Beyond UDA: Examining Temporal and Frequency Representations in Time Series Transfer

Ching-Chieh Tsao * CHINGCHIEH@IIR.CSIE.NCKU.EDU.TW
Fang-Yi Su * FANGYI@IIR.CSIE.NCKU.EDU.TW
Jung-Hsien Chiang JCHIANG@MAIL.NCKU.EDU.TW
National Cheng Kung University
**** Co-first authors.***

Editors: Hung-yi Lee and Tongliang Liu

Abstract

In time-series unsupervised domain adaptation (UDA), the adaptation between temporal and frequency domain features has been relatively underexplored. To address this gap, we conduct a comprehensive series of experiments to revisit the roles of these domains in UDA. Our findings reveal that the temporal domain contains more diverse features, offering higher discriminability, while the frequency domain is more domain-invariant, providing better transferability. Combining the strengths of both domains, we propose TF-DAN, a UDA framework that synergistically integrates temporal and frequency domain features. TF-DAN enhances feature extraction and captures subtle, class-specific features without relying on traditional alignment strategies. By utilizing simple hyperparameter adjustments and using frequency embeddings from the source domain as reference points for domain adaptation, TF-DAN achieves nearly a 10% improvement across five benchmark datasets in time-series UDA. This research highlights the unique strengths of both domains and marks a paradigm shift in UDA methods, showcasing TF-DAN's robust performance in real-world applications. Codes can be found in the additional material folder.

Keywords: Time-Series; Unsupervised Domain Adaptation; TSUDA.

1. Introduction

Time series datasets showcase the prowess of neural networks (Ravuri et al., 2021; Lundberg et al., 2018), but their vulnerability to domain shifts poses deployment challenges (Singhal et al., 2023; Painblanc et al., 2023; Zhang et al., 2021). These shifts, stemming from nuanced differences in test distributions, hinder model generalization (Koh et al., 2021; Luo et al., 2018; Zhang et al., 2013). This challenge manifests across diverse real-world applications, from machine fault detection and EEG classification to human activity recognition through wearable devices, where variations in machines, environments, and individual characteristics inevitably lead to significant domain shifts. Addressing this, domain adaptation (DA) techniques, such as leveraging unlabeled data (Garg et al., 2021; Ganin et al., 2016), emerge as essential to ensure robust model performance in real-world scenarios. In addition, DA for time series is even more difficult (Wilson and Cook, 2020; Ozyurt et al., 2023; He et al., 2023), as it has to deal with both the domain discrepancy and the temporal dynamics that may cause feature shift and label shift.

Unsupervised Domain Adaptation (UDA) is pivotal for enhancing the generalization of machine learning models, aiming to train a model on a labeled source domain that can

effectively perform on a related yet unlabeled target domain (Garg et al., 2021; Ganin et al., 2016). While UDA methods have flourished in computer vision (Huo et al., 2022; Tang et al., 2021; Pan et al., 2020; Tzeng et al., 2019), their application to time series, though feasible with feature extractor adjustments, often falls short in fully harnessing time-series properties. In the domain of time series, a limited number of works have explicitly addressed UDA, they mostly focus on temporal information. Even when the frequency domain is considered, it is typically combined with temporal features and treated as general information during training.

To clarify the characteristics of the time and frequency domains, we conducted a series of experiments leading to the following conclusions: **the temporal domain provides broader information with stronger classification discriminability, while the frequency domain, though simpler, offers more domain-invariant features that serve as reference points between the source and target domains** (Section 3.1).

Our research integrates the strengths of both the temporal and frequency domains, moving beyond the prior focus on "how to align two inconsistent distributions" to explore ***how to identify features that represent classes across domains?*** The difference lies in that the former approach pays little attention to the features extracted by the model, focusing instead on alignment methods and classifier performance. This overemphasis on alignment leads to overly sensitive and inflexible classifiers, particularly when dealing with data with large domain gaps or longer time series. The latter approach avoids these pitfalls by enabling the model to utilize class-representative features early in training, ensuring more robust performance.

We propose TF-DAN (Temporal-Frequency Domain Adaptation Network), a simple framework for UDA in time series that leverages both temporal and frequency domain characteristics to achieve strong performance. Our model integrates information from both domains to capture subtle, class-specific features, enhancing feature extraction. By focusing on the domain-invariant properties of the frequency domain, we use a frequency embedding table from the source domain as reference points, along with simple hyperparameter adjustments, to enable the model to find the most suitable embeddings for target domain data during adaptation, ultimately assigning the appropriate class labels.

Our Contributions:

1. We introduce **TF-DAN (Temporal–Frequency Domain Adaptation Network)**, a novel framework for time-series UDA that explicitly separates and leverages the complementary roles of temporal (class-discriminative) and frequency (domain-invariant) representations.

2. We empirically demonstrate that **temporal features enhance discriminability**, while **frequency features improve transferability**, revealing their distinct contributions to domain adaptation.

3. A lightweight frequency embedding module enables efficient adaptation to unlabeled target domains, achieving up to **10% performance gain** across five benchmark datasets.

2. Related Work

2.1. Unsupervised Domain Adaptation

Unsupervised domain adaptation (UDA) leverages labeled source domain data to predict labels in an unlabeled target domain by minimizing domain discrepancy and reducing target error. UDA methods fall into three categories: (1) Metric-based methods, such as DDC (Tzeng et al., 2019), Deep CORAL (Sun and Saenko, 2016), DeepJDOT (Damodaran et al., 2018), HoMM (Chen et al., 2020), and MMDA (Rahman et al., 2020), which minimize domain gaps using distance metrics. (2) Adversarial-based methods use domain discriminators to learn domain-invariant features, like DANN (Ganin et al., 2016), CDAN (Long et al., 2018), and DIRT-T (Shu et al., 2018). (3) Contrastive methods, such as CAN (Kang et al., 2019), CLDA (Singh, 2021), and IDCo (Zhang et al., 2023), align source and target embeddings via contrastive loss, using pseudo-labels for target samples. While UDA is well-studied in computer vision, research on UDA for time-series data remains limited.

2.2. Time-Series Unsupervised Domain Adaptation

Despite successes in computer vision, there has been a notable gap in research focusing on adaptation methods tailored for time-series data. Few methods have been specifically crafted for time-series domain adaptation. (1) Adversarial training for time-series UDA involves using adversarial methods to learn domain-invariant temporal relationships, such as VRADA (Purushotham et al., 2017) and CoDATS (Wilson et al., 2020). (2) Statistical divergence methods for time-series UDA focus on aligning the statistical properties of source and target domains. Examples include SASA (Cai et al., 2021), AdvSKM (Liu and Xue, 2021a) and (Ott et al., 2022). (3) Self-supervision methods for time-series UDA extract domain-invariant and domain-specific features. DAF (Jin et al., 2022) uses a shared attention module with a reconstruction task. Contrastive methods like (Ozyurt et al., 2023), CoTMix (Eldele et al., 2023), and CALDA (Wilson et al., 2023) use augmentations to enhance prediction. RAINCOAT (He et al., 2023) addresses feature and label shifts by aligning them across domains. Despite their potential, they rely on access to source data, which may not always be feasible due to privacy concerns.

2.3. Vector Quantised Variational AutoEncoder (VQ-VAE)

The VQ-VAE (Van Den Oord et al., 2017) functions as a communication system where an encoder maps inputs to vectors that are quantized using a shared codebook, and a decoder reconstructs the data from these discrete representations. This discretization process, coupled with straight-through gradient estimation for training, proves particularly advantageous in UDA scenarios by effectively capturing domain-specific features through self-supervised learning while maintaining robustness against model degeneration in both temporal and frequency domains.

3. Problem Formulations

We are given two distributions of time-series data: one from the source domain D_s and the other from the target domain D_t. In this setup, define labeled *i.i.d.* samples from the

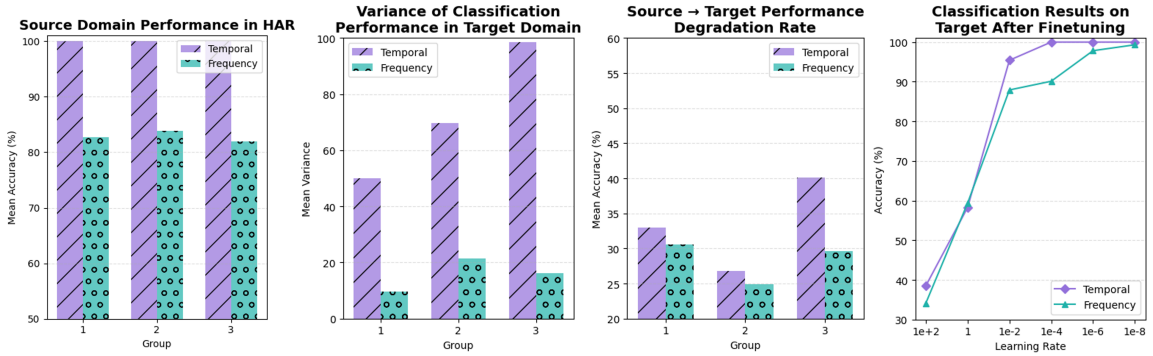


Figure 1: We randomly selected 15 source ↦ target pairs from the HAR dataset and divided them into three groups for analysis, focusing on the following metrics: (a) mean accuracy in the source domain, (b) mean variance in classification performance within the target domain, and (c) the performance degradation rate when testing the source domain pre-trained model on the target domain. Additionally, we evaluated the impact of hyperparameters on transferability in both the time and frequency domains by assessing (d) the mean accuracy after fine-tuning with different learning rates. Additional experimental results on other datasets can be found in Supplementary A.

source domain as $S = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}_{i=1}^{N_s} \sim D_s$, where $\mathbf{x}_i^s$ represents a sample from the source domain, $\mathbf{y}_i^s \in \{1, ..., H\}$, where H is the number of classes, and $\mathbf{y}_i^s$ denotes the label for the corresponding sample, and N_s denotes the total number of *i.i.d.* samples in the source domain. Conversely, consider **unlabeled** *i.i.d.* samples from the target domain denoted by $T = \{\mathbf{x}_i^t\}_{i=1}^{N_t} \sim D_t$. Here, $\mathbf{x}_i^t$ denotes an individual sample from the target domain, and N_t represents the total number of *i.i.d.* samples collected from the target domain. Furthermore, each $\mathbf{x}_i$, whether originating from D_s or D_t, constitutes a sample of a multivariate time series denoted by $\mathbf{x}_i = \{\mathbf{x}_{i,t}\}_{t=1}^{L} \in \mathbb{R}^{M \times L}$, where L represents the number of time steps, and $\mathbf{x}_{i,t} \in \mathbb{R}^M$ signifies M observations for the respective time step.

Problem (UDA for Time Series). *Given are a labeled source dataset D_s and an unlabeled target dataset D_t, the time step is L. The goal is to extract the highly generalizable representations from both D_s and D_t by utilizing unsupervised learning strategies and make the model adapt well to the target samples.*

3.1. Preliminary Study

We designed a series of experiments on both the temporal and frequency domains. To minimize model influence, we follow prior research (Liu and Xue, 2021b; Cheng et al., 2024) by constructing a 3-layer CNN as a temporal feature extractor and a frequency feature

extractor that combines a fast Fourier transform with a 1-layer linear network. Both are followed by a 1-layer linear classifier for simplicity.

The key question we explore is: ***What kind of feature information do the temporal and frequency domains provide?*** We pre-train three models on the source domain until they converge and observe their performance on the target domain. During the temporal model experiments, we observed a noteworthy phenomenon: despite achieving nearly 100% accuracy in the source domain (Fig. 1(a)) with different model parameters, the performance on the target domain exhibits considerable fluctuation. As shown in Fig. 1(b), the performance variance of the three temporal models is larger than that of the frequency models.

Transferability. When we examine transferability, Fig. 1(c) shows that the temporal models experience a more significant performance drop, with a statistically significant difference from the frequency models (p-value = 0.0491). We hypothesize that this is because the temporal domain contains a wider variety of information, enabling the model to classify based on multiple dimensions. Nevertheless, this diverse information also includes more features specific to the source domain or confounders, meaning that when domain shifting occurs, the model's focus may no longer be on the relevant class features of the target domain, resulting in poorer transferability.

In contrast, the frequency domain, after undergoing Fourier transformation, filters out much of the extraneous information, such as signal start and end points or noise, resulting in fewer feature dimensions. However, this allows the frequency models to focus more on the overall structure of the information, making them more domain-invariant. Fig. 1(c) supports this, showing that although the frequency models do not perform as well as the temporal models in source domain classification, their transferability is superior.

This raises another concern: ***Is the frequency domain truly more domain-invariant?*** To investigate, we design another experiment where we only adjust the extent of feature updates (here, we choose to adjust the learning rate) during the fine-tuning phase. Our assumption is that if merely tweaking the learning rate significantly improves model performance, it indicates that the frequency domain contains domain-agnostic features that are specific to each class of data rather than just irrelevant features that do not contribute to the model's effectiveness.

As shown in Fig. 1(d), the frequency models require a very small learning rate to fine-tune correctly. Larger learning rates prevent the frequency models from converging to the optimal point. Interestingly, the temporal models are much less sensitive to hyperparameter adjustments compared to the frequency models. In Fig. 1(d), despite averaging accuracy across 15 source $\mapsto$ target experiments, the temporal models fine-tune to 100% accuracy across learning rates ranging from 1×10^{-4} to 1×10^{-8}. This could be explained by the high feature diversity in the temporal domain, allowing different model parameters to reach optimal solutions depending on the learning rate. Meanwhile, the frequency models retain robust domain-invariant features between source and target domains, making them better suited to fine-tuning with smaller steps.

Empirical insights. The analysis reveals two key insights regarding time-series domain adaptation: (1) the time domain excels at classification, but its transferability is hindered by an excess of confounding factors, and (2) the frequency domain, though containing more uniform and less diverse information, offers better domain-invariant features, leading to

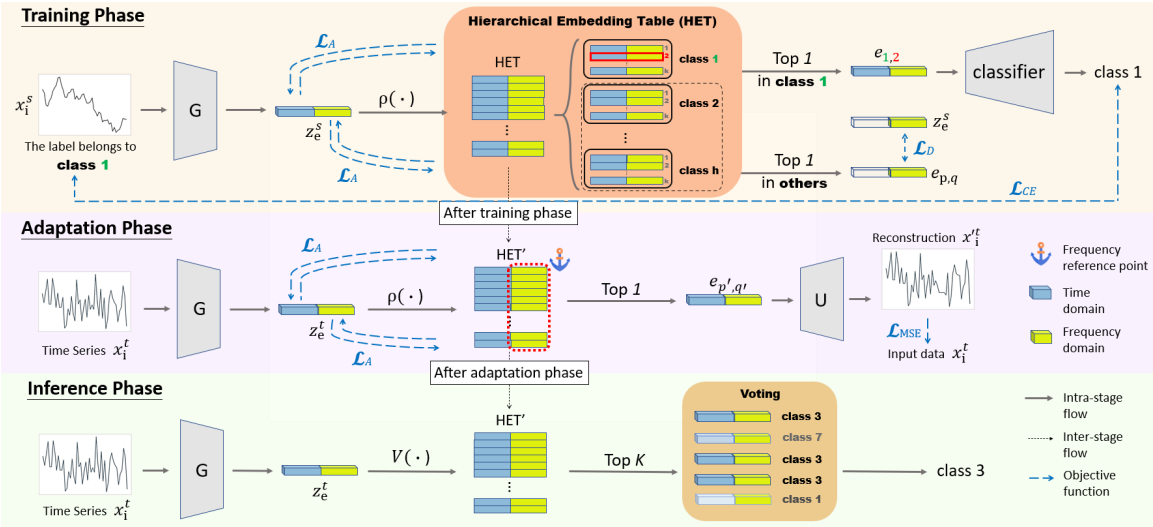


Figure 2: The TF-DAN framework operates in three phases: (a) During training, input $\mathbf{x}_i^s$ is processed by the Dual-Stream Encoder G to produce a combined temporal-frequency feature $\mathbf{z}_e^s$, with embeddings retrieved from HET using labels and classified via the nearest embedding function ρ. (b) In adaptation, HET embeddings are refined using frequency references and a decoder U for reconstruction to address domain shifts. (c) At inference, a voting mechanism ranks embedding similarities with $\mathbf{z}_e^t$ to improve classification.

stronger transferability. Based on these observations, we design a simple model framework that leverages the rich features of the time domain while using the frequency domain as a reference point to bridge the source and target domains. Our experimental results demonstrate that combining the strengths of both domains yields improved performance.

4. Our Approach

TF-DAN consists of three key modules: a dual-stream encoder G, a hierarchical embedding table (HET), and a decoder U, along with a 1-layer linear classifier for training. The encoder G extracts temporal and frequential features (Sec. 4.1). HET is initialized and operates differently across phases (Sec. 4.2). A voting mechanism refines predictions after the nearest-neighbor algorithm in inference (Sec. 4.3). Following VQ-VAE, selected embeddings serve as inputs to U. The objective functions guiding training and adaptation are detailed in Sec. 4.4.

4.1. Dual-stream encoder G

G encodes both time and frequency representations, and the source temporal and frequential features are denoted as $\mathbf{z}_{temp,i}^s$ and $\mathbf{z}_{freq,i}^s$, while the target features are denoted as $\mathbf{z}_{temp,i}^t$ and $\mathbf{z}_{freq,i}^t$. We will employ the simplified terms $\mathbf{z}_{temp}$ and $\mathbf{z}_{freq}$ to collectively represent features

from both D_s and D_t in the subsequent explanations. By including frequency information, the encoder enhances its ability to adapt across domains by potentially identifying common features. The encoder parameterizes a posterior distribution $q(\mathbf{z}|\mathbf{x})$ over the latent variables $\mathbf{z}_{temp}$ and $\mathbf{z}_{freq}$ based on the input. This posterior captures relationships between the input and latent representations, informed by both temporal and frequency patterns extracted from the input, the following:

$$G(\mathbf{x}) = \text{Concat}[\mathbf{z}_{temp}(\mathbf{x}), \mathbf{z}_{freq}(\mathbf{x})], \quad \forall \mathbf{x} \in D, \tag{1}$$

where D is either D_s or D_t, and Concat is the abbreviation of concatenation.

4.2. Hierarchical embedding table (HET)

Initialization. We introduce a 2-layer top-down embedding table and the initial layer is organized based on task labels, consisting of H categories. The subsequent layer of the hierarchical embedding table comprises independent latent embedding spaces for each $\mathbf{e}_h$, denoted as $\mathbf{e}_h \in R^{K \times \Psi}$, where K represents the number of the discrete latent variables of each category and Ψ is the dimensionality of each embedding vector. To sum up, there are $H \times K$ embeddings in the hierarchical embedding table and we initialize the embeddings using uniform distribution.

Training phase. We perform a nearest neighbor search in the whole embedding space, focusing on the category in the source domain that corresponds to the input $\mathbf{x}$ as outlined in Eq. 2. The probabilities of the posterior categorical distribution $q(G(\mathbf{x})|\mathbf{x})$ are defined as one-hot encoded, following:

$$q(G(\mathbf{x}) = k|\mathbf{x}) = \begin{cases} 1 & \text{for } k = \arg\min_j \|G(\mathbf{x}) - \mathbf{e}_{h,j}\|_2, \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

where h denoted to the same category as $\mathbf{x}$ and j is the candidates of the category h.

Adaptation phase. Due to the lack of labels in D_t, the model cannot search for the most similar embeddings within the respective categories. Therefore, we take advantage of the distinctive characteristics of the frequency domain and partially freeze the frequency blocks of HET. This deliberate constraint, achieved through significantly different learning rates, establishes a clear reference point for the encoded latent representations. Consequently, both the time and frequency modules can efficiently navigate the gradient map, leading to the identification of optimal solutions with appropriately adjusted update steps. Accordingly, we can modify Eq. 2 to be agnostic to the category h:

$$q(G(\mathbf{x}) = k|\mathbf{x}) = \begin{cases} 1 & \text{for } k = \arg\min_j \|G(\mathbf{x}) - \mathbf{e}_j\|_2, \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

where j is the embeddings of HET and there is no category h in this equation.

4.3. Voting mechanism

After the training and adaptation phases, the embeddings in HET have formed H distinctive clusters. This implies that, while the embedding in HET is discrete, the majority possess

representative features specific to their respective categories h. Subsequently, we employ a nearest-neighbor algorithm to determine the top K categories (where K=5) represented by the embeddings. Through a voting mechanism, we ascertain the category to which the input data should belong. This enhances the robustness of TF-DAN. The algorithm of the voting mechanism can be seen in Supplementary B.

4.4. Objective functions

In TF-DAN, we utilize three types of objective functions during the training phase: (1) classification loss, (2) dissimilarity loss, and (3) feature-embedding consistency loss. While there are two types of objective functions during the adaptation phase: (1) reconstruction loss and (2) feature-embedding consistency loss.

Classification loss $\mathcal{L}_{\text{CE}}$. We utilize cross-entropy loss as the loss function for our classification task during training.

Dissimilarity loss $\mathcal{L}_{\text{D}}$. This objective function is designed to prevent the model from generating nearly identical embeddings among categories during the training phase. To achieve this, we identify the closest embedding to $\mathbf{z}_{freq}$ from all embeddings in the frequency block, which is more domain-agnostic than temporal features and does not belong to the same category as $\mathbf{y}_i^s$. The repulsive effect is introduced by calculating the dissimilarity loss. It is worth noting that, while TF-DAN searches for the closest representative in the embedding table within the same category as $\mathbf{x}_i^s$, this approach may result in the model learning a common feature across all categories, neglecting latent features that distinguish between different categories. To address this, we utilize the following equation (Eq. 4) to guide the model explicitly in generating a better latent representation.

$$\mathcal{L}_{\text{D}} = 1 - \|\text{sg}[\mathbf{e}_{freq_{[h\neq\mathbf{y}]}}] - \mathbf{z}_{freq}\|_2^2, \tag{4}$$

where $\mathbf{e}_{freq_{[h\neq\mathbf{y}]}}$ is the chosen embedding from the frequency block on the hierarchical embedding table, and its category h cannot be the same label of the input data $\mathbf{x}_i^s$. Additionally, $\text{sg}(\cdot)$ represents the stop-gradient operator, which functions as an identity during forward computation and possesses zero partial derivatives.

Feature-embedding consistency loss $\mathcal{L}_{\text{A}}$. Taking inspiration from VQ-VAE, TF-DAN incorporates vector quantization algorithms, guiding the embedding encoder outputs towards proximity through L2 error, thus effectively learning the embedding space. The hierarchical structure of the embedding table, divided into temporal and frequency blocks, assigns each block to handle specific features. Consequently, they do not share the same optimizer but are updated independently. Additionally, to address a concern highlighted by VQ-VAE about the lack of dimensionality constraints on the embedding space, which could potentially lead to uncontrolled growth, TF-DAN adjusts the weight of this constraint to α and β for both temporal and frequency blocks. The objective function is expressed as:

$$\begin{aligned}\mathcal{L}_{\text{A}} = \alpha \overbrace{\left(\|\text{sg}[\mathbf{e}_{freq}] - \mathbf{z}_{freq}\|_2^2\right) + \|\mathbf{e}_{freq} - \text{sg}[\mathbf{z}_{freq}]\|_2^2}^{\text{Frequency block}} \\ + \beta \underbrace{\left(\|\text{sg}[\mathbf{e}_{temp}] - \mathbf{z}_{temp}\|_2^2\right) + \|\mathbf{e}_{temp} - \text{sg}[\mathbf{z}_{temp}]\|_2^2}_{\text{Temporal block}}.\end{aligned} \tag{5}$$

Reconstruction loss $\mathcal{L}_{\text{MSE}}$. During the adaptation phase, since the representative chosen from the hierarchical embedding table does not provide the model with a real gradient, we employ the straight-through estimator (Van Den Oord et al., 2017). This allows us to pass the gradient generated by the decoder back to the encoder directly. We opt not to use the subgradient through the quantization operation, as VQ-VAE has demonstrated that a simple estimator can achieve effective training outcomes. As the output representation of the encoder and the input to the decoder exist in the same D-dimensional space, the gradients carry valuable information on how the encoder needs to adjust its output to minimize the reconstruction loss.

Overview of TF-DAN. During training, we employ the classification loss for our classification task. The total loss function is defined with three components in the objective function, as outlined below:

$$\mathcal{L}_{\text{training}} = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{A}} + \mathcal{L}_{\text{D}}. \tag{6}$$

During adaptation, we replace the classification task with a reconstruction task, which leads us to modify our objective function as shown in Eq. 7. This design enables TF-DAN to outperform other time-series UDA methods. Last but not least, an overview algorithm of TF-DAN is in Supplementary B.

$$\mathcal{L}_{\text{adaptation}} = \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{A}}. \tag{7}$$

5. Experiments

5.1. Experimental setup

Datasets. We employ a comprehensive evaluation strategy, consisting of two main aspects. First, extensive experiments are conducted using five well-established benchmark datasets in UDA tasks, from three distinct problem types: (1) Human Activity Recognition: HAR (Anguita et al., 2013), HHAR (Stisen et al., 2015), WISDM (Kwapisz et al., 2011); (2) Sleep Stage Classification: Sleep-EDF (Goldberger et al., 2000); (3) Machine Fault Diagnosis: MFD (Lessmeier et al., 2016). In human activity recognition datasets, we treat sensor measurements from each participant as distinct domains. To ensure robust assessment, we randomly select 10 source-target domain pairs for evaluation, a methodology widely adopted in previous works on UDA in time-series research (He et al., 2023; Ozyurt et al., 2023; Cai et al., 2021; Wilson et al., 2020). For the sleep stage classification task, following the approach of (Ragab et al., 2023), we utilize the Sleep-EDF dataset, comprising EEG readings from 20 healthy subjects, and we specifically choose EEG in alignment with previous studies (Eldele et al., 2021). The machine fault diagnosis dataset has been collected under four different operating conditions, and we treat them as separate domains. In contrast to datasets used for human activity recognition being multi-variate, the data used in Sleep-EDF and MFD consist of a single univariate channel following previous works. (He et al., 2023; Ragab et al., 2023) Further details on datasets are given in Supplementary C.

Baselines. We evaluate nine domain adaptation methods, including general UDA approaches: deep correlation alignment (Deep Coral) (Sun and Saenko, 2016), decision boundary iterative refinement training with a teacher (DIRT-T) (Shu et al., 2018), HoMM

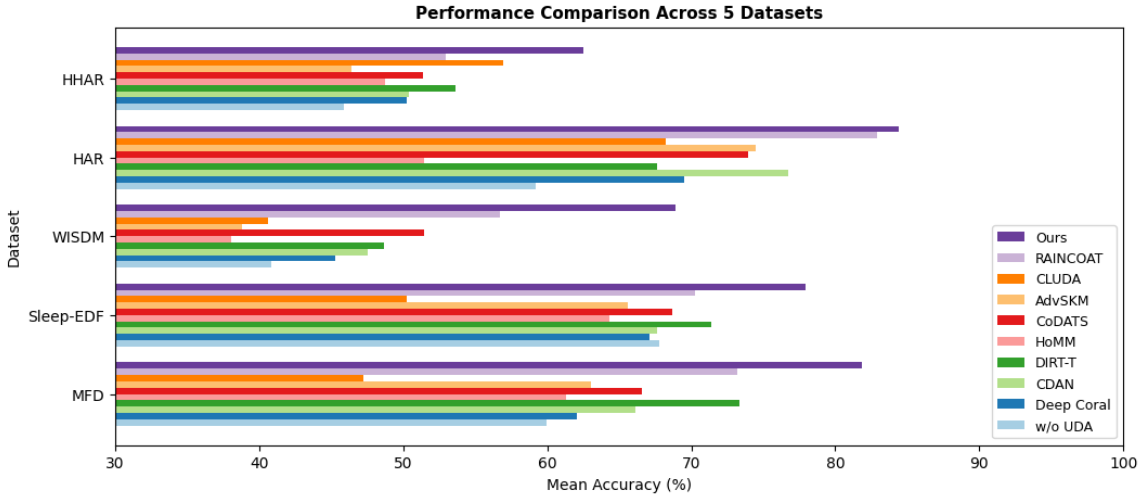


Figure 3: Average performance of multiple DA methods across 5 real-world time-series datasets. TF-DAN consistently outperforms all other methods in accuracy on test sets drawn from the target domain dataset.

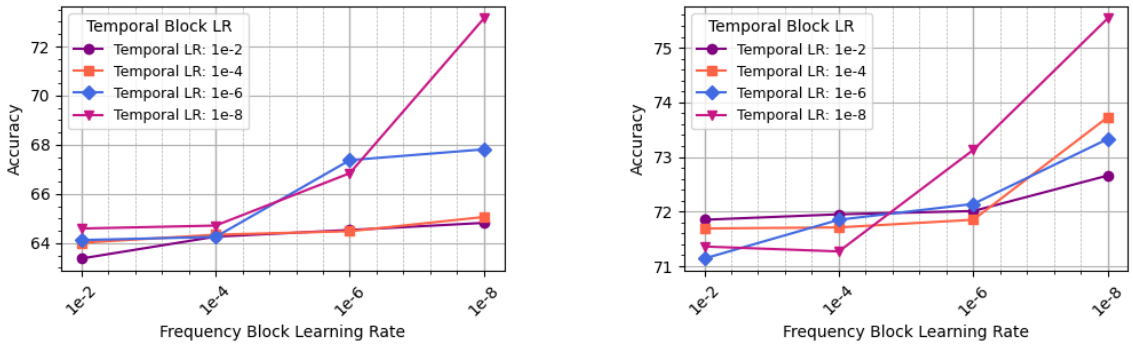
(Chen et al., 2020), and CDAN (Long et al., 2018). Additionally, we include four UDA methods specifically designed for time series: CoDATS (Wilson et al., 2020), adversarial frequency kernel matching for unsupervised time-series domain adaptation (AdvSKM) (Liu and Xue, 2021a), contrastive learning for unsupervised domain adaptation of time series (CLUDA) (Ozyurt et al., 2023), and RAINCOAT (He et al., 2023). As a baseline, we also consider source-domain-only training (no transfer) using the time-frequency encoder as RAINCOAT (He et al., 2023) and a 1-layer classifier.

Evaluation. We present accuracy and macro-F1 scores computed based on the target test datasets. In the experiment, we assign the values of 1 to both parameters α and β, treating the time domain and frequency blocks as equally important. More hyperparameter settings can be seen in Supplementary E.

5.2. Results

5.2.1. Classification performance on DA benchmark datasets

In Fig. 3, the average accuracy of each method is presented across 10 sources $\mapsto$ target domain pairs on the HAR, HHAR, WISDM, Sleep-EDF, and MFD datasets. On the HAR dataset, our model surpasses the best baseline accuracy achieved by RAINCOAT by 1.93% (0.844 vs. 0.828). For the HHAR dataset, our model outperforms the best baseline accuracy of CLUDA by 5.5% (0.624 vs. 0.569). In the case of the WISDM dataset, our model excels by surpassing the best baseline accuracy of RAINCOAT by 21.34% (0.688 vs. 0.567). Moving on to the Sleep-EDF dataset, our model exceeds the best baseline accuracy of DIRT-T by 9.1% (0.779 vs. 0.714). Similarly, on the MFD dataset, our model beats the best baseline accuracy of DIRT-T by 11.73% (0.819 vs. 0.733). Despite our model's simplicity compared



(*a*) MFD (*b*) Sleep-EDF

Figure 4: Accuracy for Frequency and Temporal Block Learning Rates in (a) and (b) datasets.

to state-of-the-art methods, it achieves the highest scores across five different datasets. The Supplementary D contains a detailed compilation of UDA results for each source $\mapsto$ target pair, accompanied by Macro-F1 scores, which further support our conclusions.

5.2.2. Different Frequency and Temporal Block Learning Rates

We further analyze the impact of different learning rates for the temporal and frequency blocks of TF-DAN during the adaptation phase. We conduct experiments using the MFD and Sleep-EDF datasets due to their large data volumes, which make performance differences more pronounced, as shown in Fig. 4. We discover some valuable findings:

1. When the learning rate for the frequency block is smaller, TF-DAN's adaptability improves. This trend aligns with the observations of our insights in Section 3.1.

2. When the learning rate for the temporal block is larger, the model's performance deteriorates. We speculate that this is due to the interaction between the encoder and the HET within TF-DAN architecture. Specifically, when the learning rates of the temporal and frequency blocks differ by four orders of magnitude, it indirectly hinders the adjustment range of one of the blocks through the encoder.

Therefore, we recommend setting the learning rates of the temporal and frequency blocks to the same value during the adaptation phase for optimal performance.

5.2.3. Embeddings in HET After Training Phase

To further understand why TF-DAN excels in UDA tasks, we use principle component analysis (PCA) to visualize the 2D embeddings of the temporal and frequency blocks. Fig. 16 in Supplementary F shows that, although both blocks are initialized uniformly, the temporal block's embeddings do not cluster as well as those of the frequency block. This is likely due to the time domain's higher feature diversity and complexity, including confounders, while the frequency block contains more uniform information, allowing it to better capture category-specific features, as seen in the PCA visualization (Fig. 16(b) in Supplementary F).

Table 1: The ablation study of TF-DAN, where performance is measured in terms of accuracy.

	Element of Our Model			MFD Dataset			
	Frequency Block	$\mathcal{L}_D$	Voting	$1 \mapsto 3$	$2 \mapsto 1$	$3 \mapsto 2$	Avg
(A)	✓			83.94	80.23	77.81	80.66
(B)			✓	58.36	65.45	69.10	64.30
(C)	✓	✓		<u>87.25</u>	<u>86.08</u>	<u>84.19</u>	<u>85.84</u>
(D)	✓		✓	83.81	85.77	82.59	84.06
(E)	✓	✓	✓	**99.84**	**91.71**	**87.22**	**92.92**

5.2.4. Ablation Study

To better understand the impact of different components in TF-DAN, we conducted ablation experiments on three key elements: the frequency block, dissimilarity loss $\mathcal{L}_{\text{D}}$, and the voting mechanism, employing five different configurations (Table 1). Given that TF-DAN relies on the frequency block as a reference point, experiments without the frequency block (Table 1 row (B)) exclusively utilized the temporal block for adaptation. Notably, experimental setups without the frequency block and with $\mathcal{L}_{\text{D}}$ were not feasible, considering that $\mathcal{L}_{\text{D}}$ is computed based on the frequency embedding table.

During the inference phase, TF-DAN utilizes a voting technique. In the ablation experiment settings, we adjust the 'without voting' configuration to directly select the category of the most similar embedding as the final prediction.

The results reveal that the absence of both the frequency block and $\mathcal{L}_{\text{D}}$ (Table 1 row (B)) leads to the poorest performance. Conversely, having only the frequency block (Table 1 row (A)) significantly improves classification accuracy. This underscores the argument presented in our preliminary study that the frequency domain's domain-invariant properties between source and target domains enable TF-DAN to generate distinct feature distributions for each category during training. The use of the well-learned frequency embedding table as a robust reference guides the classification of target domain data into the correct categories. Furthermore, incorporating $\mathcal{L}_{\text{D}}$ or adopting the voting technique enhances performance. The most optimal performance is achieved when all three components are used simultaneously, surpassing the second-place configuration (Table 1 row (C)) by nearly 8% in average performance.

6. Conclusion

This research uncovers the complementary roles of temporal and frequency domains in time-series UDA, demonstrating that temporal features provide broader discrimination while frequency components yield domain-agnostic transferability. Our proposed TF-DAN framework leverages these insights, achieving nearly 10% performance gains across five benchmarks through effective frequency embedding integration and simple hyperparameter tuning, without relying on traditional alignment methods. This paradigm shift towards extracting domain-consistent features, rather than conventional alignment approaches, offers a more robust and practical solution for real-world time-series UDA applications.

7. Acknowledgements

The authors have no conflicts of interest to declare. This research was supported by the National Science and Technology Council (NSTC), Taiwan, R.O.C., under Grant MOST NSTC 113-2634-F-006-003 - and NSTC 114-2321-B-006-011 -. The authors would like to express their sincere appreciation to Po-Chih Lin, Ben Phan, Celia Wu, and Yu-Jei Zhou for their meticulous proofreading and valuable feedback.

References

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, Jorge Luis Reyes-Ortiz, et al. A public domain dataset for human activity recognition using smartphones. In *Esann*, volume 3, page 3, 2013.

Ruichu Cai, Jiawei Chen, Zijian Li, Wei Chen, Keli Zhang, Junjian Ye, Zhuozhang Li, Xiaoyan Yang, and Zhenjie Zhang. Time series domain adaptation via sparse associative structure alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6859–6867, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

Mingyue Cheng, Jiqian Yang, Tingyue Pan, Qi Liu, and Zhi Li. Convtimenet: A deep hierarchical fully convolutional model for multivariate time series analysis. *arXiv preprint arXiv:2403.01493*, 2024.

Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 447–463, 2018.

Emadeldeen Eldele, Zhenghua Chen, Chengyu Liu, Min Wu, Chee-Keong Kwoh, Xiaoli Li, and Cuntai Guan. An attention-based deep learning approach for sleep stage classification with single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:809–818, 2021.

Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, Chee-Keong Kwoh, and Xiaoli Li. Contrastive domain adaptation for time-series via temporal mixup. *IEEE Transactions on Artificial Intelligence*, 2023.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Saurabh Garg, Sivaraman Balakrishnan, Zico Kolter, and Zachary Lipton. Ratt: Leveraging unlabeled data to guarantee generalization. In *International Conference on Machine Learning*, pages 3598–3609. PMLR, 2021.

Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.

Huan He, Owen Queen, Teddy Koker, Consuelo Cuevas, Theodoros Tsiligkaridis, and Marinka Zitnik. Domain adaptation for time series under feature and label shifts. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 12746–12774. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/he23b.html.

Xinyue Huo, Lingxi Xie, Hengtong Hu, Wengang Zhou, Houqiang Li, and Qi Tian. Domain-agnostic prior for transfer semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7075–7085, 2022.

Xiaoyong Jin, Youngsuk Park, Danielle Maddix, Hao Wang, and Yuyang Wang. Domain adaptation for time series forecasting via attention sharing. In *International Conference on Machine Learning*, pages 10280–10297. PMLR, 2022.

Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4893–4902, 2019.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International conference on machine learning*, pages 5637–5664. PMLR, 2021.

Jennifer R Kwapisz, Gary M Weiss, and Samuel A Moore. Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*, 12(2):74–82, 2011.

Christian Lessmeier, James Kuria Kimotho, Detmar Zimmer, and Walter Sextro. Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification. In *PHM Society European Conference*, volume 3, 2016.

Qiao Liu and Hui Xue. Adversarial spectral kernel matching for unsupervised time series domain adaptation. In *IJCAI*, pages 2744–2750, 2021a.

Qiao Liu and Hui Xue. Adversarial spectral kernel matching for unsupervised time series domain adaptation. In *IJCAI*, pages 2744–2750, 2021b.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31, 2018.

Scott M Lundberg, Bala Nair, Monica S Vavilala, Mayumi Horibe, Michael J Eisses, Trevor Adams, David E Liston, Daniel King-Wai Low, Shu-Fang Newman, Jerry Kim, et al. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery, 2018.

Y. Luo, L. Zheng, T. Guan, J. Yu, and Y. Yang. Taking A closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. *CoRR*, abs/1809.09478, 2018. URL http://arxiv.org/abs/1809.09478.

Felix Ott, David Rügamer, Lucas Heublein, Bernd Bischl, and Christopher Mutschler. Domain adaptation for time-series classification to mitigate covariate shift. In *Proceedings of the 30th ACM international conference on multimedia*, pages 5934–5943, 2022.

Yilmazcan Ozyurt, Stefan Feuerriegel, and Ce Zhang. Contrastive learning for unsupervised domain adaptation of time series. *ICLR*, 2023.

F. Painblanc, L. Chapel, N. Courty, C. Friguet, C. Pelletier, and R. Tavenard. Match-and-deform: Time series domain adaptation through optimal transport and temporal alignment, September 2023.

Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3764–3773, 2020.

Sanjay Purushotham, Wilka Carvalho, Tanachat Nilanon, and Yan Liu. Variational recurrent adversarial deep domain adaptation. In *International conference on learning representations*, 2017.

Mohamed Ragab, Emadeldeen Eldele, Wee Ling Tan, Chuan-Sheng Foo, Zhenghua Chen, Min Wu, Chee-Keong Kwoh, and Xiaoli Li. Adatime: A benchmarking suite for domain adaptation on time series data. *ACM Transactions on Knowledge Discovery from Data*, 17 (8):1–18, 2023.

Mohammad Mahfujur Rahman, Clinton Fookes, Mahsa Baktashmotlagh, and Sridha Sridharan. On minimum discrepancy estimation for deep domain adaptation. *Domain Adaptation for Visual Understanding*, pages 81–94, 2020.

Suman Ravuri, Karel Lenc, Matthew Willson, Dmitry Kangin, Remi Lam, Piotr Mirowski, Megan Fitzsimons, Maria Athanassiadou, Sheleem Kashem, Sam Madge, et al. Skilful precipitation nowcasting using deep generative models of radar, 2021.

Rui Shu, Hung H Bui, Hirokazu Narui, and Stefano Ermon. A dirt-t approach to unsupervised domain adaptation. *arXiv preprint arXiv:1802.08735*, 2018.

Ankit Singh. Clda: Contrastive learning for semi-supervised domain adaptation. *Advances in Neural Information Processing Systems*, 34:5089–5101, 2021.

Peeyush Singhal, Rahee Walambe, Sheela Ramanna, and Ketan Kotecha. Domain adaptation: challenges, methods, datasets, and applications. *IEEE access*, 11:6973–7020, 2023.

Allan Stisen, Henrik Blunck, Sourav Bhattacharya, Thor Siiger Prentow, Mikkel Baun Kjærgaard, Anind Dey, Tobias Sonne, and Mads Møller Jensen. Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition.

In *Proceedings of the 13th ACM conference on embedded networked sensor systems*, pages 127–140, 2015.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*, pages 443–450. Springer, 2016.

Shixiang Tang, Peng Su, Dapeng Chen, and Wanli Ouyang. Gradient regularized contrastive learning for continual domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2665–2673, 2021.

Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. arxiv 2014. *arXiv preprint arXiv:1412.3474*, 2019.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Garrett Wilson and Diane J Cook. A survey of unsupervised deep domain adaptation, 2020.

Garrett Wilson, Janardhan Rao Doppa, and Diane J Cook. Multi-source deep domain adaptation with weak supervision for time-series sensor data. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1768–1778, 2020.

Garrett Wilson, Janardhan Rao Doppa, and Diane J Cook. Calda: Improving multi-source time series domain adaptation with contrastive adversarial learning. *IEEE transactions on pattern analysis and machine intelligence*, 2023.

Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domain adaptation under target and conditional shift. In *International conference on machine learning*, pages 819–827. Pmlr, 2013.

Xiang Zhang, Marko Zeman, Theodoros Tsiligkaridis, and Marinka Zitnik. Graph-guided network for irregularly sampled multivariate time series. *CoRR*, abs/2110.05357, 2021. URL https://arxiv.org/abs/2110.05357.

Yixin Zhang, Zilei Wang, Junjie Li, Jiafan Zhuang, and Zihan Lin. Towards effective instance discrimination contrastive loss for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11388–11399, 2023.