SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2025, Vol. 68, Iss. 1, 112103:1–112103:18
https://doi.org/10.1007/s11432-022-3860-y

Domain generalization with semi-supervised learning for people-centric activity recognition

Jing LIU[1,2†], Wei ZHU[1†], Di LI[2], Xing HU[3] & Liang SONG[1,2*]

[1]*Academy for Engineering & Technology, Fudan University, Shanghai 200433, China;*
[2]*Shanghai East-bund Research Institute on Networking Systems of AI, Shanghai 202162, China;*
[3]*School of Optoelectronic Information and Computer Engineering, University of Shanghai for Science & Technology, Shanghai 200093, China*

Received 22 September 2022/Revised 30 January 2023/Accepted 9 August 2023/Published online 18 December 2024

Abstract People-centric activity recognition is one of the most critical technologies in a wide range of real-world applications, including intelligent transportation systems, healthcare services, and brain-computer interfaces. Large-scale data collection and annotation make the application of machine learning algorithms prohibitively expensive when adapting to new tasks. One way of circumventing this limitation is to train the model in a semi-supervised learning manner that utilizes a percentage of unlabeled data to reduce the labeling burden in prediction tasks. Despite their appeal, these models often assume that labeled and unlabeled data come from similar distributions, which leads to the domain shift problem caused by the presence of distribution gaps. To address these limitations, we propose herein a novel method for people-centric activity recognition, called domain generalization with semi-supervised learning (DGSSL), that effectively enhances the representation learning and domain alignment capabilities of a model. We first design a new autoregressive discriminator for adversarial training between unlabeled and labeled source domains, extracting domain-specific features to reduce the distribution gaps. Second, we introduce two reconstruction tasks to capture the task-specific features to avoid losing information related to representation learning while maintaining task-specific consistency. Finally, benefiting from the collaborative optimization of these two tasks, the model can accurately predict both the domain and category labels of the source domains for the classification task. We conduct extensive experiments on three real-world sensing datasets. The experimental results show that DGSSL surpasses the three state-of-the-art methods with better performance and generalization.

Keywords activity recognition, deep learning, domain generalization, semi-supervised learning, adversarial training

Citation Liu J, Zhu W, Li D, et al. Domain generalization with semi-supervised learning for people-centric activity recognition. Sci China Inf Sci, 2025, 68(1): 112103, https://doi.org/10.1007/s11432-022-3860-y

1 Introduction

Mobile devices nowadays have higher computing power and lesser power consumption than desktop computers used in the past decades. With the rapid growth of wearables and the Internet of Things, various lightweight terminals (e.g., smartphones, unmanned aerial vehicles, smart watches, and smart glasses) are becoming popular with different computing and communication capabilities [1]. Regardless of how technologies evolve, these devices and applications are all centered around people. In this context, people-centric activity recognition is one of the most critical technologies in many real-world applications, including intelligent transportation systems [2], healthcare services [3], smart homes [4], and brain-computer interface [5]. The goal of people-centric activity recognition, in general, is to understand the behavior of participants via machine learning algorithms and person-related sensing data. Based on the devices used for data collection, two types of scenarios exist: wearable sensor- [6–8] and video-based [9–11] activities. In this study, we focus on wearable sensor-based activity recognition, considering the advantages of flexibility, richness, efficiency, and privacy security for sensing data [12].

Building a model that works for different people is becoming progressively important with the increasing pervasiveness of various smart sensing applications. One possible approach to model construction is to collect data from different people and use these to train a generic model that requires a substantial amount of effort in annotation. However, data annotation is unreasonable because manual annotation is expensive

* Corresponding author (email: songl@fudan.edu.cn)
† These authors contributed equally to this work.

and prone to privacy violations [13]. Several studies attempted to automatically label activity data, but this task was proven difficult [14]. Accordingly, some state-of-the-art methods [15–17] were presented to tackle this problem through semi-supervised learning (SSL), which efficiently trains models by learning a certain proportion of the labeled and unlabeled samples. However, a previous study [18] showed that, when testing samples with different distributions, semi-supervised methods have the risk of becoming extremely unreliable, and the model performance plummets. This kind of model lacks generalization to different people and is quite impractical in real-world applications because we cannot annotate all unseen subjects and retrain the model.

Most existing SSL methods typically assume that labeled and unlabeled data are drawn from similar distributions, which is not applicable to dynamic real-world applications. Different people have different behavior patterns and biometric characteristics; hence, the data collected from them obey different distributions. This is known as the domain shift problem [19]. Many transfer learning and domain adaptation (DA) methods [20, 21] were proposed to address the distribution gaps among different people. Transfer learning aims to train a model on multiple source domains and adapt it to the target domains. Negative transfer is avoided by using some information on the target domain, such as data label and label proportion [22]. Many results reported in the literature [23–25] demonstrated that transfer learning effectively reduces the distribution gaps. However, this method assumes that only the marginal distribution of the input data shifts and ignores the latent similarities associated with the output task, thereby making the model see only the gaps between the input data of the source and target domains.

Domain adaptation has been proposed to work on reducing the distribution gaps between the source and target domains. The most popular of the previous efforts employed the maximum mean discrepancy (MMD) [26] as a distance metric that measures the similarity of two distributions by matching the statistical moments of different orders. Another paradigm is inspired by the generative adversarial network (GAN) [27] that utilizes adversarial learning between the feature extractor and the domain discriminator to learn the domain-invariant features and alleviate the domain shift. However, applying DA to sensor-based activity recognition remains challenging. First, most of the existing DA methods [14, 24, 28] have been designed for visual tasks, and extending them to time series tasks is usually a sub-priority. Second, previous works typically relied on ImageNet [29] to pre-train the model, which is not suitable for sensor-based activity recognition tasks.

In the presence of multiple target domains, these DA methods require model retraining for each target domain. Domain generalization learns generalizable features from several relevant source domains to enable the learned model to work well on unseen domains [19, 30]. Based on the abovementioned observations and the uniqueness of people's activities, we treat each dataset participant in this work as a single domain. In most real-world applications, labeling a large number of samples as source domains for training is difficult. Previous DA methods also assumed that the target domain is available during the training phase and rarely considered the case of combining labeled and unlabeled source domain samples. Therefore, we assume that both unlabeled and labeled samples from different domains with different distributions can be used to train a model only once to generalize to unseen samples.

To address the domain shift across different people, we propose herein a novel model, called domain generalization with semi-supervised learning (DGSSL), for people-centric activity recognition. First, all training data subjects are divided into unlabeled and labeled source domains. Next, the corresponding latent representations are generated by a shared weight encoder. Finally, the model is collaboratively optimized by two kinds of features, namely, domain- and task-specific features. Ideally, domain-specific features capture the common information across people, enabling the model to understand the distribution gaps between the labeled and unlabeled data. In contrast, task-specific features reflect the inherent factors of the same activity that lead to variations between subjects, such as a person's age and gender and behavior patterns in a particular environment. As shown in Figure 1, we utilize adversarial training and reconstruction as the two novel tasks for effectively disentangling the two types of features, such that the model can improve its understanding of the distribution gaps while maintaining a representation of the inherent characteristics of a specific distribution. Consequently, the model effectively predicts the domain and category labels of the source domains in the classification tasks. The main contributions of this study are as follows.

• We propose a domain generalization training method with SSL that improves the model's capabilities for representation and transfer learning. DGSSL aims to learn powerful embedding in a unified framework, which will enhance the understanding of distribution gaps while maintaining the inherent consistency of a specific distribution.

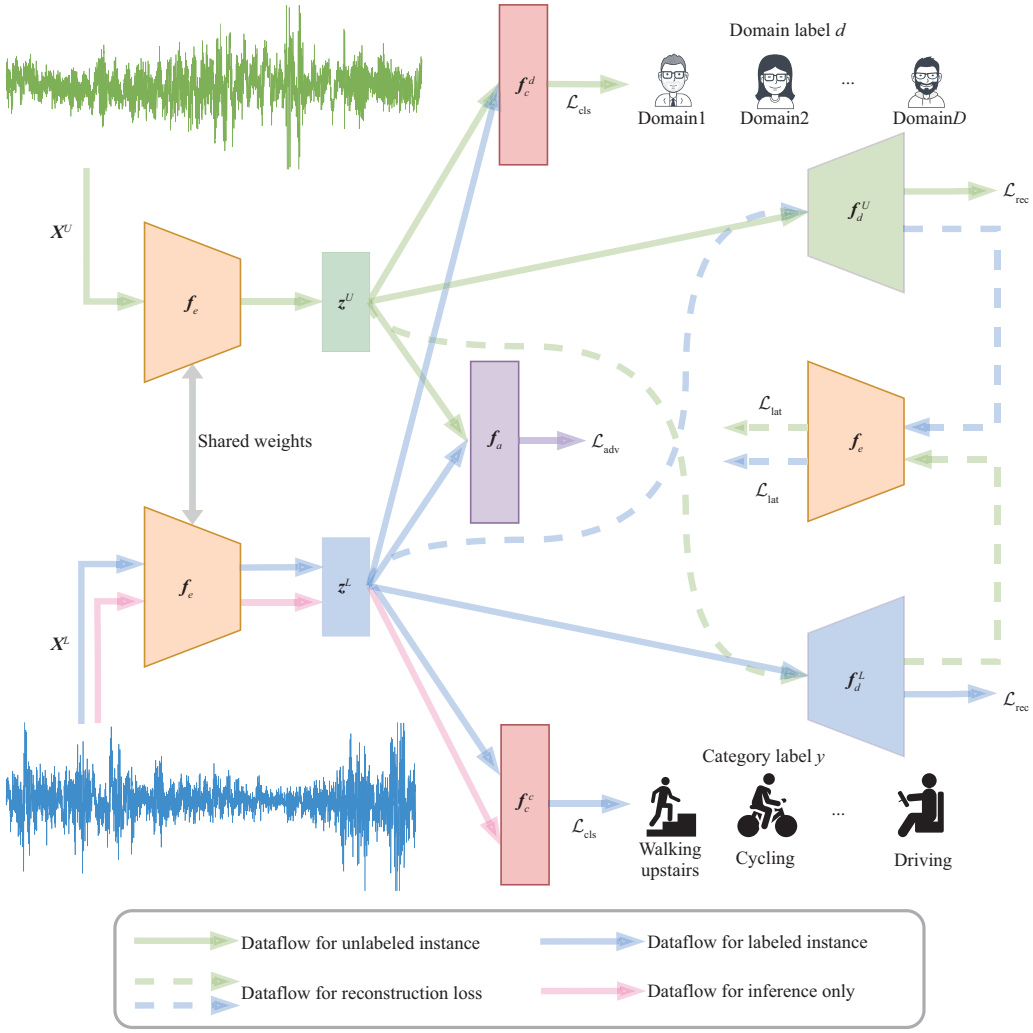

Figure 1 (Color online) Overview of the proposed DGSSL. Our model is trained in a semi-supervised learning manner with two types of tasks, i.e., adversarial training and reconstruction tasks. Firstly, the unlabeled domain $\boldsymbol{X}^U$ and labeled domain $\boldsymbol{X}^L$ of source domains are fed as inputs to the feature extractor f_e to generate the corresponding latent representation $\boldsymbol{z} \in \{\boldsymbol{z}^U, \boldsymbol{z}^L\}$. Then, we introduce an autoregressive discriminator f_a to identify whether the latent representation is mapped from the unlabeled domain $\boldsymbol{X}^U$ or labeled domain $\boldsymbol{X}^L$ and alleviate the domain shift problem. Meanwhile, to preserve the relevance of the same activity in different domain samples, the task of sequence reconstruction and latent representation reconstruction are performed by two decoders $\{f_d^U, f_d^L\}$ that take corresponding latent representation $\boldsymbol{z}$ as input. Finally, we leverage the domain label and category label of data to guide the learning of latent representations via two classifiers $\{f_d^U, f_d^L\}$. The latent representations $\boldsymbol{z}^U$ and $\boldsymbol{z}^L$ are fed to the domain classifier f_c^d to predict the domain label, and $\boldsymbol{z}^L$ is the input for category classifier f_c^c to predict the activity label. In the inference stage, only the feature extractor f_e and category classifier f_c^c are activated, while the other components are not available. The purpose is to predict the activity label of target domain samples, as shown in the red line arrow.

• We alleviate the domain shift problem caused by different distributions by utilizing adversarial training with SSL to capture domain-specific features. By considering the alignment of the marginal distributions of the unlabeled and labeled source domains, the model forces the latent representations generated by two generative tasks to lie in the same space.

• The task-specific features are extracted via two reconstruction tasks, namely sequence and latent representation reconstruction. The reconstruction task aims to maintain consistency with the classification task and avoid losing the common information related to representation learning during the domain alignment.

• We conduct comprehensive experiments on three people-centric sensing datasets to evaluate our proposed model. The experimental results demonstrate that our model outperforms the three state-of-the-art methods with better performance and generalization.

The remainder of this paper is structured as follows: the related studies on activity recognition, SSL, and DA related to the proposed DGSSL are introduced in Section 2; our proposed DGSSL is described

in detail in Section 3; the experiments conducted on three publicly available datasets are elaborated in Section 4 to demonstrate the effectiveness of DGSSL; and finally, the concluding remarks are presented in Section 5.

2 Related work

The key aspects of the proposed DGSSL are sensor-based activity recognition, SSL, and DA. Therefore, we individually present in this section the recent literature on these three topics.

2.1 Sensor-based activity recognition

Most sensor-based human activity recognition (HAR) solutions utilize classical machine learning algorithms, such as support vector machine (SVM), random forest, and k-nearest neighbor (kNN), to recognize various patterns from sensing data. Nweke et al. [3] proposed a method for identifying transportation and vehicle patterns using an accelerometer, a gyroscope, and a magnetometer in smartphone leveraging assembly learning that integrated decision tree, kNN, and SVM. With the development of deep learning and sensor technologies in recent years, sensor-based HAR has received widespread attention from the academic community [31]. Accordingly, many researchers adopted deep learning techniques for HAR and achieved satisfactory results. Yao et al. [32] utilized convolutional (CNN) and recurrent (RNN) neural networks to learn the local interactions of each sensor and the global interactions between different sensors to find the most robust features to adapt to the user behavior. In addition, some hybrid models [33–35] based on the CNN-RNN combination were also proposed, predicting different activities by exploring the spatiotemporal relationships from sensing data. Recent studies attempted to employ attention mechanisms to obtain a better performance. Hammerla et al. [36] and Ma et al. [37] combined an attention network and a CNN-RNN model to identify human activity by measuring the relationships within and between sensor channels and select information that is helpful for the same activity.

2.2 SSL

Semi-supervised learning aims to train neural networks jointly leveraging a small amount of labeled data and a large amount of unlabeled data. It is widely adopted in research areas, such as computer vision [38–40] and natural language processing [41, 42]. SSL approaches, including FixMatch [43], HMVSSL [44], SANI [45], and Tri-Net [15], recently underwent great progress. Despite the absence of technical barriers to applying SSL to HAR, only a few works tried to solve HAR tasks (e.g., FedHAR [7], AVAE [46], ASM2TV [47], DynaLAP [48], and AdaptNet [17]). They resorted to deep generative models like the restricted Boltzmann machine [49] and the autoencoder [50]) to train the model by utilizing joint tasks with large amounts of unlabeled data and obtain a well-trained feature extractor. However, these works seldom considered the variation of different data sources prevalent in the real world when applying the model to practical applications. Moreover, most discriminative models cannot directly participate well in the SSL framework because the labeled data are too small to use in training an efficient feature extractor. SSL for HAR with a small amount of labeled data used to address the domain shift problem must be carefully studied.

2.3 DA

To realize transfer learning across domains, many methods train the neural network on the source domain and fine-tune the model using the training data of the target domain [51]. Most DA efforts learn the domain-invariant features by minimizing the discrepancy between domains. These discrepancy reduction-based methods are divided into two categories: (1) metric-based methods: learning invariant features between domains by learning a certain metric distance that reduces the discrepancy, and (2) domain discrimination-based methods: distinguishing the source and target domains by an adversarial loss-based discriminator. The former are basically based on the MMD. For example, deep domain adaptation [52] and deep domain confusion [53] are classically adopted to reduce the discrepancy between two domains by minimizing the MMD of the feature distribution. Chao et al. [54] proposed higher-order moment matching by further considering the higher-order metric employing third- or fourth-order polynomial MMDs. The latter (e.g., domain-adversarial neural network (DANN) [24]) try to minimize the domain discrepancies by discriminating the domains. The convolutional deep DA model for the time series data (CoDATS) [25]

introduced weak supervision into the domain adaptive methods that learn domain-invariant features by adversarial learning between the feature extractor and the classifier. CALDA [21] leveraged the label information of cross-domain samples by approximating the same labeled data while pushing away samples with different labels, thereby reshaping the feature space via contrastive learning. The abovementioned methods basically require the usage of the target domain information in the training phase, which limits their potential capabilities. Our method considers partially available source and domain label information across domains and focuses on learning the domain-invariant features by adversarial training in an SSL manner to reduce the distribution gaps caused by different people.

3 Methodology

3.1 Problem formulation

Our method aimed to solve the people-centric activity recognition task given the unlabeled $\mathcal{D}_{\text{au}}$ and labeled $\mathcal{D}_{\text{al}}$ source domain sets. In our task setting, each domain was defined as a joint distribution $\mathbb{P}^d(x, y)$ on $\mathcal{X} \times \mathcal{Y}$, where $d \in \mathcal{D}_d = \{1, \ldots, D\}, \mathcal{X}$, and $\mathcal{Y}$ denote the index of the domain, activity instance space, and activity category space, respectively. We first divided the data collected from the participants into the unlabeled $\{(\boldsymbol{X}^U, d) \in \mathcal{D}_{\text{au}}\}_{d=1}^{D/2}$ and labeled $\{(\boldsymbol{X}^L, y, d) \in \mathcal{D}_{\text{al}}\}_{d=1}^{D/2}$ source domain sets for training using a participant identifier. We left the target domain $\boldsymbol{X}^{\hat{T}}$ as an unseen sample for testing. All domains shared the same label space, that is, $y \in \{1, \ldots, n_c\}$, where n_c is the number of activity categories on the dataset. Note that only the category label of the labeled domain $\boldsymbol{X}^L$ was available in the training phase. Each domain instance $\boldsymbol{X}_i \in \mathbb{R}^{C \times N_i}$ with C channels and N_i time steps. We assumed that $\boldsymbol{X}_i^U$ and $\boldsymbol{X}_j^L$ corresponded to distributions $\mathbb{P}^i(x, y) \neq \mathbb{P}^j(x, y), \forall i \neq j$, where $i, j \in \mathcal{D}$, that is, gaps existed between the distributions of the data collected by the two participants. We, however, believed that a potential consistency can be found between participants in the same activity. Therefore, we aimed to learn the latent representations $\boldsymbol{z}^U$ and $\boldsymbol{z}^L$ corresponding to the unlabeled and labeled domains from the D domains via a deep learning model parameterized as f_e to enable the model to map cross-domain samples to a uniform space and to generalize well for the identification of the unlabeled target samples $\boldsymbol{X}^{\hat{T}}$ on a new space. Compared with the standard machine learning setting, we trained the model in an SSL manner and leveraged the easily obtained domain label data and privacy security by simply and anonymously recording the participants' identity as the domain index.

3.2 Overview of DGSSL

As shown in Figure 1, the proposed DGSSL is decomposed into four components based on the abovementioned assumptions.

• **Feature extractor** f_e. an encoder for feeding the unlabeled $\boldsymbol{X}^U$ and labeled $\boldsymbol{X}^L$ domain to generate the corresponding latent representation $\boldsymbol{z}$.

• **Domain discriminator** f_a. an autoregressive discriminator for predicting whether or not the latent representation comes from the unlabeled $\boldsymbol{X}^U$ or labeled $\boldsymbol{X}^L$ domain.

• **Reconstructors** f_d^U and f_d^L. two decoders for feeding latent representation $\boldsymbol{z}^U$ or $\boldsymbol{z}^L$ for the sequence reconstruction task to reconstruct the sequence $\hat{\boldsymbol{X}}^U$ or $\hat{\boldsymbol{X}}^L$ and for the latent representation reconstruction task to reconstruct the latent representation $\hat{\boldsymbol{z}}^U$ or $\hat{\boldsymbol{z}}^L$.

• **Classifiers** f_c^d and f_c^c. classifiers with the same structure composed of the softmax function and a fully connected layer for predicting the domain and category labels.

Given the unlabeled and labeled domain training samples, the model aimed to train a powerful cross-domain feature extractor f_e and a category classifier f_c^L. To do this, we first generated a latent representation using the feature extractor f_e. We then discriminated whether or not the latent representation was mapped from the unlabeled $\boldsymbol{X}^U$ or labeled $\boldsymbol{X}^L$ domain by the autoregressive discriminator. At the same time, we utilized two decoders to reconstruct the original sequence and the new latent representation using the latent representation $\boldsymbol{z}$. Finally, the latent representation was used to predict the sample's domain or category in the classification task. The inference phase components included only the feature extractor f_e and the category classifier f_c^L. The process fed the target domain samples into f_e to predict the category label by f_c^c.

The model performed backpropagation by combining multiple tasks in a semi-supervised manner. The training targeted the minimization of the joint loss of these tasks. The joint loss function included four

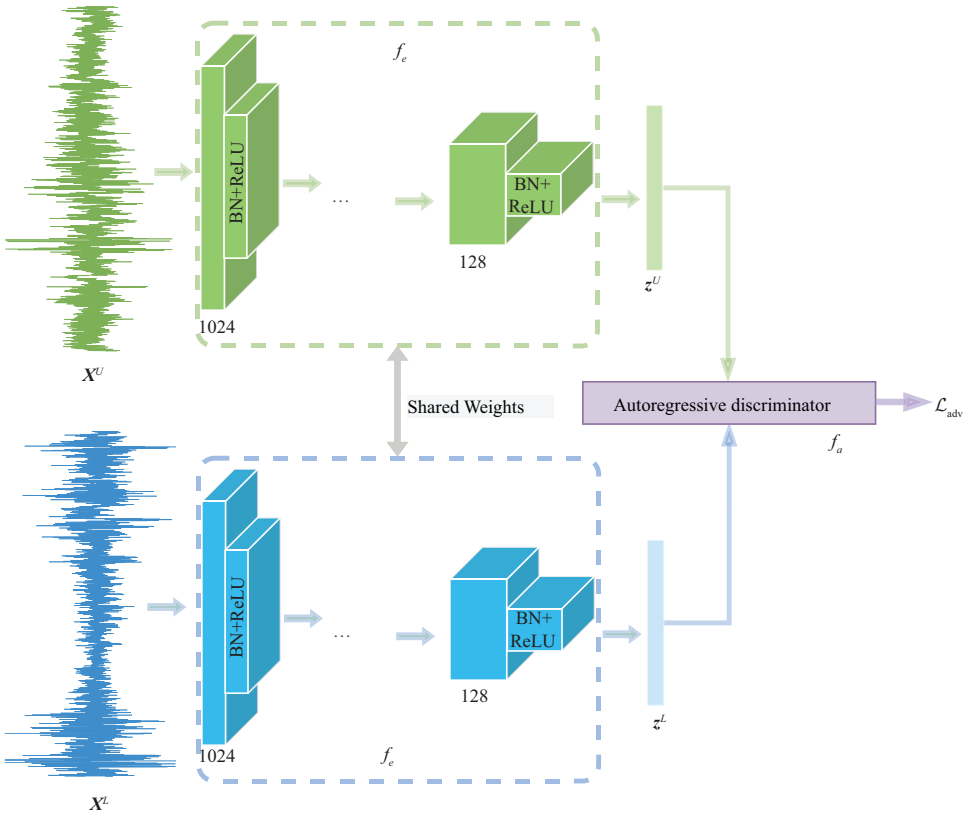

Figure 2 (Color online) Illustration of adversarial training.

parts: (1) domain-adversarial loss $\mathcal{L}_{\text{adv}}$ used to reduce the distribution gap of the unlabeled and labeled domains; (2) sequence reconstruction loss $\mathcal{L}_{\text{rec}}$ used to reduce the difference between the original and reconstructed sequences mapped from the latent representation $\boldsymbol{z}$; (3) latent representation reconstruction loss $\mathcal{L}_{\text{lat}}$ used as a constraint to prevent losing task-relevant information; and (4) classification loss $\mathcal{L}_{\text{cls}}$ used to encourage the feature extractor f_e to learn powerful features for predicting the corresponding labels. The joint loss function is formulated as follows:

$$\begin{aligned}\mathcal{L}_{\text{all}} = &\mathcal{L}_{\text{adv}}\left(f_e, f_a; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right) + \mathcal{L}_{\text{rec}}\left(f_e, f_d^U, f_d^L; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right)\\ &+ \mathcal{L}_{\text{lat}}\left(f_e, f_d^U, f_d^L; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L, \hat{\boldsymbol{X}}^U, \hat{\boldsymbol{X}}^L,\right) + \mathcal{L}_{\text{cls}}\left(f_e, f_c; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right). \end{aligned} \tag{1}$$

3.3 Domain-specific features via adversarial training

The samples collected from different participants typically had distribution gaps, which may be caused by many factors, including the behavior pattern of people conducting activities and the environmental constraints of an activity. Hence, the model must be enabled to generalize across domains and effectively identify different factors from the original sensing data. As shown in Figure 2, we modeled the generation process of the observed activity data and learned the underlying factors leading to the distribution gaps (i.e., domain-specific features) by adversarial training.

We aimed to learn robust latent representations $\boldsymbol{z}^U$ and $\boldsymbol{z}^L$ via a shared weight feature extractor f_e that maps them to a uniform space. However, the feature extractor f_e and the classifier f_c^c were not sufficient for distinguishing the variation between individuals. Inspired by [55], we boost the latent representation by introducing an autoregressive discriminator f_a. Figure 3 depicts the autoregressive discriminator that comprised two main components, that is, an autoregressive network for encoding the latent representations of the unlabeled and labeled domains into vector representations and a binary classification network to summarize the vector representations for distinguishing the unlabeled and labeled domains. The abovementioned process is formulated as follows:

$$\boldsymbol{z}^U = f_e(\boldsymbol{X}^U; \theta_e), \tag{2}$$

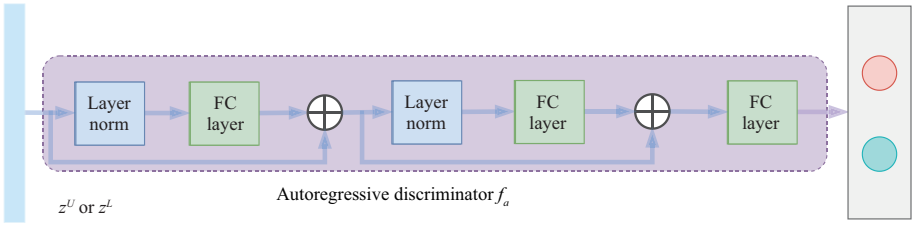

Figure 3 (Color online) Structure of autoregressive discriminator.

$$\boldsymbol{z}^L = f_e(\boldsymbol{X}^L; \theta_e), \tag{3}$$

$$\mathcal{L}_{\text{adv}} = \mathcal{L}_{\text{adv}}\left(f_e, f_a; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right), \tag{4}$$

where θ_e is the learnable parameter of the feature extractor f_e. The process in (4) was related to the GAN [27]. The feature extractor f_e for DGSSL was similar to the generative model used to generate instances to reduce the distribution gaps. The autoregressive discriminator worked to enlarge the gaps. By introducing f_a, adversarial learning was applied to model training, resulting in more discriminative representations of the discrepancy metric and leading to a more accurate distribution alignment.

We also need a sufficiently strong classifier to distinguish the users from the latent representations, of which features were drawn from similar distributions. This step was done by fixing parameter θ_e of f_e while maximizing (4) by updating parameter θ_a of f_s. The process of minimizing the adversarial loss was optimized as follows:

$$\min_{\theta_e}\max_{\theta_a}\mathcal{L}_{\text{adv}} = \min_{\theta_e}\mathcal{L}_{\text{adv}}\left(f_e, f_a; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right). \tag{5}$$

Eq. (4) is probabilistically formulated by the following objective function:

$$\begin{aligned}\mathcal{L}_{\text{adv}} &\approx \mathbb{E}_{x\sim\mathbb{P}_{\text{au}}}\left[\log f_a\left(f_e(\boldsymbol{X}^U)\right)\right] + \mathbb{E}_{x\sim\mathbb{P}_{\text{al}}}\left[\log\left(1 - f_a\left(f_e(\boldsymbol{X}^L)\right)\right)\right] \\ &= \mathbb{E}_{z\sim\mathbb{P}_{\text{au}}}\left[\log f_a(\boldsymbol{z}^U)\right] + \mathbb{E}_{z\sim\mathbb{P}_{\text{al}}}\left[\log\left(1 - f_a(\boldsymbol{z}^L)\right)\right].\end{aligned} \tag{6}$$

The maximization of (6) was related to the Jensen-Shannon divergence [56] of distributions $\mathbb{P}_{\text{al}}$ and $\mathbb{P}_{\text{au}}$. Therefore, Eq. (6) is represented as

$$\max_{\theta_a}\mathcal{L}_{\text{adv}} = 2\,\text{JSD}\left(\mathbb{P}_{\text{au}}(\boldsymbol{z}^U)\|\mathbb{P}_{\text{al}}(\boldsymbol{z}^L)\right), \tag{7}$$

where $\text{JSD}\left(\cdot\|\cdot\right)$ denotes the Jensen-Shannon divergence. If we ignore the parameter constant, the adversarial loss optimization is represented as finding the optimal parameter θ_e such that the distribution gaps $\mathbb{P}_{\text{al}}(z)$ and $\mathbb{P}_{\text{au}}(z)$ are minimized

$$\min_{\theta_e}\max_{\theta_a}\mathcal{L}_{\text{adv}} = \min_{\theta_e}\text{JSD}\left(\mathbb{P}_{\text{al}}(\boldsymbol{z}^U)\|\mathbb{P}_{\text{au}}(\boldsymbol{z}^L)\right). \tag{8}$$

3.4 Task-specific features via reconstructions

Although the component presented above can learn domain-specific features, we did not evaluate the task relevance for different domain samples with the same activity in our semi-supervised training. We further improved the task consistency for the activity (e.g., same activity from $\boldsymbol{X}^U$ and $\boldsymbol{X}^L$) by utilizing two decoders $\{f_d^U, f_d^L\}$ to generate the paired data by the corresponding latent representation $\boldsymbol{z}$ (Figure 4). Given an input source domain sample $\boldsymbol{X}^U$, the latent representation $\boldsymbol{z}^U$ was first generated by using the feature encoder f_e with parameter θ_e. The sequence $\hat{\boldsymbol{X}}^U$ was then reconstructed by employing the decoder f_d^U with parameter θ_d^U. This process is formulated as follows:

$$\boldsymbol{X}^U \approx f_d^U(f_e(\boldsymbol{X}^U; \theta_e); \theta_d^U) \approx f_d^U(\boldsymbol{z}^U; \theta_d^U) \approx \hat{\boldsymbol{X}}^U. \tag{9}$$

This was the same for the labeled domain sample

$$\boldsymbol{X}^L \approx f_d^L(f_e(\boldsymbol{X}^L; \theta_e); \theta_d^L) \approx f_d^L(\boldsymbol{z}^L; \theta_d^L) \approx \hat{\boldsymbol{X}}^L. \tag{10}$$

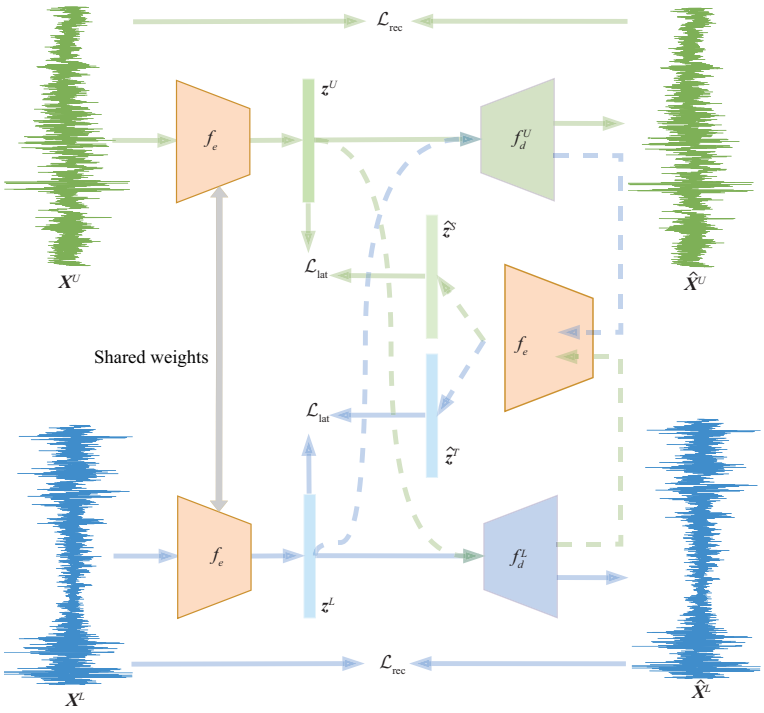

Figure 4 (Color online) Illustration of reconstruction tasks.

Eqs. (9)–(10) minimized the sequence reconstruction loss $\mathcal{L}_{\text{rec}}$ as follows:

$$\begin{aligned}\mathcal{L}_{\text{rec}} &= \mathcal{L}_{\text{rec}}\left(f_e, f_d^U, f_d^L; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right)\\ &= \frac{1}{N_u}\sum_{i=1}^{N_u}\left\|\boldsymbol{X}_i^U - f_d^U\left(f_e\left(\boldsymbol{X}_i^U\right)\right)\right\|^2 + \frac{1}{N_l}\sum_{i=1}^{N_l}\left\|\boldsymbol{X}_i^L - f_d^L\left(f_e\left(\boldsymbol{X}_i^L\right)\right)\right\|^2\\ &= \frac{1}{N_u}\sum_{i=1}^{N_u}\left\|\boldsymbol{X}_i^U - f_d^U\left(\boldsymbol{z}^U\right)\right\|^2 + \frac{1}{N_l}\sum_{i=1}^{N_l}\left\|\boldsymbol{X}_i^L - f_d^L\left(\boldsymbol{z}^L\right)\right\|^2,\end{aligned} \tag{11}$$

where N_u and N_l are the numbers of the unlabeled and labeled samples, respectively, and $\|\cdot\|^2$ denotes the L2 norm between two variables. The reconstruction task and the adversarial training were performed separately in the training phase. In other words, the parameters of the two decoders $\{f_d^U, f_d^L\}$ were updated only when the sequence $\mathcal{L}_{\text{rec}}$ and latent representation $\mathcal{L}_{\text{lat}}$ reconstruction losses were minimized to avoid distracting the domain-specific features. The latent representations of the unlabeled and labeled domains were inversely fed into another decoder and formulated as follows to ensure the consistency of the same activity from different domains:

$$\begin{aligned}\mathcal{L}_{\text{lat}} &= \mathcal{L}_{\text{lat}}\left(f_e, f_d^U, f_d^L; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L, \hat{\boldsymbol{X}}^U, \hat{\boldsymbol{X}}^L\right)\\ &= \frac{1}{N_u}\sum_{i=1}^{N_u}\left\|f_e\left(\boldsymbol{X}_i^U\right) - f_e\left(f_d^L\left(f_e\left(\boldsymbol{X}_i^U\right)\right)\right)\right\|^2 + \frac{1}{N_l}\sum_{i=1}^{N_l}\left\|f_e\left(\boldsymbol{X}_i\right)^L - f_e\left(f_d^U\left(f_e\left(\boldsymbol{X}_i^L\right)\right)\right)\right\|^2\\ &= \frac{1}{N_u}\sum_{i=1}^{N_u}\left\|\boldsymbol{z}_i^U - f_e\left(\hat{\boldsymbol{X}}_i^U\right)\right\|^2 + \frac{1}{N_l}\sum_{i=1}^{N_l}\left\|\boldsymbol{z}_i^L - f_e\left(\hat{\boldsymbol{X}}_i^L\right)\right\|^2.\end{aligned} \tag{12}$$

3.5 Classification task learning

Although the learning process of the two abovementioned tasks was efficient, it was not sufficient to guarantee that the feature extractor f_e will learn the non-overlapping and disentangled features. In some extreme cases, the latent representations $\boldsymbol{z}^U$ and $\boldsymbol{z}^L$ may learn the same features. Therefore, we employed the domain f_c^d and category f_c^c classifiers in the training phase to guide the learning of the

Table 1 Description of all datasets used in our experiments. T9–T13 represent the 9th–13th subjects in the TMD dataset. Likewise, U1–U5 and H1–H5 denote the first 5 subjects in the UAH and UCIHAR datasets. Note that the column target domain is used as the target domain sample for testing.

Dataset	# Subject	# Class	# Sample	Frequency (Hz)	# Channel	Target domain
TMD [57]	13	5	735621	20	15	(T9, T10, T11, T12, T13)
UAH [58]	6	3	359903	10	36	(U1, U2, U3, U4, U5)
UCIHAR [59]	30	6	914283	50	9	(H1, H2, H3, H4, H5)

latent representations using the domain d and category y labels. Figure 1 depicts the process. To fully incorporate the category and domain information, the two classifiers $\{f_c^d, f_c^c\}$ with the $\{\theta_d, \theta_c\}$ parameters predicted the corresponding labels with the $\boldsymbol{z}^U$ and $\boldsymbol{z}^L$ latent representations as the input, respectively. The process of minimizing the model's classification loss is formulated as follows:

$$\begin{aligned}\mathcal{L}_{\text{cls}} &= \mathcal{L}_{\text{cls}}\left(f_e, f_c^d, f_c^c; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right)\\ &= \frac{1}{N_i}\sum_{i=1}^{N_i}\ell\left(d, f_c^d\left(f_e\left(\boldsymbol{X}_i;\theta_e\right);\theta_d\right)\right) + \frac{1}{N_l}\sum_{i=1}^{N_l}\ell\left(y_i, f_c^c\left(f_e\left(\boldsymbol{X}_i^L;\theta_e\right);\theta_c\right)\right)\\ &= \frac{1}{N_i}\sum_{i=1}^{N_i}\ell\left(d, f_c^d\left(\boldsymbol{z}_i;\theta_d\right)\right) + \frac{1}{N_l}\sum_{i=1}^{N_l}\ell\left(y_i, f_c^c\left(\boldsymbol{z}_i^L;\theta_c\right)\right),\end{aligned} \tag{13}$$

where $\ell(\cdot)$ is a cross-loss function like $\ell(y,\hat{y}) = -\sum_{i=1}^{N_l} y_i \log f_c^c(f_e(\boldsymbol{X}^L))$. The model was able to encourage the domain- and task-specific features to consistently contribute to better classification results.

3.6 Optimization

Our method optimized the feature extractor f_e. The abovementioned components constructed the overall objective function $\mathcal{L}_{\text{all}}$ of the model formulated in a weighted summation as follows:

$$\begin{aligned}\mathcal{L}_{\text{all}} = {}&\lambda_{\text{adv}}\cdot\mathcal{L}_{\text{adv}}\left(f_e, f_a; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right) + \lambda_{\text{rec}}\cdot\mathcal{L}_{\text{rec}}\left(f_e, f_d^U, f_d^L; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right)\\ &+ \lambda_{\text{lat}}\cdot\mathcal{L}_{\text{lat}}\left(f_e, f_d^U, f_d^L; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L, \hat{\boldsymbol{X}}^U, \hat{\boldsymbol{X}}^L\right) + \mathcal{L}_{\text{cls}}\left(f_e, f_c^d, f_c^c; \boldsymbol{X}^U, \boldsymbol{X}^L, \boldsymbol{z}^U, \boldsymbol{z}^L\right),\end{aligned} \tag{14}$$

where λ_{adv}, λ_{rec}, and λ_{lat} are the trade-off parameters for $\mathcal{L}_{\text{all}}$. Compared to the previous work [17], our method introduced adversarial training while considering two reconstruction tasks, thereby resulting in a more distinguished representation of the learned latent representation. DGSSL also leveraged the available domain and activity labels as additional information for model training, making the learned features more meaningful and informative. Empirically, we found that setting two loss functions separately for each task will result in better training (i.e., two decoders were set instead of one with shared weights). The potential reason for this result may be that setting the loss functions separately was more helpful for feature disentanglement. In the inference phase, only the feature extractor f_e and the category classifier f_c^c were activated to predict the activity label of the unseen target domain samples. The other components were not made available. DGSSL is an elegant model that efficiently utilizes adversarial training and reconstruction tasks for training in an SSL manner, keeping it simple in the inference phase.

4 Experiments

4.1 Datasets

We verified the generalizability of the proposed method by evaluating DGSSL on three people-centric activity recognition datasets, namely, transportation mode detection (TMD), UAH-DriveSet (UAH), and UCIHAR. Table 1 [57–59] presents the descriptive information of all datasets used in our experiments. We selected multiple subjects in each dataset to form domain-level datasets for training and testing to better conduct the experiments.

TMD dataset [57]**.** The TMD dataset used smartphones to collect the multimodal sensing data from 13 subjects (i.e., three females and 10 males). This dataset not only contained acceleration, gravity, orientation, and GPS data but also provided details, such as gender, occupation, device model, and

user's age. It included 5894 trips for the five transportation modes of Still, Walking, Car, Train, and Bus. Each user was assigned to perform different activities in real-world situations and record the sensor measurement data. To guarantee the diversity of the training samples, we selected the first eight subjects (T1–T8) from the TMD dataset as the training set and the last five subjects (T9–T13) as the target domain for testing. The total number of samples was 735621 with a 20 Hz frequency. The feature channel was 15.

UAH dataset [58]**.** This dataset collected the driving behavior data of six drivers with different ages and genders. The vehicle driving data were recorded via the embedded smartphone sensors. These drivers simulated three driving styles (i.e., Normal, Aggressive, and Drowsy) on two road types (i.e., highway and secondary roads), generating over 500 min of raw data on naturalistic driving. We only had a total of six experimental subjects. We chose to test five target domains (U1–U5) by taking turns with four subjects for training and the remaining subjects as the target domain samples for testing. We obtained 359903 samples with a 10 Hz frequency after pre-processing. The instance dimension was 36.

UCIHAR dataset [59]**.** The UCIHAR dataset contained six activities, namely, Walking, Walking downstairs, Sitting, Walking upstairs, Laying, and Standing, which were collected by 30 volunteers aged 19–48 years old. They wore smartphones at their waists to record the linear acceleration, angular velocity, and orientation data in three axes collected by the accelerometer and gyroscope embedded in these smartphones. We chose the first five subjects (H1–H5) as the target domain samples in the testing phase and the remaining eight subjects (H6–H13) as the training samples. We had 914283 samples with a 50 Hz frequency after pre-processing. The feature dimension was 9.

4.2 Comparing methods

We compared our proposed method, called DGSSL, with the baseline and state-of-the-art methods. These methods are basically for time series-related datasets and classified into deep learning- (i.e., CNN [60], VAE [61], and DeepConvLSTM [62]), domain adaptation-based (i.e., DANN [24], CoDATS [25], and CALDA [21]) and semi-supervised learning-based methods (i.e., Tri-Net [15] and AdaptNet [17]). We reproduced the CNN and VAE models as the baseline methods to fit the datasets used in our experiments. CALDA and AdaptNet are the latest state-of-the-art time series-related methods in the fields of DA and semi-supervision. Instead of reproducing them in Pytorch [63], we used the experimental results of the abovementioned models if they are available on the dataset.

4.3 Experimental settings

We conducted experiments on each dataset, which included model training using the multiple unlabeled and labeled domains of the available source domain set $\mathcal{D}_a$. The remaining domains were partially treated as an unseen target domain $\boldsymbol{X}^{\hat{T}}$ for testing. The $\mathcal{D}_a$ set was evenly divided into two parts, that is, the unlabeled $\mathcal{D}_{\text{au}}$ and labeled $\mathcal{D}_{\text{al}}$ domain sets. We used accuracy (Acc) and F1-score (F1) as the evaluation metrics in the experiments. We applied a different sliding window strategy for each instance of the dataset in the data pre-processing stage. To segment the original dataset into instances, we set 5 s for the TMD and UAH datasets and 2.56 s for the UCIHAR dataset. All data segments had a 50% overlap rate.

The feature extractor f_e was an encoder with a five-layer structure, including four convolution and max-pooling layers. Each layer was followed by a batch normalization and rectified linear unit operation. The last layer was a fully connected (FC) layer for refining the latent representation. A dropout layer with 0.2 probability was applied before the FC layer. The kernels of the convolution, max-pooling, and FC layers were set to 1×3, 1×2, and 128, respectively. The two decoders were mirror structures of the encoder (i.e., one FC, four un-max-pooling, and deconvolution layers). For the classifier, we utilized one FC layer and a softmax function to compute the classification output.

We followed the experimental setup commonly used in [17]. We set the batch size to 64 and the max training epoch to 100. The loss function was minimized at a 1×10^{-4} learning rate by employing the stochastic gradient descent and the Adam optimizer [64]. All experiments were conducted in Pytorch with an Nvidia RTX 2080 Ti GPU.

4.4 Results and analysis

Tables 2–4 present the experimental results of DGSSL and other methods on the TMD, UAH, and UCIHAR datasets, in which the best performance was presented in bold, and the second-best performance

Table 2 The ovreall performance (%) of target domains (T9–T13) on TMD dataset. The best and second-best values are indicated in bold and underlined.

Method	T9		T10		T11		T12		T13		mAcc	mF1
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1		
CNN	53.71	52.51	50.25	48.30	61.74	58.03	57.44	59.56	58.77	56.38	56.38	54.96
VAE	65.74	66.79	59.05	62.71	72.18	71.03	70.12	69.67	69.85	68.93	67.39	67.83
DeepConvLSTM	86.13	85.37	85.88	87.28	90.63	89.85	89.66	88.85	89.46	87.59	88.35	87.79
DANN	81.38	82.13	80.49	79.18	87.00	88.06	85.52	86.97	83.56	82.62	83.59	83.79
CoDATS	83.08	82.58	80.18	82.31	85.87	87.40	84.93	85.29	85.62	85.83	83.94	84.68
CALDA	84.07	83.61	85.35	84.73	91.30	90.65	89.15	88.56	87.16	86.31	87.41	86.77
Tri-Net	86.70	87.60	83.02	84.16	89.52	91.72	88.10	90.03	89.12	88.40	87.29	88.38
AdaptNet	**89.66**	**88.63**	87.22	85.87	92.32	92.65	91.54	92.38	92.91	91.95	90.73	90.30
DGSSL (ours)	89.08	87.46	**88.03**	**89.24**	**94.56**	**95.55**	**93.96**	**94.48**	**94.19**	**93.93**	**91.96**	**92.13**

Table 3 The ovreall performance (%) of target domains (U1–U5) on UAH dataset. The best and second-best performances are indicated in bold and underlined.

Method	U1		U2		U3		U4		U5		mAcc	mF1
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1		
CNN	81.82	77.17	46.87	48.89	59.79	61.12	78.94	79.88	76.61	74.28	68.81	68.27
VAE	84.30	83.98	68.56	65.55	67.17	69.23	79.46	80.49	71.62	76.31	74.22	75.11
DeepConvLSTM	91.75	91.76	85.16	84.70	81.40	78.61	91.23	91.82	92.52	92.84	88.41	87.95
DANN	87.51	88.15	68.75	70.37	72.90	69.41	85.74	86.37	81.59	80.52	79.30	78.96
CoDATS	90.48	90.33	82.37	83.21	84.78	83.56	89.14	88.47	88.77	88.31	87.11	86.78
CALDA	91.15	89.28	83.72	84.39	86.27	84.68	90.78	88.03	89.57	87.78	88.30	86.83
Tri-Net	93.69	92.36	84.23	83.00	82.66	85.61	93.13	92.24	91.60	90.67	89.06	88.78
AdaptNet	95.96	93.14	87.85	89.47	89.37	90.40	93.07	95.27	95.17	95.07	92.28	92.73
DGSSL (ours)	**97.97**	**96.92**	**89.82**	**91.46**	**92.77**	**91.79**	**95.15**	**96.46**	**95.38**	**94.64**	**94.22**	**94.25**

Table 4 The ovreall performance (%) of target domains (H1–H5) on UCIHAR dataset. The best and second-best performances are indicated in bold and underlined.

Method	H1		H2		H3		H4		H5		mAcc	mF1
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1		
CNN	57.27	51.53	43.45	42.66	64.56	68.60	41.28	36.26	39.96	43.32	49.30	48.47
VAE	62.03	58.81	47.43	50.46	63.35	70.12	43.89	44.39	51.71	49.68	53.68	54.69
DeepConvLSTM	72.09	69.54	**83.06**	**82.64**	78.93	83.50	73.90	79.32	83.96	85.45	78.39	80.09
DANN	78.32	75.62	72.97	67.66	75.95	76.60	80.49	78.49	74.44	76.08	76.43	74.89
CoDATS	82.77	78.12	61.88	65.43	79.75	75.20	63.51	61.30	67.79	68.45	71.14	69.70
CALDA	83.54	80.34	63.58	66.38	74.40	79.08	68.57	71.71	74.74	73.27	72.97	74.16
Tri-Net	81.78	79.62	69.02	70.77	78.09	81.11	83.68	84.92	78.86	79.31	78.29	79.15
AdaptNet	84.37	85.31	78.73	80.89	83.30	85.92	91.47	89.39	85.65	87.04	84.70	85.71
DGSSL (ours)	**86.87**	**85.58**	81.78	82.29	**84.27**	**86.75**	**92.72**	**90.56**	**88.14**	**89.09**	**86.76**	**86.85**

was underlined. DGSSL exhibited the best average performance on all three datasets, yielding the best performance in at least four out of the five scenarios in each dataset. For the scenarios in which our method did not achieve the best performance, it showed no more than 1.5% lower F1-score compared to the state-of-the-art method. The experimental results on each dataset will be discussed and analyzed below.

• **Results on the TMD dataset.** In Table 2, DGSSL achieved the best performance in four out of five scenarios on the TMD dataset. In the first scenario, our method exhibited only 0.58% Acc and 1.17% F1-score, which were lower than those for AdaptNet, which showed the best performance with 89.66% Acc and 88.63% F1. The domain adaptation-based methods (i.e., DANN, CoDATS, and CALDA) generally performed better than the deep learning-based approaches (i.e., CNN and VAE), presenting a maximum gap of 31.82% mF1 (CALDA vs. CNN). However, for some special cases (e.g., DeepConvLSTM), mF1 was up to 87.79%, a value that was higher than those of all domain adaptation-based methods. It may be true that the feature dimension of the TMD dataset is only 9; hence, DeepConvLSTM, which is dedicated to learning powerful features, contributes a better classification performance.

Table 5 The ablation study of the different components in the first scenario (T9, U1 and H1) of each dataset. Model1 represents the model consisting of feature extractor f_e and classifiers $\{f_c^d, f_c^c\}$ with $\mathcal{L}_{\text{cls}}$, and Model2 is a model based on Model1 with adversarial training module. Model6 is our proposed model DGSSL, whose variants Model3, Model4, and Model5 exclude the adversarial training, sequence reconstruction and latent representation reconstruction modules, respectively.

Model	Ablation				TMD		UAH		UCIHAR	
	$\mathcal{L}_{\text{adv}}$	$\mathcal{L}_{\text{rec}}$	$\mathcal{L}_{\text{lat}}$	$\mathcal{L}_{\text{cls}}$	Acc	F1	Acc	F1	Acc	F1
Model1				✓	81.07	80.28	88.14	87.91	77.48	75.21
Model2	✓			✓	83.44	81.19	91.02	90.81	80.36	78.71
Model3		✓	✓	✓	86.55	85.01	96.00	93.32	84.09	82.60
Model4	✓		✓	✓	85.04	82.82	93.42	92.12	81.67	79.56
Model5	✓	✓		✓	85.16	84.11	94.59	94.23	82.17	81.03
Model6	✓	✓	✓	✓	**89.08**	**87.46**	**97.97**	**96.92**	**86.87**	**85.58**

• **Results on the UAH dataset.** Table 3 shows the experimental results in five scenarios on the UAH dataset. DGSSL achieved the best performance in all scenarios. The semi-supervised learning-based methods (i.e., AdaptNet and Tri-Net) generally outperformed the other two methods. The potential reason for this result was that the original data of the UAH dataset contained 36 channels, which prevented methods dedicated to extracting robust features from effective learning patterns. Note that Tri-Net only obtained results similar to DeepConvLSTM, even after resorting to unlabeled data $\mathcal{D}_{\text{au}}$. In contrast, our method and AdaptNet obtained better results by utilizing SSL while considering the domain shift and working to mitigate this drift.

• **Results on the UCIHAR dataset.** Table 4 presents the results of five domains (H1–H5) on the UCIHAR dataset, wherein DGSSL ranked first in four scenarios in terms of F1 and Acc by obtaining slightly higher values compared to AdaptNet (i.e., 2.05% and 1.14%, respectively). In the second scenario, DeepConvLSTM achieved the best performance in extracting spatiotemporal features. This result suggests that applying DA and SSL methods with lower sampling rates may be challenging compared to methods dedicated to extracting powerful features.

Overall, our proposed DGSSL method outperformed all baseline methods. The experimental results favorably demonstrated that DGSSL can generalize from multiple source domains to the unseen target domain by reducing the domain-specific discrepancy and enhancing the reconstruction task-specific consistency.

4.5 Ablation study of components

We verified the efficacy of the proposed model's components by conducting an ablation study of each component on all three datasets. The DGSSL variants are defined as follows.

• Model1 ($\mathcal{L}_{\text{cls}}$). A model that contains only the classifier and the feature extractor f_e to verify the efficacy of the adversarial training, sequence reconstruction, and latent representation reconstruction modules.

• Model2 ($\mathcal{L}_{\text{cls}}$ w/ $\mathcal{L}_{\text{adv}}$). A model with only the adversarial training module based on Model1. This model was used to verify the efficacy of the other two reconstruction modules.

• Model3 ($\mathcal{L}_{\text{all}}$ w/o $\mathcal{L}_{\text{adv}}$). A model that excluded the autoregressive discriminator from the proposed DGSSL and was mainly used to verify the performance of the autoregressive discriminator or reconstruction module.

• Model4 ($\mathcal{L}_{\text{all}}$ w/o $\mathcal{L}_{\text{rec}}$). A model implemented after excluding the sequence reconstruction module from DGSSL and used to verify the performance of the sequence reconstruction module.

• Model5 ($\mathcal{L}_{\text{all}}$ w/o $\mathcal{L}_{\text{lat}}$). A model similar to Model4 (i.e., a full DGSSL model that excluded the latent representation reconstruction module) used to verify the performance of the latent representation reconstruction module.

• Model6 ($\mathcal{L}_{\text{all}}$). A model containing all components (i.e., our proposed model, DGSSL).

Table 5 shows the evaluation results of these models in the first scenario (T9, U1, and H1) of each dataset. Model1 exhibited the largest performance degradation, yielding 8.01%, 9.83%, and 9.39% Acc on the TMD, UAH, and UCIHAR datasets, respectively. This performance degradation was basically consistent in terms of the F1-score. The results indicate the effectiveness of the joint training of multiple modules, which did not trap the model into a local optimum but collectively contributed to the abovementioned performance gap.

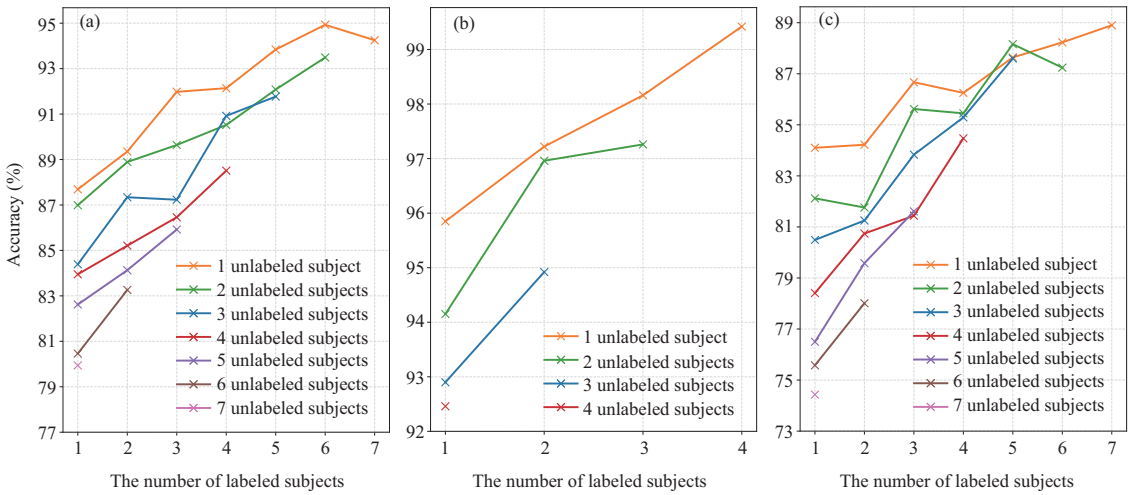

Figure 5 (Color online) Ablation study results of the labeled and unlabeled subjects. We selected eight, five, and eight subjects for training. T9, U1, and H1 were set as the target domains from the (a) TMD, (b) UAH, and (c) UCIHAR datasets, respectively. The X-axis represents the number of labeled subjects. The colored lines indicate the different numbers of unlabeled subjects.

Model2 showed a performance improvement of approximately 3% Acc and F1-score over Model1 on every dataset. The reconstruction modules can contribute at least a 5% boost compared to Model6, demonstrating the importance of the adversarial training module for mitigating the domain shift.

Model3 removed the adversarial training module from the full model, placing it at the second rank (i.e., the second-lowest drop in performance). This model showed a performance drop of approximately 3% Acc and F1-score compared to Model6. This result was almost identical to the conclusion on the efficacy of the reconstruction modules in Model2. The outcome argued that the reconstruction tasks did not bring much improvement when the domain shift was not considered.

We separately removed the sequence and latent representation reconstructions from Model6, which significantly affected the performance and placed the ranking at the fourth and third places, respectively. Compared to that of Model5, the performance of Model4 slightly decreased by approximately 1% Acc and F1-score, showing the importance of the reconstruction task for improving representation learning.

4.6 Sensitivity to labeled and unlabeled subjects

Unlike many SSL methods that perform an ablation study on the labeled:unlabeled data ratio, we investigated herein the performance variation of our proposed method with different numbers of labeled and unlabeled subjects. As described in Subsection 4.3, we evenly divided the training set into the unlabeled $\mathcal{D}_{\text{au}}$ and labeled $\mathcal{D}_{\text{al}}$ source domain sets. That is, we considered that $\mathcal{D}_{\text{au}}$ and $\mathcal{D}_{\text{al}}$ obey two different distributions. However, different subjects actually obey different distributions; hence, both $\mathcal{D}_{\text{au}}$ and $\mathcal{D}_{\text{al}}$ contain multiple distributions. The problem turns into an investigation of the sensitivity of our proposed model in this case.

In the training phase, we increased the number of labeled subjects from one to seven for the TMD and UCIHAR datasets and one to four for the UAH dataset. Likewise, we increased the number of unlabeled subjects in the same manner until the sum of the labeled and unlabeled subjects did not exceed eight, five, and eight on the TMD, UAH, and UCIHAR datasets, respectively. This was done so that $\mathcal{D}_{\text{au}}$ and $\mathcal{D}_{\text{al}}$ do not have overlapping data in the settings and to avoid violating the overall domain shift principle.

Figure 5 depicts the experimental results, wherein the X-axis represents the number of labeled subjects and different colors denote different numbers of unlabeled subjects. The performance gradually improved as the number of labeled subjects increased. In contrast, the performance decreased as the number of unlabeled subjects increased. This result was consistent with the view in [65] that argued the following: labeled data can improve the distribution generalization ability, while significantly scattered unlabeled data may confuse the model. We present in Appendix A of the Supporting information more analyzes on models 1–6.

Table 6 The performance (%) of parameter sensitivity analysis for weight coefficients λ_{adv}, λ_{rec} and λ_{lat} on three datasets. * denotes the default value of the proposed model, and the best values are indicated in bold.

Parameter	Value	TMD		UAH		UCIHAR	
		Acc	F1	Acc	F1	Acc	F1
λ_{adv}	10^{-3}	88.32	88.86	89.82	89.41	80.98	81.00
	10^{-2}	91.84	90.98	92.81	92.24	83.96	83.74
	10^{-1*}	**92.98**	**92.13**	94.22	94.25	**86.76**	**86.45**
	10^{0}	91.15	91.61	**94.71**	**94.97**	85.43	84.87
	10^{1}	84.73	84.93	85.29	84.94	79.12	78.92
λ_{rec}	0	88.78	88.62	90.45	90.24	80.51	81.16
	1	90.45	90.97	93.83	**94.39**	84.42	84.59
	2^{*}	**92.98**	**92.13**	**94.22**	94.25	**86.76**	**86.45**
	3	91.02	90.40	93.01	93.26	84.12	84.98
	4	90.57	90.35	91.64	91.91	82.59	83.16
	5	90.51	89.95	91.94	92.01	84.73	84.40
λ_{lat}	0	89.32	89.36	90.97	90.88	82.63	82.27
	1^{*}	**92.98**	92.13	**94.22**	**94.25**	86.76	**86.45**
	2	92.16	**92.71**	93.84	93.89	**87.11**	86.17
	3	90.07	90.49	91.33	91.18	85.47	84.96
	4	88.82	88.42	89.67	89.87	83.66	83.83
	5	89.54	89.83	92.34	91.96	85.37	85.41

4.7 Parameter sensitivity analysis of the weight coefficients

The three key parameters, namely, λ_{adv}, λ_{rec}, and λ_{lat}, of DGSSL may have significant effects on the model performance. We investigated herein the effect of these key parameters on the performance. Table 6 shows the results of the weight coefficients on the three datasets. λ_{adv} denotes the adversarial intensity of the domain. λ_{rec} and λ_{lat} denote the trade-off for the sequence and latent representation losses of the reconstruction tasks, respectively. We set the parameter values from 10^{-3} to 10^{1}, 0 to 5, and 0 to 5, respectively, and marked the default values of the model with *.

The default values of the model clearly yielded the best performance in most instances. For example, λ_{adv} showed the best performance at 10^{-1} or 10^{0}, remained stable between 10^{-2} and 10^{0}, and then sharply dropped at 10^{0}. This result may be attributed to the fact that an excessively weak f_a does not significantly help the model, and an excessively strong adversarial ability reduces the gap between the domains but may distract the model from the reconstruction and classification tasks.

The value for parameter λ_{rec} only exhibited a small drop at 0. The performance did not significantly fluctuate between 1 and 5, demonstrating the effectiveness of the sequence reconstruction task. Our model was insensitive to the changes in λ_{rec}. Based on the results presented in rows two and three of Table 6 show that λ_{lat} played an almost equally important role as λ_{rec}, wherein the performance variations between the two were quite consistent. Therefore, we inferred that a strong reconstruction capability was crucial to the representation learning of the proposed DGSSL.

4.8 Visualization analysis

We selected two subjects as the target domains for testing to better understand the effectiveness of our proposed model. Figure 6 illustrates the visualization results of the latent representation via t-SNE [66].

The first row in Figure 6 shows the classification results of each dataset. Different colors depict different categories. In all datasets, the aggregation within a cluster was significant, while the distance between the clusters was large. Samples with the same activity tended to converge into the same cluster, making the number of clusters learned by DGSSL exactly equal to the number of categories on each dataset. The UAH dataset showed the best classification effect when compared to the TMD and UCIHAR datasets, which was consistent with the experimental results in Tables 2–4.

The second row in Figure 6 shows the visualization results of Domain1 and Domain2. Domain1 is represented by the pink triangles, while Domain2 is denoted by the gray squares. We can distill the following observation: the mixing between Domain1 and Domain2 was significant. DGSSL provided a very efficient domain mixture while maintaining category aggregation. The model's latent representation

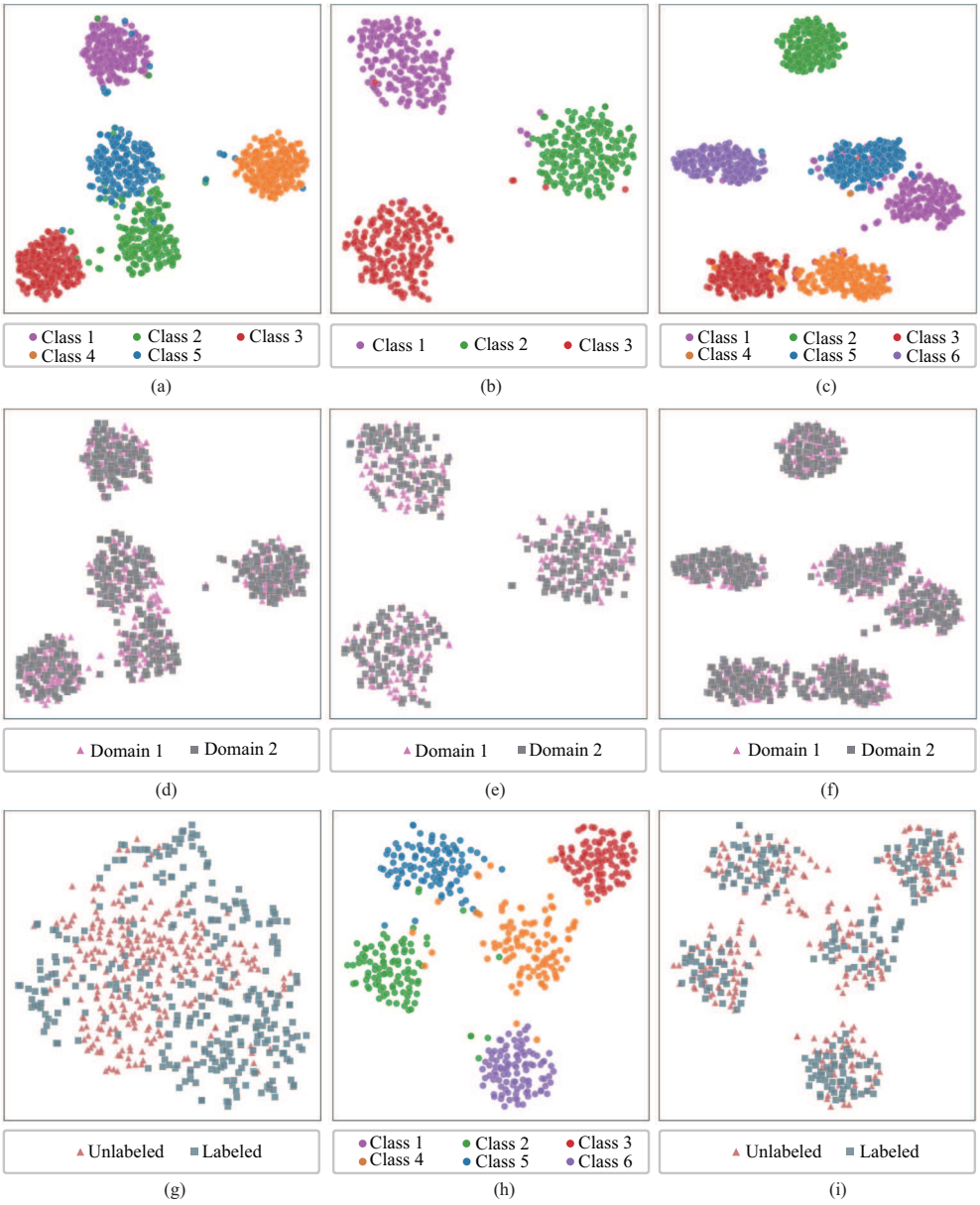

Figure 6 (Color online) Visualization of latent representations through t-SNE on TMD, UAH, and UCIHAR datasets. The first to third rows show the results of the target domain classifications, the mixture of two domains, and the classifications of labeled and unlabeled domains, respectively. Best viewed in color. (a) TMD classification; (b) UAH classification; (c) UCIHAR classification; (d) TMD domain; (e) UAH domain; (f) UCIHAR domain; (g) TMD raw; (h) TMD classification; (i) TMD domain.

was not affected by domain-specific factors.

We better visualized the differences between the latent representations $\boldsymbol{z}^U$ and $\boldsymbol{z}^L$ by excluding a significant portion (80%) of the T7 and T8 training data from the TMD dataset to serve as the labeled and unlabeled testing domains. The visualization results of the original data and the latent representations are presented in the third row of Figure 6. The original data were nearly inseparable, while the latent representations extracted by f_e achieved better classification results. The classification of the two domains in Figure 6(i) indicated that the labeled domain was more compact than the unlabeled domain (light red) (i.e., the latter was more scattered). The reason for this finding may be that labeled instances often contain additional information that is relevant to the labels, which the model can leverage to learn better and represent the data.

In summary, our proposed method, DGSSL, has good generalization across multiple datasets. It can efficiently classify activities and match samples from different domains.

5 Conclusion

This study introduced the novel DGSSL method for people-centric activity recognition. The proposed DGSSL approach disentangles domain- and task-specific features via a collaborative optimization of adversarial learning and reconstruction tasks that empower the model to improve the understanding of distribution gaps while maintaining the ability to represent the inherent characteristics of a specific distribution. In an adversarial training manner, we first performed the domain alignment using a well-designed autoregressive discriminator to alleviate the domain shift problem caused by different people. We supported the classification task better by further considering the task consistency of the same activity from different domains through the sequence and latent representation reconstruction tasks. Consequently, the model benefited from the reconstruction tasks and avoided the loss of information related to representation learning. The results of the extensive experiments on the three activity recognition datasets showed that our proposed DGSSL outperforms the three state-of-the-art methods with better performance and generalization.

We believe that our method can promote activity recognition and time-series processing communities but may still suffer from limitations. For example, DGSSL assumes the existence of relatively sufficient and available labeled data, which is usually laborious in real-world applications. To this end, our future work will include the design of a representation learning method that only needs to learn a few labeled data and a large amount of unlabeled data, with a performance that approximates or even surpasses that of supervised learning-based methods.

Acknowledgements This work was supported in part by China Mobile Research Fund of the Chinese Ministry of Education (Grant No. KEH2310029), Specific Research Fund of the Innovation Platform for Academicians of Hainan Province (Grant No. YSPTZX202314). The work was also supported by the Shanghai Key Research Laboratory of NSAI and the Joint Laboratory on Networked AI Edge Computing Fudan University-Changan. We sincerely thank all the editors and anonymous reviewers for their careful work and thoughtful suggestions that have helped improve this paper substantially.

Supporting information Appendix A. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Song L, Hu X, Zhang G, et al. Networking systems of AI: on the convergence of computing and communications. IEEE Int Things J, 2022, 9: 20352–20381
2 Yang K, Liu J, Yang D, et al. A novel efficient multi-view traffic-related object detection framework. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, 2023. 1–5
3 Nweke H F, Teh Y W, Al-garadi M A, et al. Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: state of the art and research challenges. Expert Syst Appl, 2018, 105: 233–261
4 You X, Zhang C, Tan X, et al. AI for 5G: research directions and paradigms. Sci China Inf Sci, 2018, 62: 21301
5 Li C, Song L. GCN-LSTM for EEG classification based on unspoken speech of bilinguals. In: Proceedings of the 24th International Conference on Digital Signal Processing (DSP), Rhodes, 2023. 1–4
6 Sena J, Barreto J, Caetano C, et al. Human activity recognition based on smartphone and wearable sensors using multiscale DCNN ensemble. Neurocomputing, 2021, 444: 226–243
7 Yu H, Chen Z, Zhang X, et al. FedHAR: semi-supervised online learning for personalized federated human activity recognition. IEEE Trans Mobile Comput, 2023, 22: 3318–3332
8 Gao J, Zhang Y, Zheng Z, et al. Ecological engineering projects shifted the dominance of human activity and climate variability on vegetation dynamics. Remote Sens, 2022, 14: 2386
9 Liu Y, Liu J, Yang K, et al. AMP-Net: appearance-motion prototype network assisted automatic video anomaly detection system. IEEE Trans Ind Inf, 2023, doi: 10.1109/TII.2023.3298476
10 Liu J, Liu Y, Tian C, et al. A survey of recent advances in driving behavior analysis. In: Proceedings of the 3rd International Symposium on Smart and Healthy Cities (ISHC), Toronto, 2021. 145–157
11 Liu Y, Liu J, Zhu X, et al. Learning task-specific representation for video anomaly detection with spatial-temporal attention. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 2022. 2190–2194
12 Gu F, Chung M H, Chignell M, et al. A survey on deep learning for human activity recognition. ACM Comput Surv, 2022, 54: 1–34
13 Liu Y, Liu J, Zhao M, et al. Collaborative normality learning framework for weakly supervised video anomaly detection. IEEE Trans Circ Syst II, 2022, 69: 2508–2512
14 Xiang X, Liu Y, Fang G, et al. Two-stage alignments framework for unsupervised domain adaptation on time series data. IEEE Signal Process Lett, 2023, 30: 698–702
15 Chen D D, Wang W, Gao W, et al. Tri-net for semi-supervised deep learning. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, Nanjing, 2018. 2014–2020
16 Lv M, Chen L, Chen T, et al. Bi-view semi-supervised learning based semantic human activity recognition using accelerometers. IEEE Trans Mobile Comput, 2018, 17: 1991–2001
17 An S, Medda A, Sawka M N, et al. AdaptNet: human activity recognition via bilateral domain adaptation using semi-supervised deep translation networks. IEEE Sens J, 2021, 21: 20398–20411
18 Saito K, Kim D, Saenko K. Openmatch: open-set semi-supervised learning with open-set consistency regularization. In: Proceedings of the Advances in Neural Information Processing Systems, 2021. 25956–25967

19 Zhou K, Liu Z, Qiao Y, et al. Domain generalization: a survey. IEEE Trans Pattern Anal Mach Intell, 2023, 45: 4396–4415
20 Meftah S, Semmar N, Tahiri M A, et al. Multi-task supervised pretraining for neural domain adaptation. In: Proceedings of the 8th International Workshop on Natural Language Processing for Social Media, 2020. 61–71
21 Wilson G, Doppa J R, Cook D J. CALDA: improving multi-source time series domain adaptation with contrastive adversarial learning. IEEE Trans Pattern Anal Mach Intell, 2023, 45: 14208–14221
22 Hu L, Kan M, Shan S, et al. Duplex generative adversarial network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 18–23
23 Liu Y, Yang D, Wang Y, et al. Generalized video anomaly event detection: systematic taxonomy and comparison of deep models. 2023. ArXiv:2302.05087
24 Ganin Y, Ustinova E, Ajakan H, et al. Domain-adversarial training of neural networks. J Mach Learn Res, 2016, 17: 2096–2030
25 Wilson G, Doppa J R, Cook D J. 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, 2020. 1768–1778
26 Tolstikhin I O, Sriperumbudur B K, Schölkopf B. Minimax estimation of maximum mean discrepancy with radial kernels. In: Proceedings of the Advances in Neural Information Processing Systems, Barcelona, 2016. 1938–1946
27 Ji Z, Yan J T, Wang Q, et al. Triple discriminator generative adversarial network for zero-shot image classification. Sci China Inf Sci, 2021, 64: 120101
28 Bousmalis K, Trigeorgis G, Silberman N, et al. Domain separation networks. In: Proceedings of the Advances in Neural Information Processing Systems, Barcelona, 2016
29 Recht B, Roelofs R, Schmidt L, et al. Do imagenet classifiers generalize to imagenet? In: Proceedings of the 36th International Conference on Machine Learning, 2019. 5389–5400
30 Wang J, Lan C, Liu C, et al. Generalizing to unseen domains: a survey on domain generalization. IEEE Trans Knowl Data Eng, 2023, 35: 8052–8072
31 Wang Y, Song W, Tao W, et al. A systematic review on affective computing: emotion models, databases, and recent advances. Inf Fusion, 2022, 83: 19–52
32 Yao S, Hu S, Zhao Y, et al. Deepsense: a unified deep learning framework for time-series mobile sensing data processing. In: Proceedings of the 26th International Conference on World Wide Web, Perth, 2017. 351–360
33 Liu J, Liu Y, Li D, et al. DSDCLA: driving style detection via hybrid CNN-LSTM with multi-level attention fusion. Appl Intell, 2023, 53: 19237–19254
34 Mutegeki R, Han D S. A CNN-LSTM approach to human activity recognition. In: Proceedings of the International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Fukuoka, 2020. 362–366
35 Liu J, Liu Y, Tian C, et al. Multi-level attention fusion for multimodal driving maneuver recognition. In: Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS), Austin, 2022. 2609–2613
36 Hammerla N Y, Halloran S, Plötz T. Deep, convolutional, and recurrent models for human activity recognition using wearables. In: Proceedings of the 25th International Joint Conference on Artificial Intelligence, 2016. 1533–1540
37 Ma H, Li W, Zhang X, et al. AttnSense: multi-level attention mechanism for multimodal human activity recognition. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence, 2019. 3109–3115
38 Wang Y, Sun Y, Song W, et al. DPCNet: dual path multi-excitation collaborative network for facial expression representation learning in videos. In: Proceedings of the 30th ACM International Conference on Multimedia, 2022. 101–110
39 Liu Y, Liu J, Lin J, et al. Appearance-motion united auto-encoder framework for video anomaly detection. IEEE Trans Circ Syst II, 2022, 69: 2498–2502
40 Liu J, Liu Y, Lin J, et al. One-dimensional convolutional neural network model for abnormal driving behaviors detection using smartphone sensors. In: Proceedings of the International Conference on Networking Systems of AI (INSAI), 2021. 143–150
41 Lim K, Lee J Y, Carbonell J, et al. Semi-supervised learning on meta structure: multi-task tagging and parsing in low-resource scenarios. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2020. 8344–8351
42 Feng X, Qin B, Liu T. A language-independent neural network for event detection. Sci China Inf Sci, 2018, 61: 092106
43 Sohn K, Berthelot D, Carlini N, et al. FixMatch: simplifying semi-supervised learning with consistency and confidence. In: Proceedings of the Advances in Neural Information Processing Systems, 2020. 596–608
44 Shi C, Lv Z, Yang X, et al. Hierarchical multi-view semi-supervised learning for very high-resolution remote sensing image classification. Remote Sens, 2020, 12: 1012
45 Hu Y, An R, Wang B, et al. Shape adaptive neighborhood information-based semi-supervised learning for hyperspectral image classification. Remote Sens, 2020, 12: 2976
46 Zhang X, Yao L, Yuan F. Adversarial variational embedding for robust semi-supervised learning. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019. 139–147
47 Chen Z, Zhang X, Cheng X. ASM2TV: an adaptive semi-supervised multi-task multi-view learning framework for human activity recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2022. 6342–6349
48 An S, Gazi A H, Inan O T. DynaLAP: human activity recognition in fixed protocols via semi-supervised variational recurrent neural networks with dynamic priors. IEEE Sens J, 2022, 22: 17963–17976
49 Qin Z, Zhang Y, Meng S, et al. Imaging and fusing time series for wearable sensor-based human activity recognition. Inf Fusion, 2020, 53: 80–87
50 Liu J, Liu Y, Donglai W, et al. Attention-based auto-encoder framework for abnormal driving detection. In: Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS), 2022. 3150–3154
51 Khan M A A H, Roy N, Misra A. Scaling human activity recognition via deep learning-based domain adaptation. In: Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom), 2018. 1–9
52 Long M, Cao Y, Wang J, et al. Learning transferable features with deep adaptation networks. In: Proceedings of the 32nd International Conference on Machine Learning, 2015. 97–105
53 Tzeng E, Hoffman J, Zhang N, et al. Deep domain confusion: maximizing for domain invariance. 2014. ArXiv:1412.3474
54 Chen C, Fu Z, Chen Z, et al. HoMM: higher-order moment matching for unsupervised domain adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2020. 3422–3429
55 Ragab M, Eldele E, Chen Z H, et al. Self-supervised autoregressive domain adaptation for time series data. IEEE Trans Neural Netw Learn Syst, 2022, doi: 10.1109/TNNLS.2022.3183252
56 Fuglede B, Topsoe F. Jensen-shannon divergence and hilbert space embedding. In: Proceedings of the International Symposium onInformation Theory, 2004
57 Carpineti C, Lomonaco V, Bedogni L, et al. Custom dual transportation mode detection by smartphone devices exploiting sensor diversity. In: Proceedings of the IEEE International Conference on Pervasive Computing and Communications

Workshops (PerCom Workshops), 2018. 367–372
58 Romera E, Bergasa L M, Arroyo R. Need data for driver behaviour analysis? Presenting the public uah-driveset. In: Proceedings of the 19th International Conference on Intelligent Transportation Systems (ITSC), 2016. 387–392
59 Anguita D, Ghio A, Oneto L, et al. Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In: Proceedings of International Workshop on Ambient Assisted Living, 2012. 216–223
60 Savelonas M, Vernikos I, Mantzekis D, et al. Hybrid representation of sensor data for the classification of driving behaviour. Appl Sci, 2021, 11: 8574
61 Kingma D P, Welling M. Auto-encoding variational bayes. 2014. ArXiv:1312.6114
62 Ordóñez F, Roggen D. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. Sensors, 2016, 16: 115
63 Paszke A, Gross S, Massa F, et al. PyTorch: an imperative style, high-performance deep learning library. In: Proceedings of the Advances in Neural Information Processing Systems, 2019
64 Liu J, Liu Y, Zhu W, et al. Distributional and spatial-temporal robust representation learning for transportation activity recognition. Pattern Recogn, 2023, 140: 109568
65 Chan A, Alaa A, Qian Z, et al. Unlabelled data improves bayesian uncertainty calibration under covariate shift. In: Proceedings of the 37th International Conference on Machine Learning, 2020. 1392–1402
66 van der Maaten L, Hinton G. Visualizing data using t-SNE. J Mach Learn Res, 2008, 9: 2579–2605