Sleep Stage Classification with Learning from Evolving Datasets

Huayu Li[1], Xiwen Chen[2], Gregory Ditzler[3], William D.S. Killgore[4], Stuart F. Quan[5, 6], Janet Roveda[1,7,8], and Ao Li[1,8]

1 Department of Electrical & Computer Engineering at the University of Arizona, Tucson, AZ, USA
2 School of Computing at Clemson University, Clemson, SC, USA
3 Department of Electrical and Computer Engineering at Rowan University, Glassboro, PA, USA
4 Department of Psychiatry at the University of Arizona, Tucson, AZ, USA
5 Departments of Medicine and Neurology, Brigham and Women's Hospital, Harvard Medical School, Boston, MA, USA
6 Asthma and Airway Disease Research Center, College of Medicine, The University of Arizona, Tucson, AZ, USA
7 Department of Biomedical Engineering at the University of Arizona, Tucson, AZ, USA
8 Bio5 Institute at the University of Arizona, Tucson, AZ, USA

Abstract

Sleep stage classification is pivotal for assessing sleep quality and diagnosing sleep disorders. While deep learning has shown promise in automating sleep stage classification, the dynamic nature of sleep research and data collection processes presents unique challenges. These include class-incremental learning, where models must adapt to newly emerging sleep patterns, and domain-incremental learning, necessitating generalization across diverse data collection conditions or varying EEG equipment. Moreover, the unavailability of old datasets, particularly between different institutions, and the lack of annotations in historical data hinder the effective training and adaptability of deep learning models. This study introduces a Dual-Incremental Learning Framework, amalgamating unsupervised domain adaptation, hierarchy-aware feature learning, and diffusion probabilistic model-based generative replay design to improve the generalizability of sleep stage classification models and enhance the re-analysis of historical data with updated classification schemes. We establish two benchmarks using two publicly available datasets to elucidate the challenges posed by evolving sleep datasets. The effectiveness and performance of the proposed framework are evaluated against the established benchmarks, showcasing enhanced generalizability and the potential for unearthing deeper insights from historical patient data. This framework also demonstrates compatibility with existing deep learning models, promoting a versatile solution for advancing sleep stage classification. To the best of our knowledge, this study is the first to address class and domain-incremental learning challenges in sleep stage classification, laying a solid foundation for future research in this domain. Additionally, the proposed Dual-Incremental Learning Framework holds potential for a wide range of healthcare applications.

Introduction

Sleep, a fundamental component of maintaining both physical and mental health, has garnered significant attention in recent years [1]. The criticality of monitoring and classifying sleep stages cannot be overstated, as it plays a pivotal role in diagnosing sleep disorders and formulating personalized treatment plans. Historically, trained sleep technicians have manually scored sleep stages by interpreting Polysomnography (PSG) signals, a notably labor-intensive process. The advent of deep learning has paved the way for automated sleep stage classification. By training on extensive datasets, deep learning models have demonstrated improved performance, offering real-time accuracy and valuable insights into sleep patterns and disorders through the automatic processing of raw electroencephalography (EEG) signals [2, 3, 4].

The evolution of sleep research and the diversity of data sources present two key challenges: class-incremental learning, where the model needs to adapt to new classes or categories as they emerge, and domain-incremental learning, where the model must generalize across diverse data domains such as varying data collection conditions or different types of EEG equipment. Class-incremental learning is exemplified by the transition from binary sleep-wake classifications to more nuanced multi-class sleep stage classifications as new sleep patterns and their clinical implications are recognized. Further, the identification of subtypes of sleep disorders, such as different forms of sleep apnea or arousal events, can also necessitate a class-incremental approach as the categorization complexity increases with emerging clinical knowledge. On the other hand, domain-incremental learning arises from the need to generalize across diverse data domains, such as varying data collection conditions or different types of EEG equipment. For instance, a domain shift may occur when transitioning from a clinical setting to a home sleep study or when the data is collected across different institutions with varying standard operating procedures and equipment calibrations. These examples underscore the multifaceted nature of incremental learning challenges in sleep stage scoring and highlight the need for a robust framework capable of navigating both class and domain incrementality cohesively. However, existing sleep stage classification approaches often struggle to adapt to these dynamic changes. One primary reason is the challenge of missing annotations in the old dataset, making supervised learning infeasible for re-analyzing historical data with updated scoring mechanisms to uncover granular insights. The bidirectional connection between old and new data is crucial: analyzing historical patient data with the new model can generate new insights without the cost-intensive and labor-intensive process of collecting new PSG data and adding new annotations while incorporating old datasets can enhance the model's stability and performance. This symbiotic utilization of data underscores the need for a more flexible and adaptive framework that can navigate both class and domain incrementality, effectively leveraging both historical and new data to advance sleep stage classification.

Given the identified challenges and limitations of existing models, particularly in adapting to dynamic changes and effectively leveraging historical data, we propose a Dual-Incremental Learning Framework. This framework harnesses the principles of unsupervised domain adaptation (UDA), hierarchy-aware feature learning, and diffusion probabilistic model-based generative replay design to facilitate the re-analysis of historical data with enhanced classification capabilities. It accommodates the evolving understanding of sleep stages and the variability in data domains, providing a novel pathway to enhanced generalizability in sleep stage classifications, while also unlocking the potential to unearth deeper insights from historical patient data. To initiate this exploration, we have established benchmarks using two sleep study datasets, offering a foundation for further research. These benchmarks serve as reference points for evaluating the efficacy of our Dual-Incremental Learning Framework in addressing the challenges posed by evolving sleep stage classification datasets. Experiments are conducted on the benchmarks to examine the performance of the proposed learning strategies, as well as to further study the impact of old annotation granularity and the absence of old datasets on the classification accuracy with new annotation settings. We also compared the performance of different generative models for addressing the challenge posed by the unavailability of old datasets between different institutes. An ablation study was conducted to assess the effect of each proposed component of the Dual-Incremental Learning Framework. Furthermore, our framework is designed to be compatible and can be seamlessly integrated with any existing deep learning model to extend its capability, thereby promoting a versatile solution for advancing sleep stage classification.

Materials and Methods

Materials and Benchmarks

Sleep stages can be broadly classified into two main types: Rapid Eye Movement (REM) and Non-REM (NREM) sleep [5]. NREM sleep can be further segmented into three stages based on brain activity and muscle tone characteristics. The initial stage of NREM, namely N1, represents the lightest sleep stage and is characterized by the transition from wakefulness to sleep. The subsequent stage, N2, is slightly deeper and typically constitutes around half of the sleep time in healthy adults. The third stage, N3, corresponds to the deepest phase of NREM sleep, often called slow-wave sleep, which is crucial for physical restoration and memory consolidation [6]. Conversely, REM sleep is characterized by vivid dreaming, rapid eye

Table 1: Details of the SleepEDF and SHHS dataset. Each Sample is a 30 Second EEG segment.

Dataset		SleepEDF	SHHS
Subjects		153	329
EEG Channel		Fpz-Cz	C4-A1
Sampling Rate		100	125
Num of Each Stage	Wake	65,951	46,319
	Sleep	129,528	278,535
	NREM	103,693	212,582
	N1	21,522	10,304
	N2	69,132	142,125
	N3	13,039	60,153
	REM	25,835	65,953
Total 30s Epochs		195,479	324,854

movements, and temporary muscle paralysis. Transitions between sleep stages occur continuously and cyclically, with each cycle lasting approximately 90 to 110 minutes. Figure 1 shows the hierarchy between different sleep stages. For constructing the coarse annotation of the old dataset, we aggregate N1, N2, and N3 stages into the NREM stage, while combining the NREM and REM stages under the category of sleep.

Figure 1: Illustration of the hierarchy between the sleep stages, showcasing the relationships among different sleep stages. The hierarchy diagram visually represents the interconnectedness and hierarchical structure of the sleep stages.

The sleep community has collected and released two publicly available datasets that enable research in this domain. Namely, the Sleep Heart Health Study (SHHS) [7, 8] and Sleep-EDF Database Expanded (Sleep-EDF) [9, 10] are used in this work. The details of these datasets are provided in Table 1. Sleep-EDF [10] comprises whole-night polysomnography (PSG) recordings, including EEG, EOG, chin EMG, and event markers. Each PSG file has two EEG channels, Fpz-Cz and Pz-Oz, sampled at 100 Hz. For our experiments, we used the Fpz-Cz channel as the input, resampling at a rate of 125 Hz to align with the sampling rate of the SHHS dataset. SHHS [7] is a multi-center cohort study conducted by the National Heart, Lung, and Blood Institute to investigate the cardiovascular and other impacts of sleep-disordered breathing. We chose the C4-A1 EEG channel from 329 PSG records out of 6,441 records as the input to our model, following the settings used in previous studies [4, 11]. The EEG signals in the SHHS dataset are sampled at 125 Hz.

In this work, we use the SHHS and Sleep-EDF datasets as benchmarks to evaluate the performance of the proposed strategies. Specifically, we designate the Sleep-EDF and SHHS as the old and new datasets, respectively. For Sleep-EDF, two label sets are assigned to incrementally increase the level of difficulty: Wake/Sleep and Wake/NREM/REM. Meanwhile, the SHHS dataset is labeled with a comprehensive five-stage label set: Wake/N1/N2/N3/REM. Further, we establish two benchmarks to account for varying degrees of accessibility to the old dataset (Sleep-EDF) for the new network. The first benchmark (**B1**) allows the new network access to the old dataset (Sleep-EDF); however, the label sets of Sleep-EDF remain unchanged and do not undergo relabeling to the five stages. The second benchmark (**B2**) presents a more challenging scenario that does not allow the new network to access the old dataset. In this scenario, a generative model is trained on the old dataset to produce synthetic samples. The new network is then trained using these generated samples. To evaluate the effectiveness of our proposed framework, the new network, once trained on the SHHS dataset with the detailed five-stage label set, is employed to perform a five-stage classification on the Sleep-EDF dataset. In summary, the proposed benchmark and the proposed framework shown in Figure 2.

Figure 2: Illustration of the benchmarks and proposed framework. Two types of coarse annotation are employed: Sleep/Wake and Wake/NREM/REM. The distinction between **B1** and **B2** lies in the accessibility of the old dataset.

Problem Formulation

We are provided with the old dataset $D^{old} := \left\{x_i^{old}, y_i^{old}\right\}_{i=1}^{N}$ comprising N input EEG signals x_i^{old}, and their corresponding sleep stages y_i^{old}. We use supervised learning to train the old neural network f^{old} on D^{old} by minimizing the cross-entropy loss, ℓ:

$$f^{old} = \arg\min_{f \in \mathcal{F}} \mathbb{E}_{D^{old}} \ell(f(x_i^{old}), y_i^{old}), \tag{1}$$

At a future time, we receive a new dataset $D^{new} = \{x_i^{new}, y_i^{new}\}_{i=1}^{M}$, where the input EEG signals x_i^{new} are sampled from a distribution different from that of x_i^{old}. This change can be due to changes in collection procedures, such as a new EEG collection device or patients from different sub-populations. Moreover, the new sleep stages y_i^{new} are annotated using different criteria than y_i^{old}. It is worth noting that some sleep stage annotations remain the same in both the old and new datasets (e.g., wake). Others are subcategories of the old annotations (i.e., N1/N2/N3 can be grouped as NREM, and N1/N2/N3/NREM can be grouped as sleep).

A new model f^{new} can easily be trained with the new dataset D^{new}; however, such an approach fails to capture knowledge learned in f^{old}. Our learning setting requires using the new network to classify the old dataset based on the new labeling. The challenges associated with the new network with different labeling than the old dataset are twofold. First, the distribution shift between the old and new EEG signals will result in poor generalization performance of f^{new} evaluated on D^{old}. Second, the absence of new annotations for D^{old} makes joint training infeasible even with access to the old dataset. Additionally, we assume that D^{old} will not be available once D^{new} is received, which makes training f^{new} even more challenging.

Aligning Feature Distributions via Unsupervised Domain Adaptation

We explore the application of UDA [12, 13, 14], a technique that facilitates the transfer of knowledge from a source domain (old dataset) to a target domain (new dataset) without reliance on labeled information. UDA proves valuable in scenarios where limited labeled data is available in the target domain, and the source and target domains exhibit distinct data distributions. Our objective with UDA is to mitigate domain discrepancies and enhance the model's generalization in the target domain by aligning data distributions through unsupervised learning. We operate on the premise that, despite differing distributions, commonalities and structural patterns exist between the domains. Here, we define the new and old EEG signal distributions as X^{new} and X^{old}, respectively. Based on these definitions, Ben-David et al. [13] established an upper bound on the error (i.e., $\epsilon_{old}(f)$) of a classifier f on the target domain by f's error on the source domain (i.e., $\epsilon_{new}(f)$) in addition to several other terms. The bound is expressed as follows:

$$\epsilon_{old}(f) \leq \epsilon_{new}(f) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(X^{new}, X^{old}) + C \tag{2}$$

where C is a constant term independent of the specific classifier f. The $d_{\mathcal{H}\Delta\mathcal{H}}$ term represents the $\mathcal{H}\Delta\mathcal{H}$-distance, which characterizes the discrepancy between the two classifier's decisions over the two domains. Formally, the $\mathcal{H}\Delta\mathcal{H}$-distance is defined as:

$$d_{\mathcal{H}\Delta\mathcal{H}}(X^{new}, X^{old}) = 2 \sup_{f_1, f_2 \in \mathcal{F}} |P_{x \sim X^{new}}[f_1(x) \neq f_2(x)] - P_{x \sim X^{old}}[f_1(x) \neq f_2(x)]|. \tag{3}$$

Minimizing the $\mathcal{H}\Delta\mathcal{H}$-distance in practice can be challenging and infeasible; however, recent work has developed methods to approximate this divergence. For example, Ganin and Lempitsky (2015)

Figure 3: Illustration of joint training with hierarchy learning. In this approach, we aggregate the output probabilities of the new sleep stages based on the hierarchy. This allows us to perform joint training on both the new and old datasets, leveraging the labels from the old dataset.

approximated this divergence with the Domain Adversarial Neural Network (DANN) framework [15] using JS-divergence, and it can be formulated as the following objective:

$$\min_{c,f}\max_{g}\mathbb{E}_{D^{old},D^{new}}[\ell(c\circ f^{'}(x_i^{new}),y_i^{new})+\lambda JSD(g\circ f^{'}(x_i^{new}),g\circ f^{'}(x_j^{old}))], \tag{4}$$

where we slightly change the notations by splitting the network f into the classifier c and feature extractor $f^{'}$. The term g is a domain classifier trained to maximize the domain classification error. Thus encouraging $f^{'}$ to learn domain-invariant features.

In [16], f-Domain-Adversarial Learning (namely, fDAL) enhances the training stability of UDA by minimizing an f-divergence. The f-divergence between two distribution functions P_s and P_t is defined as $D_\phi(P_s||P_t)=\int p_t(x)\phi(\frac{p_s(x)}{p_t(x)})dx$, where p_s and p_t represent the densities of P_s and P_t, respectively. The f-divergence can also be reformulated using variational forms [17] as:

$$D_\phi(P_s||P_t)\geq\sup_{T\in\mathcal{T}}\mathbb{E}_{x\in P_s}[T(x)]-\mathbb{E}_{x\in P_t}[\phi^*(T(x))], \tag{5}$$

where $T:X\rightarrow dom(\phi^*)$ is an arbitrary measurable function of the set $\mathcal{T}$, and ϕ^* is the conjugate function of ϕ. More specifically, we can incorporate fDAL into our benchmark using the following min-max objective:

$$\begin{aligned}\min_{c,f'}\max_{g}&\mathbb{E}_{D^{old},D^{new}}[\ell(c\circ f^{'}(x_i^{new}),y_i^{new})+\\&\hat{\ell}(g\circ f^{'}(x_i^{new}),c\circ f^{'}(x_i^{new}))-(\phi^*\circ\hat{\ell})(g\circ f^{'}(x_i^{old}),g\circ f^{'}(x_i^{old}))].\end{aligned} \tag{6}$$

We let $\hat{\ell}(c,b)=a(b_{\arg\max_c})$, where $a(.)$ is a monotonically increasing function. By incorporating fDAL, the new network can generate domain-invariant features for both EEGs from the old and new datasets.

Exploit Old Labels with Hierarchy Aware Features Learning

In continual learning, joint training is commonly used to establish an upper bound for neural network performance. Joint training can be considered a near-optimal solution when the old dataset is fully annotated according to our desired specifications. Unfortunately, in our case, the old dataset's EEG signals have a coarser labeling level than the new dataset, making vanilla joint training impractical. We can still leverage the old dataset's annotations to enhance the new network's performance. Given that a significant challenge in our scenario is the domain shift between the two datasets, the network must acquire valuable features from the old dataset. Hence, we can address this requirement by applying Hierarchy Aware Feature Learning [18], which integrates hierarchical information encoded in label structures into the learning process. This empowers classifiers to make semantically meaningful mistakes while minimizing overall errors by considering relationships between labels at varying levels of granularity. Joint training in our setting can be achieved as follows. The new network provides output probabilities of the new annotations to the replay data with coarse sleep stage labels. This process is illustrated in Figure 3. Specifically, the cross-entropy loss computed on the old annotations can be defined as $l(Hp,y)=-\sum_{i=1}^{C}(Hp)_i\log(y_i)$, where p represents the softmax probability vectors, and H is a matrix utilized to calculate the summation of fine-to-coarse probabilities. The construction of matrix H involves setting $H_{i,j}=1$ if the ith sleep stage in the new annotations falls under the jth sleep stage of the old annotations; otherwise, $H_{i,j}=0$.

We use knowledge distillation (KD) loss [19] to incorporate the old network's information using the old network's soft labels. KD facilitates knowledge transfer from the old (i.e., teacher) to the new (i.e., student) network. Specifically, we use the matrix H to aggregate the new network's output probabilities and then compute the Kullback-Leibler (KL) divergence $KL^\tau=\sum_i q_i^\tau\log\frac{p_i^\tau}{q_i^\tau}$ using the old network's soft labels with a temperature factor, τ. Then we apply a softmax activation function to the logits z to get probabilities to obtain $p_i^\tau = \frac{\exp(z_i/\tau)}{\sum_j \exp(z_j/\tau)}$. The final objective for the joint training with hierarchy is as follows:

$$\min_f \ \mathbb{E}_{(D^{new}, D^{old})}[l(Hf(x_i), y_i^{old})] + \mathbb{E}_{X^{old}}[KL^\tau(Hf(x_i)), f_{old}(x_i)]. \tag{7}$$

In this objective, we optimize the expectation over the new and old datasets by minimizing the cross entropy loss $l(Hf(x_i), y_i^{old})$ for the old annotations, and the KL divergence $D_{KL}^\tau(Hf(x_i)), f_{old}(x_i)$ for the old network's soft labels. By jointly considering the new and old datasets, we aim to improve the new network's performance by leveraging the knowledge acquired by the old network, and enhancing the transfer of information across different sleep stage annotations.

By incorporating UDA and joint training with label hierarchy as:

$$\begin{aligned}\min_{c,f'} \max_g \ \mathbb{E}_{(D^{new}, D^{old})}[l(Hc \circ f^{'}(x_i), y_i^{old})] + \ell(c \circ f^{'}(x_i^{new}), y_i^{new}) + \\ KL^\tau(Hc \circ f^{'}(x_i), f_{old}(x_i))] + \\ \hat{\ell}(g \circ f^{'}(x_i^{new}), c \circ f^{'}(x_i^{new})) - (\phi^* \circ \hat{\ell})(g \circ f^{'}(x_i^{old}), g \circ f^{'}(x_i^{old})).\end{aligned} \tag{8}$$

Our approach seeks to leverage the benefits of both techniques. Through UDA, we address the domain shift between the old and new datasets, allowing the new network to learn domain-invariant features and adapt to the characteristics of the target domain. Simultaneously, the joint training with hierarchy enables the new network to use the information from the old network, leveraging the fine-to-coarse annotations to enhance the performance on the coarse sleep stages. This combination of UDA and joint training with hierarchy provides a comprehensive framework that addresses the challenges posed by our sleep-stage classification scenarios. Further, our approach enables the new network to learn discriminative features from the new dataset, while benefiting from the knowledge distilled from the old network. Our approach enhances the generalization capability of the new network by jointly optimizing the UDA and hierarchy, thus, facilitating improved classification accuracy for both the old and new sleep stages.

Generative Samples with Wasserstein GAN

We have discussed the strategies used to address the distribution shift between datasets and the absence of annotations. Here, we tackle the challenge of working in a continual learning context [20, 21, 22, 23] where historical data is often unavailable in scenario **B2**. To address this, we investigate Deep Generative Replay (DGR) [24], a methodology that trains a generative model on historical data in parallel with classification models and employs synthetic replay samples when new task data becomes available. One potential solution is to use Generative Adversarial Networks (GANs) to generate realistic synthetic EEG signals [25]. GANs have demonstrated effectiveness in various biomarker classification tasks, such as augmenting imbalanced datasets for electrocardiogram (ECG) classification [26]. A GAN consists of two neural networks: a generator G and a discriminator C. These networks collaborate within a game-theoretic framework to learn the underlying distribution of the training data and generate new samples that closely resemble the data distribution. Generally, GANs are trained using the following objectives:

$$\min_G \max_C \mathbb{E}_{x \sim P_{data}}[\log C(x)] + \mathbb{E}_{z \sim P_{noise}}[\log(1 - C(G(z))]. \tag{9}$$

In this formulation, x (z) represents real (noise) data samples, P_{data} denotes the distribution of the real data, and P_{noise} denotes the distribution of the noise.

Wasserstein GANs (WGAN) [27] are a variant of the original GAN that addresses challenges associated with training. WGAN introduces a different loss function from (9). Specifically, the Wasserstein distance, also known as the Earth Mover's Distance, to measure the dissimilarity between the generated and real distributions. This modification improves the stability and reliability of GAN training (e.g., reduction of mode collapse). The objective of WGAN can be expressed as follows:

$$\min_G \max_C \mathbb{E}_{x \sim P_{data}}[C(x)] - \mathbb{E}_{z \sim P_{noise}}[C(G(z)]. \tag{10}$$

Wasserstein GAN with Gradient Penalty (WGAN-GP) [28] is an enhanced version of WGAN that incorporates gradient penalty regularization (GP) to enforce the Lipschitz continuity constraint on the discriminator. This modification improves training stability and prevents mode collapse by controlling the discriminator's power. The following equation represents the objective function of WGAN-GP:

$$\min_G \max_C \mathbb{E}_{x\sim P_{data}}[C(x)] - \mathbb{E}_{z\sim P_{noise}}[C(G(z)] + \lambda \mathbb{E}_{\hat{x}\sim Pinterp}[(||\nabla_{\hat{x}} C(\hat{x})||_2 - 1)^2] \tag{11}$$

where λ is a hyperparameter to control the strength of the gradient penalty. Here, $\hat{x}$ represents a point along the straight line connecting a real sample and a sample generated from P_{interp}. While the original WGAN-GP is an unconditional model with non-conditional probability distributions in the loss function, we aim to generate synthetic samples conditioned on labels. To achieve this, we introduce random synthetic labels y' and denote the true labels of real samples as y. Accordingly, $\hat{y}$ represents a point along the straight line connecting the real and synthetic labels, sampled from P_{interp}. The conditional version of WGAN-GP is trained using the following objective:

$$\min_G \max_C \mathbb{E}_{x\sim P_{data}}[C(x|y)] - \mathbb{E}_{z\sim P_{noise}}[C(G(z)|y^{'})] + \lambda \mathbb{E}_{\hat{x}\sim Pinterp}[(||\nabla_{\hat{x}} C(\hat{x}|\hat{y})||_2 - 1)^2], \tag{12}$$

We can generate synthetic samples conditioned on specific classes by introducing conditional labels y and y'. Thus, improving the applicability of the WGAN-GP framework to our task. This modification allows us to generate EEG signals with label-specific characteristics, facilitating more targeted analysis and classification tasks.

Generative Samples with a modified Denoising Diffusion Probabilistic Model

In the field of deep generative models, the Denoising Diffusion Probabilistic Model (DDPM) [29] belongs to a category of models that focus on converting noise into realistic data samples by progressively eliminating noise through a denoising procedure. In this approach, the training data are iteratively corrupted by introducing Gaussian noise, and the model is trained to reverse this process and restore the original data. As a result, a well-trained DDPM can generate novel data by applying a denoising process to randomly generated noise.

Specifically, DDPM encompasses two main processes: the forward process, also known as the diffusion process, where data is progressively diffused to a well-behaved distribution by adding noise, and the reverse process, which transforms noise back into a sample from the target distribution. In the forward process, a distribution denoted as q gradually introduces noise to a given data point $x_0 \sim q(x_0)$. DDPM implements the diffusion process using a fixed Markov Chain with conditional Gaussian translation at each step, defined as follows:

$$q\left(x_{1:T}|x_0\right) := \prod_{t=1}^{T} q\left(x_t|x_{t-1}\right), \quad q\left(x_t|x_{t-1}\right) := \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1},\ \beta_t\mathbf{I}\right), \tag{13}$$

where $\beta_1, \ldots, \beta_T$ represents a variance schedule, and $\mathcal{N}$ denotes the Gaussian distribution with parameters μ and Σ. In contrast, the reverse process aims to recover the initial data point x_0 from a given state x_t by reversing the diffusion process. Starting with pure Gaussian noise sampled from $p(x_T) := \mathcal{N}(x_T, \mathbf{0}, \mathbf{I})$, the reverse process is defined by the following Markov chain:

$$p_\theta\left(x_{0:T}\right) := p\left(x_T\right) \prod_{t=1}^{T} p_\theta\left(x_{t-1}|x_t\right), x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{14}$$

$$p_\theta\left(x_{t-1}|x_t\right) := \mathcal{N}\left(x_{t-1}; \boldsymbol{\mu}_\theta\left(x_t, t\right), \sigma_\theta\left(x_t, t\right)\mathbf{I}\right). \tag{15}$$

In this process, the time-dependent parameters of the Gaussian transitions are learned.

In the context of DDPM, a specific parameterization for $p_\theta\left(x_{t-1}|x_t\right)$ is proposed:

$$\boldsymbol{\mu}_\theta\left(x_t, t\right) = \frac{1}{\alpha_t}\left(x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\boldsymbol{\epsilon}_\theta\left(x_t, t\right)\right), \tag{16}$$

$$\sigma_\theta\left(x_t, t\right) = \sqrt{\tilde{\beta}_t}, \text{ where } \tilde{\beta}_t = \begin{cases} \frac{1-\alpha_{t-1}}{1-\alpha_t}\beta_t & t > 1 \\ \beta_1 & t = 1 \end{cases} \tag{17}$$

where $\boldsymbol{\epsilon}_\theta(\cdot,\cdot)$ is a learnable denoising function that estimates the noise vector $\boldsymbol{\epsilon}$ added to a noisy input x_t. The parameterization leads to an alternative loss function:

$$L(\theta) := \mathbb{E}_{t,x_0,\epsilon}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t\right)\right\|^2\right], \tag{18}$$

where $\bar{\alpha}_t$ represents the schedule of values for α_t.

Given the synthetic labels y', we proposed a modified conditional version of DDPM by estimating the true conditional data distribution $q(x_0|y')$ through modeling the conditional distribution $p_\theta(x_0|y')$. Consequently, the *reverse process* is extended as follows:

$$\begin{aligned} p_\theta\left(x_{0:T}|y^{'}\right) &:= p\left(x_T\right)\prod_{t=1}^{T} p_\theta\left(x_{t-1}|x_t, y^{'}\right), x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \\ p_\theta\left(x_{t-1}|x_t, y^{'}\right) &:= \mathcal{N}\left(x_{t-1}; \boldsymbol{\mu}_\theta\left(x_t, t|y^{'}\right), \sigma_\theta\left(x_t, t|y^{'}\right)\mathbf{I}\right). \end{aligned}$$

To accommodate the conditional aspect, a conditional denoising function $\boldsymbol{\epsilon}_\theta$ is introduced, which is conditioned on y'. This allows for the definition of the conditional loss function as follows:

$$\mathbb{E}_{x_0,\bar{\alpha},\epsilon}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}}x_0 + \sqrt{1-\bar{\alpha}}\boldsymbol{\epsilon}, y^{'}, \bar{\alpha}\right)\right\|^2\right]. \tag{19}$$

In our case, DDPM offers several advantages. DDPM can generate new conditional data samples by leveraging the learned denoising process and the synthetic labels y'. DDPM effectively captures the conditional dependencies between the generated data and the corresponding labels by modeling the conditional distribution $p_\theta(x_0|y')$. Consequently, it generates realistic and diverse samples that align with specific label conditions. Further, the conditional loss function ensures that the generated samples exhibit similarity to the noise vector while conforming to the conditioning information. Thus, the modified DDPM provides a powerful framework for conditional data generation, which makes it highly suitable for our particular application.

Experimental Configurations

This study used two neural networks: ResNet [30] and AttnSleep [4] as the old and new networks, respectively. AttnSleep was chosen for its architecture, which incorporates transformer layers [31] to effectively capture long-range dependencies. It is worthy to note that the main goal of this paper is to discover the solutions for the challenges of sleep stage classification with learning from evolving datasets. Therefore, choosing different networks of two representative architectures could better demonstrate the practical scenarios. Both neural networks were trained with a batch size of 128 using the Adam optimizer [32]. The initial learning rate was set to 1×10^{-3} and then reduced to 1×10^{-4} after ten epochs. To mitigate overfitting, a weight decay of 1×10^{-3} was applied within Adam. The class-aware cross-entropy loss is used to train both networks on their respective datasets [4]. This loss function, denoted as $\ell(p, q) = \sum_i w_k q_i \log p_i$, includes precalculated weights w_k for each class k. These weights are crucial to address class imbalance and ensure balanced learning during training.

We introduced an additional batch of size 128, sampled from either the old or synthetic datasets, to train the new network. For **B1**, the entire old dataset is used, while for **B2**, the generative model generates 100 batches which sum up to 12800 samples with equal numbers of each stage. This approach allowed us to evaluate the new network's performance under different conditions. The central learning objective is a linear combination of Equations 6 and 7. We adopted the same configuration of fDAL as described in the original paper [16], by using the Pearson χ^2 function and its conjugate. The generator and discriminator architectures in the WGAN are reported in Table 2. The training procedure proposed by [33] is used to train the WGAN. Additionally, the architecture of the UNet Backbone [34] used in the modified DDPM (see Figure 4). The training procedures used in our previous work [35] for ECG reconstruction were replicated.

In the Ablation Study, we focused on Benchmark **B1** and used the wake/REM/NREM old label set to explore the impact of different components of the proposed framework on the model's performance. We

Table 2: Architecture of WGAN-GP. The configuration of Conv1D is specified by the following parameters: Output Channel, Kernel Size, Stride, and Padding.

Generator		Discriminator	
Embedding 5→28	Randn 1×100	Embedding 5→100 Linear 3750	EEG signal 1×3750
Input size: 1×128		Input size: 2×3750	
Linear 128 BatchNorm,ReLU		Conv1D 32,4,2,0 InstanceNorm,ReLU	
Linear 256 BatchNorm,ReLU		Conv1D 64,4,2,0 InstanceNorm,ReLU	
Linear 512 BatchNorm,ReLU		Conv1D 128,4,2,0 InstanceNorm,ReLU	
Linear 3750 BatchNorm,ReLU		Conv1D 256,4,2,0 InstanceNorm,ReLU	
Conv1D 32,4,1,2 BatchNorm,ReLU		Conv1D 512,4,2,0 InstanceNorm,ReLU	
Conv1D 64,4,1,2 BatchNorm,ReLU		Conv1D 1,1,1,0 InstanceNorm,ReLU	
Conv1D 128,4,1,2 BatchNorm,ReLU			
Conv1D 1,4,1,0 BatchNorm,ReLU			

Figure 4: UNet backbone architecture for modified DDPM. The main structure (a) is composed of downsampling modules (c) and upsampling modules (d), which include the double convolution module (b).

evaluated five distinct strategies: **S1**, which represents the proposed duo-incremental learning framework combining UDA and hierarchy joint training; **S2**, denoting hierarchy joint training alone; **S3**, reflecting the usage of UDA exclusively; **S4**, representing hierarchy joint training without KD loss; and **S5**, a variation of the proposed duo-incremental learning framework without KD loss.

All experiments were run on an NVIDIA RTX 3090 GPU. To ensure fair comparisons, fixed random seeds were used throughout the experiments. Further, we performed five-fold cross-validation and report the average performance metric.

Evaluation Metrics

We use multiple figures of merit to assess performance. These metrics include per-class F1-score (F1), accuracy, the area under the Receiver Operating Characteristic curve (AUROC), and the area under the Precision-Recall curve (AUPRC). Each metric provides valuable insights into the strengths and limitations of the model [36]. We denote true positive predictions as TP, true negative predictions as TN, false positive predictions as FP, and false negative predictions as FN. Precision (P) is calculated as $\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$, while Recall (R) is calculated as $\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$.

The *F1 Score*, a widely recognized metric for binary classification tasks, balances precision and recall, and is calculated as $\mathrm{F1} = 2 \times \frac{\mathrm{P}\times\mathrm{R}}{\mathrm{P}+\mathrm{R}}$. This score is the harmonic mean of precision and recall. We compute the F1 score for each sleep stage and then average F1 score across all stages (MF1). The *Accuracy* is another commonly used metric, representing the proportion of correct predictions out of the total number of predictions. The accuracy is calculated as $\mathrm{Acc} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}$, providing a high-level assessment of the model's correctness. Further, we also use *AUROC*, which assesses performance at distinguishing between positive and negative instances across various classification thresholds. AUROC is computed by integrating the Receiver Operating Characteristic (ROC) curve. In scenarios with class imbalance, such as sleep stage classification, *AUPRC* holds significance. It considers the precision and recall at different thresholds, comprehensively evaluating the classifier's performance. AUPRC is calculated by integrating the precision-recall curve. By using this diverse set of evaluation metrics, we gain an understanding of our model's performance and suitability for automatic sleep stage classification.

Results

We conducted a comprehensive evaluation of the AttnSleep model's performance in five-stage classification using the Sleep-EDF dataset. The evaluation was carried out in two distinct settings: **B1** and **B2**. We compared our approach against two different baselines: Joint Training and SHHS Only. These baselines provide upper and lower bounds for assessing our model's performance. Joint Training serves as a loose upper bound, while SHHS Only represents a lower bound. The lower bound baseline is designed to establish a performance threshold that represents the minimal achievable results. This baseline typically involves naive approaches that do not incorporate advanced techniques or use all available resources. Comparing our method against the lower bound helps us understand how much our approach surpasses or outperforms the minimal expectations. On the other hand, the upper bound represents ideal or optimal performance. By comparing our method against the upper bound, we can identify any gaps or limitations in our approach and determine areas for further improvement.

In **B1**, our proposed strategy demonstrates impressive performance, as shown in Table 3, surpassing the lower bound without the need for manual relabeling of the old dataset. Notably, when the old dataset is annotated with sleep/wake stages, our approach yields gains ranging from 0.02 to 0.1 across various metrics. These gains increase to 0.05 to 0.2 when the old dataset has wake/REM/NREM annotations. These outcomes underscore the effectiveness of our approach, which capitalizes on its ability to leverage knowledge from the old dataset.

Benchmark **B2** is a more challenging setting as it assumes limited or no access to historical data, making it valuable for practical applications where acquiring old data may be costly or unfeasible. In this scenario, our approach continues to deliver favorable results by using a generative model to synthesize the old data. Particularly, we compare the performance using two different generators: WGAN-GP and DDPM.

Table 3: Main Results on **B1**.

	Per-class F1					Overall Metrics			
	Wake	N1	N2	N3	REM	Accuracy	AUROC	AUPRC	MF1
SHHS Only (Baseline)	0.8473	0.1738	0.6468	0.5716	0.4923	66.20	0.7727	0.6371	0.6386
Joint training (Upper Bound)	0.8971	0.4282	0.8100	0.7672	0.7060	76.92	0.8221	0.7577	0.7610
with sleep/wake	0.8948	0.1980	0.7215	0.6781	0.5804	71.28	0.7937	0.6944	0.7060
with wake/REM/NREM	0.9070	0.2381	0.7686	0.7119	0.6977	75.88	0.8069	0.7315	0.7580

Table 4: Main Results on **B2**.

		WGAN-GP		DDPM	
		with sleep/wake	with wake/REM/NREM	with sleep/wake	with wake/REM/NREM
Per-class F1	Wake	0.8273	0.8729	0.8617	0.8990
	N1	0.1588	0.1648	0.2141	0.2158
	N2	0.5768	0.6901	0.6921	0.7231
	N3	0.4778	0.5929	0.6582	0.6142
	REM	0.4194	0.6223	0.4659	0.6772
Overall Metrics	Accuracy	58.83	69.03	67.12	72.25
	AUROC	0.7399	0.7812	0.7719	0.7957
	AUPRC	0.6148	0.6800	0.6568	0.7033
	MF1	0.5855	0.6916	0.6661	0.7227

In **B2**, as shown in Table 4, DDPM outperforms WGAN-GP. When the old dataset is labeled with sleep/wake stages, our approach with DDPM achieves a 0.02 MF1 gain over SHHS, signifying its effectiveness in mitigating the challenges posed by the absence of historical data. Notably, when the old dataset is annotated with wake/REM/NREM stages, the performance improvement is even more substantial, with a 0.08 MF1 gain over SHHS. Examining class-wise evaluations in **B2**, we observe that DDPM maintains performance levels similar to those in B1, with only minor per-class F1 decreases (ranging from 0.01 to 0.02) on classes such as Wake and REM. These results underline the resilience of our approach in facilitating hierarchy-aware feature learning for the new task while preserving valuable information from the old dataset.

In both benchmarks, our method consistently outperforms the lower bound and demonstrates competitive performance against the upper bound, even when faced with evolving data distributions, label granularity changes, and the unavailability of historical data. This highlights the potential of our approach in the context of sleep stage classification with learning from evolving datasets.

The results of the ablation study, as presented in Table 5, reveal the influence of altering or removing different components of our proposed strategy in **S2** to **S5**. In all scenarios where elements of our approach were replaced or omitted, there was a noticeable reduction in both class-wise and overall performance.

For instance, the experiments conducted with **S2** to **S5** resulted in a degradation of the Mean F1 Score (MF1) ranging from 0.02 to 0.2. The impact is particularly pronounced in the class-wise evaluation, with the most significant decline observed in class N1. In this specific class, **S1** achieved per-class F1 scores that were notably larger, showing an improvement ranging from 2x to 7x when compared to the outcomes obtained using **S2**, **S5**, and **S4**.

These findings underscore the critical importance of our comprehensive design to effectively address the challenges of classification in the presence of shifting data distributions. The integration of UDA, hierarchy joint training, and KD loss, as embodied in S1, proved to be the most effective strategy for enhancing the model's performance in adapting to evolving datasets. This study emphasizes the significance of each component and their synergy in achieving superior results in sleep stage classification.

Table 5: Ablation Study on the Integration of Proposed Learning Strategies.

		S1	S2	S3	S4	S5
Per-class F1	Wake	0.9070	0.9042	0.5777	0.8994	0.8936
	N1	0.2381	0.0793	0.1781	0.0299	0.1075
	N2	0.7686	0.5995	0.6328	0.6152	0.7579
	N3	0.7119	0.5133	0.6713	0.5166	0.6955
	REM	0.6977	0.6916	0.4231	0.6774	0.6668
Overall Metrics	Accuracy	75.88	66.73	55.47	66.92	73.89
	AUROC	0.8069	0.8029	0.7270	0.7962	0.8019
	AUPRC	0.7315	0.7159	0.5852	0.6982	0.7123
	MF1	0.7580	0.6663	0.5481	0.6690	0.7352

Discussion

This research introduces a Dual-Incremental Learning Framework poised to tackle the inherently dynamic nature of sleep stage classification, focusing on evolving datasets. The proposed framework addresses the complexities introduced by changing label sets and data distributions, a prevalent challenge in machine learning models applied to real-world scenarios. Our empirical analysis substantiates that this framework can reach performance metrics on par with those achieved through joint training on old datasets that have been reannotated. This finding is significant as it showcases the framework's ability to update and fine-tune its learning in response to new data without the necessity for cumbersome dataset relabeling.

An ablation study was pivotal in our research, shedding light on the unique contribution of each component within our learning strategies. By isolating and evaluating the effects of different elements, it becomes evident that the synergistic integration of the various strategies culminates in superior classification performance. This integration proves to be particularly effective in scenarios where access to historical datasets is constrained, thus enhancing the practical applicability of the model. Notably, the implementation of the modified DDPM illustrates a viable pathway to circumvent these constraints, allowing the classifier to assimilate valuable historical data insights while adapting to the present conditions.

The necessity for an adaptive approach in sleep stage classification is accentuated by the continuously evolving definitions and criteria in the field, as noted by [37] and recent trends in classification tasks [38]. Changes between institutes, often spurred by new technologies and demographic shifts, present additional layers of complexity. Traditionally, the sleep stage classification models are trained once and then deployed, which limits their ability to adapt to these dynamic changes. The Dual-Incremental Learning Framework, which can be seamlessly integrated with existing sleep stage classification approaches, is designed to better accommodate the dynamic and evolving nature of real-world scenarios. Additionally, the framework's capability to navigate through privacy considerations, policy barriers, and discontinuation of collaborations highlights its robustness and the versatility of its application.

Moreover, the potential utility of this study extends beyond sleep stage classification, serving as a template for addressing broader issues that permeate various fields where data evolution is a constant. The demand for algorithms that can dynamically adapt to shifts in data characteristics and classification contexts is burgeoning, given the accelerating changes brought about by technological advancements, behavioral shifts, and environmental influences.

However, the quest for refinement persists. The suboptimal performance observed in distinguishing the N1 sleep stage warrants further inquiry into specialized methodologies aimed at enhancing discernment in this and similarly challenging categories. Additionally, the performance discrepancy when using generative models versus actual historical data prompts an exploration into more sophisticated data synthesis techniques that could bridge this gap.

The avenues for future research are manifold. The exploration of the Dual-Incremental Learning Framework's applications across various healthcare domains promises to unearth insights beneficial to disease diagnosis, patient monitoring, and more. Such endeavors will involve tailoring the learning strategies to the nuanced requirements of different datasets and classification tasks within those domains, a pursuit that holds the potential to amplify the impact of this research significantly.

Conclusion

In conclusion, this study confronts the intricacies of sleep stage classification in the presence of evolving datasets, constructing benchmarks from prevalent sleep datasets to simulate real-world scenarios and rigorously evaluate algorithmic performance. Through a confluence of UDA, Hierarchy-Aware Feature Learning, and advanced deep generative models, we have delineated a duo-incremental learning framework that adeptly negotiates the vagaries of data evolution. The promising results obtained here underline the framework's capacity for precise and adaptive classification in the face of shifting data landscapes and incomplete historical data. Ultimately, this work marks a substantive stride forward in sleep stage classification and lays a foundation for similar approaches to be applied across an array of healthcare-related classification challenges, setting the stage for further investigative forays into the realm of dynamic data adaptation.

References

[1] Luyster FS, Strollo Jr PJ, Zee PC, Walsh JK. Sleep: a health imperative. Sleep. 2012;35(6):727-34. (document)

[2] Supratak A, Dong H, Wu C, Guo Y. DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2017;25(11):1998-2008. (document)

[3] Mousavi S, Afghah F, Acharya UR. SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach. PloS one. 2019;14(5):e0216456. (document)

[4] Eldele E, Chen Z, Liu C, Wu M, Kwoh CK, Li X, et al. An attention-based deep learning approach for sleep stage classification with single-channel EEG. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2021;29:809-18. (document)

[5] Patel AK, Reddy V, Araujo JF. Physiology, sleep stages. In: StatPearls [Internet]. StatPearls Publishing; 2022. . (document)

[6] Wagner U, Born J. Memory consolidation during sleep: interactive effects of sleep stages and HPA regulation. Stress. 2008;11(1):28-41. (document)

[7] Quan SF, Howard BV, Iber C, Kiley JP, Nieto FJ, O'Connor GT, et al. The sleep heart health study: design, rationale, and methods. Sleep. 1997;20(12):1077-85. (document)

[8] Zhang GQ, Cui L, Mueller R, Tao S, Kim M, Rueschman M, et al. The National Sleep Research Resource: towards a sleep data commons. Journal of the American Medical Informatics Association. 2018;25(10):1351-8. (document)

[9] Kemp B, Zwinderman AH, Tuk B, Kamphuisen HA, Oberye JJ. Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the EEG. IEEE Transactions on Biomedical Engineering. 2000;47(9):1185-94. (document)

[10] Goldberger AL, Amaral LA, Glass L, Hausdorff JM, Ivanov PC, Mark RG, et al. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. circulation. 2000;101(23):e215-20. (document)

[11] Fonseca P, Den Teuling N, Long X, Aarts RM. Cardiorespiratory sleep stage detection using conditional random fields. IEEE journal of biomedical and health informatics. 2016;21(4):956-66. (document)

[12] Ben-David S, Blitzer J, Crammer K, Pereira F. Analysis of representations for domain adaptation. Advances in neural information processing systems. 2006;19. (document)

[13] Ben-David S, Blitzer J, Crammer K, Kulesza A, Pereira F, Vaughan JW. A theory of learning from different domains. Machine learning. 2010;79:151-75. (document)

[14] Ganin Y, Ustinova E, Ajakan H, Germain P, Larochelle H, Laviolette F, et al. Domain-adversarial training of neural networks. The journal of machine learning research. 2016;17(1):2096-30. (document)

[15] Ganin Y, Lempitsky V. Unsupervised domain adaptation by backpropagation. In: International conference on machine learning. PMLR; 2015. p. 1180-9. (document)

[16] Acuna D, Zhang G, Law MT, Fidler S. f-domain adversarial learning: Theory and algorithms. In: International Conference on Machine Learning. PMLR; 2021. p. 66-75. (document)

[17] Nguyen X, Wainwright MJ, Jordan MI. Estimating divergence functionals and the likelihood ratio by convex risk minimization. IEEE Transactions on Information Theory. 2010;56(11):5847-61. (document)

[18] Garg A, Sani D, Anand S. Learning Hierarchy Aware Features for Reducing Mistake Severity. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIV. Springer; 2022. p. 252-67. (document)

[19] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv preprint arXiv:150302531. 2015. (document)

[20] Goodfellow IJ, Mirza M, Xiao D, Courville A, Bengio Y. An empirical investigation of catastrophic forgetting in gradient-based neural networks. arXiv preprint arXiv:13126211. 2013. (document)

[21] Kirkpatrick J, Pascanu R, Rabinowitz N, Veness J, Desjardins G, Rusu AA, et al. Overcoming catastrophic forgetting in neural networks. Proceedings of the national academy of sciences. 2017;114(13):3521-6. (document)

[22] Lee SW, Kim JH, Jun J, Ha JW, Zhang BT. Overcoming catastrophic forgetting by incremental moment matching. Advances in neural information processing systems. 2017;30. (document)

[23] Li Z, Hoiem D. Learning without forgetting. IEEE transactions on pattern analysis and machine intelligence. 2017;40(12):2935-47. (document)

[24] Shin H, Lee JK, Kim J, Kim J. Continual learning with deep generative replay. Advances in neural information processing systems. 2017;30. (document)

[25] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. Communications of the ACM. 2020;63(11):139-44. (document)

[26] Adib E, Afghah F, Prevost JJ. Arrhythmia Classification Using CGAN-Augmented ECG Signals. In: 2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE; 2022. p. 1865-72. (document)

[27] Arjovsky M, Chintala S, Bottou L. Wasserstein generative adversarial networks. In: International conference on machine learning. PMLR; 2017. p. 214-23. (document)

[28] Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville AC. Improved training of wasserstein gans. Advances in neural information processing systems. 2017;30. (document)

[29] Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems. 2020;33:6840-51. (document)

[30] Humayun AI, Sushmit AS, Hasan T, Bhuiyan MIH. End-to-end sleep staging with raw single channel EEG using deep residual convnets. In: 2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI). IEEE; 2019. p. 1-5. (document)

[31] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. Advances in neural information processing systems. 2017;30. (document)

[32] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:14126980. 2014. (document)

[33] Adib E, Fernandez A, Afghah F, Prevost JJ. Synthetic ECG Signal Generation using Probabilistic Diffusion Models. arXiv preprint arXiv:230302475. 2023. (document)

[34] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. Springer; 2015. p. 234-41. (document)

[35] Li H, Ditzler G, Roveda J, Li A. DeScoD-ECG: Deep Score-Based Diffusion Model for ECG Baseline Wander and Noise Removal. IEEE Journal of Biomedical and Health Informatics. 2023. (document)

[36] Fawcett T. An introduction to {ROC} analysis. Pattern Recognition Letters. 2006;27:861-74. (document)

[37] Berry RB, Brooks R, Gamaldo C, Harding SM, Lloyd RM, Quan SF, et al.. AASM scoring manual updates for 2017 (version 2.4). American Academy of Sleep Medicine; 2017. (document)

[38] Cay G, Ravichandran V, Sadhu S, Zisk AH, Salisbury A, Solanki D, et al. Recent Advancement in Sleep Technologies: A Literature Review on Clinical Standards, Sensors, Apps, and AI Methods. IEEE Access. 2022. (document)