Title
DEGSleepNet: A Dual Evolving Graph Network for EEG-Based Single-Channel Automatic Sleep Staging

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 48(0)

Authors
Wu, Xi
Xu, Yiwei
Cui, Xinzhong
et al.

Publication Date
2026

DEGSleepNet: A Dual Evolving Graph Network for EEG-Based Single-Channel Automatic Sleep Staging

Xi Wu[1], Yiwei Xu[2], Xinzhong Cui[1], Zhiying Long[1](friskying@163.com)
[1]School of Artificial Intelligence, Beijing Normal University, Beijing, China
[2]State Key Laboratory for Fine Exploration and Intelligent Development of Coal Resources, China University of Mining and Technology-Beijing, Beijing, China

Abstract

The development of brain–computer interfaces (BCI) has provided a solid data foundation for sleep staging. Transformer-based methods have achieved moderate sleep staging by capturing temporal dependencies within time-frequency representations. However, Transformers with substantial computational overhead are limited to capturing pairwise temporal dependencies rather than group-wise temporal dependencies. To address these issues, we propose a Dual Evolving Graph Network (DEGSleepNet) for sleep staging. DEGSleepNet consists of multiple Mamba with one-dimension inverse discrete cosine transform (MambaIDCT) blocks and dual evolving graph (DEvoGraph) blocks. DEvoGraph consists of a time EvoGraph and a frequency EvoGraph. Time EvoGraph sequentially captures both group-wise and pairwise temporal dependencies, while frequency EvoGraph does the same for frequency dependencies. Extensive experiments demonstrate DEGSleepNet achieves the best performance on public sleep datasets. Notably, DEGSleepNet has few model parameters, making it suitable for sleep monitoring on wearable devices.

Keywords: DEvoGraph; Sleep Staging; Group-wise; Mamba; Time-frequency

Introduction

Sleep health has become a central issue in global public health (Grandner & Fernandez, 2021; Lim et al., 2023). Accurate sleep staging is the key prerequisite for evaluating sleep quality. Polysomnography (PSG), as a standard BCI tool for sleep recording, includes sleep-related signals such as EEG, EOG, among others, providing a multimodal foundation for sleep staging (Bruyneel & Ninane, 2014). The irregularity of sleep patterns causes expert annotations to lack objectivity and timeliness. Thus, there is an urgent need for a high-precision automatic sleep staging method to overcome these problems.

Sleep staging based on 30s single-channel EEG signals for lightweight design has become a prominent research focus (Birrer et al., 2024). Many deep learning approaches based on CNNs or RNNs have been proposed for sleep staging using raw time-domain data (Eldele et al., 2021; Sun et al., 2018; Supratak & Guo, 2020; Supratak et al., 2017). CNNs are capable of capturing local temporal dependencies by adapting convolutional kernels over short time windows, whereas RNNs further capture sequential dependencies across features. These approaches achieve limited performance due to ignoring the frequency information inherent in data. Some Transformer-based works have started using time-frequency images as an alternative to time-domain signals to capture global pairwise temporal dependencies based on attention mechanism (Dai et al., 2023; Phan et al., 2022). However, they cannot capture group-wise higher-order dependencies that multiple time-point or frequency-point features exhibit collaborative interactions for sleep staging.

Graph-based structures exhibit significant advantages in modeling pair-level and group-level interactions within EEG (Klepl et al., 2024). Normal Graph (NorGraph) relies on predefined or learnable adjacency relationships to model pairwise dependencies between nodes (Chen et al., 2025; Zhang et al., 2025). However, such structures struggle to capture group-wise high-order dependencies between nodes. Several studies adopted HyperGraph to improve modeling of high-order dependencies (Dong et al., 2025; Ji et al., 2022). While HyperGraph captures group-wise dependencies, it falls short in direct pairwise interactions between nodes, thereby failing to emphasize strong associations and limiting performance.

The work focuses on the design of an EvoGraph model, aiming to optimize time-frequency learning by leveraging group-wise dependencies of HyperGraph module and pairwise dependencies of NorGraph module. The works by Eldele et al. (2021), Sun et al. (2018), Supratak and Guo (2020), and Supratak et al. (2017) focus on capturing local temporal dependencies within time-domain data using CNNs. The works by Phan et al. (2022) and Dai et al. (2023) employ Transformers to jointly consider both temporal and frequency information within time-frequency data. However, they essentially focus on capturing global pairwise temporal dependencies based on attention mechanism. While the present study is related to recent approaches based on graph networks (Chen et al., 2025; Dong et al., 2025; Ji et al., 2022; Li et al., 2025; Zhang et al., 2025), it capitalizes on integrating the advantages of NorGraph module and HyperGraph module, which was not considered in these studies.

In this paper, our main contributions are: 1) We construct MambaIDCT module to integrate the ability of capturing temporal by Mamba and time-frequency by IDCT dependencies. 2) Two kinds of EvoGraph within designed DEvoGraph sequentially capture group-wise by HyperGraph and pairwise by NorGraph dependencies. 3) We conduct extensive experiments on SleepEDF20, SleepEDF78 and SHHS sleep datasets to verify and visualize the effectiveness of our DEGSleepNet.

Corresponding author: Zhiying Long (friskying@163.com). Xi Wu and Yiwei Xu are co-first authors and contribute equally.

In G. Boleda, I. Dautriche, M. Maldonado, C. Saldana and T. Torrent (Eds.), *Proceedings of the 48th Annual Conference of the Cognitive Science Society*

Methodology

In this section, we introduce the proposed DEGSleepNet. First, we describe the overall architecture. Then, we provide details of MambaIDCT block and DEvoGraph block.

Overall Architecture

The architecture of the proposed DEGSleepNet model is illustrated in Figure. 1. The DEGSleepNet model consists of two core blocks. In the MambaIDCT block, we use a Mamba module to capture global temporal dependencies and apply a 1DIDCT module to capture time-frequency dependencies. In the DEvoGraph block, we design a time-wise EvoGraph module to capture group-wise and pairwise temporal dependencies, and a frequency-wise EvoGraph module to capture group-wise and pairwise frequency dependencies. Detailed design of these modules is described in the following sections.

MambaIDCT Block

The MambaIDCT block consists of two main modules: Mamba module and 1DIDCT module. The two modules are linked through residual connections, thereby fusing the ability to capture temporal and time–frequency dependencies.

Mamba module: The correlations among time-point features within time–frequency data are not fully connected. The self-attention mechanism of Transformers has high computational complexity and requires considering correlations among all nodes during computation, making it difficult to directly ignore irrelevant ones. In contrast, the selective mechanism of Mamba can suppress irrelevant nodes and highlight more informative nodes, thereby achieving more efficient temporal dependencies modeling. In a continuous system, structured state space sequence models map a 1D sequence $x(t) \in \mathbb{R} \mapsto y(t) \in \mathbb{R}$ through an implicit latent state $h(t) \in \mathbb{R}^N$. The system uses $A \in \mathbb{R}^{N\times N}$ as the evolution parameter, $B \in \mathbb{R}^{N\times N}$ and $C \in \mathbb{R}^{N\times N}$ as projection parameters. After zero-order hold or other discretization, we obtain the corresponding discretization parameters $(\bar{A}, \bar{B}, C)$, thereby formulating the model as follows:

$$\begin{aligned} h_t &= \bar{A}h_{t-1} + \bar{B}x_t \\ y_t &= Ch_t \end{aligned} \tag{1}$$

The model can be reformulated via a global convolution:

$$\begin{aligned} \bar{K} &= (C\bar{B}, C\bar{A}\bar{B}, \ldots, C\bar{A}^k\bar{B}, \ldots) \\ y &= x * \bar{K} \end{aligned} \tag{2}$$

where k is the length of the sequence x, $\bar{K}$ represents the convolution kernel and $*$ denotes the convolution operator.

1DIDCT module: In the time–frequency representation, each time-point feature corresponds to a frequency distribution. Since different time-point features contribute unequally to sleep staging, we adjust the importance of time points by leveraging their corresponding frequency distributions, thereby modeling time–frequency dependencies. First, the 1DIDCT performs the transformation of time-point feature from frequency domain to time domain based on the IDCT with an selected frequency range, expressed as follows:

$$\ddot{x}_i\,(l) = \sum_{z\in Coe} x_i(z) \cdot \phi_z\,(l) \tag{3}$$

where $\ddot{x}_i$ denotes the recovered time-domain feature of the i-th time point within time-frequency feature $x \in \mathbb{R}^{T\times L}$. l denotes the time-domain index. z denotes a frequency index and Coe denotes the set of sleep-related frequency indices.

$$\phi_z\,(l) = \cos\left(\frac{\pi}{L}\left(l + \frac{1}{2}\right)z\right) \tag{4}$$

where ϕ_z denotes a basis function of the z-th frequency index.

Let $U = [u_1, u_2, \ldots, u_{T-1}] \in \mathbb{R}^T$ denote the amplitude sum vector, where $u_i = \sum_l \ddot{x}_i\,(l)$ represents the amplitude sum of $\ddot{x}_i$. Then, a time-wise weight vector $W = [w_1, w_2, \ldots, w_{T-1}] \in \mathbb{R}^T$ is generated from the U via a two-layer fully connected layers, formalized by:

$$W = Sigmoid(FC(Relu(FC(U)))) \tag{5}$$

Let $\vec{x} = [\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_{T-1}] \in \mathbb{R}^{T\times L}$ denote time-wise adjusted feature, where $\vec{x}_i^{(c)}$ at the i-th time point is computed:

$$\vec{x}_i = Conv1D(w_i \cdot x_i) \tag{6}$$

DEvoGraph Block

The DEvoGraph block consists of two main modules: time EvoGraph module and frequency EvoGraph module.

Time EvoGraph: Sleep staging relies on frequency variations over time, while the contributions of different time points exhibit complex and synergistic relationships. The learning of time EvoGraph mainly consists of four stages: hyperedge construction, hypergraph learning, edge degradation and normal graph (norgraph) learning.

Hyperedge construction: First, we define a hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where V denotes set of nodes and $\mathcal{E}$ denotes set of edges. Each time-point feature x_i from time-frequency feature is used as the feature of a hypergraph node. Then we initialize a set of learnable hyperedge embeddings which form a hyperedge codebook $HC = \{hc_1, hc_2, \ldots, hc_R\}$. Then, we compute the similarity matrix $S \in \mathbb{R}^{T\times R}$ based on time-wise nodes and hyperedge codes as follows:

$$s_{i,r} = cos\,(x_i, hc_r) = \frac{x_i^T hc_r}{\|x_i\|_2\,\|hc_r\|_2} \tag{7}$$

where $s_{i,r} \in S$ denotes the similarity between hypergraph node x_i and code hc_r. To prevent hyperedges from connecting an excessive number of nodes, we apply a Top-K sparsification to similarity matrix, resulting in a sparse incidence matrix H. The process of constructing hyperedges can be viewed as a clustering-like procedure. Each hyperedge is assumed to group time-point features that exhibit high similarity.

HyperGraph learning: Let $M \in \mathbb{R}^{T\times L}$ denote the node feature matrix and $H \in \mathbb{R}^{T\times R}$ denote the incidence matrix. We first aggregate node features into hyperedge features by:

$$M_H = H^T M \tag{8}$$

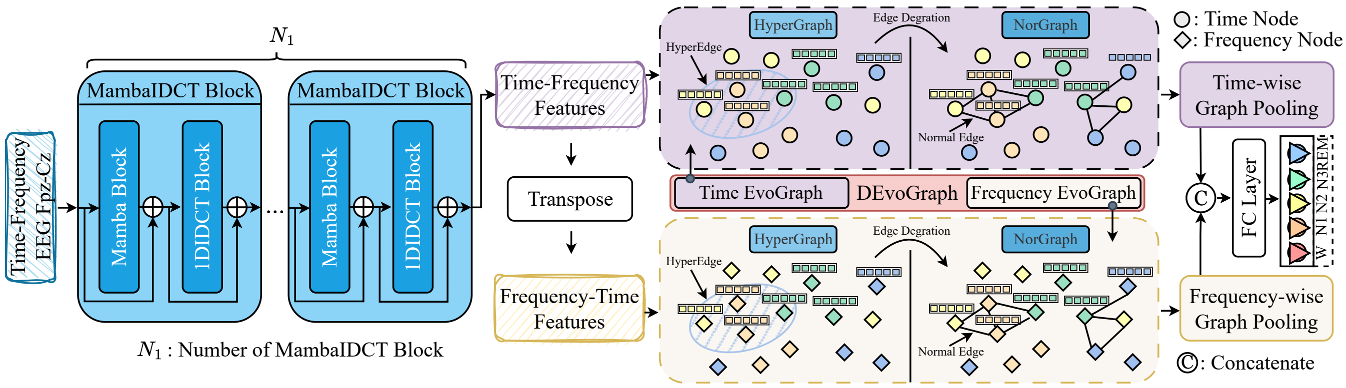


Figure 1: The overall architecture of the proposed DEGSleepNet for sleep stage classification.

where $M_H \in \mathbb{R}^{R\times L}$ denotes the hyperedge feature matrix. Then, the hyperedge features are propagated back to the connected nodes computed by:

$$\bar{M} = HM_H = HH^T M \tag{9}$$

where $\bar{M} \in \mathbb{R}^{T\times L}$ denotes the updated node features matrix.

Edge degradation: The fundamental difference between hypergraph structure and normal graph structure lies in the number of nodes associated with each edge. A hyperedge can connect multiple nodes simultaneously, whereas a normal edge can only connect a pair of nodes. As a result, the hypergraph can capture group-wise interactions, while the normal graph can capture pairwise interactions. We construct a norgraph $\mathcal{G} = (\mathcal{V}', \mathcal{E}')$, where $\mathcal{V}'$ denotes set of nodes and $\mathcal{E}'$ denotes set of edges. The adjacency matrix $A \in \mathbb{R}^{T\times T}$ of the corresponding normal graph $\mathcal{G}$ is obtained by, $A = HH^T$.

NorGraph learning: Let $\bar{M} \in \mathbb{R}^{T\times L}$ denote the node feature matrix and $\bar{A} \in \mathbb{R}^{T\times T}$ denote the normalized adjacency matrix. Node features are updated by aggregating information from neighboring nodes using the learned adjacency matrix. The updated feature matrix $\widetilde{M} \in \mathbb{R}^{T\times L}$ is computed as:

$$\widetilde{M} = \bar{A}\bar{M} \tag{10}$$

Then, a time-wise graph pooling (mean operation) is applied to the feature matrix to obtain the time feature $f_{time} \in \mathbb{R}^L$.

Frequency EvoGraph: The same frequency varies over time, while different frequencies cooperate in determining the sleep stages. Therefore, we construct frequency–time features through matrix transposition. Each row in a frequency–time feature represents a frequency-point feature. The frequency EvoGraph is constructed in the same manner as the Time EvoGraph, except that it uses frequency-point features as graph node features. The learned frequency–time features are then processed with frequency-wise graph pooling to obtain the final frequency feature $f_{freq} \in \mathbb{R}^T$.

Finally, the concatenated time and frequency features serve as the input to a fully connected layer for classification.

Table 1: Distribution of sleep stages on three sleep datasets.

Dataset	W	N1	N2	N3	REM	Total
SleepEDF20	9118	2804	17799	5703	7717	43141
	21.1%	6.5%	41.3%	13.2%	17.9%	–
SleepEDF78	66822	21522	69132	13039	25835	196350
	34.0%	11.0%	35.2%	6.6%	13.2%	–
SHHS	46319	10304	142125	60153	65953	324854
	14.3%	3.2%	43.8%	18.5%	20.3%	–

Experiments

Datasets and Implementation Details

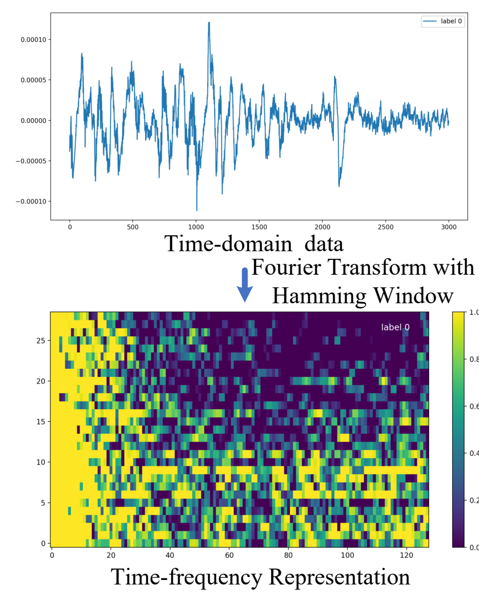


Figure 2: Preprocessing methods and data views for the Wake (label 0) Stage.

To assess the effectiveness of the proposed DEGSleepNet, we conduct experiments on three public sleep datasets. Table 1 shows the distribution of sleep stages on applied datasets.

SleepEDF20 (Kemp et al., 2000) dataset was re-annotated according to the AASM standard into five sleep stages for 20 healthy subjects. In this experiment, we used Fpz-Cz EEG and EOG channel with the highest relevance to sleep stages. Only the 30-minute W stages preceding sleep onset and following awakening were retained. Due to its limited size, this dataset is more suitable for initial model development and validation.

SleepEDF78 (Kemp et al., 2000) dataset consists of 78 subjects, and is an expanded version of SleepEDF20. Similar to SleepEDF20, the same sleep label adjustments and channel selection are applied. This dataset provides more diverse data, making it more suitable for evaluating the generalizability of the model.

SHHS (Quan et al., 1997) dataset consists of 329 subjects with regular sleep. Similar to SleepEDF20, the same label adjustments are applied. The C4-A1 EEG channel was adopted. This provides extensive data from different scenarios, with a lower proportion of N1, making it well-suited for rigorous validation.

As shown in the Figure 2, the raw time-domain data is further transformed into the time-frequency representation during the data preprocessing. Time–frequency data captures the temporal evolution of frequency distributions, and it is precisely these fluctuations that constitute the key distinctions between different sleep stages. For preprocessing, 30-second time-domain segments are transformed into time-frequency representations by Fourier Transform using a Hamming window (size: 200, overlap: 100). We use epoch-wise K-fold cross-validation with the results averaged, (20 for SleepEDF20, SHHS and 10 for SleepEDF78). The model is trained for 200 epochs using AdamW optimizer (learning rate: 5e-6, batchsize: 64) with early stopping. The number N_1 is 16. Each EvoGraph module includes one HyperGraph layer and two NorGraph layers.

Baselines

In this experiment, we compared the proposed model with five baseline models. A brief introduction to each baseline is provided below:

1)Mamba (Gu & Dao, 2023): The model uses the vanilla Mamba module for feature extraction from time–frequency representations, and the extracted features are processed through FC layers to achieve automatic sleep staging.

2)Transformer (Phan et al., 2022): The model adapts the original architecture for single-channel tasks, using multiple Transformer blocks to optimize features from time–frequency representations for sleep staging.

3)AttnSleep (Eldele et al., 2021): The model employs a multi-resolution CNN with adaptive feature recalibration to extract low- and high-frequency features, followed by multi-head attention to capture temporal dependencies among the features from time–domain representations.

4)TinySleepNet (Supratak & Guo, 2020): The model integrates CNN and LSTM to achieve a trade-off in sleep staging while maintaining minimal parameters and computational overhead from time–domain representations.

5)ResnetLSTM (Sun et al., 2018): The model leverages a residual network to extract multi-level features from time–domain representations and employs LSTM to learn the state transition dynamics within sleep signals.

6)DeepSleepNet (Supratak et al., 2017): The model employs a convolutional neural network (CNN) to extract time-invariant features from time–domain representations and a bidirectional long short-term memory network (BiLSTM) to automatically learn the transition rules between sleep stages within EEG.

Evaluation Metrics

We rely on confusion matrix based statistics to assess our model. True Positive (TP) counts samples correctly predicted as a class, False Positive (FP) counts samples incorrectly predicted as that class, False Negative (FN) counts samples that belong to the class but are predicted otherwise, and True Negative (TN) counts samples that do not belong to the class and are correctly not predicted as it.

Per-class precision (PR) measures the proportion of correctly predicted samples among all samples predicted as belonging to class c. It reflects how reliable the model's positive predictions are for each class:

$$\mathrm{PR}_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FP}_c} \tag{11}$$

Per-class recall (RE) evaluates the proportion of correctly predicted samples among all ground-truth samples of class c. It reflects the model's ability to detect all instances of class c:

$$\mathrm{RE}_c = \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FN}_c} \tag{12}$$

Per-class F1 score is the harmonic mean of precision and recall for class c. It provides a balanced measure of both correctness and completeness of predictions:

$$\mathrm{F1}_c = \frac{2 \cdot \mathrm{PR}_c \cdot \mathrm{RE}_c}{\mathrm{PR}_c + \mathrm{RE}_c} \tag{13}$$

Overall accuracy (Acc) measures the proportion of correctly predicted samples among all samples:

$$\mathrm{Acc} = \frac{\sum_{c=1}^{C} \mathrm{TP}_c + \mathrm{TN}_c}{\sum_{c=1}^{C} (\mathrm{TP}_c + \mathrm{FP}_c + \mathrm{FN}_c + \mathrm{TN}_c)} \tag{14}$$

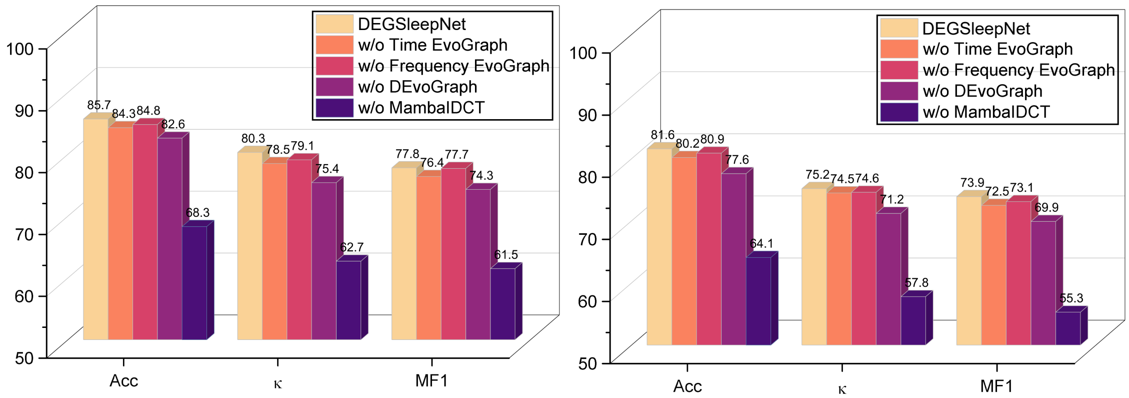
Overall Cohen's Kappa (κ) evaluates the agreement between predictions and ground truth while considering chance agreement:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{15}$$

where p_0 is the observed agreement and p_e is the expected agreement by chance.

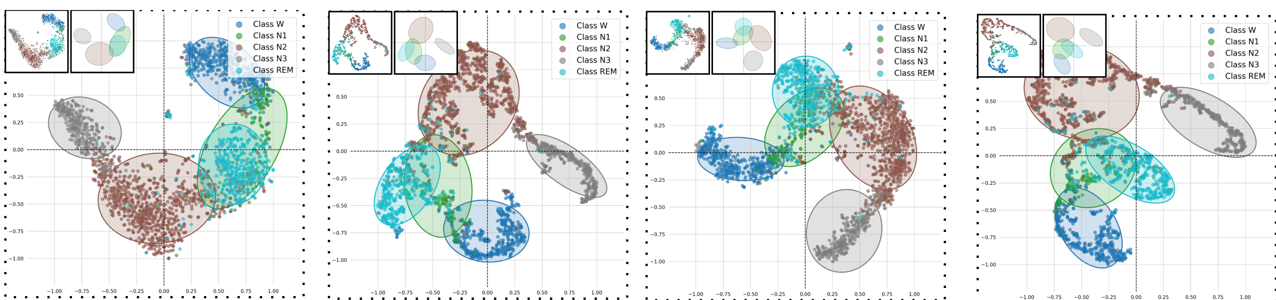
Macro F1 (MF1) computes the mean of per-class F1 scores:

$$\mathrm{MF1} = \frac{1}{C} \sum_{c=1}^{C} \mathrm{F1}_c \tag{16}$$



(a) Modules Influence on EEG Fpz-Cz (b) Modules Influence on EOG

Figure 3: The ablation studies on SleepEDF20 dataset.



(a) Time-Frequency Features From MambaIDCT Blocks (b) Time Features (c) Frequency Features (d) Concatenated Features

Figure 4: The t-SNE feature visualization of the DEGSleepNet.

Results and Analysis

We re-implemented existing models based on 30-second inputs, using the same data strategy and parameter settings for a fair comparison. Table 2 shows the performance comparison of DEGSleepNet against several representative sleep staging methods across three public datasets (SleepEDF20, SleepEDF78, and SHHS). Overall, DEGSleepNet consistently achieves the best or highly competitive performance under different EEG channels. On SleepEDF20 (Fpz-Cz), our method obtains an accuracy of 85.7%, κ of 0.80, and MF1 of 77.8%, outperforming strong Transformer-based, CNN-based and LSTM-based baselines such as Transformer (84.8% Acc) and AttnSleep (84.6% Acc). Similar improvements are observed in per-class F1 scores, particularly in challenging stages such as N3 and REM, demonstrating stronger discriminative capability for minority and transitional sleep stages. On SleepEDF78, DEGSleepNet further achieves 82.8% accuracy and 76.1% MF1, again surpassing all compared methods, including Mamba and AttnSleep, while maintaining more balanced performance across sleep stages. For the larger-scale SHHS dataset, the proposed model achieves 83.3% accuracy and 68.7% MF1, consistently outperforming prior methods under the more challenging sample setting. These improvements can be attributed to the proposed dual evolving graph network architecture. Unlike Transformer-based approaches that primarily model pairwise temporal dependencies with high computational overhead, DEGSleepNet explicitly captures group-wise and pairwise temporal and frequency interactions through the proposed time EvoGraph and frequency EvoGraph modules. Furthermore, together with MambaIDCT blocks, the model effectively integrates long-range temporal dynamics with frequency-aware representations, enabling more robust sleep stage discrimination. Notably, DEGSleepNet achieves these results with only 0.132M parameters and 0.833G FLOPs, significantly lower than most competing methods, highlighting its suitability for resource-constrained wearable sleep monitoring applications.

In addition, we performed sufficient ablation experiments on the Fpz-Cz and EOG channels of the SleepEDF20 dataset for demonstrating the influence of the proposed modules. We observed that removing DEvoGraph leads to a significant degradation in model performance in Figure. 3, indicating that the proposed DEvoGraph structure is valuable for sleep staging. And we observed that removing the basic MambaIDCT leads to a more pronounced decline in performance, indicating that an appropriate feature extraction structure is necessary to provide solid features for the DEvograph. We observed that the

Table 2: Performance comparison with other models on three sleep datasets.

Dataset	Model	Channels	Overall metrics			Per-class F1 score					FLOPs	Parameters
			Acc	κ	MF1	W	N1	N2	N3	REM		
SleepEDF20	**DEGSleepNet**	Fpz-Cz	**85.7**	**0.80**	**77.8**	**92.0**	37.1	**89.7**	**89.2**	**81.0**	0.833G	0.132M
	Mamba (Gu & Dao, 2023)	Fpz-Cz	80.0	0.73	72.5	85.2	32.8	86.3	83.9	74.4	0.265G	3.813M
	Transformer (Phan et al., 2022)	Fpz-Cz	84.8	0.79	77.6	91.7	39.7	88.6	88.1	79.7	4.153G	5.918M
	AttnSleep (Eldele et al., 2021)	Fpz-Cz	84.6	0.78	77.7	91.2	41.7	88.8	87.1	79.7	3.896G	0.471M
	TinySleepNet (Supratak & Guo, 2020)	Fpz-Cz	76.0	0.67	63.3	82.3	6.4	82.1	73.2	72.5	2.178G	0.684M
	ResnetLSTM (Sun et al., 2018)	Fpz-Cz	70.6	0.59	59.1	77.7	9.5	77.7	74.5	56.2	-	20.762M
	DeepSleepNet (Supratak et al., 2017)	Fpz-Cz	77.1	0.69	69.5	86.0	37.0	82.3	65.0	77.3	-	9.574M
	DEGSleepNet	EOG	**81.6**	**0.75**	73.9	**88.1**	34.2	**85.2**	**82.9**	**79.2**	0.833G	0.132M
	Mamba (Gu & Dao, 2023)	EOG	75.6	0.67	67.1	80.5	32.1	79.7	75.1	67.9	0.265G	3.813M
	Transformer (Phan et al., 2022)	EOG	80.5	0.73	74.3	86.9	42.6	83.9	80.9	77.5	4.153G	5.918M
	AttnSleep (Eldele et al., 2021)	EOG	78.1	0.70	70.4	81.1	35.3	83.5	76.1	76.2	3.896G	0.471M
	TinySleepNet (Supratak & Guo, 2020)	EOG	73.9	0.64	65.9	80.0	31.0	79.0	74.7	64.8	2.178G	0.684M
	ResnetLSTM (Sun et al., 2018)	EOG	70.6	0.58	61.8	76.4	25.6	77.5	75.2	54.2	-	20.762M
	DeepSleepNet (Supratak et al., 2017)	EOG	65.7	0.51	50.8	69.4	0.90	74.2	65.6	44.0	-	9.574M
SleepEDF78	**DEGSleepNet**	Fpz-Cz	**82.8**	**0.76**	**76.1**	**92.9**	41.2	**85.7**	**83.6**	**76.8**	0.833G	0.132M
	Mamba (Gu & Dao, 2023)	Fpz-Cz	80.3	0.74	72.7	91.0	36.0	84.9	78.2	73.3	0.265G	3.813M
	Transformer (Phan et al., 2022)	Fpz-Cz	81.7	0.74	74.5	92.4	40.0	85.1	80.5	74.5	4.153G	5.918M
	AttnSleep (Eldele et al., 2021)	Fpz-Cz	81.9	0.75	75.2	92.6	41.6	85.2	80.6	76.0	3.896G	0.471M
	TinySleepNet (Supratak & Guo, 2020)	Fpz-Cz	78.2	0.69	69.0	90.7	28.0	81.9	76.6	67.6	2.178G	0.684M
	ResnetLSTM (Sun et al., 2018)	Fpz-Cz	76.0	0.68	67.6	89.7	31.4	80.1	74.2	62.6	-	20.762M
	DeepSleepNet (Supratak et al., 2017)	Fpz-Cz	79.5	0.71	72.3	91.0	38.8	83.0	77.0	71.8	-	9.574M
SHHS	**DEGSleepNet**	C4-A1	**83.3**	**0.77**	68.7	**88.1**	3.5	84.9	**86.3**	80.6	0.833G	0.132M
	Mamba (Gu & Dao, 2023)	C4-A1	79.9	0.71	66.4	82.3	18.2	81.7	75.3	74.5	0.265G	3.813M
	Transformer (Phan et al., 2022)	C4-A1	83.2	0.77	71.7	86.9	20.5	84.9	85.3	81.0	4.153G	5.918M
	AttnSleep (Eldele et al., 2021)	C4-A1	83.0	0.76	71.6	86.4	20.4	85.0	85.6	80.8	3.896G	0.471M
	TinySleepNet (Supratak & Guo, 2020)	C4-A1	75.4	0.65	63.0	80.7	10.3	79.1	72.5	72.6	2.178G	0.684M
	ResnetLSTM (Sun et al., 2018)	C4-A1	71.9	0.61	61.5	78.2	14.1	78.3	74.1	62.8	-	20.762M
	DeepSleepNet (Supratak et al., 2017)	C4-A1	78.5	0.69	65.5	82.3	17.2	81.4	71.3	75.5	-	9.574M

performance on EOG channel is lower than that on Fpz-Cz channel, indicating that Fpz-Cz channel is more relevant for sleep stages, in line with expert knowledge.

To gain deeper insight into the feature extraction mechanism of DEGSleepNet, we visualize the t-SNE feature maps of different layers. Figure 4 projects the features of five sleep stages into a two-dimensional space using distinct colors, enabling an intuitive comparison of feature separability. In Figure 4(a), the initial time–frequency features extracted by the MambaIDCT blocks exhibit only coarse clustering patterns. Although basic class structures begin to emerge, substantial overlap remains between stages with similar physiological characteristics, particularly between N1 and REM. This observation is consistent with sleep physiology, where N1 and REM both exhibit low-amplitude mixed-frequency EEG patterns. Although REM is additionally associated with rapid eye movements and reduced muscle tone, these signals are not explicitly captured. Furthermore, while frequency-domain features capture characteristic patterns such as sleep spindles in N2 and dominant delta activity in N3, they are less effective in separating stages with similar spectral profiles of N1 and REM. In contrast, time features provide relatively clearer class boundaries, suggesting their advantage in modeling stage transitions. To address these limitations, the Time EvoGraph and Frequency EvoGraph branches further refine the feature space, as illustrated in Figures 4(b) and 4(c). The Time EvoGraph enhances intra-class compactness and produces clearer boundaries for stages such as W and N3 by integrating group-wise and pairwise temporal dependencies. This enables the network to better capture the gradual and sequential transitions across sleep stages. In parallel, the Frequency EvoGraph focuses on spectral dependencies, extracting complementary frequency-specific patterns and forming a structured yet distinct topological distribution in the feature space. Finally, Figure 4(d) presents the t-SNE visualization of the fused features obtained by concatenating time and frequency features. Compared to individual branches, the fused feature space demonstrates significantly improved class separability, with enlarged inter-class margins and reduced intra-class variance. Notably, the previously ambiguous boundary between transitional stages, such as N1 and REM, becomes much clearer after fusion. Overall, these results demonstrate that while the initial MambaIDCT representations provide a useful foundation, they are insufficient for fully capturing complex sleep dynamics. The introduction of EvoGraph modules enables the modeling of higher-order dependencies in both temporal and frequency domains, and their synergistic fusion further enhances feature discriminability. Consequently, DEGSleepNet effectively transforms complex sleep data into a well-structured and highly separable embedding space, thereby improving staging performance.

Conclusion

In this paper, we propose a DEGSleepNet model for high-precision automatic sleep staging. Two kinds of novel EvoGraph module within DEvoGraph block transition from a HyperGraph to a NorGraph, sequentially capturing group-wise and pairwise temporal or frequency dependencies. Extensive comparative experiments and ablation studies demonstrate that leveraging the multi-level dependencies within sleep data is highly valuable. Using epoch-wise K-fold cross-validation, the proposed single-channel model achieves the highest accuracy on SleepEDF20, SleepEDF78 and SHHS sleep datasets.

References

Birrer, V., Elgendi, M., Lambercy, O., & Menon, C. (2024). Evaluating reliability in wearable devices for sleep staging. *NPJ Digital Medicine*, *7*(1), 74.

Bruyneel, M., & Ninane, V. (2014). Unattended home-based polysomnography for sleep disordered breathing: Current concepts and perspectives. *Sleep medicine reviews*, *18*(4), 341–347.

Chen, C. P., Chen, B., & Zhang, T. (2025). Adamgraph: Adaptive attention-modulated graph network for eeg emotion recognition. *IEEE Transactions on Cybernetics*.

Dai, Y., Li, X., Liang, S., Wang, L., Duan, Q., Yang, H., Zhang, C., Chen, X., Li, L., Li, X., et al. (2023). Multichannelsleepnet: A transformer-based model for automatic sleep stage classification with psg. *IEEE Journal of Biomedical and Health Informatics*, *27*(9), 4204–4215.

Dong, C., Sun, D., Zhang, Z., & Luo, B. (2025). Eeg-based patient-specific seizure prediction based on spatial–temporal hypergraph attention transformer. *Biomedical Signal Processing and Control*, *100*, 107075.

Eldele, E., Chen, Z., Liu, C., Wu, M., Kwoh, C.-K., Li, X., & Guan, C. (2021). An attention-based deep learning approach for sleep stage classification with single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *29*, 809–818.

Grandner, M. A., & Fernandez, F.-X. (2021). The translational neuroscience of sleep: A contextual framework. *Science*, *374*(6567), 568–573.

Gu, A., & Dao, T. (2023). Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Ji, J., Ren, Y., & Lei, M. (2022). Fc–hat: Hypergraph attention network for functional brain network classification. *Information Sciences*, *608*, 1301–1316.

Kemp, B., Zwinderman, A. H., Tuk, B., Kamphuisen, H. A., & Oberye, J. J. (2000). Analysis of a sleep-dependent neuronal feedback loop: The slow-wave microcontinuity of the eeg. *IEEE Transactions on Biomedical Engineering*, *47*(9), 1185–1194.

Klepl, D., Wu, M., & He, F. (2024). Graph neural network-based eeg classification: A survey. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *32*, 493–503.

Li, N., Wang, Z., Zhang, Z., Huang, J., Zhu, J., Li, X., & Hu, B. (2025). Improving depression diagnosis using a brain module-based weighted hypergraph convolutional network framework. *Neurocomputing*, 131074.

Lim, D. C., Najafi, A., Afifi, L., Bassetti, C. L., Buysse, D. J., Han, F., Högl, B., Melaku, Y. A., Morin, C. M., Pack, A. I., et al. (2023). The need to promote sleep health in public health agendas across the globe. *The Lancet Public Health*, *8*(10), e820–e826.

Phan, H., Mikkelsen, K., Chén, O. Y., Koch, P., Mertins, A., & De Vos, M. (2022). Sleeptransformer: Automatic sleep staging with interpretability and uncertainty quantification. *IEEE Transactions on Biomedical Engineering*, *69*(8), 2456–2467.

Quan, S. F., Howard, B. V., Iber, C., Kiley, J. P., Nieto, F. J., O'Connor, G. T., Rapoport, D. M., Redline, S., Robbins, J., Samet, J. M., et al. (1997). The sleep heart health study: Design, rationale, and methods. *Sleep*, *20*(12), 1077–1085.

Sun, Y., Wang, B., Jin, J., & Wang, X. (2018). Deep convolutional network method for automatic sleep stage classification based on neurophysiological signals. *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 1–5.

Supratak, A., Dong, H., Wu, C., & Guo, Y. (2017). Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg. *IEEE transactions on neural systems and rehabilitation engineering*, *25*(11), 1998–2008.

Supratak, A., & Guo, Y. (2020). Tinysleepnet: An efficient deep learning model for sleep stage scoring based on raw single-channel eeg. *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 641–644.

Zhang, T., Hu, T., Wu, M., Xu, Z., & Chen, C. P. (2025). Acm-gnn: Adaptive cluster-oriented modularity graph neural network for eeg depression detection. *IEEE Transactions on Computational Social Systems*.