%-pair (b) $M \rightarrow U$, 20%-instance

Fig. 2. Failures of existing FDA and WFDA methods in the presence of label noise in the source domain, taking MNIST→USPS and a noise rate of 20% as an example. The y-axis represents the testing accuracy on the target domain. (a) When **FDA methods** meet the class-dependent source noise, they are prone to overfit the noisy data, making it hard to converge on the target domain. (b) When **WFDA methods** (e.g., QAN [14] and LNL) meet more realistic and intractable instance-dependent label noise, they struggle to counteract the adverse effects of this irregular noise, leading to performance degradation.

of more severe source noise (e.g., noise rate $\geq 40\%$) and more realistic and intractable source noise (e.g., instance-dependent source noise).

Overall, our contributions can be summarized in four ways, shown in detail below.

- In this paper, we investigate an important problem, termed as *wildly few-shot domain adaptation* (WFDA). WFDA aims to train a target model with *noisy* labeled data from the source domain and *few* labeled data from the target domain. Insufficient data and noisy labels are common in real scenarios and bring non-trivial challenges for many learning tasks.
- We theoretically analyze the WFDA problem in principle, deriving 1) how the source label noise negatively affects the source model and how to mitigate it and 2) an upper bound of the excess risk on the target domain.
- Inspired by our theoretical results, we find that useful source data representations help address WFDA, and acquiring them is free. To this end, we propose a simple and lightweight method, NT-FAN. Our method can obtain clean representations from noisy source data and align them with the data-insufficient target domain in the representation space.
- Our method consistently performs well in the presence of 1) more realistic and intractable instance-dependent source noise and 2) more severe source noise (e.g., 40% noise rate) without requiring any modification.

2. Related work

This section reviews the recent development of few-shot learning, label-noise learning, few-shot domain adaptation, semi-supervised domain adaptation, and representation learning, which are related to WFDA.

2.1. Few-shot learning

Few-shot learning (FSL) is inspired by a human's learning and reasoning ability, aiming at completing a learning task with few experiences given some performance measure [48,84,91]. From the methodology perspective, FSL can be divided into three directions, i.e., data augmentation, transfer learning, and meta-learning. Data augmentation is the most direct way to address FSL, trying to estimate the target distribution and generating more samples within it [12,16,107]. However, given the limited data, estimating the entire target distribution is difficult. Thus, the generated samples have large biases and poor diversity. FSL algorithms generally require an auxiliary dataset to pretrain a model, and we can use few target data to finetune it like transfer learning [53,56,94]. For example, DeepEMD [106] adopted the Earth Mover's distance (EMD) to measure images' structural distances, and proposed a structured fully connected layer to classify EMD-induced representations accordingly. TEEN [89] proposed calibrating new-class prototypes by fusing weighted base prototypes to improve class-incremental learning. Meta-learning is the most common method address FSL, which tends to learn the metaknowledge (e.g., initial parameters, hyper-parameters, etc.) shared by auxiliary data and target data [20,60,76]. Thus, with the help of meta-knowledge, a pre-trained model can be quickly adapted to a FSL task even with limited data. For example, AVIATOR [104] improved MAML [20] by initializing models with task-level characteristics to facilitate the model optimization process. Relation Network [80] learned a deep distance metric to classify images of new classes based on distances between new-class images and query images during meta-learning. With the development of big models [11,18], FSL has gradually become the mainstream paradigm in artificial intelligence. However, compared with naive FSL, WFDA handles domain shift and label noise on the source domain.

2.2. Label-noise learning

Label-noise learning (LNL) is an important branch of weak-supervised learning, aiming at completing a learning task with samples containing wrong labels [1]. Given a limited budget, data annotation is often finished by non-expert [77] or crowdsourcing [108],

which makes it hard to avoid noisy labels. Existing works mainly involve the following techniques to address LNL. 1) Some works try to select clean data based on the small-loss trick and detach noisy data in training procedure [26,31,52,59,78,98]. 2) Some works estimate the transition matrix of noisy data and correct them through this matrix [13,57,101,103]. 3) A well-designed robust loss function can also help to alleviate the impact of noisy labels [22,51,93]. 4) LNL can be viewed as semi-supervised learning [40,44,100] through viewing noisy data as unlabeled data, and using related semi-supervised learning methods to address LNL. 5) Causal representation learning [72] provides a new perspective to understand LNL and induces corresponding solutions [96,102]. 6) Some works modify loss values to weaken the impact of noisy data before optimizing model [2,28,54,64,79]. 7) Useful regularization can help to overcome the strong memory of deep networks for noisy labels [5,6,42,82,97]. Compared with naive LNL, WFDA also takes care of domain shift and data scarcity.

2.3. Few-shot domain adaptation

Few-shot domain adaptation (FDA) is proposed to address data shortage beyond the restriction of sharing data distribution like FSL, aiming at completing a learning task with few target experiences and abundant source experiences given some performance measure [56]. FADA [56] proposes a novel data augmentation method of pairing data based on domain and category jointly to address data shortage, and then follows the adversarial domain adaptation framework [21]. Teshima et al. [83] formulate FDA with a structural causal model and propose to learn the invariant mechanism that decides data generation from auxiliary data. Due to the assumption of domain similarity, this mechanism can be adapted to target FSL tasks. Xu et al. [99] combine the mixup mechanism and optimal transport to learn the cross-domain alignment matrix and domain-invariant classifier. Tseng et al. [85] use affine transformation to augment data features for simulating various feature distributions, and use meta-learning to search the hyper-parameters of transformation. Thus, this meta-transformation can generate more features given the data of a new domain. Zhao et al. [111] propose to enhance class separation before domain-invariant feature learning to solve FSL and DA jointly. Compared with the FDA, the WFDA also considers label noise on the source domain.

2.4. Semi-supervised domain adaptation

Compared to the FDA, the *semi-supervised domain adaptation* (SSDA) is a similar but different setting. The main difference is that the target domain of SSDA contains both few labeled data and many unlabeled data, while the target domain of FDA only has few labeled data. There have been some deep learning-based methods of SSDA in recent years. Satio et al. [71] first formalize and study the SSDA and propose a mini-max optimization method towards the conditional entropy of unlabeled target data. [8] addresses SSDA by guessing low-entropy labels for data-augmented unlabeled examples and mixing the labeled and unlabeled data using mix-up. [75] uses contrastive learning to bridge the intra-domain discrepancy between the labeled and unlabeled target data and the inter-domain gap between source and unlabeled target distribution in SSDA.

2.5. Representation learning

Representation learning focuses on learning effective representations of data, which capture underlying patterns and structures to enhance performance on downstream tasks. The goal is to transform raw input data into a more informative and compact format, facilitating tasks such as classification, regression, clustering, and anomaly detection. Deep learning techniques, particularly autoencoders and neural networks, have been instrumental in advancing this field, as they can learn complex, hierarchical representations directly from data, often outperforming traditional feature-based approaches. Khemakhem et al. [32] explored the potential for identifying the true joint distribution over observed and latent variables in deep latent-variable models, specifically within the context of variational autoencoders (VAEs). The proposed identifiable VAE (iVAE) framework provides a moral form of disentanglement. It bridges the gap between VAEs and nonlinear ICA, offering a unified view of these two methods in unsupervised representation learning. Mita et al. [55] argued that achieving disentanglement in a fully unsupervised setting was impossible without inductive biases, and proposed the *Identifiable Double VAE* (IDVAE) that incorporated auxiliary variables to impose structure on the latent space, leading to a model with theoretical guarantees on identifiability and disentanglement. Kivva et al. [35] proved the identifiability of a broad class of deep generative models that possessed universal approximation capabilities and were commonly used as decoders in VAEs. They introduced an identifiability hierarchy that generalized previous results and demonstrated how different assumptions lead to varying strengths of identifiability. Our algorithm is also established on representation learning to obtain clean domain-invariant representation.

3. Preliminary

In this section, we first introduce the notations in this paper. Then, given a clean dataset, we explain the generation process of class-dependent label [68] noise and instance-dependent label [9] noise.

3.1. Notations

We list the notations and their meanings as follows.

- a data space $\mathcal{X} \subset \mathbb{R}^d$ and a label set $\mathcal{Y} = \{1, \dots, K\}$;
- noisy source data $D_s = \{(x_s, \tilde{y}_s)\} \sim \tilde{p}_s(x_s, \tilde{y}_s)$ and few target data $D_t = \{(x_t, y_t)\} \sim p_t(x_t, y_t)$ with $|D_s| = m_1$ and $|D_t| = m_2$, where $|\cdot|$ denotes the cardinal number of a set;
- density function of source domain $\tilde{p}_s(x_s, \tilde{y}_s)$ *w.r.t.* multiple random variable $(X_s, \tilde{Y}_s)$ and density function of target domain $p_t(x_t, y_t)$ *w.r.t.* multiple random variable (X_t, Y_t) defined on $\mathcal{X} \times \mathcal{Y}$;
- $\tilde{f}_s$, f_s, $f_t : \mathcal{X} \to \mathcal{Y}$ denote the real source labeling function, ground-truth source labeling function, and real target labeling function, respectively, where $\tilde{f}_s(x_s) = \tilde{y}_s$, $f_s(x_s) = y_s$, and $f_t(x_t) = y_t$;
- δ^s, $\tilde{\delta}^s : \mathcal{X} \to \Delta_K$ denote the class probability functions of data in the clean source domain and the noisy source domain, respectively, where Δ_K is a K-simplex and K is the dimension. Specifically, $\delta^s_i(x) = \mathbb{P}(Y_s = i | X_s = x)$ and $\tilde{\delta}^s_i(x) = \mathbb{P}(\tilde{Y}_s = i | X_s = x)$;
- noise rate ρ, indicating the ratio of data with noisy labels in a dataset;
- hypothesis space $\mathcal{F} = \{f : \mathcal{X} \to \mathcal{Y}\}$;
- loss function $\ell : \mathcal{F} \times (\mathcal{X} \times \mathcal{Y}) \to \mathbb{R}$, where $\ell(f(x), y)$ represents the loss value of data point x with label y;
- empirical risk $\mathcal{R}(f) = \mathbb{E}_{(x,y)\sim p}[\ell(f(x), y)]$, $\forall h \in \mathcal{H}$ given a distribution p and a loss function ℓ.

3.2. Generation of two types of label noise

We manually generate specific label noise to align with the WFDA setup. In this paper, we consider two types of frequent label noise: 1) class-dependent label noise [68]; 2) instance-dependent label noise [9].

3.2.1. Class-dependent label noise

For class-dependent label noise, we employ transition matrix Q to model the label corruption process, where $Q_{ij} = \mathbb{P}(\tilde{y} = j | y = i)$ indicates the probability that ground-truth y is flipped to noisy label $\tilde{y}$. One type of class-dependent corruption method is symmetric flipping, where the ground truth is flipped to all other classes equally. Symmetric flipping can be formulated through the transition matrix given noise rate ρ,

$$Q = \begin{bmatrix} 1-\rho & \frac{\rho}{K-1} & \cdots & \frac{\rho}{K-1} & \frac{\rho}{K-1} \\ \frac{\rho}{K-1} & 1-\rho & \frac{\rho}{K-1} & \cdots & \frac{\rho}{K-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{\rho}{K-1} & \cdots & \frac{\rho}{K-1} & 1-\rho & \frac{\rho}{K-1} \\ \frac{\rho}{K-1} & \frac{\rho}{K-1} & \cdots & \frac{\rho}{K-1} & 1-\rho \end{bmatrix}.$$

Another type of class-dependent corruption method is pair flipping, where the ground truth is only flipped to its next category. Pair flipping can also be formulated through a transition matrix given noise rate ρ,

$$Q = \begin{bmatrix} 1-\rho & \rho & 0 & \cdots & 0 \\ 0 & 1-\rho & \rho & & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1-\rho & \rho \\ \rho & 0 & \cdots & 0 & 1-\rho \end{bmatrix}.$$

3.2.2. Instance-dependent label noise

In real-world scenarios, an instance is more likely to be mislabeled by both its class and features, denoted as instance-dependent label noise. Thus, the transition matrix turns into $Q_{ij}(x) = \mathbb{P}(\tilde{y} = j | y = i, x)$, $\forall x \in \mathcal{X}$. Specially, $\tilde{y} = \bar{h}(x)$, where $\bar{h} \in \mathcal{H}$ is a noisy labeling function that depends on feature x.

4. Wildly few-shot domain adaptation

In this section, we first give *wildly few-shot domain adaptation* (WFDA) [14] a precise definition. Then, we empirically explore why previous FDA methods fail to address WFDA in a more realistic and challenging setting.

Problem 1 *(Wildly few-shot domain adaptation).* Let $(X_s, \tilde{Y}_s)$ be a multiple random variable defined on $\mathcal{X} \times \mathcal{Y}$ with respective to density function $\tilde{p}_s(x_s, \tilde{y}_s)$, and (X_t, Y_t) be a multiple random variable defined on $\mathcal{X} \times \mathcal{Y}$ with respective to density function $p_t(x_t, y_t)$, where $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y} = \{1, \dots, K\}$ represent feature space and label space, respectively. Given *i.i.d.* noisy source data $D_s = \{(x_s, \tilde{y}_s)\} \sim \tilde{p}_s(x_s, \tilde{y}_s)$ and few target data $D_t = \{(x_t, y_t)\} \sim p_t(x_t, y_t)$, in wildly few-shot domain adaptation, we aim to learn a model to accurately classify data drawn from p_t with D_s and D_t, where $m_1 = |D_s| \gg |D_t| = m_2$.

Remark 1. In Problem 1, $\tilde{p}_s(x_s, \tilde{y}_s)$ describes the real source domain with noisy labels. Moreover, we denote $p_s(x_s, y_s)$ as the source domain whose labels are all ground-truths.

After formally defining WFDA, we show that previous FDA methods will *fail* to address WFDA. As shown in Fig. 2a, previous related methods are negatively affected by source noise to severely fluctuate and cannot converge to an optimal solution in the

training procedure. Moreover, when encountering the more complex source noise (e.g., instance-dependent label noise), the previous WFDA method (i.e., QAN [14]) also performs poorly, as shown in Fig. 2b. The previous WFDA method fluctuates severely in the training process because it cannot handle complex source noise and tends to overfit them.

5. Theoretical analysis of WFDA

WFDA aims to learn a target classifier f with D_s and D_t to minimize the following risk:

$$\mathcal{R}_t(f) := \mathbb{E}_{(x_t,y_t)\sim p_t}[\ell(f(x_t), y_t)],$$

where ℓ is a proper loss function with ℓ and $\nabla\ell$ bounded. General adversarial domain adaptation [21,95] is to learn domain-invariant representations (DIR) and achieve good performance on the source domain. With DIR, a classification model that performs well on the source domain also can perform well on the target domain. Thus, the learning objective is

$$\min_{f\in F} \underbrace{\mathbb{E}_{(x_s,y_s)\sim p_s}[\ell(f(x_s), y_s)]}_{\text{source domain risk } \mathcal{R}_s(f)} - \lambda \underbrace{\mathbb{E}_{x_s\sim p_s^X, x_t\sim p_t^X}[\ell_d(x_s, x_t, y_d)]}_{\text{domain discrimination risk}},$$

where ℓ_d denotes the loss function of the domain discriminator (i.e., a binary classifier), y_d denotes the domain label, and λ is a trade-off hyper-parameter. p_s^X and p_t^X are marginal densities of source domain and target domain on $\mathcal{X}$, respectively. However, in WFDA, the ground-truth source domain p_s is corrupted to $\tilde{p}_s$ with label noise, thus the actual learning objective becomes

$$\min_{f\in F} \mathbb{E}_{(x_s,\tilde{y}_s)\sim \tilde{p}_s}[\ell(f(x_s), \tilde{y}_s)] - \gamma \mathbb{E}_{x_s\sim p_s^X, x_t\sim p_t^X}[\ell_d(x_s, x_t, y_d)].$$

As noisy labels in the first term of the above equation corrupt the source domain to $\tilde{p}_s$, thus the distribution shift between the source domain and target domain sharply increases. In detail, for general FDA, there only exists a small covariate shift (i.e., $p_s^{Y|X}(y|x) = p_t^{Y|X}(y|x)$ and $p_s^X(x) \neq p_t^X(x)$) between two domains. While in WFDA, they have a large joint distribution shift (i.e., $p_s^{Y|X}(\tilde{y}|x) \neq p_t^{Y|X}(y|x)$ and $p_s^X(x) \neq p_t^X(x)$) after corruption, i.e.,

$$p_s^{Y|X}(\tilde{y}|x)p_s^X(x) \neq p_t^{Y|X}(y|x)p_t^X(x),$$

and this indicates

$$p_s(x, \tilde{y}) \neq p_t(x, y),$$

where both $p_s^X(x)$ (*resp.* $p_t^X(x)$) and $p_s^{Y|X}(\tilde{y}|x)$ (*resp.* $p_t^{Y|X}(y|x)$) simultaneously lead to the domain discrepancy. Thus, such a large domain discrepancy makes naive adversarial domain adaptation disabled (Fig. 2a).

5.1. Correct source noise through transition matrix

Next, we formally use the transition matrix to model label noise. Let $\delta^s, \tilde{\delta}_s : \mathcal{X} \to \Delta_K$ denote the class probability functions of the clean source domain and noisy source domain, respectively, where Δ_k denotes the K-simplex. We can derive $f_s(x) = \arg\max_{1\le i\le K} \delta_i^s$ and $\tilde{f}_s(x) = \arg\max_{1\le i\le K} \tilde{\delta}_i^s$. Given a transition matrix Q defined in 3.2, we have

$$\begin{aligned}
\tilde{\delta}_i^s(x) &= \mathbb{P}(\tilde{Y}_s = i | X_s = x) \\
&= \sum_j \mathbb{P}(\tilde{Y}_s = i | Y_s = j, X_s = x)\mathbb{P}(Y_s = j | X_s = x) \\
&= \sum_j Q_{ji}(x)\delta_j^s(x) \\
&= Q_i^\top(x) \cdot \delta^s(x), \quad \forall x \sim X_s.
\end{aligned}$$

Furthermore, we have $\tilde{\delta}^s(x) = Q^\top(x)\delta^s(x)$. Thus, we can directly derive

$$\delta^s(x) = (Q^\top(x))^{-1}\tilde{\delta}^s(x), \tag{1}$$

if $Q^\top(x)$ is invertible. Otherwise, the Moore-Penrose pseudo left inverse $(Q(x)Q^\top(x))^{-1}Q(x)$ is alternative. For simplicity, we assume that $Q(\cdot)$ is invertible. Eq. (1) provides an estimated corrected source class probability function $\hat{\delta}^s(x)$ given a transition matrix $Q(\cdot)$ and an estimated noisy source class probability function $\hat{\tilde{\delta}}^s(x)$. Based on [109], we will provide an excess risk bound on the clean source domain for corrected model f_s learned on noisy source domain, i.e., $f_s(x) = \arg\max_{1\le i\le K}((Q^\top(x))^{-1}\tilde{\delta}^s(x))_i$. We equivalently define the loss function $\mathcal{I}(\delta(x), y) := \ell(f(x), y) = \ell\left(\arg\max_{1\le i\le K} \delta_i(x), y\right)$.

Theorem 1. *Let* $f_s^* = \arg\min_{f\in\mathcal{F}} \mathbb{E}_{(x_s,y_s)\sim p_s}[\ell(f(x_s),y_s)]$ *with* $f_s^* = \arg\max_{1\le i\le K} \delta_i^{s*}$ *and* $\mathcal{I}(x,y)$ *be* L*-Lipschitz continuous for* x*, where* $L \ge 0$*. For any* $f_s = \arg\max_{1\le i\le K} \delta_i^s \in \mathcal{F}$ *trained on the noisy source domain, we have*

$$\mathcal{R}_s(f_s) \le \mathcal{R}_s(f_s^*) + L \sup_{x\in\mathcal{X}} \|Q^\top(x)^{-1}\|_2 \mathbb{E}_{p_s^X} \|\tilde{\delta}^s(x) - \tilde{\delta}^{s*}(x)\|_2,$$

where $\tilde{\delta}^s(x) = Q^\top(x)\delta^s(x)$ *and* $\tilde{\delta}^{s*}(x) = Q^\top(x)\delta^{s*}(x)$.

Proof.

$$\begin{aligned}
&\mathcal{R}_s(f_s) - \mathcal{R}_s(f_s^*)\\
&= \mathbb{E}_{p_s}[\ell(f_s(x),y) - \ell(f_s^*(x),y)]\\
&= \mathbb{E}_{p_s}[\mathcal{I}(\delta^s(x),y) - \mathcal{I}(\delta^{s*}(x),y)]\\
&= \mathbb{E}_{p_s}[\mathcal{I}((Q^\top(x))^{-1}\tilde{\delta}^s(x),y) - \mathcal{I}((Q^\top(x))^{-1}\tilde{\delta}^{s*}(x),y)].
\end{aligned}$$

Thus, we have

$$\begin{aligned}
&\mathcal{R}_s(f_s) - \mathcal{R}_s(f_s^*)\\
&= |\mathcal{R}_s(f_s) - \mathcal{R}_s(f_s^*)|\\
&= |\mathbb{E}_{p_s}[\mathcal{I}((Q^\top(x))^{-1}\tilde{\delta}^s(x),y) - \mathcal{I}((Q^\top(x))^{-1}\tilde{\delta}^{s*}(x),y)]|\\
&\le \mathbb{E}_{p_s}|\mathcal{I}((Q^\top(x))^{-1}\tilde{\delta}^s(x),y) - \mathcal{I}((Q^\top(x))^{-1}\tilde{\delta}^{s*}(x),y)|\\
&\le \mathbb{E}_{p_s}[L\cdot|(Q^\top(x))^{-1}\cdot(\tilde{\delta}^s(x) - \tilde{\delta}^{s*}(x))|]\\
&= \mathbb{E}_{p_s^X}[L\cdot|(Q^\top(x))^{-1}\cdot(\tilde{\delta}^s(x) - \tilde{\delta}^{s*}(x))|]\\
&\le \mathbb{E}_{p_s}[L\cdot\|Q^\top(x)^{-1}\|_2\cdot\|\tilde{\delta}^s(x) - \tilde{\delta}^{s*}(x)\|_2]\\
&\le L\cdot\sup_{x\in\mathcal{X}}\|Q^\top(x)^{-1}\|_2\cdot\mathbb{E}_{p_s^X}\|\tilde{\delta}^s(x) - \tilde{\delta}^{s*}(x)\|_2.
\end{aligned}$$

Hence, we prove this theorem. □

Remark 2. Theorem 1 shows that the excess risk of a source model trained on the noisy source domain is upper bounded by $L\cdot\sup_{x\in\mathcal{X}}\|Q^\top(x)^{-1}\|_2\cdot\mathbb{E}_{p_s^X}\|\tilde{\delta}^s(x) - \tilde{\delta}^{s*}(x)\|_2$. If the model fits noisy source data very well, $\mathbb{E}_{p_s^X}\|\tilde{\delta}^s(x) - \tilde{\delta}^{s*}(x)\|_2$ will be very small, thereby the excess risk $\mathcal{R}_s(f_s) - \mathcal{R}_s(f_s^*)$ will tend to 0. This result shows that directly pre-training a source model with noisy source data also makes sense if the learned class probability distribution $\tilde{\delta}^s$ is (nearly) equal to the ground-truth $\tilde{\delta}^{s*}$ on the noisy source domain. Based on Eq. (1), we can acquire a clean class probability distribution with the help of Q.

5.2. An excess risk bound for WFDA

After analyzing how the source noise negatively affects the source model and how to mitigate this, we turn to the few-shot domain adaptation process. Following [7], we employ the $\mathcal{F}\Delta\mathcal{F}$-divergence to measure domain discrepancy that is dependent on hypothesis space $\mathcal{F}$. The symmetric difference hypothesis space $\mathcal{F}\Delta\mathcal{F} = \{g : \mathcal{X}\to\{0,1\} : g(x) = f(x)\oplus f'(x), \exists f, f'\in\mathcal{F}\}$ and $\oplus$ represents the XOR operation.

Definition 1 *($\mathcal{F}\Delta\mathcal{F}$-divergence [7]).* Given two probability densities p, p' defined on $\mathcal{X}$ and a hypothesis space $\mathcal{F}$, the $\mathcal{F}\Delta\mathcal{F}$-divergence is defined as

$$d_{\mathcal{F}\Delta\mathcal{F}}(p,p') = 2\sup_{g\in\mathcal{F}\Delta\mathcal{F}} |\mathbb{P}_{x\sim p}(\{x : g(x) = 1\}) - \mathbb{P}_{x'\sim p'}(\{x' : g(x') = 1\})|,$$

where $\mathbb{P}$ denotes the probability measure.

Remark 3. For any $g\in\mathcal{F}\Delta\mathcal{F}$, it indicates the set of data points with different predicting results by two hypotheses in $\mathcal{F}$, i.e., the symmetric difference set derived by two hypotheses. Furthermore, $\mathcal{F}\Delta\mathcal{F}$-divergence represents the largest possible difference of measure value between two symmetric difference sets on two domains, serving as a divergence measure of such two domains.

Now, we return to the problem of WFDA itself. With the help of $\mathcal{F}\Delta\mathcal{F}$-divergence, we try to derive a PAC-learning bound of the excess risk on the target domain.

In realistic WFDA problem, we can only calculate the empirical value of $\mathcal{F}\Delta\mathcal{F}$-divergence with source data $D_s \overset{i.i.d.}{\sim} \tilde{p}_s$ and target data $D_t \overset{i.i.d.}{\sim} p_t$. Thus, we will give an error bound of true value and empirical value of $\mathcal{F}\Delta\mathcal{F}$-divergence. This error bound is derived through the following two lemmas.

Lemma 1 *(Theorem 3.4 in [33]). Assume the VC dimension of hypothesis space $\mathcal{F}$ is d. Let $\tilde{p}_s$ be the probability density of the noisy source and p_t be the probability density of the target domain. D_s and D_t are samples drawn from $\tilde{p}_s$ and p_t, respectively, with $|D_s| = m_1$ and $|D_t| = m_2$. Then*

$$\mathbb{P}^{m_1+m_2}(|d_{\mathcal{F}\Delta\mathcal{F}}(\tilde{p}_s, p_t) - \hat{d}_{\mathcal{F}\Delta\mathcal{F}}(D_s, D_t)| > \epsilon) \leq (2m_1)^d \exp(-m_1\epsilon^2/16) + (2m_2)^d \exp(-m_2\epsilon^2/16),$$

where $\mathbb{P}^{m_1+m_2}$ means the (m_1+m_2)th power of $\mathbb{P}$.

Lemma 2. *Assume the VC dimension of hypothesis space $\mathcal{F}$ is d. Let $\hat{d}_{\mathcal{F}\Delta\mathcal{F}}(D_s, D_t)$ be the empirical $\mathcal{F}\Delta\mathcal{F}$-divergence with $|D_s| = m_1$ and $|D_t| = m_2$ $(m_1 \gg m_2)$. Then for any $\eta > 0$, with probability at least $1-\eta$,*

$$d_{\mathcal{F}\Delta\mathcal{F}}(\tilde{p}_s, p_t) \leq \hat{d}_{\mathcal{F}\Delta\mathcal{F}}(D_s, D_t) + 4\sqrt{\frac{(m_1+m_2)\log\frac{1}{\eta} + d\log(2m_1)}{m_1}}. \tag{2}$$

Proof. Based on Lemma 1, let

$$(2m_1)^d \exp(-m_1\epsilon^2/16) + (2m_2)^d \exp(-m_2\epsilon^2/16) := \eta^{m_1+m_2},$$

then we equivalently have

$$\eta^{m_1+m_2} = (2m_1)^d \left(\exp\left(\frac{-m_1\epsilon^2}{16}\right) + \exp\left(\frac{-m_2\epsilon^2}{16}\right)\left(\frac{m_2}{m_1}\right)^d\right).$$

Generally, the VC dimension of $\mathcal{F}$ is very large, especially for deep neural networks used in our method. For an arbitrary ReLU neural network, let W be the number of weights and L be the number of layers; its VC dimension is $O(WL\log(W))$. For instance, ResNet-18 [27], its VC dimension is nearly 10^9. As m_2/m_1 is very small, we have $(m_2/m_1)^d \to 0$ and

$$\eta^{m_1+m_2} = (2m_1)^d \exp(-m_1\epsilon^2/16).$$

By solving the above equation, we have

$$\epsilon = 4\sqrt{\frac{(m_1+m_2)\log\frac{1}{\eta} + d\log(2m_1)}{m_1}}.$$

According to the results of Lemma 1, with probability at least $1-\eta$, we can obtain Eq. (2). Hence, we prove this lemma. □

Remark 4. Limited by the number of target data, there always exists a gap between the true and empirical values of $\mathcal{F}\Delta\mathcal{F}$-divergence, because just a few arbitrary target data are hard to represent the whole target domain. As $m_1 \to \infty$, $d_{\mathcal{F}\Delta\mathcal{F}}(\tilde{p}_s, p_t) - \hat{d}_{\mathcal{F}\Delta\mathcal{F}}(D_s, D_t) \to 4\sqrt{\log\frac{1}{\eta}}$, which is reasonable in FSL. Through a meta-learning strategy with auxiliary target data, this upper bound can be further reduced [24].

Based on the above conclusions, we can propose an excess risk bound on the target domain for WFDA in the following theorem.

Theorem 2. *Let f_s^* defined as Theorem 1 and $\|I\|_2 \leq L$. Let $f^* = \arg\min_{f\in\mathcal{F}} \mathcal{R}_{\tilde{s}}(f) + \mathcal{R}_t(f)$ and $\lambda = \mathcal{R}_{\tilde{s}}(f^*) + \mathcal{R}_t(f^*)$. Assume the VC dimension of hypothesis space $\mathcal{F}$ is d. Let $\hat{d}_{\mathcal{F}\Delta\mathcal{F}}(D_s, D_t)$ be the empirical $\mathcal{F}\Delta\mathcal{F}$-divergence with $|D_s| = m_1$ and $|D_t| = m_2$ $(m_1 \gg m_2)$. Then for any $\eta \in (0,1)$, with probability at least $1-\eta$, for every $f \in \mathcal{F}$,*

$$\mathcal{R}_t(f) \leq \underbrace{|\mathcal{R}_{\tilde{s}}(f) - \mathcal{R}_s(f)|}_{\text{(i) error from source noise}} + \underbrace{L\sup_{x\in\mathcal{X}} \|Q^\top(x)^{-1}\|_2 \mathbb{E}_{p_s^X} \|\tilde{\delta}(x) - \tilde{\delta}^{s*}(x)\|_2}_{\text{(ii) error from correct source hypothesis}} + \underbrace{2\sqrt{\frac{(m_1+m_2)\log\frac{1}{\eta} + d\log(2m_1)}{m_1}}}_{\text{(iii) error from data shortage}}$$
$$+ \underbrace{\mathcal{R}_s(f_s^*) + \lambda}_{\text{(iv) error from } \mathcal{F}} + \underbrace{\frac{1}{2}\hat{d}_{\mathcal{F}\Delta\mathcal{F}}(D_s, D_t)}_{\text{(v) error from empirical domain discrepancy}}. \tag{3}$$

Proof. We prove this theorem in the following.

$$
\begin{aligned}
&\mathcal{R}_t(f)\\
&= \mathcal{R}_t(f) - \mathcal{R}_t(f^*) + \mathcal{R}_t(f^*) - \mathcal{R}_{\tilde{s}}(|f - f^*|) + \mathcal{R}_{\tilde{s}}(|f - f^*|)\\
&\le \mathcal{R}_t(f^*) + \mathcal{R}_t(|f - f^*|) - \mathcal{R}_{\tilde{s}}(|f - f^*|) + \mathcal{R}_{\tilde{s}}(|f - f^*|)\\
&\le \mathcal{R}_t(f^*) + \mathcal{R}_{\tilde{s}}(|f - f^*|) + \frac{1}{2} d_{\mathcal{F}\Delta\mathcal{F}}(p_{\tilde{s}}, p_t) \quad \% \text{ by Lemma 3 of [7]}\\
&\le \mathcal{R}_t(f^*) + \mathcal{R}_{\tilde{s}}(f) + \mathcal{R}_{\tilde{s}}(f^*) + \frac{1}{2} d_{\mathcal{F}\Delta\mathcal{F}}(\tilde{p}_s, p_t)\\
&= \mathcal{R}_{\tilde{s}}(f) + \frac{1}{2} d_{\mathcal{F}\Delta\mathcal{F}}(\tilde{p}_s, p_t) + \lambda\\
&\le |\mathcal{R}_{\tilde{s}}(f) - \mathcal{R}_s(f)| + |\mathcal{R}_s(f) - \mathcal{R}_s(f_s^*)| + \mathcal{R}_s(f_s^*) + \frac{1}{2} d_{\mathcal{F}\Delta\mathcal{F}}(\tilde{p}_s, p_t) + \lambda\\
&\le |\mathcal{R}_{\tilde{s}}(f) - \mathcal{R}_s(f)| + L \sup_{x \in \mathcal{X}} \|Q^{\top}(x)^{-1}\|_2 \mathbb{E}_{p_s^X} \|\tilde{\delta}(x) - \tilde{\delta}^{s*}(x)\|_2 + \frac{1}{2} \hat{d}_{\mathcal{F}\Delta\mathcal{F}}(D_s, D_t) + 2\sqrt{\frac{(m_1 + m_2)\log\frac{1}{\eta} + d\log(2m_1)}{m_1}}\\
&+ \mathcal{R}_s(f_s^*) + \lambda \quad \% \text{ by Theorem 1 and Lemma 2}
\end{aligned}
$$

Hence, we prove this theorem. □

Remark 5. In Theorem 2, we prove that target risk mainly comes from five parts. Error (i) results from the noises in the source domain. Error (ii) results from the corrected source hypothesis, and $\mathbb{E}_{p_s^X} \|\tilde{\delta}(x) - \tilde{\delta}^{s*}(x)\|_2 \to 0$ if f fully fits noisy source data. Error (iii) results from the shortage of target data and can be alleviated through data augmentation [16], meta-learning [20], etc. Error (iv) results from the complexity of pre-defined hypothesis space $\mathcal{F}$, which can be reduced to 0 if $\mathcal{F}$ has strong learning ability. Error (v) results from the empirical discrepancy between the source domain and the target domain, depending on the difficulty of the WFDA task itself.

5.3. The inspiration of theory to method

If the source domain is contaminated by label noise, FDA methods will be degraded by the learned distorted source data representations. The distorted source data representations further lead to distorted target data representations and bad task performance.

According to Theorem 2, a clean source domain is *not* always a necessary condition for few-shot domain adaptation. The item (*i*) and item (*ii*) in Eq. (3) represent the target risks brought by source label noise. The item (*i*) represents the difference of risks on clean and noisy source domains given a hypothesis f and can be reduced by domain alignment. The item (*ii*) implies the difference of class probability distributions on the noisy source domain, induced by the optimal source hypothesis $\tilde{\delta}^{s*}$ and any a hypothesis $\tilde{\delta}$. Thus, reducing the item (*ii*) will be the key for addressing WFDA. Specifically, a hypothesis that fully fits the noisy source domain (i.e., $\tilde{\delta}^{s*}$ is very close to $\tilde{\delta}$) can directly reduce the target risk. Inspired by our theories and the above analysis, we propose our method in the next section.

6. Method: noise-tolerant few-shot adaptation network

Based on the above inspirations, this section will present our method *noise-tolerant few-shot adaptation network* (NT-FAN). NT-FAN is divided into three steps (Fig. 3), and we will introduce them in detail.

6.1. Pre-train source model

Inspired by a general pipeline of domain adaptation [21,50,105], we should pre-train a useful source model with abundant source data for the next adaptation task. However, in WFDA, source data has label noise, destroying the original ground-truth source domain. If we directly follow the standard training strategy [21] without modification, the sick pre-trained source model may result in severe negative transfer for the target domain. Theorem 1 indicates that training with noisy data does not always hurt model performance if given proper correction, e.g., correcting class probability distribution with transition matrix [109] as we have proven, noise-corrected model estimator [88], and so on. All these methods seek useful knowledge from noisy data that is not badly affected.

From the perspective of adversarial domain adaptation [14,21,50], the key factor is to acquire *domain-invariant representations* (DIR) of source and target data. Thus, our attention lies in learning useful data representations under source noise through correction techniques. Fortunately, Li et al. [41] pointed out that if an architecture "suits" one task, training with noisy labels can induce useful hidden representations. Through this conclusion, we can obtain nearly clean representations of source data without any more budget.

Given source model (i.e., a deep network) $\tilde{f}_s = \arg\max \tilde{\delta}^s$ with $\tilde{\delta}^s = h_s \circ g_s$, we can decompose $\tilde{\delta}^s$ as $\tilde{\delta}^s = \tilde{\delta}_n^s \circ \tilde{\delta}_{n-1}^s \circ \cdots \circ \tilde{\delta}_1^s$, where $\tilde{\delta}_z^s$ denotes the z-th layer of deep network $\tilde{\delta}^s$. Then, we will train $\tilde{f}_s$ to fully fit noisy source data D_s with the following loss function,

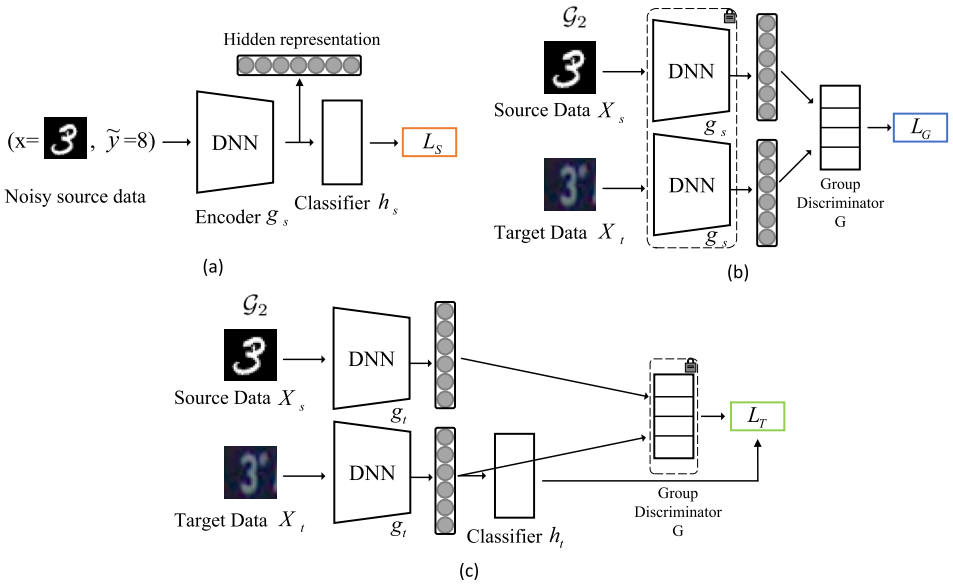


Fig. 3. Overview of the *noise-tolerant few-shot adaptation network* (NT-FAN). (a) Pre-train source model. We train a deep network $f_s = h_s \circ g_s$ to fully fit noisy source data, where g_s and h_s denote source encoder and source classifier, respectively. The next steps use the representations induced by one proper hidden layer of g_s. (b) Pre-train group discriminator. We use source encoder g_s to induce hidden representations for data pairs and train the group discriminator to identify them. (c) Update the target model and confuse the group discriminator simultaneously. We initialize target encoder $g_t = g_s$ and randomly initialize target classifier h_t. $h_t \circ g_t$ is updated to fit target data and confuse group discriminator simultaneously. Note that in steps (b) and (c), training inputs are both data pairs (i.e., $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{G}_3$, and $\mathcal{G}_4$) defined in Section 6 and we take $\mathcal{G}_2$ as an example in this figure.

$$\mathcal{L}_S = \frac{1}{|D_s|} \sum_{(x_s,\tilde{y}_s)\in D_s} \ell_{ce}(\tilde{\delta}^s(x_s), \tilde{y}_s) = -\frac{1}{|D_s|} \sum_{(x_s,\tilde{y}_s)\in D_s} \tilde{y}_s \cdot \log(\tilde{\delta}^s(x_s)),$$

where ℓ_{ce} denotes the standard cross entropy loss and $\tilde{y}_s$ is the noisy label with one-hot version. Without loss of generality, we assume to use the z-th layer of $\tilde{\delta}^s$ to induce hidden data representations. Specifically, given a data point x, the hidden representation induced by the z-th layer of $\tilde{\delta}^s$ is denoted as $\psi^s_z(x) := \tilde{\delta}^s_z(\tilde{\delta}^s_{z-1}(\cdots \tilde{\delta}^s_1(x)))$.

We can link hidden representation and noise transition matrix naturally. In this work, we choose the last layer of the source encoder with U units as the hidden layer to induce hidden representations. Assume source encoder g_s has n_1 layers, the final class probability distribution of data point x can be written as $\tilde{\delta}^s(x) = h_s(\psi^s_{n_1}(x))$. Regardless of bias, linear transformation $h_s \in \mathbb{R}^{K\times U}$ is a real matrix with a shape of $K \times U$. Previous works [26,93] pointed out that neural networks tend to learn easy knowledge first, and label noise is mainly fitted by very latter layers of deep neural networks. Based on these empirical studies, noise information in D_s is mainly fitted by the last layer of $\tilde{\delta}^s$, i.e., h_s. As mentioned in Section 5, we can use the Moore-Penrose pseudo left inverse of h_s to obtain nearly clean data representations. If the weight matrix of h_s is full rank, then

$$\psi^s_{n_1}(x) = (h_s^\top h_s)^{-1} h_s^\top \cdot \tilde{\delta}^s(x).$$

If h_s is not full rank, we have to leverage the singular value decomposition for h_s, i.e., $h_s = Z\Sigma V^\top$, for orthogonal matrices $Z \in \mathbb{R}^{K\times K}$, $V \in \mathbb{R}^{U\times U}$ and diagonal non-negative real matrix $\Sigma \in \mathbb{R}^{K\times U}$. Then,

$$\psi^s_{n_1}(x) = V\Sigma^+ Z^\top \cdot \tilde{\delta}^s(x),$$

where Σ^+ is the pseudo inverse of Σ, which is formed by replacing every non-zero diagonal entry with its reciprocal and transposing the resulting matrix. Note that if the last layer of the encoder has the same number of neurons as the classifier, the Moore-Penrose pseudo left inverse will degrade into the general matrix inverse. Based on this motivation, employing hidden data representations implies correcting class probability distribution with transition noise matrix, which naturally links our theory (i.e., the term (*ii*) in Eq. (3) of Theorem 2) and our method. Therefore, the inverse of the last layer's parameter matrix can be viewed as an approximated noise transition matrix. Although strictly proving this claim using deep learning theory is hard, we alternatively demonstrate it from an empirical perspective. The detailed result is presented in Section 7.10.

6.2. Pre-train group discriminator

As mentioned above, adversarial domain adaptation aims to map source data and target data to the same distribution in embedding space, making the domain discriminator unable to identify their domains. Thereby, the source classifier is also suited for the target domain.

In WFDA, however, target data is very scarce and insufficient to represent the entire target domain. Thus, directly performing domain adversary will cause serious overfitting. We augment data by pairing it to 4 groups to alleviate data shortage based on whether their categories and domains are the same simultaneously, following [56]. In this way, category information is incorporated into domain information. Specifically, $\mathcal{G}_1$ consists of data pairs from the same domain and the same category, $\mathcal{G}_2$ consists of pairs from different domains (one comes from the noisy source domain and the other comes from the target domain) but from the same category, $\mathcal{G}_3$ consists of pairs from the same domain but from different categories, and $\mathcal{G}_4$ consists of pairs from various domains (one comes from the noisy source domain and the other comes from the target domain) and from different categories.

However, due to label noise on the source domain, we cannot simply use current noisy labels to pair data. Thus, we use the small-loss criterion [25,26] to select low-risk source data. Data with small loss is seen as an easy sample and is more likely to be clean. The detailed selection method is presented in Implementation Details (Section 7.2). For each class, we calculate the loss of each data point $(x_s, \tilde{y}_s) \in D_s$ with warmed-up source model $\tilde{\delta}^s$, i.e., $\ell_{ce}(\tilde{\delta}^s(x_s), \tilde{y}_s)$. According to the required number of data, we select the data with the smallest loss values and employ their nearly clean hidden representations to pre-train group discriminator G. The loss function is defined as,

$$\mathcal{L}_G = -\hat{\mathbb{E}}\left[\sum_{j=1}^{4} \tilde{y}_{\mathcal{G}_j} \log(G(\phi_s(\mathcal{G}_j)))\right],$$

where $\phi_s : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{2U}$ is a concatenation function of data pair (x_1, x_2) with

$$\phi_s(x_1, x_2) := [\psi^s_{n_1}(x_1), \psi^s_{n_1}(x_2)].$$

After pre-training, group discriminator G can identify whether the domains and categories of data pairs are the same.

6.3. Update target model and confuse group discriminator simultaneously

Ultimately, we aim to learn a target-specific model f_t with the help of pre-trained source model $\tilde{\delta}^s$ and group discriminator G. $f_t = \arg\max \delta^t$, where $\delta^t = h_t \circ g_t$. On the one hand, we hope f_t will fully fit labeled target data to leverage the limited supervision information of the target domain. On the other hand, we hope the hidden representations induced by f_t can confuse the group discriminator to perform adversarial training. We initialize target encoder g_t by source encoder g_s and train target classifier h_t from scratch, because the source classifier h_s is negatively affected. Based on the above two goals and non-saturating game [23,56], the loss function of target model is defined as

$$\mathcal{L}_T = -\frac{1}{|D_t|} \sum_{(x_t, y_t) \in D_t} y_t \cdot \log(\delta^t(x_t)) - \tau \mathbb{E}\left[y_{\mathcal{G}_1} \log(G(\phi_t(\mathcal{G}_2))) + y_{\mathcal{G}_3} \log(G(\phi_t(\mathcal{G}_4)))\right], \tag{4}$$

where τ is a hyper-parameter to trade-off target classification and confusing group discriminator, and y_t is the label of target data with a one-hot version. ϕ_t is the concatenation function with $\phi_t(x_1, x_2) := [\psi^t_{n_1}(x_1), \psi^t_{n_1}(x_2)]$, where $\psi^t_{n_1}(\cdot)$ denotes the hidden representations induced by target encoder g_t.

6.4. Adversarial training

We will alternate steps (b) and (c) of NT-FAN to perform adversarial training. In detail, we first train the group discriminator to identify the category and domain of data pairs accurately. Then, we train the target model to confuse the group discriminator and classify the categories of target data simultaneously. These two procedures are performed alternately. As the procedure of adversarial training goes on, the group discriminator will be unable to identify their domains of source data representations and target data representations. This ideal situation indicates that the hidden representations of source and target data are subject to the same distribution in embedding space.

7. Experiments

This section compares our proposed NT-FAN with six strong baselines on both digital and objective datasets. The detailed experimental settings, results, and analysis are presented below.

7.1. Datasets and tasks

We evaluate our method and baselines on both digital datasets and objective datasets.

For digital datasets, following previous works [15,56], we choose MNIST (M) [37], SVHN (S) [58], and USPS (U) [29]. Then, these three datasets can form six basic WFDA tasks, i.e., MNIST→SVHN (M→S), SVHN→MNIST (S→M), MNIST→USPS (M→U), USPS→MNIST (U→M), SVHN→USPS (S→U), and USPS→SVHN (U→S). Since these datasets are originally clean, we need to corrupt them manually. Based on Subsection 3.2, we use a noise transition matrix to generate class-dependent label noise (including pair flipping and symmetric flipping) and instance-dependent label noise with a noise rate of ρ. Here we choose noise rate ρ as 20% following previous works [14,46] and also evaluate our method on larger ρ (e.g., 40%). As the number of target data is very scarce in

WFDA, we randomly select $\{1,2,3,4,5,6,7\}$ data per class from the target dataset of each task, and thus only $\{10,20,30,40,50,60,70\}$ data are available in the target domain.

For objective datasets, following previous works [14,50,56], we evaluate our method on Office-31 [70], ImageCLEF-DA,[2] PACS [39], and VisDA 2017.[3]

- **Office-31** includes three domains, i.e., Amazon (A), Dslr (D), and Webcam (W). These three domains also can form six WFDA tasks, i.e., $A\rightarrow D$, $D\rightarrow A$, $A\rightarrow W$, $W\rightarrow A$, $D\rightarrow W$, and $W\rightarrow D$. Constrained by the data volume of each domain, we set the number of target data to $\{3,5,7\}$ for task $\{\{D\rightarrow A, D\rightarrow W\},\{W\rightarrow A, W\rightarrow D\},\{A\rightarrow D, A\rightarrow W\}\}$ following previous work [14].
- **ImageCLEF-DA** is a challenge that addresses domain adaptation problems in visual recognition tasks. It contains 12 classes from three different domains, including Caltech 256 (C), ImageNet 2012 (I), and PASCAL VOC 2012 (P). Permuting all three domains can build six WFDA tasks: $C\rightarrow I$, $I\rightarrow C$, $C\rightarrow P$, $P\rightarrow C$, $I\rightarrow P$, and $P\rightarrow I$. The number of target data for each class is set to 7.
- **PACS** is a popular benchmark for domain generalization and is also available for domain adaptation. It contains seven classes from four different domains, including Photo (P), Art painting (A), Cartoon (C), and Sketch (S). Permuting all four domains can build 12 WFDA tasks: $A\rightarrow C$, $C\rightarrow A$, $A\rightarrow P$, $P\rightarrow A$, $A\rightarrow S$, $S\rightarrow A$, $C\rightarrow P$, $P\rightarrow C$, $C\rightarrow S$, $S\rightarrow C$, $P\rightarrow S$, and $S\rightarrow P$. The number of target data for each class is set to 7, but the tasks $\cdot\rightarrow S$ are set to 5 constrained by the data volume of the Sketch domain.
- **VisDA 2017** is a visual challenge focusing on domain adaptation, including classification and segmentation. It contains 12 classes from two domains: the source domain is synthesized by rendering CAD models from 12 different classes; the target domain is more complex and diverse real images from the MSCOCO dataset, making the adaptation challenging. VisDA 2017 builds only one WFDA task, i.e., *Synthetic→Real*, with the number of target data for each class set to 10.

Since the above objective datasets are also originally clean, thus we also need to corrupt them manually. We also corrupt the source domain of each task by two types of label noise (i.e., the class-dependent variant and the instance-dependent variant), respectively, with noise rate ρ of 20%.

7.2. Implementation details

We conducted all experiments with default parameters by PyTorch 1.7.1 with Python 3.7.6 and NVIDIA TITAN V GPUs. For tasks in $\{M\rightarrow S, M\rightarrow U, U\rightarrow M, U\rightarrow S\}$, we use LeNet-5 as encoder. For tasks in $\{S\rightarrow M, S\rightarrow U\}$ and tasks built on Office-31 and ImageCLEF-DA, we use ResNet-18 [27] as their encoder. For all tasks built on VisDA 2017 and PACS, we use ResNet-50 [27] as their encoder. For all tasks, we use 3-layer fully connected network as the group discriminator and 1-layer fully connected network as the classifier. For all tasks, we choose Adam optimizer [34] with an initial learning rate of 0.0003. In the source pre-training procedure, the batch size is set to 256, while in the training procedure of the group discriminator and target model, the batch size is set to 40. For the three steps of NT-FAN, the numbers of training epochs are 1000, 100, and 500, respectively. We choose the last layer of the encoder as the hidden layer to induce hidden data representations.

Low-risk source data selection During the pre-training of the group discriminator, we used the small-loss criterion, and the details are introduced here. We rank all source data from smallest to largest according to their loss values and randomly sample source data from the top $\Delta\%$ ones. τ depends on the noise rate ρ. We treat ρ as prior knowledge. Otherwise, we can estimate it using validation set [49]. A larger noise rate implies that fewer top-$\Delta\%$ data is clean, so we need a smaller Δ. Finally, we choose $\Delta(\rho)=100-90*\rho$, where ρ is the noise rate, and the scaling factor 0.9 is used for addressing extreme scenes, such as ρ is very close to 1. Without the scaling factor, the $\Delta\approx 0$, making our method unable to proceed.

Choice of hyper-parameter τ in Eq. (4) In our implementation, we choose τ as 1 across all tasks. Our method is not sensitive to the choice of τ. To demonstrate it, we take three digital tasks ($M\rightarrow U$, $S\rightarrow U$, $S\rightarrow M$) as an example. The label corruption method is 20%-symmetric flip, and the number of target data per class is 7. The result is shown in Fig. 6. We find that the final accuracy among the three tasks is nearly unaffected by the $\tau\in\{0.1,0.5,1,5\}$, demonstrating that our method is robust to τ.

7.3. Baselines

We compare NT-FAN with six important and strong baselines, including two basic domain adaptation baselines (without adaptation, finetuning), two FDA methods (FADA, TPN+ATA), one label correction method (LNL), one existing WFDA method.

- **Without adaptation (WA)**. We train a source model with noisy source data directly and use this well-trained source model to classify data on the target domain.
- **Finetuning (FT)**. We first pre-train a source model with noisy source data, then we further tune this warmed-up model with few labeled target data.

[2] https://www.imageclef.org/2014/adaptation.
[3] http://ai.bu.edu/visda-2017/.

Table 1
Classification accuracy±standard deviation (%) on 6 digital WFDA tasks under class-dependent label noise (pair flipping) with noise rate of 20%. "WA" denotes without adaptation, "FT" denotes finetuning, and "T+A" denotes TPN+ATA.

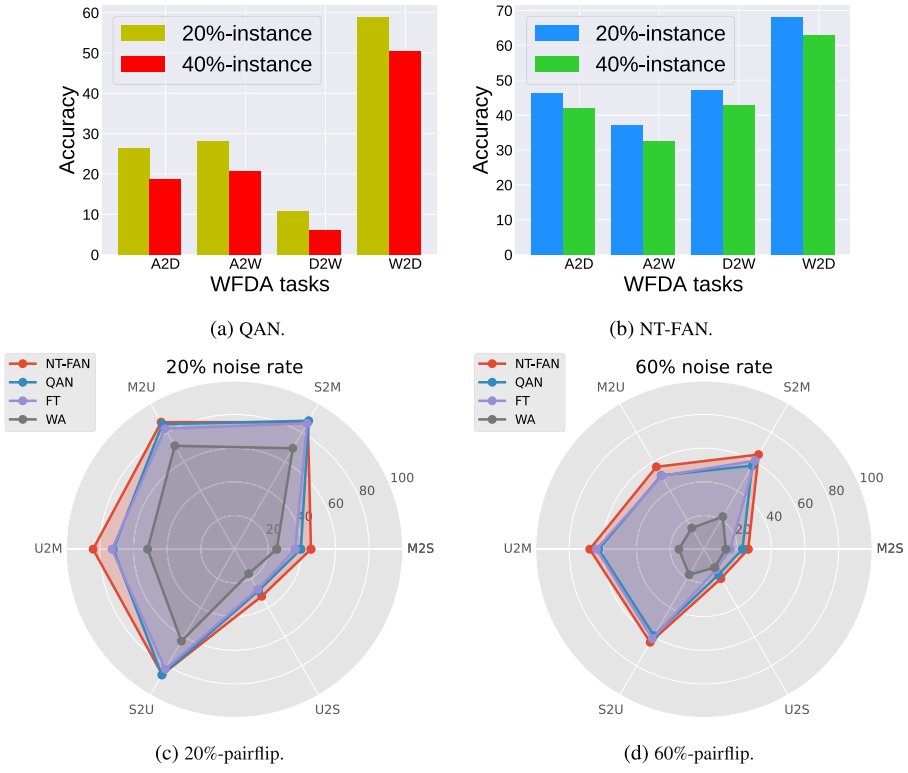
Tasks	WA	WFDA Methods	Number of Target Data per Class							Avg.
			1	2	3	4	5	6	7	
$M \rightarrow S$	25.1	FT	26.7±1.0	28.3±1.4	30.8±0.8	33.6±1.0	35.7±2.1	36.7±1.0	36.9±1.6	32.7
		FADA	25.3±2.0	28.2±0.8	30.1±0.9	30.3±1.4	32.4±1.6	34.7±2.3	36.0±1.2	31.0
		T+A	26.1±1.2	27.8±0.5	30.1±2.0	30.0±1.7	31.9±1.4	34.8±0.9	35.5±1.0	30.9
		LNL	30.0±0.5	31.3±0.3	33.9±0.6	34.6±0.5	33.9±0.2	38.4±0.4	38.1±0.3	34.3
		QAN	**30.6±1.2**	**32.9±0.7**	31.1±0.5	31.8±1.4	34.2±2.0	37.6±1.2	39.4±1.7	33.9
		NT-FAN	27.5±1.6	32.6±1.3	**34.7±0.7**	**39.4±0.8**	**40.4±0.8**	**41.4±0.7**	**45.4±0.5**	**37.3**
$S \rightarrow M$	69.2	FT	71.3±1.2	76.3±2.5	74.2±1.8	77.0±3.1	84.9±2.0	88.4±2.6	90.1±2.2	80.3
		FADA	77.1±1.1	79.0±0.7	82.1±1.5	83.0±1.4	83.3±0.7	84.9±2.3	86.3±3.1	82.2
		T+A	75.4±1.2	78.0±3.6	80.1±1.4	81.7±1.6	83.2±2.0	83.8±1.3	85.1±2.3	81.0
		LNL	73.1±1.4	78.8±1.4	77.0±1.5	79.5±2.7	85.3±1.8	**88.6**±1.6	**90.6**±1.4	81.8
		QAN	**79.0±0.7**	**81.8±2.2**	**83.5±1.6**	**84.8±2.4**	85.4±1.9	88.3±3.1	88.1±1.2	**84.4**
		NT-FAN	77.5±0.8	81.5±1.4	82.1±1.3	83.6±1.4	**85.6±1.2**	85.8±0.7	87.1±0.5	83.3
$M \rightarrow U$	70.9	FT	76.8±0.7	76.1±1.2	78.4±1.0	81.8±1.7	80.2±2.0	83.9±1.3	83.8±1.6	80.1
		FADA	73.9±2.5	74.4±3.0	74.8±1.4	78.7±1.0	76.4±3.1	80.0±0.7	82.6±1.5	77.3
		T+A	78.4±1.3	78.8±0.7	80.4±2.0	**83.6±2.1**	84.8±1.3	85.1±2.2	85.6±0.9	82.4
		LNL	76.1±0.5	76.1±0.3	76.7±0.3	77.8±0.1	78.8±0.3	79.8±0.1	80.3±0.2	77.9
		QAN	78.1±1.2	80.9±2.1	**82.1±0.8**	82.1±3.1	84.4±1.8	85.4±0.6	85.7±1.3	82.7
		NT-FAN	**78.6±1.4**	**81.7±1.0**	81.5±0.8	83.5±0.7	**85.8±1.1**	**86.2±0.8**	**87.1±0.7**	**83.5**
$U \rightarrow M$	51.6	FT	53.2±1.0	56.2±1.4	59.0±2.4	64.5±2.2	62.7±3.2	65.6±2.6	66.7±1.8	61.1
		FADA	59.8±1.5	60.1±2.6	62.7±0.5	66.4±2.0	68.1±2.1	70.0±1.1	72.7±1.8	65.7
		T+A	61.2±0.8	61.6±1.2	64.5±2.9	65.2±1.7	64.3±1.0	67.1±0.9	69.5±1.8	64.8
		LNL	56.6±1.6	59.1±1.3	59.7±0.7	62.2±1.4	62.5±1.4	64.0±0.8	66.8±1.0	61.6
		QAN	60.5±2.6	65.5±1.4	66.6±0.4	68.2±1.7	69.9±2.6	70.6±1.0	72.0±1.8	67.6
		NT-FAN	**68.3±1.5**	**75.7±1.2**	**79.7±1.3**	**81.1±1.1**	**82.5±0.6**	**82.1±0.8**	**84.0±1.1**	**79.1**
$S \rightarrow U$	63.0	FT	64.4±2.0	66.5±1.5	66.9±1.0	70.0±2.4	74.7±1.0	78.4±1.7	80.2±1.4	71.6
		FADA	68.1±0.9	70.2±1.3	74.5±2.0	77.0±2.7	81.2±2.3	82.6±3.7	82.2±2.4	76.5
		T+A	68.4±2.7	72.1±0.8	70.6±1.0	74.7±0.9	79.3±2.6	81.8±2.4	82.0±1.7	75.6
		LNL	69.7±2.0	71.4±1.6	76.9±1.8	79.0±1.9	79.2±2.5	83.3±2.0	83.7±1.8	77.6
		QAN	73.1±1.9	**77.8±0.7**	78.5±2.2	**83.8±1.5**	**83.9±3.4**	**86.3±2.6**	**86.3±1.8**	**81.4**
		NT-FAN	**75.2±1.3**	77.1±1.2	**79.0±0.8**	81.2±1.5	82.3±1.0	84.5±1.2	85.9±1.2	80.7
$U \rightarrow S$	16.8	FT	17.2±1.8	21.4±2.0	21.7±1.1	24.5±2.3	23.9±1.7	25.9±2.4	25.6±0.9	22.9
		FADA	19.2±1.5	21.3±1.0	21.8±1.4	23.7±2.1	24.2±1.0	25.9±1.6	27.5±2.4	23.4
		T+A	**22.6±0.8**	23.1±1.9	24.3±0.7	24.8±0.4	25.0±0.6	25.9±1.2	26.4±0.5	24.6
		LNL	22.2±1.1	23.5±0.8	25.5±1.0	25.9±0.6	25.6±0.3	26.1±0.3	26.9±0.2	25.1
		QAN	21.5±0.6	23.5±0.8	23.8±0.5	25.3±1.0	25.8±0.7	27.7±0.7	28.5±1.3	25.2
		NT-FAN	21.8±1.4	**24.5±1.3**	**28.4±1.5**	**30.3±1.3**	**32.7±1.0**	**31.8±1.2**	**32.3±1.4**	**28.8**

- **FADA**. *Few-shot adversarial domain adaptation* (FADA) [56] proposed to augment the target data by pairing data based on their different categories and domains to four groups and perform adversarial domain adaptation, which is an important foundation of NT-FAN.
- **TPN+ATA**. TPN+ATA [87] proposed to consider the hardest samples around the source domain, and proposed an adversarial task augmentation method to generate inductive bias-adaptive hard tasks to improve the robustness of the algorithm to the inductive bias.
- **LNL**. We first employ a representative label-noise learning method, Co-teaching [26], to correct the noisy labels of the source domain, and use the corrected source data to pre-train a source model. Then, we transfer this source model to the target domain with few target data.
- **QAN**. QAN [14] proposed to maintain four groups of adaptation model (two groups from the source domain and the other two groups from the target domain) to filter noisy data for others in the training procedure collaboratively, and use highly trustworthy data to update four groups of the model, together with data augmentation strategy in FADA.

7.4. NT-FAN under class-dependent source noise

We evaluate the effectiveness of NT-FAN under class-dependent label noise on digital WFDA tasks. We use pair flipping (Table 1, Table 5, and Table 4) and symmetric flipping (Table 2) to corrupt source data.

It can be easily found that NT-FAN outperforms baselines on most of the sub-tasks, except for $S \rightarrow M$ and $S \rightarrow U$. When target data are too few, we find that NT-FAN does not always beat other baselines. The main reason is that NT-FAN relies on minimizing the empirical risk of target data (i.e., the first term in $\mathcal{L}_T$), and too few target data cannot provide enough information about the target domain. However, when target data are relatively more (i.e., the number of target data per class is more than 5), NT-FAN can significantly outperform other baselines. In real scenarios, the data volume is hardly very small (e.g., the number of data per



(a) QAN. (b) NT-FAN.

(c) 20%-pairflip. (d) 60%-pairflip.

Fig. 4. Robustness of NT-FAN against **serious** source-domain label noise. (a) and (b) show the performance drops of NT-FAN and QAN when the instance-dependent noise rate increases from 20% to 40% on the Office dataset. (c) and (d) show performance comparisons of NT-FAN and other methods under the 20%-pairflip and 60%-pairflip on the digital tasks.

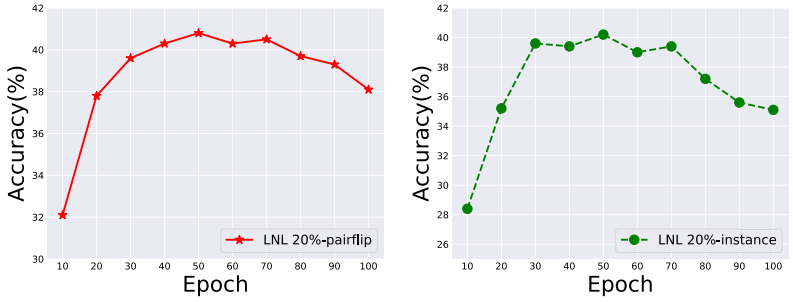


Fig. 5. Vulnerability of label-noise learning method on WFDA. LNL is evaluated on $M \rightarrow S$ with 7 target data per class.

class is less than 2). The performance of NT-FAN on $S \rightarrow M$ and $S \rightarrow U$ is not good enough. This phenomenon results from the first pre-training stage. Specifically, fully fitting the corrupted SVHN dataset is very hard for medium-sized CNN-based models, and we can only achieve the training accuracy of nearly 90% at most. Thus, obtaining high-quality hidden representations is hard, leading to performance degradation.

In Table 4 and Table 5, NT-FAN also significantly outperforms the baseline methods on most of the sub-tasks. However, for three sub-tasks of the PACS dataset (Table 4), LNL is marginally better. LNL mostly relies on the noisy label correction by a label-noise learning algorithm, and Co-teaching achieves a better correction effect on Photo (P) and Cartoon (C) than others. For more difficult sub-tasks, NT-FAN is consistently better.

7.5. NT-FAN under instance-dependent source noise

We evaluate the effectiveness of NT-FAN under instance-dependent label noise on objective WFDA tasks, i.e., six sub-tasks on the Office-31 (Table 3), twelve sub-tasks on the PACS (Table 4), six sub-tasks on the ImageCLEF-DA, and one sub-task on the VisDA (Table 5). The way of generating instance-dependent label noise has been presented in subsection 3.2. From Table 3, Table 4 and Table 5, we find that NT-FAN significantly outperforms baselines and achieves a huge lead. This result is because NT-FAN does

Table 2
Classification accuracy±standard deviation (%) on 6 digital WFDA tasks under class-dependent label noise (symmetric flipping) with noise rate of 20%. "WA" denotes without adaptation, "FT" denotes finetuning, and "T+A" denotes TPN+ATA.

Tasks	WA	WFDA Methods	Number of Target Data per Class							Avg.
			1	2	3	4	5	6	7	
$M \to S$	14.3	FT	14.4±1.0	18.2±1.4	20.5±2.0	24.4±0.9	26.1±1.3	26.8±2.4	26.4±1.9	22.4
		FADA	23.1±0.7	26.3±2.0	25.2±1.9	29.9±0.7	32.5±0.8	34.0±1.7	34.8±1.6	29.4
		T+A	20.1±1.1	24.0±1.8	25.7±2.4	28.6±1.6	30.2±0.7	29.7±2.8	33.2±1.4	27.4
		LNL	**31.9±1.0**	**35.7±0.7**	**38.7±1.1**	38.8±1.0	38.5±0.6	41.5±0.7	40.0±0.6	37.9
		QAN	28.3±0.3	31.2±1.1	32.8±0.8	33.2±1.9	33.4±2.5	36.1±1.8	37.7±1.3	33.2
		NT-FAN	28.1±1.2	32.3±1.0	35.4±1.3	**40.0±0.6**	**44.3±0.8**	**44.9±0.6**	**47.0±1.0**	**38.9**
$S \to M$	65.0	FT	71.8±0.8	74.8±1.2	79.4±2.0	84.0±1.3	**88.9±2.6**	88.2±1.4	**89.9±1.4**	82.4
		FADA	**81.3±0.3**	81.8±1.0	82.7±0.9	86.4±1.6	84.2±0.7	86.8±2.4	87.2±2.0	84.3
		T+A	75.9±2.5	76.1±1.8	82.0±2.9	80.3±1.5	82.7±0.8	81.0±2.7	84.6±1.6	80.4
		LNL	74.2±2.2	78.9±1.5	80.2±1.9	84.9±0.9	83.1±1.3	86.4±1.0	87.7±1.1	82.2
		QAN	80.3±0.5	**84.8±1.2**	83.2±0.8	86.6±1.1	85.6±2.1	**88.6±1.8**	89.5±0.9	**85.5**
		NT-FAN	74.8±1.3	83.5±1.0	**84.6±1.7**	**86.7±0.6**	87.1±0.7	88.0±0.4	88.0±0.6	84.7
$M \to U$	63.3	FT	67.6±1.0	67.9±0.5	74.8±2.0	79.5±1.3	76.1±3.1	82.0±1.2	82.5±1.6	75.8
		FADA	70.1±1.3	73.4±0.8	74.9±1.0	77.1±2.2	80.3±0.7	81.8±1.9	81.4±0.5	77.0
		T+A	69.8±2.6	73.1±2.0	76.9±3.1	77.2±0.9	80.9±1.6	82.3±0.7	81.8±1.9	77.4
		LNL	79.2±1.9	79.1±1.4	79.4±2.1	80.0±1.2	81.2±1.7	85.0±1.2	82.5±1.2	80.9
		QAN	79.7±0.7	**84.2±0.9**	85.5±0.5	85.2±1.2	87.1±0.7	87.2±1.0	87.4±0.6	**85.2**
		NT-FAN	**79.9±1.2**	80.7±1.0	**85.9±1.4**	**86.2±1.0**	**87.6±0.5**	**88.2±0.6**	**88.0±0.9**	**85.2**
$U \to M$	59.0	FT	61.3±0.6	66.7±1.6	68.4±0.8	70.9±1.3	72.4±1.6	75.2±2.0	76.8±1.1	70.2
		FADA	63.6±1.3	65.6±2.6	70.1±1.0	70.6±1.6	69.8±2.0	74.0±1.3	74.5±0.7	69.7
		T+A	60.5±1.1	65.3±1.0	66.8±2.2	69.5±0.8	70.0±1.4	72.6±1.8	75.0±0.4	68.5
		LNL	51.3±0.9	55.8±1.3	58.7±1.4	64.5±0.6	62.9±1.4	67.3±1.8	69.2±0.6	61.4
		QAN	63.1±1.3	69.1±1.8	75.2±0.8	73.3±1.0	74.0±3.4	76.5±2.1	77.9±1.3	72.7
		NT-FAN	**67.8±1.0**	**77.6±1.5**	**81.0±1.3**	**81.8±1.6**	**83.5±0.7**	**82.7±1.5**	**84.7±0.8**	**79.9**
$S \to U$	67.3	FT	74.3±1.1	76.5±1.5	76.7±1.7	79.3±1.1	83.1±2.3	86.3±0.5	89.0±1.4	80.7
		FADA	76.5±2.0	79.1±1.6	84.2±3.1	82.9±2.5	86.1±1.4	86.5±1.2	87.4±1.9	83.2
		T+A	77.1±0.6	77.7±1.4	80.0±0.6	82.7±1.4	82.9±2.1	83.6±1.2	86.1±2.0	81.4
		LNL	71.3±1.5	77.8±1.3	80.4±0.8	82.1±1.7	83.8±1.8	86.9±0.9	86.0±1.1	81.2
		QAN	**82.0±2.1**	**83.2±1.5**	**85.7±2.0**	**85.9±1.1**	**86.3±3.1**	**88.4±2.3**	**90.4±2.2**	**86.0**
		NT-FAN	77.5±1.1	78.1±0.8	79.0±0.8	80.1±1.3	82.6±1.3	83.5±0.7	86.2±1.0	81.0
$U \to S$	16.3	FT	18.7±1.8	23.6±2.7	23.8±1.6	24.6±1.4	24.6±1.2	24.8±0.7	29.1±1.3	24.2
		FADA	21.1±1.2	22.5±0.4	25.1±2.0	24.0±1.2	25.9±1.4	26.4±0.7	27.5±1.1	24.6
		T+A	20.3±0.4	21.9±1.1	23.6±0.7	25.1±0.5	25.7±1.2	26.3±1.0	27.2±0.7	24.3
		LNL	17.9±1.3	24.1±1.0	24.6±1.3	25.5±0.4	25.0±1.1	25.6±0.6	28.9±1.0	24.5
		QAN	18.9±0.4	20.2±0.2	20.8±0.7	22.0±0.5	23.6±1.2	26.3±0.8	30.7±1.3	23.2
		NT-FAN	**23.5±0.8**	**25.7±1.1**	**25.9±0.6**	**28.5±1.3**	**31.0±1.1**	**29.8±1.6**	**31.5±0.7**	**28.0**

Table 3
Classification accuracy±standard deviation (%) on Office-31 dataset under instance-dependent label noise with noise rate of 20%. "WA" denotes without adaptation, "FT" denotes finetuning, and "T+A" denotes TPN+ATA.

Methods	$A \to D$	$A \to W$	$D \to A$	$D \to W$	$W \to A$	$W \to D$	Avg.
WA	13.0	7.5	3.8	33.4	5.5	25.1	14.7
FT	39.7±3.2	24.8±2.6	11.7±1.4	35.9±1.4	12.9±1.5	62.2±2.9	31.2
FADA	23.4±2.1	17.1±2.4	8.6±1.3	29.4±3.3	11.4±0.8	37.1±2.0	21.2
T+A	26.2±2.1	30.2±1.7	11.4±0.4	36.0±2.8	13.6±1.3	62.2±2.4	29.9
LNL	40.4±2.5	28.0±2.1	12.9±2.1	38.1±1.3	15.6±2.2	64.6±1.8	33.3
QAN	25.7±1.4	26.5±2.2	14.2±0.5	41.7±2.2	14.7±1.3	59.0±1.5	30.3
NT-FAN	**46.3±1.7**	**37.2±1.3**	**17.6±0.7**	**47.3±0.9**	**20.1±1.2**	**68.0±1.2**	**39.4**

not rely on a specific structure of the noise transition matrix, and thus can effectively address more complex scenarios, such as instance-dependent label noise on the source domain. For QAN, it cannot handle irregular label noise, and the adaptation performance naturally degrades. It is worth noting that finetuning achieves a relatively good adaptation performance, and we think it depends on its simpleness and stability. Thus, compared with the previous WFDA method QAN, NT-FAN can effectively address more realistic and intractable label noise on the source domain and therefore is highly practical.

Table 4
Results on the PACS dataset. The noise rate on the source domain is 20%. For the task $\cdot \rightarrow S$, 5 target data per class is available, while 7 target data per class is available for the other tasks. We report the results of WFDA under both class-dependent and instance-dependent source noises.

Method	$A \rightarrow C$	$C \rightarrow A$	$A \rightarrow P$	$P \rightarrow A$	$A \rightarrow S$	$S \rightarrow A$	$C \rightarrow P$	$P \rightarrow C$	$C \rightarrow S$	$S \rightarrow C$	$P \rightarrow S$	$S \rightarrow P$	Avg.
Class-dependent source noise (pairflip)													
WA	25.9	17.1	38.6	14.1	24.4	15.1	12.3	6.2	14.7	12.9	8.9	5.1	18.6
FT	46.0	37.1	48.9	40.5	31.5	33.7	50.3	40.4	30.7	37.5	25.8	44.3	37.2
FADA	41.6	39.4	52.7	38.5	34.0	30.7	51.2	37.0	35.3	38.5	25.1	40.3	38.1
T+A	44.8	36.9	50.0	42.6	30.4	32.5	49.5	41.8	32.2	39.5	26.7	44.3	37.7
LNL	50.9	39.2	51.1	39.0	31.8	34.0	**55.5**	**47.7**	42.4	40.7	**28.2**	43.8	40.7
QAN	52.6	43.5	57.3	47.1	30.5	33.8	51.8	42.0	38.1	39.8	25.5	46.9	40.9
NT-FAN	**60.3**	**45.4**	**83.0**	**47.9**	**38.7**	**35.5**	53.4	44.6	**42.5**	**42.9**	26.0	**47.7**	**48.0**
Instance-dependent source noise													
WA	25.0	22.9	39.8	15.1	18.3	11.2	33.0	10.3	9.5	16.1	9.5	8.5	20.6
FT	47.7	36.6	59.7	43.4	28.5	30.7	55.1	44.2	27.0	43.3	31.3	35.2	39.4
FADA	38.2	31.1	53.5	37.0	24.3	27.5	58.4	49.3	26.0	44.6	29.0	37.9	36.7
T+A	48.3	40.6	58.0	40.9	29.6	33.5	61.2	45.0	33.4	49.6	31.7	39.2	42.8
LNL	51.3	37.8	55.7	44.2	28.5	33.8	56.7	52.3	27.2	49.3	32.7	40.4	40.9
QAN	49.6	47.1	60.4	40.6	29.9	32.4	66.7	**53.7**	35.4	45.0	30.8	40.7	43.2
NT-FAN	**52.9**	**52.2**	**68.8**	**46.3**	**30.1**	**35.1**	**67.6**	50.4	**39.4**	**59.4**	**33.6**	**45.8**	**47.9**

7.6. Robustness of NT-FAN under serious source noise

We also evaluate NT-FAN under more serious label noise. We take objective tasks and instance-dependent label noise with the noise rate of 40% as an example. We compare the performance of NT-FAN and QAN [14] (i.e., a previous WFDA method) under the noise rates of 20% and **40%**, respectively, and the results can be seen in Fig. 4. Fig. 4b shows that the performance of NT-FAN decreases little as the noise rate increases from 20% to **40%**, while the performance of QAN decreases much more as shown in Fig. 4a. These two figures reflect the robustness of NT-FAN under serious source noise and the vulnerability of QAN to severe instance-dependent label noise.

In addition, we further evaluate NT-FAN under the **60%** noise rate. Taking digital tasks as examples, we choose the number of target data per class as 7, and use the pairflip to generate label noise on the source domain. The comparison results are shown in Fig. 4c and 4d. Although the 60% noise rate will degrade both NT-FAN (ours) and other baseline methods significantly, NT-FAN still consistently outperforms other methods under such a very serious noise scenario.

7.7. Vulnerability of LNL methods for WFDA

Label-noise learning (LNL) is an important and direct two-step baseline for WFDA, and its details are introduced in 7.3. However, such a simple method cannot address WFDA effectively. In Fig. 5, we show the training processes of the LNL method under different types of source noise. We take $M \rightarrow S$ with 7 target data per class as an example. We can easily find that although the LNL method can achieve performance improvement at the early stage, it will significantly degrade at a later stage for both types of source noise. As the existing label-noise learning methods cannot completely correct the noisy labels, the ones that have not been corrected will persistently degrade the source model, leading to serious overfitting. In contrast, NT-FAN does not correct the noisy labels but tries to obtain useful data representations directly from noisy data. This operation prevents the classifier from overfitting the data whose label has not been corrected.

7.8. Ablation study

In this section, we perform an ablation study to verify the effectiveness of hidden representations in NT-FAN. Specifically, as shallow layers cannot capture the high-level semantic features of images [112], we employ the penultimate layer of ResNet-18 as the hidden layer and compare it with the last layer. We choose digital tasks and 7 target data per class as examples, and the results are shown in Table 6. We can easily see that NT-FAN with hidden representations achieves better performance, and their standard deviations are smaller, indicating that hidden representations in NAT-FAN are cleaner and more robust. Thus, this ablation study verifies the effectiveness of hidden representations in NT-FAN based on various noises and tasks.

To further evaluate the usefulness of hidden representations, we select another downstream task, linear probing (an effective approach to evaluate the quality of data representations), to make an assessment. We use the data representations induced by the last and penultimate layers to perform linear probing. We use these two types of data representations to train logistic regression models with ground-truth labels, respectively. For simplicity, we choose CIFAR-10 and CIFAR-100 as examples. The deep model is ResNet-50, and the training data involves 20% instance-dependent label noise. The results are shown in Table 7. The representations induced by the penultimate layer perform better on the linear probing task. This result serves as another evidence that the hidden representations are significantly high-quality.

Table 5
Results on the ImageCLEF-DA and VisDA 2017. The noise rate on the source domain is 20%. 7 target data per class is available for all tasks from ImageCLEF-DA. 10 target data per class is available for VisDA 2017. We report the results of WFDA under both class-dependent and instance-dependent source noises.

Method	*I→P*	*P→I*	*I→C*	*C→I*	*C→P*	*P→C*	Avg.	*Synthetic→Real*
Class-dependent source noise (pairflip)								
WA	18.7	27.1	32.8	19.0	20.2	21.3	23.2	10.5
FT	33.2	31.2	48.5	38.2	31.0	50.5	37.0	41.6
FADA	33.0	38.1	56.6	37.2	31.6	47.5	36.6	35.3
T+A	31.2	26.8	49.5	33.0	29.4	51.7	34.6	33.7
LNL	33.1	37.8	51.5	33.4	33.3	52.3	35.9	42.8
QAN	35.4	39.8	52.3	42.0	27.5	**54.9**	38.8	40.5
NT-FAN	**37.5**	**42.2**	**60.1**	**45.9**	**33.4**	53.1	**41.6**	**45.6**
Instance-dependent source noise								
WA	25.8	24.9	29.9	17.7	19.1	16.7	22.3	7.2
FT	32.5	34.9	47.1	33.8	28.2	49.4	35.7	39.0
FADA	30.6	28.4	41.0	31.7	26.0	39.4	35.3	34.3
T+A	32.5	37.6	50.9	31.9	32.7	48.6	34.6	38.5
LNL	35.3	38.2	54.9	34.9	33.2	48.5	35.9	42.1
QAN	35.0	38.4	51.5	38.3	36.9	51.1	38.3	38.6
NT-FAN	**39.0**	**44.2**	**62.1**	**45.3**	**38.7**	**54.4**	**44.2**	**44.2**

Table 6
Ablation study. As examples, we choose digital tasks and **7** target data per class. "HR" denotes the hidden representation, and "w/o" is the abbreviation of without. "Symmetric", "Pair", and "Instance" mean symmetric flipping, pair flipping, and instance-dependent label noise, respectively.

Noise Types	Methods	Tasks					
		M→S	*S→M*	*M→U*	*U→M*	*S→U*	*U→S*
Symmetric	NT-FAN w/o HR	42.7±0.4	81.9±0.8	82.7±0.6	79.8±0.9	82.8±1.0	24.8±1.1
	NT-FAN	**47.0±0.5**	**88.0±0.6**	**88.0±0.9**	**84.7±0.8**	**86.2±1.0**	**31.5±0.6**
Pair	NT-FAN w/o HR	32.7±0.3	70.6±0.5	79.7±1.2	79.2±1.0	69.0±1.4	30.8±0.6
	NT-FAN	**45.4±0.8**	**87.1±0.4**	**87.1±0.7**	**84.0±1.0**	**85.9±0.8**	**32.3±1.1**
Instance	NT-FAN w/o HR	40.5±0.8	72.3±1.5	86.1±2.0	75.0±1.9	75.9±1.6	23.3±0.5
	NT-FAN	**44.6±1.4**	**76.5±1.9**	**88.3±1.7**	**82.7±0.9**	**83.4±2.2**	**28.3±1.3**

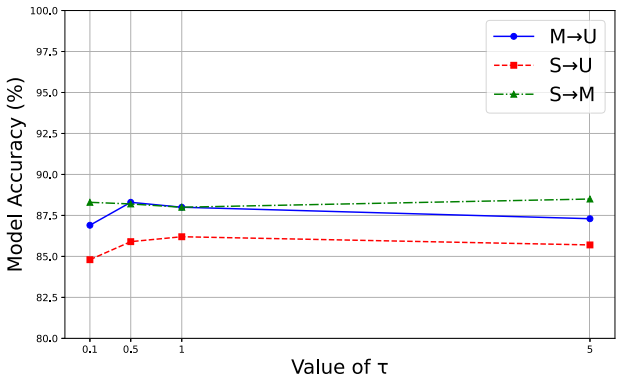


Fig. 6. A analysis about the choice of τ in Eq. (4).

Table 7
A quality comparison between the data representations induced by the penultimate layer and the last layer, using linear probing. We take CIFAR-10/100 as an example, and the label corruption method is 20%-instance.

	penultimate layer	last layer
CIFAR-10	**78.5%**	69.7%
CIFAR-100	**46.1%**	29.7%

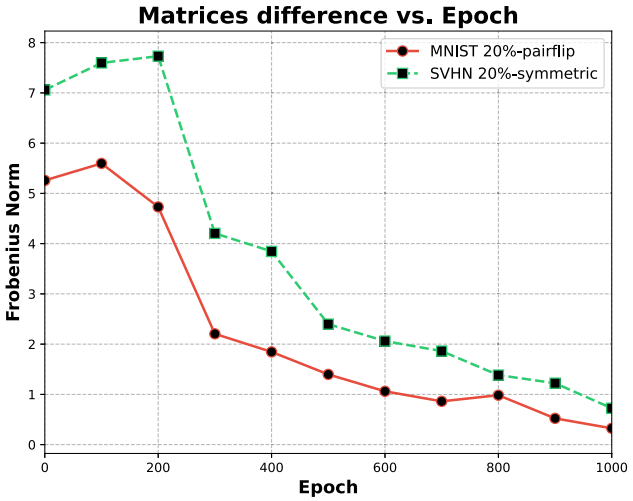


Fig. 7. The variation trend in the difference between the ground-truth noise transition matrix and the approximated one.

Table 8
Classification accuracy±standard deviation (%) on 3 tasks without source label noise (i.e., degrade to **FDA**). The results of FT and FADA refer to [56]. "WA" denotes without adaptation, "FT" denotes finetuning, and "T+A" denotes TPN+ATA.

Tasks	WA	FDA Methods	Number of Target Data per Class						
			1	2	3	4	5	6	7
$M \to S$	20.3	FT	29.7	31.2	36.1	36.7	38.1	38.3	39.1
		FADA	**37.7**	**40.5**	**42.9**	46.3	46.1	46.8	47.0
		T+A	36.4	35.7	37.8	40.1	40.6	41.0	41.7
		NT-FAN	36.0	39.8	40.6	**46.7**	**48.3**	**48.8**	**50.2**
$S \to M$	60.1	FT	65.5	68.6	70.7	73.3	74.5	74.6	75.4
		FADA	72.8	81.8	82.6	85.1	86.1	86.8	87.2
		T+A	68.1	76.3	78.5	80.1	80.8	82.6	82.2
		NT-FAN	**79.5**	**86.0**	**87.4**	**89.0**	**89.3**	**90.5**	**91.4**
$U \to S$	15.3	FT	19.9	22.2	22.8	24.6	25.4	25.4	25.6
		FADA	**27.5**	**29.8**	**34.5**	**36.0**	**37.9**	41.3	42.9
		T+A	17.5	22.0	25.8	23.7	27.3	30.1	32.7
		NT-FAN	26.6	28.3	33.9	35.0	37.6	**41.5**	**43.2**

7.9. Consistent effectiveness on the clean source domain

To demonstrate the wide applicability of NT-FAN except for the WFDA, we also evaluate NT-FAN on the condition of clean source domain (i.e., FDA). As shown in Table 8, we compare NT-FAN with other competitive FDA methods, taking the tasks of $M \to S$, $S \to M$, $U \to S$ as examples. We find that NT-FAN outperforms the baselines on most of the sub-tasks, especially when the number of target data is larger than 4 per class. When the target data amount is too little (e.g., less than 3), NT-FAN may not perform as well as other specialized FDA methods. This phenomenon mainly results from the robustness of source data representations induced by NT-FAN, making them harder to transfer than data representations induced by vanilla training given very limited target data. However, the above results imply that NT-FAN can address different variants of the FDA problem, regardless of source noise.

7.10. Relationship between the last layer and noise transition matrix

To link our theoretical result to our algorithm NT-FAN, we demonstrate that the inverse of the last layer's parameter matrix approximates the ground-truth noise transition matrix empirically. For convenience, we set the number of the penultimate layer's neurons equal that of the last layer, i.e., the number of classes. In our experimental setup, we inherently have access to the ground truth since the noisy labels are simulated using the noise transition matrix. We use the Frobenius norm to evaluate the difference between the ground-truth noise transition matrix and the approximated one. We use the normalized matrix to compute the Frobenius norm for a fair comparison.

Here we choose two cases, including "MNIST 20%-pairflp" and "SVHN 20%-symmetric". We present the variation trend in the difference between these two matrices as the fitting procedure progresses, shown in Fig. 7.

It is clear that the inverse of the last layer's parameter matrix gradually approximates the ground-truth noise transition matrix if the source model fully fits noisy source data. This result implies that NT-FAN implicitly learns and leverages the noise transition matrix and perfectly aligns with the Theorem 1. NT-FAN skillfully uses this good property to obtain clean source data representations in a simple and easy-to-implement way.

8. Conclusion

This paper focuses on a realistic and challenging problem, which learns a target model with noisy source data and few target data, called *wildly few-shot domain adaptation* (WFDA). To reduce training costs and accumulation of residual noises in previous works, we propose a simple yet effective method termed *noise-tolerant few-shot adaptation network* (NT-FAN). NT-FAN trains a deep network to fully fit noisy source data, whose proper hidden layers can induce nearly clean data representations. As hidden data representations are subject to the ground-truth source domain, we take them to perform domain adversary, thus source-specific knowledge is adapted to the target domain through domain-invariant representations. We also give theoretical analysis about correcting source noise in the training procedure and an excess risk bound on the target domain, which also motivates our method. Experiments verify the effectiveness of the more lightweight NT-FAN on various kinds of source noise, especially for more realistic instance-dependent noise. In addition, NT-FAN can remain robust when encountering severe source noise.

CRediT authorship contribution statement

Wenjing Yang: Writing – original draft, Methodology, Formal analysis, Conceptualization. **Haoang Chi:** Writing – review & editing, Writing – original draft, Validation, Software. **Yibing Zhan:** Writing – review & editing, Formal analysis. **Bowen Hu:** Writing – review & editing, Validation, Formal analysis. **Xiaoguang Ren:** Writing – review & editing, Validation. **Dapeng Tao:** Writing – review & editing, Validation, Supervision, Investigation. **Long Lan:** Writing – review & editing, Writing – original draft, Validation, Supervision, Project administration, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships that may be considered as potential competing interests: Long lan reports financial support was provided by National Natural Science Foundation of China (62376282).

Acknowledgement

The work was supported by the National Natural Science Foundation of China (No. 62376282, No. 62372459).

Data availability

Data will be made available on request.

References

[1] D. Angluin, P. Laird, Learning from noisy examples, Mach. Learn. (1988).
[2] E. Arazo, D. Ortego, P. Albert, N. O'Connor, K. McGuinness, Unsupervised label noise modeling and loss correction, in: ICML, 2019, pp. 312–321.
[3] K. Arndt, A. Ghadirzadeh, M. Hazara, V. Kyrki, Few-shot model-based adaptation in noisy conditions, IEEE Robot. Autom. Lett. 6 (2021) 4193–4200, https://doi.org/10.1109/LRA.2021.3068104.
[4] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: ICLR, 2015.
[5] Y. Bai, E. Yang, B. Han, Y. Yang, J. Li, Y. Mao, G. Niu, T. Liu, Understanding and improving early stopping for learning with noisy labels, in: NeurIPS, 2021.
[6] O. Bar, A. Drory, R. Giryes, A spectral perspective of dnn robustness to label noise, in: AISTATS, 2022, pp. 3732–3752.
[7] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, J.W. Vaughan, A theory of learning from different domains, Mach. Learn. 79 (2010) 151–175.
[8] D. Berthelot, N. Carlini, I.J. Goodfellow, N. Papernot, A. Oliver, C. Raffel, Mixmatch: a holistic approach to semi-supervised learning, in: NeurIPS, 2019, pp. 5050–5060.
[9] A. Berthon, B. Han, G. Niu, T. Liu, M. Sugiyama, Confidence scores make instance-dependent label-noise learning possible, in: ICML, 2021, pp. 825–836.
[10] A. Bitarafan, M.S. Baghshah, M. Gheisari, Incremental evolving domain adaptation, IEEE Trans. Knowl. Data Eng. 28 (2016) 2128–2141.
[11] T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, in: NeurIPS, 2020, pp. 1877–1901.
[12] Z. Chen, Y. Fu, Y. Zhang, Y.G. Jiang, X. Xue, L. Sigal, Multi-level semantic feature augmentation for one-shot learning, IEEE Trans. Image Process. 28 (2019) 4594–4605, https://doi.org/10.1109/TIP.2019.2910052.
[13] D. Cheng, T. Liu, Y. Ning, N. Wang, B. Han, G. Niu, X. Gao, M. Sugiyama, Instance-dependent label-noise learning with manifold-regularized transition matrix estimation, in: CVPR, 2022, pp. 16609–16618.
[14] H. Chi, S. Li, W. Yang, L. Lan, A robust quadruple adaptation network in few-shot scenarios, Knowl.-Based Syst. 232 (2021) 107506, https://doi.org/10.1016/j.knosys.2021.107506.
[15] H. Chi, F. Liu, W. Yang, L. Lan, T. Liu, B. Han, W. Cheung, J. Kwok, Tohan: a one-step approach towards few-shot hypothesis adaptation, in: NeurIPS, 2021, pp. 20970–20982.
[16] W.H. Chu, Y.J. Li, J.C. Chang, Y.C.F. Wang, Spot and learn: a maximum-entropy patch sampler for few-shot image classification, in: CVPR, 2019, pp. 6244–6253.
[17] C.Y. Chuang, A. Torralba, S. Jegelka, Estimating generalization under distribution shifts via domain-invariant representations, in: ICML, 2020, pp. 1984–1994.
[18] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, Bert: pre-training of deep bidirectional transformers for language understanding, in: NAACL, 2019, pp. 4171–4186.

[19] R. Dong, F. Liu, H. Chi, T. Liu, M. Gong, G. Niu, M. Sugiyama, B. Han, Diversity-enhancing generative network for few-shot hypothesis adaptation, in: ICML, 2023, pp. 8260–8275.
[20] C. Finn, P. Abbeel, S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in: ICML, 2017, pp. 1126–1135.
[21] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V.S. Lempitsky, Domain-adversarial training of neural networks, J. Mach. Learn. Res. 17 (2016) 59.
[22] A. Ghosh, H. Kumar, P.S. Sastry, Robust loss functions under label noise for deep neural networks, in: AAAI, 2017, pp. 1919–1925.
[23] I. Goodfellow, Nips 2016 tutorial: generative adversarial networks, in: NIPS 2016 Tutorial, 2016.
[24] J. Guan, Z. Lu, T. Xiang, J.-Rong Wen, Few-shot learning as domain adaptation: algorithm and analysis, arXiv:2002.02050 [abs], 2020.
[25] X.J. Gui, W. Wang, Z.H. Tian, Towards understanding deep learning from noisy labels with small-loss criterion, in: IJCAI, 2021, pp. 2469–2475.
[26] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I.W. Tsang, M. Sugiyama, Co-teaching: robust training of deep neural networks with extremely noisy labels, in: NeurIPS, 2018, pp. 8536–8546.
[27] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: CVPR, 2016, pp. 770–778.
[28] D. Hendrycks, M. Mazeika, D. Wilson, K. Gimpel, Using trusted data to train deep networks on labels corrupted by severe noise, in: NeurIPS, 2018.
[29] J. Hull, A database for handwritten text recognition research, IEEE Trans. Pattern Anal. Mach. Intell. 16 (1994) 550–554.
[30] B.B. Jia, M.L. Zhang, Maximum margin multi-dimensional classification, IEEE Trans. Neural Netw. Learn. Syst. 33 (2022) 7185–7198.
[31] L. Jiang, Z. Zhou, T. Leung, L. Li, L. Fei-Fei, Mentornet: learning data-driven curriculum for very deep neural networks on corrupted labels, in: ICML, 2018, pp. 2309–2318.
[32] I. Khemakhem, D.P. Kingma, R.P. Monti, A. Hyvärinen, Variational autoencoders and nonlinear ICA: a unifying framework, in: AISTATS, 2020.
[33] D. Kifer, S. Ben-David, J. Gehrke, Detecting change in data streams, in: VLDB, 2004, pp. 180–191.
[34] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, CoRR, 2015.
[35] B. Kivva, G. Rajendran, P. Ravikumar, B. Aragam, Identifiability of deep generative models without auxiliary information, in: NeurIPS, 2022.
[36] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: NeurIPS, 2012.
[37] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (1998) 2278–2324.
[38] K. Lee, X. He, L. Zhang, L. Yang, Cleannet: transfer learning for scalable image classifier training with label noise, in: CVPR, 2018, pp. 5447–5456.
[39] D. Li, Y. Yang, Y. Song, T.M. Hospedales, Deeper, broader and artier domain generalization, in: ICCV, 2017, pp. 5543–5551.
[40] J. Li, R. Socher, S.C.H. Hoi, Dividemix: learning with noisy labels as semi-supervised learning, in: ICLR, 2020.
[41] J. Li, M. Zhang, K. Xu, J. Dickerson, J. Ba, How does a neural network's architecture impact its robustness to noisy labels?, in: NeurIPS, 2021.
[42] M. Li, M. Soltanolkotabi, S. Oymak, Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks, in: AISTATS, 2020, pp. 4313–4324.
[43] S. Li, X. Xia, S. Ge, T. Liu, Selective-supervised contrastive learning with noisy labels, in: CVPR, 2022, pp. 316–325.
[44] S. Li, X. Xia, S. Ge, T. Liu, Selective-supervised contrastive learning with noisy labels, in: CVPR, 2022, pp. 316–325.
[45] B.Y. Liu, L. Huang, C.D. Wang, J.H. Lai, P.S. Yu, Multiview clustering via proximity learning in latent representation space, IEEE Trans. Neural Netw. Learn. Syst. (2021) 1–14, https://doi.org/10.1109/TNNLS.2021.3104846.
[46] F. Liu, J. Lu, B. Han, G. Niu, G. Zhang, M. Sugiyama, Butterfly: a panacea for all difficulties in wildly unsupervised domain adaptation, in: NeurIPS Workshop, 2019.
[47] H. Liu, M. Shao, Z. Ding, Y. Fu, Structure-preserved unsupervised domain adaptation, IEEE Trans. Knowl. Data Eng. 31 (2019) 799–812.
[48] L. Liu, T. Zhou, G. Long, J. Jiang, C. Zhang, Many-class few-shot learning on multi-granularity class hierarchy, IEEE Trans. Knowl. Data Eng. 34 (2022) 2293–2305.
[49] T. Liu, D. Tao, Classification with noisy labels by importance reweighting, IEEE Trans. Pattern Anal. Mach. Intell. 38 (2016) 447–461, https://doi.org/10.1109/TPAMI.2015.2456899.
[50] M. Long, Z. Cao, J. Wang, M.I. Jordan, Conditional adversarial domain adaptation, in: NeurIPS, 2018, pp. 1647–1657.
[51] Y. Lyu, I.W. Tsang, Curriculum loss: robust learning and generalization against label corruption, in: ICLR, 2020.
[52] E. Malach, S. Shalev-Shwartz, Decoupling "when to update" from "how to update", in: NeurIPS, 2017, pp. 960–970.
[53] A. Mehrotra, A. Dukkipati, Generative adversarial residual pairwise networks for one shot learning, arXiv:1703.08033 [abs], 2017.
[54] A.K. Menon, A.S. Rawat, S.J. Reddi, S. Kumar, Can gradient clipping mitigate label noise?, in: ICLR, 2019.
[55] G. Mita, M. Filippone, P. Michiardi, An identifiable double VAE for disentangled representations, in: ICML, 2021.
[56] S. Motiian, Q. Jones, S.M. Iranmanesh, G. Doretto, Few-shot adversarial domain adaptation, in: NeurIPS, 2017, pp. 6670–6680.
[57] N. Natarajan, I.S. Dhillon, P.K. Ravikumar, A. Tewari, Learning with noisy labels, in: NeurIPS 26, 2013.
[58] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A. Ng, Reading digits in natural images with unsupervised feature learning, in: NeurIPS Workshop on Deep Learning and Unsupervised Feature Learning, 2011.
[59] D.T. Nguyen, C.K. Mummadi, T.P.N. Ngo, T.H.P. Nguyen, L. Beggel, T. Brox, Self: learning to filter noisy labels with self-ensembling, in: ICLR, 2020.
[60] A. Nichol, J. Achiam, J. Schulman, On first-order meta-learning algorithms, arXiv:1803.02999 [abs], 2018.
[61] C.G. Northcutt, A. Athalye, J. Mueller, Pervasive label errors in test sets destabilize machine learning benchmarks, in: NeurIPS Datasets and Benchmarks, 2021.
[62] J. Oldfield, Y. Panagakis, M.A. Nicolaou, Adversarial learning of disentangled and generalizable representations of visual attributes, IEEE Trans. Neural Netw. Learn. Syst. (2021) 1–12, https://doi.org/10.1109/TNNLS.2021.3053205.
[63] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (2010) 1345–1359.
[64] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, L. Qu, Making deep neural networks robust to label noise: a loss correction approach, in: CVPR, 2017, pp. 2233–2241.
[65] S. Ravi, H. Larochelle, Optimization as a model for few-shot learning, in: ICLR, 2017.
[66] J.C. Reinhold, A. Carass, J.L. Prince, A structural causal model for mr images of multiple sclerosis, in: MICCAI, 2021, pp. 782–792.
[67] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in: NeurIPS, 2015, pp. 91–99.
[68] B. Rooyen, R. Williamson, A theory of learning with corrupted labels, J. Mach. Learn. Res. 18 (2018) 1–50.
[69] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. (2015) 211–252.
[70] K. Saenko, B. Kulis, M. Fritz, T. Darrell, Adapting visual category models to new domains, in: ECCV, 2010, pp. 213–226.
[71] K. Saito, D. Kim, S. Sclaroff, T. Darrell, K. Saenko, Semi-supervised domain adaptation via minimax entropy, in: ICCV, 2019, pp. 8049–8057.
[72] B. Schölkopf, F. Locatello, S. Bauer, N.R. Ke, N. Kalchbrenner, A. Goyal, Y. Bengio, Toward causal representation learning, Proc. IEEE 109 (2021) 612–634.
[73] F. Schroff, A. Criminisi, A. Zisserman, Harvesting image databases from the web, in: 2007 IEEE 11th International Conference on Computer Vision, 2007, pp. 1–8.
[74] S. Shi, F. Nie, R. Wang, X. Li, Multi-view clustering via nonnegative and orthogonal graph reconstruction, IEEE Trans. Neural Netw. Learn. Syst. 34 (2023) 201–214.
[75] A. Singh, CLDA: contrastive learning for semi-supervised domain adaptation, in: NeurIPS, 2021, pp. 5089–5101.
[76] J. Snell, K. Swersky, R.S. Zemel, Prototypical networks for few-shot learning, in: NeurIPS, 2017.
[77] R. Snow, B. O'Connor, D. Jurafsky, A. Ng, Cheap and fast – but is it good? Evaluating non-expert annotations for natural language tasks, in: EMNLP, 2008, pp. 254–263.

[78] H. Song, M. Kim, J.G. Lee, SELFIE: refurbishing unclean samples for robust deep learning, in: ICML, 2019, pp. 5907–5915.
[79] H. Song, M. Kim, J.G. Lee, Selfie: refurbishing unclean samples for robust deep learning, in: ICML, 2019, pp. 5907–5915.
[80] F. Sung, Y. Yang, L. Zhang, T. Xiang, P.H.S. Torr, T.M. Hospedales, Learning to compare: relation network for few-shot learning, in: CVPR, 2018, pp. 1199–1208.
[81] H. Tan, R. Cheng, S. Huang, C. He, C. Qiu, F. Yang, P. Luo, Relativenas: relative neural architecture search via slow-fast learning, IEEE Trans. Neural Netw. Learn. Syst. 34 (2023) 475–489.
[82] R. Tanno, A. Saeedi, S. Sankaranarayanan, D.C. Alexander, N. Silberman, Learning from noisy labels by regularized estimation of annotator confusion, in: CVPR, 2019, pp. 11244–11253.
[83] T. Teshima, I. Sato, M. Sugiyama, Few-shot domain adaptation by causal mechanism transfer, in: ICML, 2020, pp. 9458–9469.
[84] P. Tian, W. Li, Y. Gao, Consistent meta-regularization for better meta-knowledge in few-shot learning, IEEE Trans. Neural Netw. Learn. Syst. 33 (2022) 7277–7288.
[85] H.Y. Tseng, H.Y. Lee, J.B. Huang, M.H. Yang, Cross-domain few-shot classification via learned feature-wise transformation, in: ICLR, 2020.
[86] A. Veit, N. Alldrin, G. Chechik, I. Krasin, A. Gupta, S. Belongie, Learning from noisy large-scale datasets with minimal supervision, in: CVPR, 2017, pp. 6575–6583.
[87] H. Wang, Z. Deng, Cross-domain few-shot classification via adversarial task augmentation, in: IJCAI, 2021, pp. 1075–1081.
[88] J. Wang, Y. Liu, C. Levy, Fair classification with group-dependent label noise, in: ACM FAccT, 2021, pp. 526–536.
[89] Q. Wang, D. Zhou, Y. Zhang, D. Zhan, H. Ye, Few-shot class-incremental learning via training-free prototype calibration, in: NeurIPS, 2023.
[90] W. Wang, H. Li, Z. Ding, F. Nie, J. Chen, X. Dong, Z. Wang, Rethinking maximum mean discrepancy for visual domain adaptation, IEEE Trans. Neural Netw. Learn. Syst. 34 (2023) 264–277.
[91] Y. Wang, Q. Yao, J.T. Kwok, L.M. Ni, Generalizing from a few examples: a survey on few-shot learning, ACM Comput. Surv. 53 (2020), https://doi.org/10.1145/3386252.
[92] Z. Wang, Y. Ni, B. Jing, D. Wang, H. Zhang, E. Xing, Dnb: a joint learning framework for deep Bayesian nonparametric clustering, IEEE Trans. Neural Netw. Learn. Syst. 33 (2022) 7610–7620.
[93] J. Wei, H. Liu, T. Liu, G. Niu, S. Masashi, Y. Liu, To smooth or not? When label smoothing meets noisy labels, in: ICML, 2022.
[94] M. Woodward, C. Finn, Active one-shot learning, in: NeurIPS Workshop, 2017.
[95] H. Wu, Y. Yan, G. Lin, M. Yang, M.K. Ng, Q. Wu, Iterative refinement for multi-source visual domain adaptation, IEEE Trans. Knowl. Data Eng. 34 (2022) 2810–2823.
[96] S. Wu, M. Gong, B. Han, Y. Liu, T. Liu, Fair classification with instance-dependent label noise, in: CLeaR, 2021, pp. 927–943.
[97] X. Xia, T. Liu, B. Han, C. Gong, N. Wang, Z. Ge, Y. Chang, Robust early-learning: hindering the memorization of noisy labels, in: ICLR, 2021.
[98] X. Xia, T. Liu, B. Han, M. Gong, J. Yu, G. Niu, M. Sugiyama, Sample selection with uncertainty of losses for learning with noisy labels, in: ICLR, 2022.
[99] B. Xu, Z. Zeng, C. Lian, Z. Ding, Few-shot domain adaptation via mixup optimal transport, IEEE Trans. Image Process. 31 (2022) 2518–2528, https://doi.org/10.1109/TIP.2022.3157139.
[100] Y. Yan, Z. Xu, I. Tsang, G. Long, Y. Yang, Robust semi-supervised learning through label aggregation, in: AAAI, 2016, pp. 2244–2250.
[101] S. Yang, E. Yang, B. Han, Y. Liu, M. Xu, G. Niu, T. Liu, Estimating instance-dependent Bayes-label transition matrix using a deep neural network, in: ICML, 2022, pp. 25302–25312.
[102] Y. Yao, T. Liu, M. Gong, B. Han, G. Niu, K. Zhang, Instance-dependent label-noise learning under a structural causal model, in: NeurIPS, 2021.
[103] Y. Yao, T. Liu, B. Han, M. Gong, J. Deng, G. Niu, M. Sugiyama, Dual t: reducing estimation error for transition matrix in label-noise learning, in: NeurIPS 33, 2020.
[104] H. Ye, X. Sheng, D. Zhan, Few-shot learning with adaptively initialized task optimizer: a practical meta-learning approach, Mach. Learn. 109 (2020) 643–664.
[105] K. You, M. Long, Z. Cao, J. Wang, M.I. Jordan, Universal domain adaptation, in: CVPR, 2019, pp. 2720–2729.
[106] C. Zhang, Y. Cai, G. Lin, C. Shen, Deepemd: few-shot image classification with differentiable Earth mover's distance and structured classifiers, in: CVPR, 2020, pp. 12200–12210.
[107] H. Zhang, J. Zhang, P. Koniusz, Few-shot learning via saliency-guided hallucination of samples, in: CVPR, 2019, pp. 2765–2774.
[108] J. Zhang, V.S. Sheng, T. Li, X. Wu, Improving crowdsourced label quality using noise correction, IEEE Trans. Neural Netw. Learn. Syst. 29 (2018) 1675–1688, https://doi.org/10.1109/TNNLS.2017.2677468.
[109] M. Zhang, J. Lee, S. Agarwal, Learning from noisy labels with no change to the training process, in: ICML, 2021, pp. 12468–12478.
[110] Y. Zhang, F. Liu, Z. Fang, B. Yuan, G. Zhang, J. Lu, Learning from a complementary-label source domain: theory and algorithms, IEEE Trans. Neural Netw. Learn. Syst. 33 (2022) 7667–7681.
[111] A. Zhao, M. Ding, Z. Lu, T. Xiang, Y. Niu, J. Guan, J.R. Wen, Domain-adaptive few-shot learning, in: WACV, 2021, pp. 1389–1398.
[112] S.C. Zhu, Statistical and computational theories for image segmentation, texture modeling and object recognition, Ph.D. thesis, Harvard University, 1996.
[113] I.M. Ziko, J. Dolz, E. Granger, I.B. Ayed, Laplacian regularized few-shot learning, in: ICML, 2020, pp. 11660–11670.