Unsupervised Domain Adaptation by Statistics Alignment for Deep Sleep Staging Networks

Jiahao Fan, Hangyu Zhu, Xinyu Jiang, Long Meng, *Graduate Student Member, IEEE*,
Chen Chen, Cong Fu, Huan Yu, Chenyun Dai, and Wei Chen, *Senior Member, IEEE*

***Abstract*— Deep sleep staging networks have reached top performance on large-scale datasets. However, these models perform poorer when training and testing on small sleep cohorts due to data inefficiency. Transferring well-trained models from large-scale datasets (source domain) to small sleep cohorts (target domain) is a promising solution but still remains challenging due to the domain-shift issue. In this work, an unsupervised domain adaptation approach, domain statistics alignment (DSA), is developed to bridge the gap between the data distribution of source and target domains. DSA adapts the source models on the target domain by modulating the domain-specific statistics of deep features stored in the Batch Normalization (BN) layers. Furthermore, we have extended DSA by introducing cross-domain statistics in each BN layer to perform DSA adaptively (AdaDSA). The proposed methods merely need the well-trained source model without access to the source data, which may be proprietary and inaccessible. DSA and AdaDSA are universally applicable to various deep sleep staging networks that have BN layers. We have validated the proposed methods by extensive experiments on two state-of-the-art deep sleep staging networks, DeepSleepNet+ and U-time. The performance was evaluated by conducting various transfer tasks on six sleep databases, including two large-scale databases, MASS and SHHS, as the source domain, four small sleep databases as the target domain. Thereinto, clinical sleep records acquired in Huashan Hospital, Shanghai, were used. The results show that both DSA and AdaDSA could significantly improve the performance of source models on target domains, providing novel insights into the domain generalization problem in sleep staging tasks.**

***Index Terms*— Unsupervised domain adaptation, deep learning, sleep staging, transfer learning, batch normalization.**

I. Introduction

SLEEP staging is an essential process in sleep disorder diagnosis. In clinical practice, whole-night polysomnography (PSG) is analyzed by sleep technicians to label each sleep duration of 30 seconds to one of either six (according to R&K rules [1]) or five (according to the AASM guideline [2]) sleep stages. The manual scoring process is expensive and time-consuming, taking a scorer hours for annotating one PSG record. Also, manual scoring is prone to human errors due to subjectivity. An average inter-rater sleep scoring agreement of 82.6% over more than 2500 scorers is reported in [3]. With the soaring development of machine learning, automatic sleep staging [4]–[10] and sleep disorders assessment techniques [11]–[13] have progressed significantly in recent years. In particular, extensive works based on deep learning have reached human-performance in sleep staging tasks.

Despite the top-performance achieved, deep sleep staging models have not been widely accepted in clinical practice. The primary reason is two-fold. First, there is no sufficient labeled training data in many cases. The training of deep sleep staging models is largely supervised and data-driven, requiring a large number of labeled training samples to establish the map between physiological signals and sleep stage labels. The performance of deep sleep staging models relies on the size of available training samples. In previous studies, a poorer performance was observed when trained the same network on a small sleep database than on large sleep cohorts [4], [9], [14]. Unfortunately, large-scale annotated sleep data is not always attainable for many sleep studies. Second, it is feasible to predict unseen sleep records (target domain) using models trained on large databases (source domain). However, due to different acquisition settings, the properties of sleep samples on target domains inevitably differ from those on the source domain. This domain-shift issue leads to performance

Manuscript received September 16, 2021; revised December 7, 2021; accepted January 2, 2022. Date of publication January 18, 2022; date of current version January 31, 2022. This work was supported in part by the Shanghai Municipal Science and Technology Major Project under Grant 2017SHZDZX01, in part by the Shanghai Municipal Science and Technology International Research and Development Collaboration Project under Grant 20510710500, in part by the National Natural Science Foundation of China under Grant 62001118, and in part by the Shanghai Committee of Science and Technology under Grant 20S31903900. *(Corresponding authors: Wei Chen; Chen Chen; Cong Fu.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Institutional Review Board of Huashan Hospital, Fudan University, under Application No. 2021-811.

Jiahao Fan and Wei Chen are with the Center for Intelligent Medical Electronics, School of Information Science and Technology, Fudan University, Shanghai 200433, China, and also with the Human Phenome Institute, Fudan University, Shanghai 201203, China (e-mail: w_chen@fudan.edu.cn).

Hangyu Zhu, Xinyu Jiang, Long Meng, and Chenyun Dai are with the Center for Intelligent Medical Electronics, School of Information Science and Technology, Fudan University, Shanghai 200433, China.

Chen Chen is with the Human Phenome Institute, Fudan University, Shanghai 201203, China (e-mail: chenchen_fd@fudan.edu.cn).

Cong Fu and Huan Yu are with the Sleep and Wake Disorders' Center, Department of Neurology, Huashan Hospital, Shanghai Medical College, Fudan University, Shanghai 200040, China (e-mail: fucong@fudan.edu.cn).

Digital Object Identifier 10.1109/TNSRE.2022.3144169

deterioration when directly applying models trained on a source domain to target domains.

Several approaches have been proposed to adapt the source models to unseen target domains. Supervised transfer learning is the most frequently used approach. Supervised transfer learning approaches generally adapt the well-trained source models on the target domain by fine-tuning the network parameters using the available annotated samples on the target domains [4], [15]–[18]. The target domain could either be a collection of sleep records from a clinical site [4], [15], [16] or sleep records from a particular individual (personalized transfer) [17], [18]. In particular, Phan *et al.* [4] conducted a comprehensive transfer learning study, providing an insight into the transferability of the deep features learned from sleep staging networks. However, supervised fine-tune requires many sleep samples to be labeled on the target domain, which is not always applicable.

From a realistic perspective, semi-supervised or unsupervised approaches are preferable. Banluesombatkul *et al.* [14] proposed a transfer learning framework, MetaSleepLearner, based on Model-Agnostic Meta-Learning (MAML) [19]. By MetaSleepLearner, the network could be fast adapted to new individuals using only a few labeled data. However, the computation cost of Meta-Learning is expensive. In addition, the authors only validated the framework on a simple network. The efficacy of this framework on state-of-the-art networks still needs to be further validated. Unsupervised Domain adaptation approaches using adversarial learning were proposed to learn domain-invariant representations of sleep samples from different databases [20]–[22]. In these works, specific network architecture was designed and trained simultaneously using samples from source and target datasets. Domain classifiers were used to drive the network to learn domain-invariant features. These approaches can enhance the sleep staging performance on the target dataset without needing sleep annotations. Nevertheless, these methods are not model-agnostic, often rely on special design in network architecture, thus are hard to generalize to other networks. These adversarial methods contain several loss functions and converge slowly. Also, these methods require the sleep samples from the source domain, which are proprietary and inaccessible in most cases.

Different from all previous works, we adapted the source models on target domains by statistics alignment. Samples for training a deep sleep staging neural network are assumed *i.i.d*, but samples across data domains are not. The marginal distribution of sleep samples varies in different acquisition setups and individual peculiarities. The inconsistency in data distribution hinders the network trained on the source domain from performing well on the target domains. Inspired by the widespread Batch Normalization (BN) technique [23], we cope with the domain-shift issues by explicitly aligning the distributions of the intermediate features of neural networks. We hypothesize that the task-specific information is stored in network weights, whereas the domain-specific information is represented by the statistics accumulated in BN layers. Therefore, well-trained models can be adapted to new target domains by merely updating the statistics in BN layers. This method, we call domain statistics alignments (DSA), is easy to implement as a preprocessing step during model transfer. Furthermore, we have extended DSA by a second step, in which we introduce cross-domain distributions to tune the degree of the statistics alignment. This method, we call adaptive domain statistics alignment (AdaDSA), has contributed to considerable performance gains over DSA.

The main contributions of this work include:

- This work provides a new perspective on bridging the gap between the distribution of source and target domains in sleep staging tasks. We explicitly align the distribution of different domains by modulating the domain-specific statistics stored in the networks. The proposed methods are easy to implement without access to source data, and they can be universally applied to various deep sleep staging models with BN layers.
- We validated the effectiveness of our approach on two state-of-the-art sleep staging networks on various transfer tasks. We used two large-scale databases as the source domain and four small sleep databases as the target domains. Thereinto, practical clinical sleep records are taken into account. The results show that our approaches can significantly improve the performance of source models on target domains.
- We have compared the proposed methods with other transfer learning works. The obtained results further demonstrated the efficacy of proposed approaches. This work provides new ideas for generalizing deep sleep staging models on sleep records with different characteristics and can boost the development of practical automatic sleep staging applications.

II. Methodologies

The conceptual framework of the proposed methods is shown in Fig.1. Let $\mathcal{X}$ be the input space (*e.g.* sleep samples) and $\mathcal{Y}$ the output space (*e.g.* sleep stages). Let $\mathcal{S} = \{(x_s^1, y_s^1), (x_s^2, y_s^2) \ldots, (x_s^{n_s}, y_s^{n_s})\}$ and $\mathcal{T} = \{x_t^1, x_t^2 \ldots, x_t^{n_t}\}$ refer to the source domain samples and the unlabeled target domain, respectively. We first trained a source model $F_s(\boldsymbol{\theta}'; \boldsymbol{\mu}_s; \boldsymbol{\sigma}_s^2)$ on $\mathcal{S}$ via empirical risk minimization (Fig.1(a)), where $\boldsymbol{\theta}'$ is the network weights represented the map: $\mathcal{X} \rightarrow \mathcal{Y}$, and $\boldsymbol{\mu}_s$ and $\boldsymbol{\sigma}_s^2$ represent the domain-specific BN statistics on $\mathcal{S}$. The primary goal of DSA and AdaDSA is to modulate the statistics in model F_s on the target domain, to build a prediction model F_t, which can perform well on the target domain (Fig.1(b) and (c)). Since our work is highly related to BN technique, we briefly introduced BN technique below.

A. Batch Normalization

Known as internal covariance shift, the distribution of internal nodes of a neural network may change during training. Batch Normalization (BN) [23] reduces this shift by applying normalization on features in each mini-batch, thus could facilitate model convergence. BN is one of the most popular techniques in deep learning applications. It has been used in most recent top-performing deep-learning-based sleep staging models, such as DeepSleepNet [9], SeqSleepNet [5], and Utime [10], *etc.*

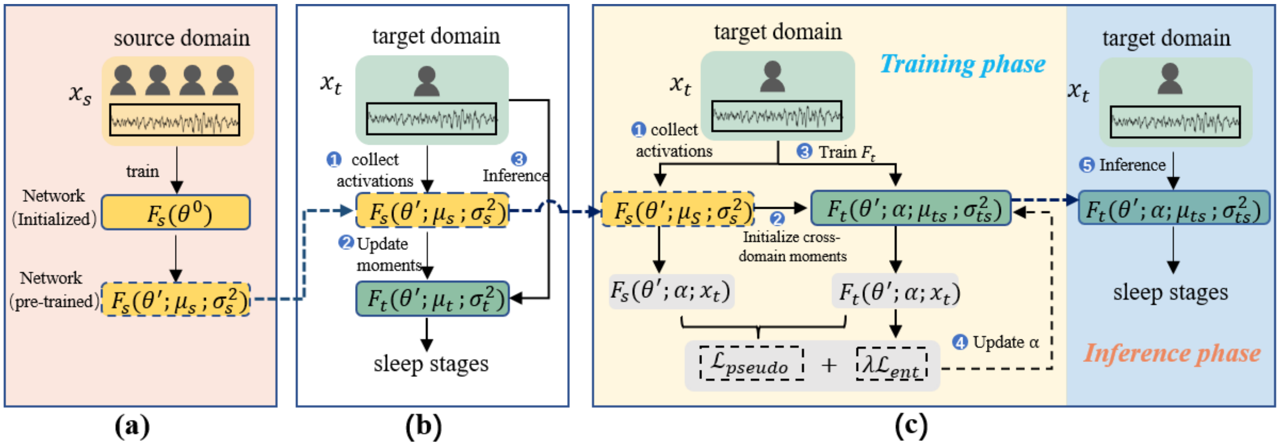
source domain
x_s
train
Network (Initialized)
$F_s(\theta^0)$
Network (pre-trained)
$F_s(\theta';\mu_s;\sigma_s^2)$
target domain
x_t
1 collect activations
3 Inference
$F_s(\theta';\mu_s;\sigma_s^2)$
2 Update moments
$F_t(\theta';\mu_t;\sigma_t^2)$
sleep stages
target domain
x_t
Training phase
1 collect activations
3 Train F_t
$F_s(\theta';\mu_s;\sigma_s^2)$
2 Initialize cross-domain moments
$F_t(\theta';\alpha;\mu_{ts};\sigma_{ts}^2)$
$F_s(\theta';\alpha;x_t)$
$F_t(\theta';\alpha;x_t)$
$\mathcal{L}_{pseudo}$ + $\lambda\mathcal{L}_{ent}$
4 Update α
target domain
x_t
5 Inference
$F_t(\theta';\alpha;\mu_{ts};\sigma_{ts}^2)$
sleep stages
Inference phase
(a)
(b)
(c)

Fig. 1. The conceptual frameworks of proposed methods. (a): Pre-training network on the source domain, the pre-trained model F_S is then transferred to the target domains. (b): Domain statistics alignment (DSA); (c): Adaptive Domain statistics alignment (AdaDSA).

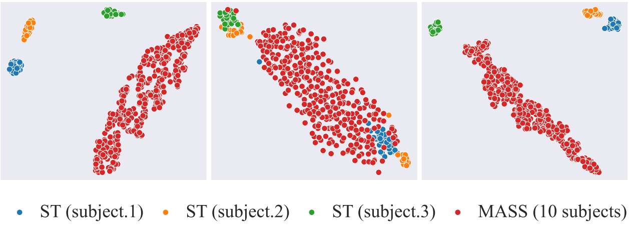

Fig. 2. The t-SNE visualization of statistics vectors from different domains on different network layers. Each point represents the statistics from a mini-batch. Samples from the MASS database are drawn from the test set. From left to right: networks layers from shallow to deep.

Given d-dimensional input $\boldsymbol{x} \in \mathbb{R}^d$ in a mini-batch. BN layer first normalizes $\boldsymbol{x}$ by making every scalar feature have the mean of 0 and the variance of 1.

$$\hat{\boldsymbol{x}}^{(d)} = \frac{\boldsymbol{x}^{(d)} - \mathbb{E}[\boldsymbol{x}^{(d)}]}{\sqrt{Var[\boldsymbol{x}^{(d)}]}} \tag{1}$$

A transformation is made to represent the identity transform for reserving the information that the network layers originally represents?

$$\boldsymbol{y}^{(d)} = \boldsymbol{\gamma}^{(d)}\hat{\boldsymbol{x}}^{(d)} + \boldsymbol{\beta}^{(d)} \tag{2}$$

Thereinto, $\beta^{(d)}$ and $\gamma^{(d)}$ are trainable parameters that can restore the representation power of the network. The global statistics, *i.e.*, mean value $\mu^{(d)}$ and variance value $(\sigma^{(d)})^2$ of each feature scalar are estimated during training by the exponential moving average across all training samples by:

$$\begin{aligned}\boldsymbol{\mu}^{(d)} &= m\boldsymbol{\mu}^{(d)} + (1-m)\mathbb{E}[\boldsymbol{x}^{(d)}],\\ (\boldsymbol{\sigma}^{(d)})^2 &= m(\boldsymbol{\sigma}^{(d)})^2 + (1-m)Var[\boldsymbol{x}^{(d)}]\end{aligned} \tag{3}$$

where m is the moving average term, the estimated global statistic is used to normalize the testing data during inference. For clarify, we use $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$, $\boldsymbol{\mu}^2$ and $\boldsymbol{\sigma}$ to represent the multi-dimensional linear transformation parameters, global mean and variance vectors, respectively, below.

B. Domain-Shift in BN Layers

When using the trained source models F_s to test the samples from $\mathcal{T}$, following the standard procedure, the network normalizes the internal activations by the source statistics $\boldsymbol{\mu}_s$ and $\boldsymbol{\sigma}_s^2$, which may lead to mismatch in data distribution of the intermediate features between $\mathcal{S}$ and $\mathcal{T}$. To demonstrate this, we used a DeepSleepNet+ model pre-trained on the MASS database, to test samples from Sleep-EDF-ST database. We computed the mean and variance for each scalar feature of the input activations on different BN layers for each mini-batch. The moments from a mini-batch were then concatenated to form a statistics feature vector. We visualized in Fig.2 the statistics vectors by t-SNE embedding [24]. It could be observed that the statistics are clustered into their own domains across network layers. Even for sleep samples of different subjects in the same database, the discrepancies still exist. This distributional shift leads to a mismatch between the network weight $\boldsymbol{\theta}'$ and the normalized network activations. As a consequence, F_s perform poorly on $\mathcal{T}$.

C. Domain Statistics Alignment (DSA)

According to the above analysis, we conjectured that the mismatch in BN statistics between the source and target domain partly causes the performance deterioration of source models on the target domain. Inspired by [25], we attempt to mitigate the domain shift issue by modulating the statistics in BN layers on the target domain.

As shown in Fig. 1 (b), before testing sleep samples in $\mathcal{T}$, we collect the input activations of each BN layer in $F_s(\boldsymbol{\theta}')$ over all the samples $\boldsymbol{x_t}$ drawn from $\mathcal{T}$. Let $\boldsymbol{o}_t = \{\boldsymbol{o}_t^1, \boldsymbol{o}_t^2, \ldots \boldsymbol{o}_t^B\}$ denotes a batch of input to a BN layer. The statistics of the current BN layer could be easily estimated on $\mathcal{T}$ by:

$$\boldsymbol{\mu_t} = \mathbb{E}[\boldsymbol{o_t}], \quad \boldsymbol{\sigma_t}^2 = Var[\boldsymbol{o_t}] \tag{4}$$

$\boldsymbol{\mu_t}$ and $\boldsymbol{\sigma_t}$ is computed layer-by-layer for each BN layer. The values can be either computed directly or using the online algorithm proposed in [26].

Finally, we replace the global statistics of BN layers in $F(\boldsymbol{\theta'}; \boldsymbol{\mu_s}; \boldsymbol{\sigma_s^2})$ with the estimated statistics from $\mathcal{T}$, yielding a domain-specific model $F(\boldsymbol{\theta'}; \boldsymbol{\mu_t}; \boldsymbol{\sigma_t^2})$. During the inference on $\mathcal{T}$, the activation of BN layers are normalized using $\boldsymbol{\mu_t}$ and $\boldsymbol{\sigma_T}$ from $\boldsymbol{\mathcal{T}}$.

$$\hat{\boldsymbol{o}}_t = \frac{\boldsymbol{o_t} - \boldsymbol{\mu_t}}{\boldsymbol{\sigma_t}} \tag{5}$$

the same linear transformation with that in the training phase is reserved to generate *i.i.d.* samples:

$$\boldsymbol{y_t} = \boldsymbol{\gamma} * \hat{\boldsymbol{o}}_t + \boldsymbol{\beta} \tag{6}$$

In this study, we consider sleep records from the same subject as a separate data domain. The reasons are manifold. First, owning data from other subjects is becoming more and more difficult with privacy and security concerns growing. Secondly, inter-subject variability in data distribution inherently exists due to the peculiarities of individuals, hence adapting the source models subject-by-subject seems to be more reasonable. Lastly, the proposed methods are straightforward to implement with minimal meta-parameters to optimize. It is feasible to adapt source models on sleep records from a particular individual before the reference.

D. Adaptive Domain Statistics Alignment (AdaDSA)

We argue that DSA only exploits the local moments of $\mathcal{T}$. However, the size of $\mathcal{T}$ is generally small, the estimations of moments based on $\mathcal{T}$ alone is prone to noisy and may be inaccurate. Therefore, leveraging the observations from a much larger dataset $\mathcal{S}$ can potentially improve the robustness of the estimation of these statistics. For each BN layer, we introduce cross-domain statistics to control the degree of statistics alignment explicitly. Let q_s and q_t denote the distribution of data domain $\mathcal{S}$ and $\mathcal{T}$, these two distributions can be mixed by:

$$q_{ts} = \alpha q_t + (1-\alpha) q_s \tag{7}$$

where $\alpha \in [0, 1]$ is the blending factor to control the weights of two domains. The moments of the cross-domain distribution can be computed via:

$$\begin{aligned} \boldsymbol{\mu_{ts}} &= \alpha \boldsymbol{\mu_t} + (1-\alpha)\boldsymbol{\mu_s} \\ \boldsymbol{\sigma_{ts}^2} &= \alpha(\boldsymbol{\sigma_t^2} + (\boldsymbol{\mu_t} - \boldsymbol{\mu_{ts}})^2) \\ &\quad + (1-\alpha)(\boldsymbol{\sigma_s^2} + (\boldsymbol{\mu_s} - \boldsymbol{\mu_{ts}})^2) \end{aligned} \tag{8}$$

where $\boldsymbol{\mu_t}$, $\boldsymbol{\sigma_t^2}$, $\boldsymbol{\mu_s}$, and $\boldsymbol{\sigma_s^2}$ are the estimated global first and second moments on $\mathcal{T}$ and $\mathcal{S}$, respectively. $\boldsymbol{\mu_{ts}}$ is the weighted mean and $\boldsymbol{\sigma_{ts}^2}$ is the pooled variance. The parameter α enables us to learn how much each domain should contribute to the estimate of cross-domain statistics, when $\alpha = 1$, the adaptation on the current BN layer is equivalent to DSA, when $\alpha = 0.5$, the moments of two domains are equally used.

As shown in Fig.1(c), given the pre-trained source model $F_s(\boldsymbol{\theta'})$, we aim at learning a domain-specific model $F_t(\boldsymbol{\theta'}; \alpha)$ on the target domain. First, we replaced the stored global moments in BN layers with cross-domain statistics. Since the cross-domain moments are decided by the blending parameter α, we need to find an optimal α for each BN layer to make F_t perform well on $\mathcal{T}$. Ideally, the outputs of F_t should be close to one-hot encoding. Thus an objective we can pursue is to reduce the degree of uncertainty of the sleep labels predicted by F_t:

$$\mathcal{L}_{ent}(\boldsymbol{\alpha}) = -\frac{1}{n_t} \sum_{\boldsymbol{x_t} \in \mathcal{T}} \sum_{y \in \mathcal{Y}} F_t(\boldsymbol{\alpha}; y; \boldsymbol{x_t}) \log F_t(\boldsymbol{\alpha}; y; \boldsymbol{x_t}) \tag{9}$$

However, the target samples may be matched to the wrong labels by F_t. We use pseudo labels generated by the conjunction outputs of F_s and F_t to prevent the predicted results of F_t from deviating too far from the real labels. The pseudo labels y_t' were generated via:

$$y_t' = \underset{y}{argmax}\{(1-\lambda)F_s(\boldsymbol{x_t})[y] + \lambda F_t(\boldsymbol{\alpha}; \boldsymbol{x_t})[y]\} \tag{10}$$

where λ is a balance weight, which will gradually increase from 0 to 1 during training, this means we initialize the pseudo labels using F_s and progressively refine them by F_t. The cross-entropy loss of y_t' and the predicted probabilities is utilized as a regularization:

$$\mathcal{L}_{pseudo}(\boldsymbol{\alpha}) = -\frac{1}{n_t} \sum_{x_t \in \mathcal{T}} \sum_{y \in \mathcal{Y}} y_t' \log F_t(\boldsymbol{\alpha}; \boldsymbol{x_t})[y] \tag{11}$$

Finally, with the network weights $\boldsymbol{\theta'}$ fixed, we optimized the statistic blending factor $\boldsymbol{\alpha}$ with the full objective:

$$\mathcal{L} = \mathcal{L}_{pseudo} + \lambda \mathcal{L}_{ent} \tag{12}$$

During training, parameter set $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \ldots, \alpha_i \ldots, \alpha_n\}$, where α_i denotes the blending factor in the i-th BN layer, is optimized to minimize the full loss $\mathcal{L}$. In particular, α_i is updated by backpropagation and gradients descent with a learning rate η:

$$\alpha_i \leftarrow \alpha_i - \eta \frac{\partial \mathcal{L}}{\partial \alpha_i} \tag{13}$$

Intuitively, the hypothesis space for searching optimal parameter $\boldsymbol{\alpha}$ is small, it is feasible to fine-tune F_t merely using the target samples, which generally has a small data size.

III. Experiments

A. Databases

The used databases are summarized in Table I. We also introduced them briefly below.

1) Source Domain: We used two large scale open-source sleep datasets as the source datasets:

MASS: The Montreal Archive of Sleep Studies (MASS) [27] is the most frequently used source dataset in sleep staging transfer learning studies. It consists of 200 whole-night PSG recordings, which were pooled from different sleep centers. There are five subsets (SS1-SS5), in which each record was annotated by experts either according to the AASM (SS1 and SS3) or R&K standard (SS2, SS4, and SS5). We followed the procedure in [4] to convert all the records annotations to meet the AASM standard. The entire dataset was used in our experiments. The C4-A1 EEG channel was used to evaluate the proposed method.

TABLE I
SUMMARY OF DATABASES

Databases	Records Number	Sample Rate (Hz)	EEG Channel	Scoring Standard	Train/vadiation/test
SHHS	5973	125	C4-A1	R&K	4534/100/1159
MASS	200	200	C4-A1	R&K or AASM	180/10/10
SLEEP-EDF-SC	153	100	Fpz-Cz	R&K	subject-wise
SLEEP-EDF-ST	22	100	Fpz-Cz	R&K	subject-wise
UCD	25	128	C3-M2	R&K	subject-wise
HSFU	26	1024	C4-M1	AASM	subject-wise

SHHS: This is the largest accessible public database. The Sleep Heart Health Study (SHHS) [28], [29] is a multicenter cohort study designed to investigate the relationship between sleep-disordered breathing and cardiovascular diseases. The database consists of two subsets, SHHS Visit 1 (SHHS-1) and SHHS Visit 2 (SHHS-2). We used SHHS-1 as recommended in [30]. SHHS-1 contains 5793 whole-night PSG recordings annotated by sleep experts according to the R&K standard. Like MASS, the sleep labels were converted to meet the AASM standard, and the C4-A1 EEG channel was used in our experiments.

2) Target Domain: We used four small sleep databases as the target domains in our studies. The enrolled datasets contain both healthy and disordered subjects.

Sleep-EDF-SC: The Sleep Cassette (SC) [31] is a publicly available database. The database consists of 153 whole-night PSG recordings acquired from 78 healthy subjects. In this study, we both used the full Sleep-EDF-SC database (referred to as *Sleep-EDF-SC-78*) and a subset of PSG records from 20 subjects (referred to as *Sleep-EDF-SC-20*), which corresponds to an earlier version of the Sleep-EDF-SC database that has been extensively studied in the literature. As recommend in [4], [22], [32], only the in-bed parts (from lights off time to light on time) of the PSG recordings were used in our study. Following [4], we converted sleep annotations to meet the AASM standard. The Fpz-Cz EEG channel is used in our experiments.

Sleep-EDF-ST: The Sleep Telemetry (ST) database is a subset of the Sleep-EDF Expanded dataset. Different from Sleep-EDF-SC, the 22 sleep recordings were collected on subjects with difficulty falling asleep. Similar to the Sleep-EDF-SC subset, the sleep annotations were converted to meet the AASM standard. Besides, only in-bed parts of the whole night sleep records were used. The Fpz-Cz EEG channel is used in the experiments.

UCD: The St. Vincent's University Hospital / University College Dublin Sleep Apnea Database (UCD) [31] contains 25 full-night PSG recordings. The recordings were collected on subjects under diagnosis for either primary snoring and sleep apnea. The sleep annotations were made by experts according to the R&K standard. We used the C3-A2 EEG channel in our experiments.

HSFU: A non-public database collected in Huashan Hospital, Fudan University, Shanghai, China, during 2019-2020. It consists of 26 clinical PSG recordings, which were acquired on patients diagnosed with obstructive sleep apnea, insomnia, and restless legs syndrome. The PSG recordings were annotated by one qualified sleep expert according to the AASM standard. We used the C4-M1 EEG channel in our study. The study was approved by the Institutional Review Board of Huashan Hospital, Fudan University (2021-811).

B. Data Preprocessing

In our experiments, all the used EEG signals were downsampled to 100 Hz. EEG signals were normalized to zero mean and standard deviation of one by z-score to ensure the scale of these EEG signals from different databases roughly matching. Following the standard procedure, all the EEG signals were split into 30 seconds epochs without overlap. Each epoch has a corresponding sleep stage label.

C. Model Specification

Two state-of-the-art networks DeepSleepNet+ [4], [9] and U-time [10],were adopted to evaluate the proposed methods. Both networks have achieved the top-performance in sleep staging tasks as reported in the prior studies.

DeepSleepNet+: DeepSleepNet uses two branches of convolution layers to learn the task-related features from raw EEG signals. The learned features are then fed into sequential learning modules to capture the transition rules of sleep epochs. Each convolution layer in DeepSleepNet is associated with a BN layer. In this study, we used DeepSleepNet+ [4], which is an end-to-end variant of DeepSleepNet with a sequence-to-sequence learning scheme.

U-time: This is a temporal fully convolutional network with a deep architecture [10], which maps sequential inputs of raw EEG signals with arbitrary length to a sequence of class labels. Departing from the recurrent architectures, U-time can be directly applied across sleep databases. Therefore, U-time is ideal to be used in transfer learning studies. BN techniques are used following convolution layers both in the encoder and decoder modules.

D. Implementation Details

In our experiments, both networks were re-implemented by Pytorch.[1] The hyper-parameters of DeepSleepNet+ and U-time are consistent with their original implementation, except that we used the sequence-to-sequence learning scheme

[1] https://pytorch.org/

for U-time in accordance with DeepSleepNet+. In all experiments, we fixed the input sequence length as $L = 20$ for both networks.

We used the MASS and SHHS database, in turn, as the source domain. On source datasets, the number of subjects in the training set, validation set, and testing set are summarized in table I. We used Adam optimizer to update the network weights by minimizing the cross-entropy loss between output probabilities and sleep stage labels. For both networks, the learning rate was fixed as 10^{-4}. The training batch size was set to 32 on the MASS database and 1024 on the SHHS database to speed up training. The training process was early stopped when the validation accuracy stopped improving for 500 training steps. The models with the best validation accuracy were used as the source model $F_s(\boldsymbol{\theta}')$ to be transferred to the target domain.

On the target domains, the subject-wise domain adaptation was conducted. For DSA, in DeepSleepNet+, the statistics of target samples were directly computed, whereas we used the online estimation algorithm [26] to calculate the moments in U-time because of memory restrictions.

For AdaDSA, the blending factors $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \ldots \alpha_n\}$ is initialized as 1 and updated via Adam optimizer with network weights $\boldsymbol{\theta}'$ fixed. To attenuate the noisy predictions of $F_t^i(\theta')$ at the early stages of the training, parameter λ is gradually increased from 0 to 1 by $\lambda_p = \frac{2}{1+exp(-ap)} - 1$ with $a = 10$ as recommended in [33], where $p \in [0, 1]$ represents the training process. The initial learning rate η_0 is set as 10^{-3}. It is annealed during the training by: $\eta_p = \frac{\eta_0}{(1+bp)^g}$, where $b = 10, g = 0.75$ [34]. All the models were trained for 40 iterations. The parameters are fixed throughout all the experiments.

E. Performance Metrics

We used metrics including the accuracy (Acc.), macro F1-score (MF1), and Cohen's kappa (κ) to evaluate the sleep staging performance. The reported values are computed by pooling all the results of the subject-wise validation folds.

Shapiro–Wilk test was applied to verify the normality of the obtained metrics. Results showed that the Gaussian distribution assumption was not satisfied. Therefore, the Friedman test was used to evaluate the effects of specific factors on the obtained performance metrics. Wilcoxon signed-rank test with Holm-Bonferroni correction was applied for multiple comparisons if necessary. The significance level is set as $p < 0.05$ in this study.

IV. Results

A. Intra-Domain and Direct Transfer Performance

We trained DeepSleepNet+ and U-time on two source databases. Fig.3 shows the intra-domain performance of the source models and the cross-domain performance by direct applying source models on four target domains. In principle, both networks perform well on the source domains, reaching an accuracy with values of above 84%. Using the same network architecture, source models trained on the MASS database obtain slightly higher accuracy than that on SHHS.

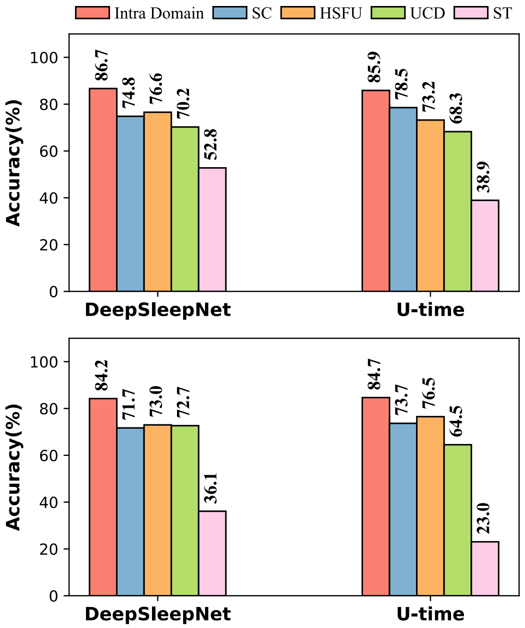

Fig. 3. The intra-domain and cross-domain sleep staging accuracy of DeepSleepNet+ and U-time trained on two source datasets: top: MASS; bottom: SHHS.

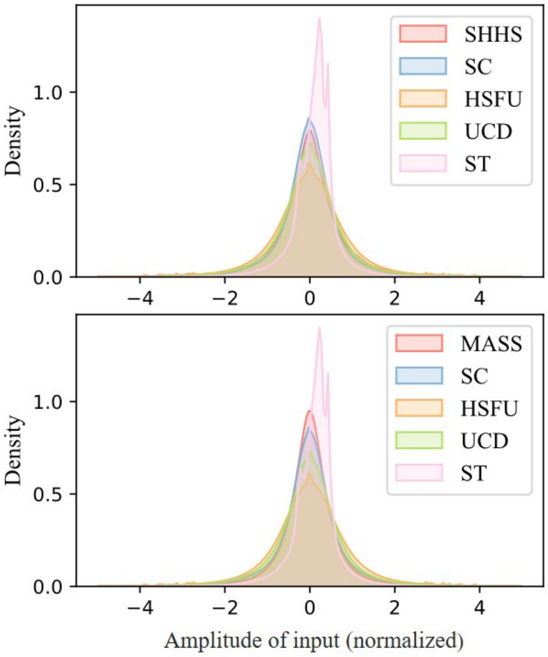

Fig. 4. The distribution of the signal amplitudes of the normalized EEG epochs on different databases. Top: the MASS databases compared to target domains. Bottom: the SHHS databases compared to target domains.

However, the trained source models without adaptation have poorer performance on the target domains than that on the source domain as indicated in Fig.3. The degree of performance degeneration differs by the target datasets. The accuracy drop on Sleep-EDF-SC-20, HSFU, UCD database is relatively modest. In contrast, the accuracy declined sharply on Sleep-EDF-ST, with a value of 23.0-52.8% obtained by different source models. Although the distribution of high-dimensional sleep samples can not be explicitly given, Fig. 4 provides

TABLE II
OVERALL PERFORMANCE

		MASS→ Sleep-EDF-SC-20			MASS→ Sleep-EDF-SC-78			MASS→ Sleep-EDF-ST			MASS→UCD			MASS→HSFU		
	Method	Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ
DeepSleepNet+	DT	74.81^{b}	66.16^{b}	0.650^{b}	65.1^{c}	57.18^{c}	0.526^{c}	52.75^{c}	42.85^{b}	0.324^{c}	70.2^{b}	67.31^{b}	0.615^{b}	76.56^{b}	73.0^{b}	0.686^{b}
	DSA	80.38^{a}	74.11^{a}	0.720^{a}	70.26^{b}	64.33^{b}	0.576^{b}	74.96^{b}	**68.84^{a}**	0.624^{b}	70.78^{b}	67.46^{b}	0.607^{b}	78.51^{b}	74.78^{b}	0.707^{b}
	AdaDSA	**81.47^{a}**	**75.03^{a}**	**0.736^{a}**	**72.73^{a}**	**66.82^{a}**	**0.616^{a}**	**75.93^{a}**	68.06^{a}	**0.641^{a}**	**73.99^{a}**	**70.66^{a}**	**0.651^{a}**	**80.06^{a}**	**77.16^{a}**	**0.732^{a}**
U-time	DT	78.53^{c}	71.45^{b}	0.706^{c}	67.76^{b}	61.31^{b}	0.567^{b}	38.92^{c}	32.03^{b}	0.202^{c}	68.25^{b}	64.25^{b}	0.589^{b}	73.21^{b}	69.4^{a}	0.646^{b}
	DSA	81.36^{b}	**74.78^{a}**	0.739^{b}	70.72^{a}	64.45^{a}	0.592^{a}	72.88^{b}	**65.08^{a}**	0.611^{b}	68.86^{b}	65.12^{b}	0.585^{b}	75.70^{b}	70.95^{a}	0.669^{ab}
	AdaDSA	**81.86^{a}**	74.71^{a}	**0.747^{a}**	**73.3^{a}**	**66.64^{a}**	**0.634^{a}**	**73.78^{a}**	64.85^{a}	**0.621^{a}**	**70.93^{a}**	**66.99^{a}**	**0.614^{a}**	**76.75^{a}**	**71.82^{a}**	**0.683^{a}**
		SHHS→ Sleep-EDF-SC-20			SHHS→ Sleep-EDF-SC-78			SHHS→ Sleep-EDF-ST			SHHS→UCD			SHHS→HSFU		
	Method	Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ
DeepSleepNet+	DT	71.69^{c}	64.30^{c}	0.589^{c}	66.68^{c}	57.69^{c}	0.53^{c}	36.10^{b}	30.68^{b}	0.20^{b}	72.66^{a}	67.38^{b}	0.630^{b}	72.99^{c}	66.03^{c}	0.621^{c}
	DSA	76.65^{b}	67.12^{b}	0.667^{b}	73.77^{b}	64.37^{b}	0.626^{b}	71.85^{a}	60.02^{a}	0.586^{a}	71.83^{a}	66.46^{b}	0.618^{b}	77.47^{b}	71.39^{b}	0.687^{b}
	AdaDSA	**79.64^{a}**	**70.94^{a}**	**0.712^{a}**	**73.93^{a}**	**65.0^{a}**	**0.633^{a}**	**72.34^{a}**	**61.18^{a}**	**0.597^{a}**	**73.73^{a}**	**69.08^{a}**	**0.646^{a}**	**78.51^{a}**	**72.96^{a}**	**0.704^{a}**
U-time	DT	73.66^{c}	65.65^{b}	0.630^{c}	68.04^{c}	59.49^{b}	0.558^{c}	23.03^{b}	16.86^{b}	0.09^{b}	64.51^{b}	55.69^{b}	0.520^{b}	76.51^{b}	69.05^{b}	0.674^{b}
	DSA	80.22^{b}	69.88^{a}	0.720^{b}	72.58^{b}	63.15^{a}	0.616^{b}	62.11^{a}	48.77^{a}	0.457^{a}	70.48^{a}	**63.40^{a}**	0.602^{a}	78.69^{b}	70.60^{a}	0.707^{a}
	AdaDSA	**81.34^{a}**	**70.72^{a}**	**0.739^{a}**	**73.42^{a}**	**63.69^{a}**	**0.631^{a}**	**62.77^{a}**	**49.81^{a}**	**0.471^{a}**	**70.69^{a}**	63.17^{a}	**0.603^{a}**	**79.01^{a}**	**70.72^{a}**	**0.712^{a}**

^{abc}: Within a column of reported metrics on a transfer tasks, means without a common superscription differ significantly ($p < 0.05$); DT: direct transfer; DSA: domain statistics alignment; AdaDSA: adaptive domain statistics alignment.

evidence of the data distribution variability by drawing the density of the amplitude of EEG signals. The distributions are in principle close to Gaussian but have different variances. In particular, the data distribution of Sleep-EDF-ST deviates significantly from that of the source databases, which may partly explain the poor DT performance.

B. Performance on Transfer Tasks

Table II shows the overall performance using different methods on 16 transfer tasks (2 network architectures × 2 source domains × 4 target domains). We mark the highest values of each metrics on each transfer task in bold. The superscription of the metrics indicates the statistical significance. The performance shows different patterns by different transfer tasks.

1) DSA: In most cases, using DSA could achieve higher performance than DT, which is indicated by Acc., MF1, and κ. The performance gains vary from different target datasets. On Sleep-EDF-SC-20, Sleep-EDF-SC-78 and Sleep-EDF-ST, DSA could achieve a significantly higher performance under all experimental settings. In particular, for Sleep-EDF-ST, which suffers from the most severe distribution mismatch with the source database, the DT performance is very low, with accuracy ranging from 23.03% to 52.75% on different transfer tasks. Such precision is unacceptable for practical applications. By DSA, the prediction accuracy could be primarily enhanced, reaching a significantly higher value ranging from 62.1% to 75.93% ($p < 0.05$). The DT performance decline on Sleep-EDF-SC-20 is not as sharp. For example, the accuracy of DT using U-time pre-trained on the MASS database has reached 78.53%. By DSA, the accuracy could be brought to a higher level of 81.36%. On HSFU, all the performance indicators obtained by DSA are higher than that by DT, but the performance differences are not significant using models pre-trained on MASS. As for the UCD dataset, the improvement by DSA is relatively marginal in most cases. Remarkable accuracy improvement could be observed using U-time pre-trained on SHHS. A subtle performance decline using DSA on DeepSleepNet+ pre-trained on SHHS could be observed compared to DT, but the difference is not significant for all indicators ($p > 0.05$). Overall, DSA contributes to better sleep staging performance on most transfer tasks compared to DT, demonstrating the effectiveness of DSA.

2) AdaDSA: From Table II, applying AdaDSA on source models could attain the best performance on almost all the transfer tasks. The best accuracy obtained by AdaDSA are 81.86%, 73.93%, 75.93%, 73.99%, and 80.06% on Sleep-EDF-SC-20, Sleep-EDF-SC-78, Sleep-EDF-ST, UCD, and HSFU, respectively, across all experimental settings. Overall, source models with AdaDSA consistently outperform their DT counterparts in all cases. The differences in the obtained performance indicators are significant on almost all the transfer tasks. Fig.5 shows the subject-wise accuracy improvement by

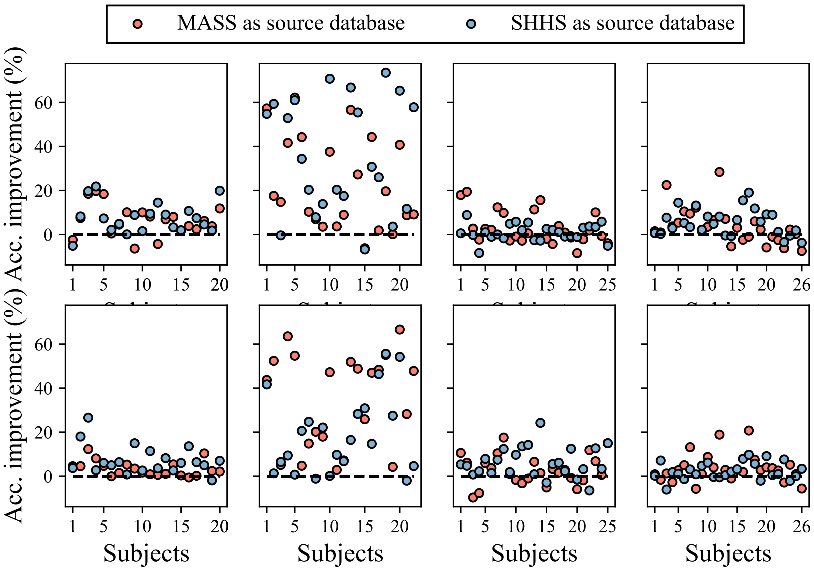

Fig. 5. The accuracy improvements from DSA to DT on each subject. Top: DeepSleepNet+; Bottom: U-time; From left to right on each row: Sleep-EDF-SC-20, Sleep-EDF-ST, UCD, HSFU.

AdaDSA with respect to DT. This figure suggests AdaDSA can improve the accuracy of DT on most subjects. Overall, the subject-wise accuracy improvements are more pronounced on Sleep-EDF-SC-20 and Sleep-EDF-ST. Although samples from most subjects on UCD and HSFU can be more accurately predicted by AdaDSA over DT. However, it is shown that the accuracy was not improved by AdaDSA in a few cases. For example, applying AdaDSA using DeepSleepNet+ on Subject.20 from UCD database leads to an accuracy drop by 9.48%. We found the performance drop is mainly due to the inaccurate estimation of the second-order moments on the first convolution layers of DeepSleepNet+. The biased estimation of moments leads to a severe distribution mismatch of the following normalized internal features between source and target domains. Consequently, the features of target samples are non-normally distributed, hence hinder the networks from mapping them into the correct labels.

AdaDSA could further enhance the performance of source models compared to DSA on most transfer tasks. AdaDSA perform slightly better than DSA on almost all the transfer tasks. The statistical significance differs by the used networks, the source databases, and the target databases. For example, when pre-trained on MASS, U-time could obtain significantly higher accuracy by AdaDSA than that by DSA on four databases ($p < 0.05$). By contrast, the statistical difference is not observed on most transfer tasks when using MASS database as the source domain. The performance improvement from DSA to AdaDSA is mainly attributed to the learned cross-domain statistics, which are decided by the blending factor α. Overall, there is a trend that the performance obtained by AdaDSA is highly relevant to that by DSA.

C. Compared With Related Works

In this section, we compared our work with representative sleep staging transfer learning studies.

TABLE III
PERFORMANCE COMPARISON WITH [4]

methods	Sleep-EDF-SC-20			Sleep-EDF-ST		
	Acc.	MF1	κ	Acc.	MF1	κ
DT [4]	74.2	66.9	0.651	66.7	61.3	0.541
LFS [4]	80.8	74.2	0.731	72.4	64.6	0.603
FT [4]	**84.4**	**78.8**	**0.781**	**81.5**	**77.5**	**0.738**
DT (ours)	74.81	66.16	0.650	52.75	42.85	0.324
DSA (ours)	80.38	74.11	0.72	74.96	68.84	0.624
AdaDSA (ours)	81.47	75.03	0.736	73.93	68.06	0.641

The values of Acc. and MF1 are presented in percentage. The results from [4] are reported in the original paper. DT: direct transfer; FT: fine-tune; LFS: learning from scratch

1) Compared to Supervised Transfer Learning: Supervised transfer learning on sleep staging tasks has been well studied in previous studies. The comprehensive transfer learning conducted by Phan *et al.* [4] gives us a standard benchmark for comparison. In this work, we have utilized the same network (DeepSleepNet+) and validated the proposed method on the same transfer tasks (*i.e.*, MASS → Sleep-EDF-SC-20 and MASS → Sleep-EDF-ST) with [4], which ensure a fair comparison with [4]. The obtained results in our experiments and that reported in [4] are summarized in table III. First, a noticeable difference in DT performance on Sleep-EDF-ST between two studies is observed. The DT performance obtained in our experiments is significantly lower than that in [4]. We found this inconsistency is due to the normalization scheme applied on EEG signals. In our study, we normalized EEG signals by z-score, whereas the authors only removed the DC components from EEG signals in [4]. Intriguingly, when using the same normalization scheme with [4], we can obtain a very close DT performance on Sleep-EDF-ST, with the Acc., MF1 and κ values corresponding to 67.51%, 59.26% and 0.521, respectively. This suggests the normalization scheme used on sleep samples

TABLE IV
PERFORMANCE COMPARISON WITH METASLEEPLEARNER

Datasets	Acc.			MF1			κ		
	[14]	DSA (ours)	AdaDSA (ours)	[14]	DSA (ours)	AdaDSA (ours)	[14]	DSA (ours)	AdaDSA (ours)
Sleep-EDF-SC-20 †	72.1	72.78	**76.25**	64.8	65.21	**65.24**	0.624	0.651	**0.668**
Sleep-EDF-SC-20	74.9	72.78	**76.25**	**68.8**	65.21	65.24	0.662	0.651	**0.668**
Sleep-EDF-ST	67.1	69.22	**69.74**	**60.8**	59.05	59.62	0.554	0.547	**0.555**
UCD	56.3	63.51	**66.13**	50.1	57.26	**59.59**	0.429	0.504	**0.544**
MASS(SS2)	77.3	83.40	**83.86**	69.9	72.65	**72.90**	0.682	0.754	**0.762**

The values of Acc. and MF1 are presented in percentage. The highest values in a row evaluated on the same transfer tasks are marked in bold. The results of MetaSleepLearner were reported in the original paper. All the results obtained by our approaches methods are based on single-channel EEG. By contrast, the reported results of MetaSleepLearner are obtained using multi-channel montage (*i.e.*, a combination of EEG, EOG, and submental EMG), except the results of Sleep-EDF-SC†, whose results were obtained by single-channel EEG only. Note that the used sleep records in Sleep-EDF-SC-20, Sleep-EDF-ST, and UCD datasets are trimmed at each end to at most 30 minutes of wake periods before and after the sleep records following [14].

may effects DT performance (see section.V-C for details). Not surprisingly, the highest performance was achieved by conducting supervised fine-tune (FT) on the target domains across all the transfer tasks, demonstrating the benefit of transferring knowledge from the source domain to the target domains (see in [4]). In contrast, the proposed methods do not perform as well as FT. This is understandable since the proposed method is unsupervised. Unlike FT, the proposed methods did not use any label information on the target domains. Nevertheless, the proposed methods consistently outperform their LFS counterparts. In particular, AdaDSA has achieved superior performance over learning from scratch (LFS) on both target domains. This, on the one hand, demonstrates the efficacy of the proposed methods, on the other hand, indicates that the learned task-specific knowledge on the source domains is largely transferable to the target domain. As long as the marginal distribution of target samples is aligned to that of the source samples, the networks can accurately predict these unseen target samples.

2) Compared to Meta-Learning: MetaSleepLearner [14] enhance the generalization of sleep staging models on target domains by learning optimal initial network weights on the source domain. MetaSleepLearner can adapt the learned network weights on unseen target domains with only a few labeled sleep samples. Although both MetaSleepLearner and the proposed methods are largely model-agnostic, to ensure a fair comparison, we re-implemented the same network originally proposed in [14]. It was not explicitly stated whether the BN technique was applied in the original paper [14], we applied BN layers after all the CNN blocks in our implementation. We kept the used source domain and target domain consistent with [14], except that we only used single-channel EEG signals as input. The source models in our experiments were trained following the non-meta-learning setup (baseline 1) in the original paper. The source models are subsequently adapted to each subject from the target domain by DSA and AdaDSA. The results comparison is summarized in table IV. With the same PSG montage (single-channel EEG), both DSA and AdaDSA outperform MetaSleepLearner on Sleep-EDF-SC-20. Compared with the results obtained by MetaSleepLearner using multiple-channel montage (*i.e.*, a combination of EEG, EOG, and submental EMG), DSA and AdaDSA still attained better performance on most transfer tasks using single-channel EEG. On UCD and MASS SS2, the accuracy gains by AdaDSA over MetaSleepLearner are 9.83% and 6.56%, respectively. These results demonstrate the proposed methods could achieve better performance than MetaSleepLearner on most transfer tasks. Most importantly, the proposed methods are far more efficient than MetaSleepLearner, which generally cost days to optimize the meta-weights.

TABLE V
PERFORMANCE COMPARISON WITH YOO *et.al* [22]

method	Sleep-EDF-SC-20			Sleep-EDF-ST		
	Acc.	MF1	κ	Acc.	MF1	κ
Yoo *et.al* [22]	80.3	70.1	0.728	71.7	63.5	0.615
DSA (ours)	80.38	74.11	0.720	74.96	**68.84**	0.624
AdaDSA (ours)	**81.47**	**75.03**	**0.736**	**75.93**	68.06	**0.641**

The values of Acc. and MF1 are presented in percentage. The highest values in a column are marked in bold. The results from Yoo *et.al* were reported in the original paper.

3) Compared to Previous Domain Adaptation Works: To reduce the labeling cost on target domains, a few unsupervised domain adaptation approaches have been proposed [20]–[22]. In principle, these approaches enhance the source models' performance on target domains by learning domain-invariant features through a set of adversarial losses. However, these methods need to access the source data during learning to adapt, which is not efficient for data transmission, and even worse, not applicable in case the source samples are inaccessible. Besides, the complicated adversarial losses generally converge slowly. In comparison, the proposed methods are source-free and have minimal parameters to optimize. As one of the state-of-the-art, Yoo *et al.* [22] used a similar experimental setting with our study (transferring pre-trained deepSleepNet+ on MASS to Sleep-EDF-SC-20 and Sleep-EDF-ST dataset), providing a benchmark for us to directly compare with. As shown in table V, both DSA and AdaDSA have attained higher performance than that reported in [22]. Despite its simplicity, DSA has achieved a higher MF1 than [22] by 4.1% and 5.34% on Sleep-EDF-SC-20 and Sleep-EDF-ST, respectively. Overall, these results suggest that the proposed method, if not better, at least as well as the complicated approaches that use adversarial losses.

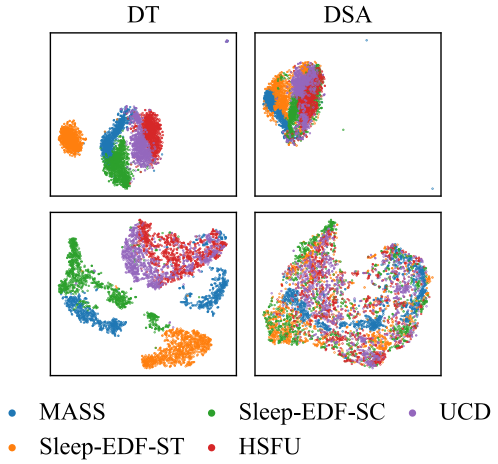

Fig. 6. Visualization of features from different domains generated by DeepSleepNet+ pre-trained on MASS. From each dataset, we randomly selected 1000 samples from one subject. Top: features from shallow layer; Bottom: features from deep layer.

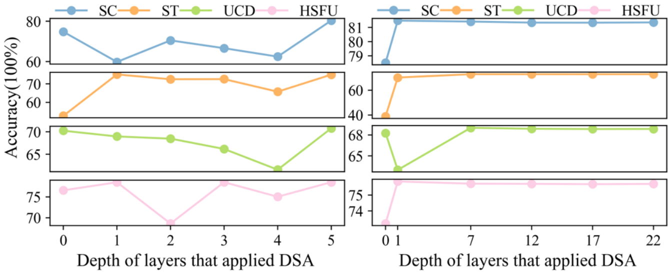

Fig. 7. The obtained accuracy by applying DSA on BN layers with different layer depths. left: DeepSleepNet+; right: U-time.

V. Discussion

A. DSA

As indicated in section IV-B, DSA enhances the DT performance in most cases. The performance improvement is most likely to be attributed to the alignment of the distribution of network features. Fig.6 gives us a visual observation on how DSA mitigates the distribution mismatch issue. In the shallow layer of DeepSleepNet+, the features from different domains have a trend to cluster into their own domains, indicating the cross-domain data distribution variability in the feature space. With the growth in layer depth, the cross-domain feature discrepancies have a tend to be more severe. The divergence in the feature space causes the negative transfer problem. By DSA, the inter-domain feature boundaries were largely blurred. With the network weights of source models unchanged, the output of the network on the target domain may become more accurate due to the distribution alignment in the feature space. We analyzed the effect of applying DSA on different BN layers of DeepSleepNet+ and U-time. As shown in Fig.7, the average accuracy of source models is improved on almost all transfer tasks when applied DSA on the first BN layers of both networks. However, when progressively incorporating more BN layers in the adaptation, the obtained accuracy shows different patterns on DeepSleepNet+ and U-time. On U-time, the accuracy shows a smooth increasing trend as the depth of adapted BN layers grows. However, the improvement trend is not observed on DeepSleepNet+. In principle, the best accuracy is always attained when all the BN layers have been adapted by DSA.

DSA was designed to adapt the pre-trained source models to sleep samples of a particular subject. Apart from the results presented in table II, to provide a more comprehensive assessment, here we have also evaluated the performance of DSA using the statistics computed on samples from the whole target sleep cohorts. As shown in table VI, by modulating the BN statistics by the moments of all the samples in the target sleep cohorts, DSA still has attained higher performance over its DT counterpart on most transfer tasks. In principle, the performance improvements by DSA under this experimental setting is on par with that of subject-wise adaptation in most cases. We observed that the results obtained by applying DSA on the whole Sleep-EDF-ST database are far less prominent than the subject-wise basis. This, we conjecture, is most likely due to the distributional-shift across different subjects in the Sleep-EDF-ST database (as evidenced in Fig.2). Overall, these results imply that it is feasible to apply DSA using the whole sleep cohorts' statistics. Nevertheless, to avoid an extreme mismatch between the estimated population moments and the data distribution of sleep samples from a particular subject, applying subject-wise DSA may be preferable. Furthermore, owing EEG data from others would be more and more difficult when privacy and security become serious concerns. In this context, we recommend applying subject-wise DSA in practical applications.

B. AdaDSA

To see how the source models were adapted on target domains by AdaDSA, we take validation folds from Sleep-EDF-SC and Sleep-EDF-ST databases as examples (illustrated in Fig.8). With the updating of parameter $\boldsymbol{\alpha}$, the training loss gradually descents. In particular, the sharp decline could be observed in $\mathcal{L}_{ent}$ in the early training stages. Nevertheless, $\mathcal{L}_{pseudo}$ either oscillate heavily or keep at near-zero during the training. As the training loss decreases, the testing accuracy on the target domains has shown an increasing trend. Finally, parameter $\boldsymbol{\alpha}$ is tuned for each BN layer. In most cases, AdaDSA could achieve better results than DSA By tuning the degree of statistic alignment.

C. The Sensitivity to Input Normalization

Another interesting finding is that the transfer performance of source models is influenced by the normalization method applied to the sleep samples. It is known that EEG signals from different sources generally have different output levels due to different acquisition devices and setups. Normalizing EEG signals can roughly uniform the scale of samples from the source and target domain. In our experiments, we found whether scaling the input signals has potential impacts on the DT performance on a few transfer tasks, as evidenced in our experiments (cf. section IV-C.1). Another example is that EEG signals from the UCD dataset are recorded in the same unit with MASS (microvolt) as indicated in the

TABLE VI
THE PERFORMANCE OF DSA USING STATISTICS OF THE TARGET SLEEP COHORTS

The MASS database as source													
Network	Method	Sleep-EDF-SC-20			Sleep-EDF-ST			UCD			HSFU		
		Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ
DeepSleepNet+	DT	74.81	66.16	0.650	52.75	42.85	0.324	70.2	67.31	**0.615**	76.56	**73.0**	**0.686**
	DSA	**77.96***	**71.82****	**0.687***	**66.02****	**58.36****	**0.503****	**70.66**	**67.34**	0.606	**76.73**	72.95	0.682
U-time	DT	78.53	71.45	0.706	38.92	32.03	0.200	68.25	64.25	**0.589**	73.21	69.4	0.656
	DSA	**80.17***	**73.75****	**0.722***	**56.69****	**48.85****	**0.399****	**68.85**	**64.85**	0.583	**74.97**	**70.70**	**0.660**
The SHHS as source database													
Network	Method	Sleep-EDF-SC-20			Sleep-EDF-ST			UCD			HSFU		
		Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ	Acc.	MF1	κ
DeepSleepNet+	DT	71.69	64.30	0.589	36.10	30.68	0.20	72.66	67.38	0.630	72.99	66.03	0.621
	DSA	**76.98***	**67.81***	**0.671***	**68.18****	**56.18****	**0.518****	**74.0**	**69.21**	**0.649**	**78.64****	**72.97****	**0.706****
U-time	DT	73.66	65.65	0.630	23.03	16.86	0.09	64.51	55.69	0.520	76.51	69.05	0.674
	DSA	**80.11****	**70.15****	**0.717****	**52.19****	**39.85****	**0.314****	**70.82****	**63.44****	**0.605****	**78.3***	**70.35***	**0.701***

The values of Acc. and MF1 are presented in percentage. The highest values of each metrics under a transfer task was marked in bold. **: statistically highly significant as $p < 0.01$. *: statistically significant as $p < 0.05$

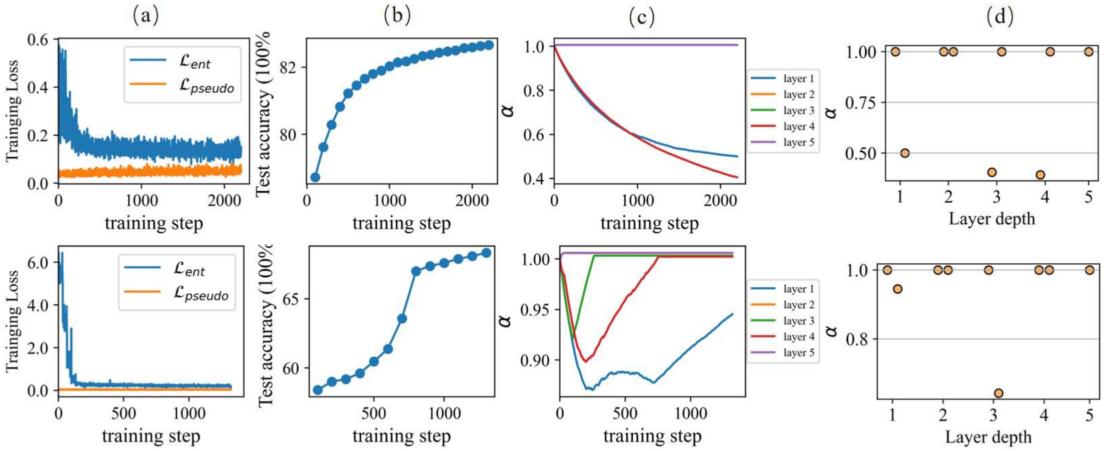

Fig. 8. Examples of the training process of AdaDSA on DeepSleepNet+. (a): training loss; (b): testing accuracy; (c): the updating process of α values of each BN layer during training (only show one branch of CNN blocks) (d): the learned α of each BN layer (two CNN blocks each for the first four layers).

meta-information. However, the output level of EEG signals from UCD datasets is orders of magnitude less than that from MASS datasets. Consequently, when direct transferring the source models, which are trained with samples without being normalized on MASS, to UCD dataset. The average accuracy is extremely low, with a value of 22.62%. The source model has been lost its predictive abilities on UCD due to the mismatch in input scale. Nevertheless, applying DSA and AdaDSA can substantially improve the accuracy value to 72.63% and 73.15%, respectively, which is on par with the obtained performance with all samples being normalized. This indicates the proposed methods are less sensitive to the normalization scheme applied to sleep data. Thus, they could be applied without making any assumptions about the preprocessing pipeline of source data. Therefore, we recommend normalizing sleep signals from both source and target datasets in transfer learning studies. In practice applications, in case only the source models are accessible, DSA and AdaDSA are recommended to use to mitigate the mismatch in the scale of EEG input.

D. Limitations and Future Works

Future work should be further investigated. First, EEG signals in PSG inherently have low-frequency noises, which vary between subjects and acquisition settings. To minimize this effect, noise removal techniques, such as Fourier decomposition [13] and variational mode decomposition techniques [35] play important roles in the signal processing pipeline. Secondly, in a few cases, the estimated moments from a small target set deviates from the real distribution. In this context, applying DSA and AdaDSA on the target domain could not reach a superior performance over DT. We should develop a framework to evaluate the accuracy of estimated moments from the target domain. Consequently, before testing on the target domain, decisions about whether to apply DSA to the source models can be made to avoid negative adaptation. Thirdly, pseudo labels play an essential role in the optimization process of AdaDSA. Under the current framework, the predicted pseudo labels are noisy due to domain shift and often provide inaccurate directions for gradient descent. In future works, we target to develop a self-supervised pseudo-labeling strategy, such as DeepCluster [36], to improve the performance of AdaDSA further. Lastly, we plan to evaluate the proposed methods on more extensive network architectures, such as the newest version of U-time, i.e. U-sleep [37].

VI. Conclusion

We have illustrated a practical domain adaptation approach for deep sleep model transfer, based on statistics alignment of the deep features. Our methodology is unsupervised on the target domain and can be applied without access to the source data. Experiments are conducted using two state-of-the-art deep sleep staging models on two large-scale source datasets and four small target sets. The results show that the proposed methods significantly improve the performance of source models on data from unseen subjects, offering a novel perspective for overcoming sleep data mismatch and generalizing deep sleep staging models to various sleep data domains.

Acknowledgment

The authors would like to thank Zhenning Mei for his supports on this work.

References

[1] J. A. Hobson, "A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects," *Electroencephalogr. Clin. Neurophysiol.*, vol. 26, no. 6, p. 644, 1969.

[2] R. B. Berry, R. Brooks, C. E. Gamaldo, S. M. Harding, C. Marcus, and B. V. Vaughn, "The AASM manual for the scoring of sleep and associated events," *Rules, Terminol. Tech. Specifications, Darien, Illinois, Amer. Acad. Sleep Med.*, vol. 176, p. 2012, Oct. 2012.

[3] R. S. Rosenberg and S. Van Hout, "The American Academy of Sleep Medicine inter-scorer reliability program: Sleep stage scoring," *J. Clin. Sleep Med.*, vol. 9, no. 1, pp. 81–87, Jan. 2013.

[4] H. Phan *et al.*, "Towards more accurate automatic sleep staging via deep transfer learning," *IEEE Trans. Biomed. Eng.*, vol. 68, no. 6, pp. 1787–1798, Jun. 2021.

[5] H. Phan, F. Andreotti, N. Cooray, O. Y. Chen, and M. De Vos, "SeqSleepNet: End-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 3, pp. 400–410, Mar. 2019.

[6] C. Sun, J. Fan, C. Chen, W. Li, and W. Chen, "A two-stage neural network for sleep stage classification based on feature learning, sequence learning, and data augmentation," *IEEE Access*, vol. 7, pp. 109386–109397, 2019.

[7] C. Sun, C. Chen, W. Li, J. Fan, and W. Chen, "A hierarchical neural network for sleep stage classification based on comprehensive feature learning and multi-flow sequence learning," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 5, pp. 1351–1366, May 2020.

[8] J. B. Stephansen, "Neural network analysis of sleep stages enables efficient diagnosis of narcolepsy," *Nature Commun.*, vol. 9, no. 1, pp. 1–15, 2018.

[9] A. Supratak, H. Dong, C. Wu, and Y. Guo, "DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 11, pp. 1998–2008, Nov. 2017.

[10] M. Perslev, M. Jensen, S. Darkner, P. J. R. Jennum, and C. Igel, "U-Time: A fully convolutional network for time series segmentation applied to sleep staging," in *Advances in Neural Information Processing Systems*, vol. 32. Red Hook, NY, USA: Curran Associates, 2019.

[11] B. Xie, "Real-time sleep apnea detection by classifier combination," *IEEE Trans. Inf. Technol. Biomed.*, vol. 16, no. 3, pp. 469–477, May 2012.

[12] B. Fatimah, P. Singh, A. Singhal, and R. B. Pachori, "Detection of apnea events from ECG segments using Fourier decomposition method," *Biomed. Signal Process. Control*, vol. 61, Aug. 2020, Art. no. 102005.

[13] P. Singh, S. D. Joshi, R. K. Patney, and K. Saha, "The Fourier decomposition method for nonlinear and non-stationary time series analysis," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 473, no. 2199, Mar. 2017, Art. no. 20160871.

[14] N. Banluesombatkul *et al.*, "MetaSleepLearner: A pilot study on fast adaptation of bio-signals-based sleep stage classifier to new individual subject using meta-learning," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 6, pp. 1949–1963, Jun. 2021.

[15] H. Phan, O. Y. Chen, P. Koch, A. Mertins, and M. D. Vos, "Deep transfer learning for single-channel automatic sleep staging with channel mismatch," in *Proc. 27th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2019, pp. 1–5.

[16] A. Guillot and V. Thorey, "RobustSleepNet: Transfer learning for automated sleep staging at scale," 2021, *arXiv:2101.02452*.

[17] K. Mikkelsen and M. de Vos, "Personalizing deep learning models for automatic sleep staging," 2018, *arXiv:1801.02645*.

[18] H. Phan *et al.*, "Personalized automatic sleep staging with single-night data: A pilot study with Kullback–Leibler divergence regularization," *Physiolog. Meas.*, vol. 41, no. 6, Jun. 2020, Art. no. 064004.

[19] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2017, pp. 1126–1135.

[20] S. Nasiri and G. D. Clifford, "Attentive adversarial network for large-scale sleep staging," in *Proc. Mach. Learn. Healthcare Conf.*, Sep. 2020, pp. 457–478.

[21] R. Zhao, Y. Xia, and Y. Zhang, "Unsupervised sleep staging system based on domain adaptation," *Biomed. Signal Process. Control*, vol. 69, Aug. 2021, Art. no. 102937.

[22] C. Yoo, H. W. Lee, and J. Kang, "Transferring structured knowledge in unsupervised domain adaptation of a sleep staging network," *IEEE J. Biomed. Health Informat.*, early access, Aug. 13, 2021, doi: 10.1109/JBHI.2021.3103614.

[23] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2015, pp. 448–456.

[24] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.

[25] Y. Li, N. Wang, J. Shi, X. Hou, and J. Liu, "Adaptive batch normalization for practical domain adaptation," *Pattern Recognit.*, vol. 80, pp. 109–117, Aug. 2018.

[26] E. K. Donald, "The art of computer programming," in *Sorting and Searching*, vol. 3. Addison Wesley, 1999, pp. 426–458.

[27] C. O'Reilly, N. Gosselin, J. Carrier, and T. Nielsen, "Montreal archive of sleep studies: An open-access resource for instrument benchmarking and exploratory research," *J. Sleep Res.*, vol. 23, no. 6, pp. 628–635, Jun. 2014.

[28] G.-Q. Zhang *et al.*, "The National Sleep Research Resource: Towards a sleep data commons," *J. Amer. Med. Inform. Assoc.*, vol. 25, no. 10, pp. 1351–1358, Oct. 2018.

[29] S. F. Quan *et al.*, "The sleep heart health study: Design, rationale, and methods," *Sleep*, vol. 20, no. 12, pp. 1077–1085, 1997.

[30] A. Sors, S. Bonnet, S. Mirek, L. Vercueil, and J.-F. Payen, "A convolutional neural network for sleep stage scoring from raw single-channel EEG," *Biomed. Signal Process. Control*, vol. 42, pp. 107–114, Apr. 2018.

[31] A. L. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.

[32] S. A. Imtiaz and E. Rodriguez-Villegas, "Recommendations for performance assessment of automatic sleep staging algorithms," in *Proc. 36th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2014, pp. 5044–5047.

[33] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, 2017.

[34] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2015, pp. 1180–1189.

[35] K. Dragomiretskiy and D. Zosso, "Variational mode decomposition," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 531–544, Feb. 2014.

[36] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 132–149.

[37] M. Perslev, S. Darkner, L. Kempfner, M. Nikolic, P. J. Jennum, and C. Igel, "U-Sleep: Resilient high-frequency sleep staging," *npj Digit. Med.*, vol. 4, no. 1, pp. 1–12, Dec. 2021.