MELD: A Multimodal Ensemble of Lightweight 1D Convolutional Neural Networks for Sleep Stage Classification

Ómar Bessi Ómarsson[1]*, **Teitur Hrólfsson**[1]*, **Emil Hardarson**[1], **María Óskarsdóttir**[1, 2]

omar23@ru.is, teitur23@ru.is, emilh@ru.is, mariaoskars@ru.is

[1]Department of Computer Science, Reykjavik University, Reykjavik, Iceland

[2]School of Mathematical Sciences, University of Southampton, Southampton, United Kingdom

Abstract

Accurate sleep stage classification from multimodal biosignals is crucial for diagnosing sleep disorders, yet manual scoring is time-consuming and error-prone. We introduce MELD, a multimodal ensemble of lightweight 1D convolutional neural networks (CNNs), each processing a single signal modality, with an evolutionary algorithm used to optimize kernel sizes. Single-signal CNNs are evolved on two datasets, namely subsets of the Sleep-EDF and SHHS datasets, then combined into a fixed multimodal ensemble for multiclass classification. The base ensemble achieves consistently high accuracy and F1 scores across all datasets, while the evolutionary optimization of kernel sizes does not improve performance, indicating that kernel size evolution alone is insufficient within a narrow search space. Our work demonstrates the robustness of carefully designed CNN ensembles and evaluates the limits of kernel-focused neuro-evolution. This framework provides a simple yet effective approach for sleep staging and other multimodal biosignal tasks, highlighting the potential for broader evolutionary search spaces and interpretability to further enhance performance and understanding.

Code — https://github.com/emilhar/signal_fusion_nas

Introduction

In recent years, the ability to automatically find suitable deep learning architectures has become increasingly important (Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter 2019; White et al. 2023; Ren et al. 2022); particularly for multiclass classification on multimodal biosignals, such as sleep stage classification (Elsken, Metzen, and Hutter 2019; Kong et al. 2023). As new measuring devices are released at a rapid pace, each with unique signal modalities, sampling rates, and characteristics, the need to quickly create accurate models is growing. However, designing optimal architectures is time-consuming and often requires expert knowledge.

Sleep stage classification is a prominent example. It plays a central role in diagnosing sleep disorders, which can significantly affect health and quality of life. The gold-standard method, polysomnography (PSG), records multiple physiological signals, such as EEG, EOG, and EMG, during sleep. While PSG is reliable, manual scoring is time-consuming, error-prone, and requires trained experts (Hardarson et al. 2023; Arnardottir, Islind, and Óskarsdóttir 2021). Automated methods, with deep learning approaches at the forefront, have therefore been actively explored, with convolutional neural networks (CNNs) showing strong results (Supratak et al. 2017; Zhang et al. 2024; Alattar, Govind, and Mainali 2024). Recent work has also applied neural architecture search (NAS) to EEG-based sleep stage classification, achieving competitive accuracy while reducing the need for manual architecture design (Kong et al. 2023; Rala Cordeiro et al. 2021).

In this paper, we present an approach for multimodal classification using a simple ensemble of 1D CNNs, with each CNN processing a single modality. We employ an evolutionary algorithm to optimize the hyperparameters of the component models. While our primary focus is on sleep stage classification, the method is applicable to a range of multimodal tasks, such as activity recognition or emotion detection.

Methods

In this section we describe our proposed methodology, as well as the datasets and metrics used to evaluate its performance. Our multiclass classification model is set up as a collection of binary classification tasks, where for each task we use a CNN architecture whose parameters are determined using evolutionary algorithms.

Single Signal Binary Classification

For each sleep stage and signal, we train a separate binary classification model inspired by the architectures of TinySleepNet (Supratak et al. 2017) and DeepSleepNet (Supratak et al. 2017; Fiorillo, Favaro, and Faraci 2021) (see Figure 1). Within the evolutionary algorithm described later, the training loss is used as the fitness function. Each candidate model is trained on a subset of the dataset for a fixed number of epochs.

The number of branches and the kernel sizes in each branch are treated as hyperparameters subject to evolutionary optimization. The convolutional neural networks are intentionally lightweight inside said optimization, with each convolutional layer containing a single filter.

*These authors contributed equally.

For each branch, additional parameters such as padding, stride, pooling sizes, pooling strides, and dropout rates are derived from the kernel sizes and input sample count. These parameters are computed according to fixed rules to ensure valid convolutional and pooling operations across branches.

Following the convolutional branches, all extracted features are concatenated and passed through a fully connected module consisting of two hidden layers: the first with 64 units and the second with 32 units. Each hidden layer is followed by a ReLU activation and a dropout layer with a rate of 0.1 is applied after the first layer.

Combined Multimodal Multiclass Classification

After the architectures for all single–signal models have been selected and the models have been fully trained on their respective tasks (full training set, 30 epochs, 32 filters), they are combined into an ensemble.

Only this ensemble is trained at this stage, while the parameters of the single–signal models remain fixed. Training uses a standard multiclass loss function and is performed on the multimodal dataset created by pairing all signals for each sample.

Figure 2 shows the structure of the ensemble: parallel branches for each signal contain pre-trained models, whose outputs are concatenated and passed to the classification module to generate the final prediction of multiple classes.

Evolutionary Neural Architecture Search

We searched for optimal architectures of the binary classification models using the `eaMuPlusLambda` algorithm as implemented in the DEAP Python framework.

Each architecture α in a population P_g of algorithms is represented as a list of numerical values describing its kernel sizes. For example, the configuration [[400,20,20],[16,2,2]] represents a two-branch model in which the first branch has kernel sizes 400, 20, and 20, while the second branch has kernel sizes 16, 2, and 2.

The fitness $f(\alpha)$ of individuals is determined by training the model architecture for a short, fixed number of epochs, E, on a random small subset of the training data

$$\mathcal{D}_{sub,\text{train}} \subset \mathcal{D}_{sub},$$

where $\mathcal{D}_{sub}$ is the training data, and evaluating the resulting model on

$$\mathcal{D}_{sub,\text{val}} \subset \mathcal{D}_{sub} \setminus \mathcal{D}_{sub,\text{train}}.$$

Parents are selected using tournament selection, with a tournament size of 5, and an elitism of 1.

The training of the single signal binary classification models and the combined multimodal multiclass classification model was performed in PyTorch using an RTX 3060 Ti.

Datasets

We used public Sleep-EDF datasets for initial experiments and the large-scale Sleep Heart Health Study (SHHS) for final evaluation.

Sleep-EDF Sleep-EDF contains PSG recordings with two EEG channels (Fpz–Cz, Pz–Oz), EOG, and chin EMG, scored under R&K rules. Key subsets:

- **Sleep-EDF-20**: 40 recordings from 20 healthy subjects.
- **Sleep-EDF-78**: 153 recordings from 78 subjects, including mild insomnia.
- **Sleep-EDF-x**: 197 recordings, combining cassette and telemetry data from healthy and mildly disordered subjects.

Sleep Heart Health Study (SHHS) A multi-center cohort with:

- SHHS-1: ∼5,793 participants.
- SHHS-2: ∼2,651 participants.

PSGs include multi-EEG, EOG, EMG, respiration, and XML hypnograms, scored per AASM rules.

We decided on only using a subset of SHHS-1 that was chosen randomly.

Evaluation Metrics

The evolutionary algorithm uses the training loss as its fitness metric, although this is not an ideal measure of generalization performance. For ensemble model evaluation, we report the per-class F1 score and overall accuracy derived from the confusion matrix.

We compare each model to its "base" version. This model is created from "smartly" chosen branches. To formalize this selection process, we first define a set of allowable parameter scales based on the number of training samples. Let n denote the number of datapoints in an epoch for a signal and define the set of denominators

$$M = \{5,\ 10,\ 30,\ 90,\ 270,\ 810,\ 2430\}.$$

The choice set is then given by

$$\mathrm{C} = \left\{ \max\left(\left\lfloor \frac{n}{m} \right\rfloor, 3 \right) \;\middle|\; m \in M \right\}.$$

The lower bound of 3 ensures a minimum sample size for any choice, while the elements of M control the range of sampling granularities.

Each element of C is then adjusted to be an odd integer, ensuring symmetry around the central filter in convolutional operations. When constructing new individuals, the selection of branch parameters proceeds in a monotonic manner: the first element is chosen from C, and all subsequent selections are drawn from elements of equal or smaller magnitude.

Results

Base Model Evaluation

We first evaluate the base models before applying the evolutionary algorithm (EA). The results are presented for each dataset Sleep-EDF-20, Sleep-EDF-78, Sleep-EDF-x, and SHHS in tables 1, 2, 3 and 4, respectively. For each dataset, we report the multiclass classification metrics (accuracy and per-class F1-scores) along with confusion matrices showing the five-class classification.

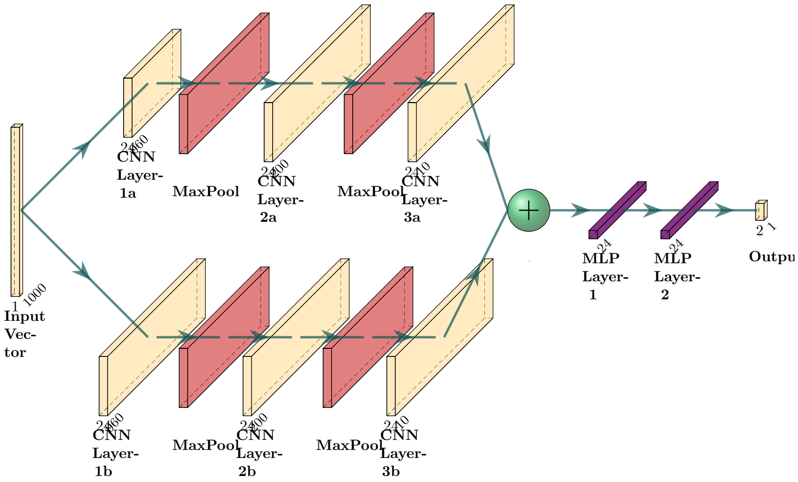

Figure 1: Example CNN architecture

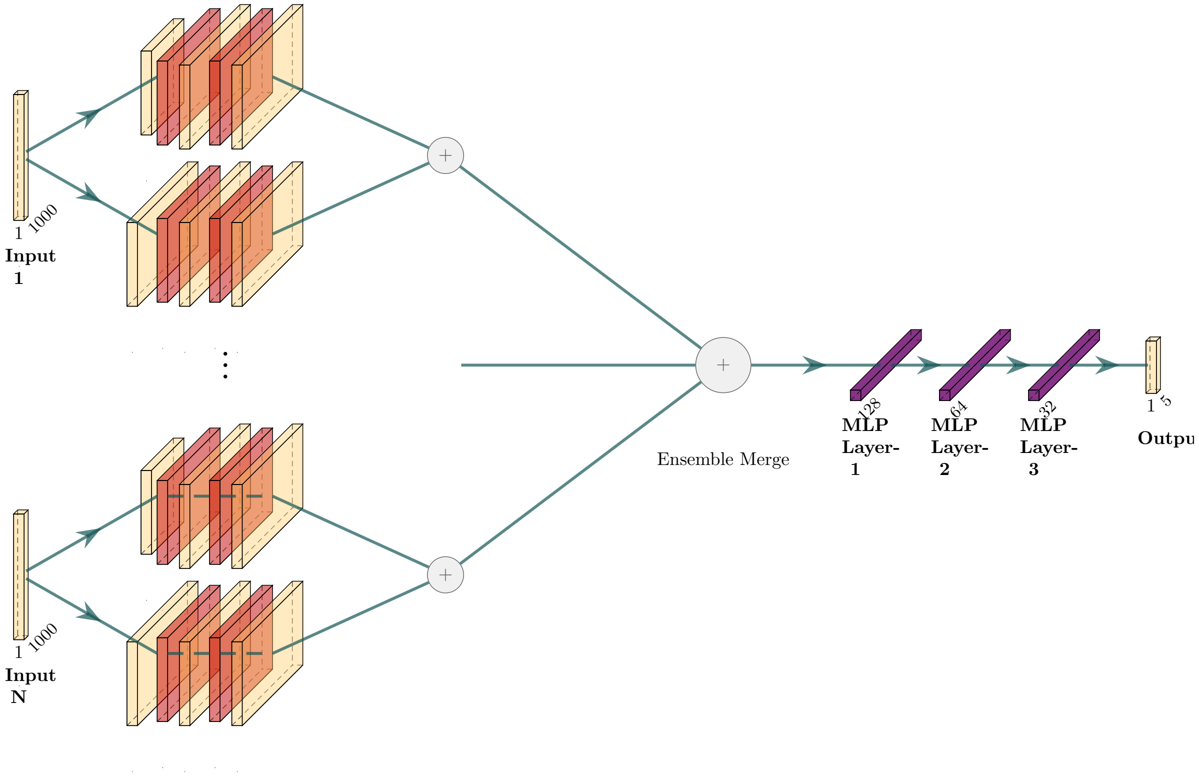

Figure 2: Ensemble model architecture

The tables show the accuracy for each sleep stage.

Table 1: Confusion matrix and performance metrics for the base model on the Sleep-EDF-20 dataset.

Sleep-EDF-20

	Estimated				
True	W	N1	N2	N3	REM
W	0.81	0.13	0.01	0.00	0.04
N1	0.17	0.36	0.08	0.00	0.38
N2	0.01	0.04	0.79	0.06	0.09
N3	0.00	0.00	0.10	0.89	0.00
REM	0.02	0.06	0.05	0.00	0.86

Metric	**Accuracy**	**Macro F1**
Value	0.7495	0.7388

Class	**F1-score**
Wake	0.8100
N1	0.4557
N2	0.7822
N3	0.9175
REM	0.7288

Table 2: Confusion matrix and performance metrics for the base model on the Sleep-EDF-78 dataset.

Sleep-EDF-78

	Estimated				
True	W	N1	N2	N3	REM
W	0.93	0.06	0.01	0.00	0.01
N1	0.16	0.48	0.23	0.00	0.12
N2	0.01	0.13	0.69	0.05	0.13
N3	0.00	0.01	0.17	0.80	0.01
REM	0.02	0.15	0.09	0.00	0.73

Metric	**Accuracy**	**Macro F1**
Value	0.7275	0.7262

Class	**F1-score**
Wake	0.8732
N1	0.5275
N2	0.6273
N3	0.8696
REM	0.7337

Table 3: Confusion matrix and performance metrics for the base model on the Sleep-EDF-x dataset.

Sleep-EDF-x

	Estimated				
True	W	N1	N2	N3	REM
W	0.79	0.19	0.01	0.00	0.02
N1	0.02	0.72	0.09	0.00	0.17
N2	0.00	0.02	0.91	0.04	0.03
N3	0.00	0.00	0.09	0.90	0.00
REM	0.00	0.01	0.08	0.00	0.91

Metric	**Accuracy**	**Macro F1**
Value	0.8460	0.8465

Class	**F1-score**
Wake	0.8681
N1	0.7423
N2	0.8349
N3	0.9326
REM	0.8545

Table 4: Confusion matrix and performance metrics. for the base model on the SHHS dataset.

SHHS

	Estimated				
True	W	N1	N2	N3	REM
W	0.82	0.11	0.05	0.00	0.02
N1	0.13	0.61	0.18	0.00	0.07
N2	0.04	0.11	0.76	0.06	0.03
N3	0.03	0.00	0.17	0.77	0.03
REM	0.03	0.22	0.12	0.02	0.61

Metric	**Accuracy**	**Macro F1**
Value	0.7154	0.7181

Class	**F1-score**
Wake	0.8000
N1	0.5980
N2	0.6667
N3	0.8324
REM	0.6932

Base Model Evaluation Summary

While the base models demonstrate strong performance on the more easily distinguishable stages such as N2 and N3, the results highlight persistent challenges in correctly identifying REM and the transitional stage N1. These difficulties are consistent across datasets and reflect common limitations in automated sleep staging, providing a meaningful baseline against which improvements from the evolutionary algorithm can be measured.

Post-EA Model Evaluation

We now present results after applying the evolutionary algorithm. The evaluation setup is identical to the base model, allowing direct before-and-after comparison. Tables 5, 6, 7 and 8 show the confusion matrices for the four datasets.

Table 5: Confusion matrix and performance of the EA-optimized model on the Sleep-EDF-20 dataset.

Sleep-EDF-20

	Estimated				
True	W	N1	N2	N3	REM
W	0.81	0.07	0.04	0.00	0.08
N1	0.21	0.20	0.18	0.01	0.40
N2	0.01	0.02	0.78	0.09	0.11
N3	0.01	0.00	0.08	0.92	0.00
REM	0.02	0.03	0.06	0.00	0.89

Metric	**Accuracy**	**Macro F1**
Value	0.7171	0.6878

Class	**F1-score**
Wake	0.7864
N1	0.3030
N2	0.7256
N3	0.9064
REM	0.7177

Table 6: Confusion matrix and performance of the EA-optimized model on the Sleep-EDF-78 dataset.

Sleep-EDF-78

	Estimated				
True	W	N1	N2	N3	REM
W	0.91	0.07	0.00	0.00	0.01
N1	0.14	0.47	0.24	0.01	0.14
N2	0.01	0.13	0.70	0.07	0.10
N3	0.00	0.01	0.13	0.85	0.01
REM	0.02	0.14	0.10	0.00	0.74

Metric	**Accuracy**	**Macro F1**
Value	0.7340	0.7317

Class	**F1-score**
Wake	0.8792
N1	0.5165
N2	0.6422
N3	0.8808
REM	0.7400

Table 7: Confusion matrix and performance of the EA-optimized model on the Sleep-EDF-x dataset.

Sleep-EDF-x

	Estimated				
True	W	N1	N2	N3	REM
W	0.80	0.18	0.01	0.00	0.01
N1	0.05	0.72	0.11	0.01	0.11
N2	0.00	0.03	0.85	0.09	0.02
N3	0.00	0.00	0.04	0.96	0.00
REM	0.00	0.02	0.06	0.00	0.91

Metric	**Accuracy**	**Macro F1**
Value	0.8514	0.8506

Class	**F1-score**
Wake	0.8649
N1	0.7385
N2	0.8252
N3	0.9320
REM	0.8922

Table 8: Confusion matrix and performance of the EA-optimized model on the SHHS dataset.

SHHS

	Estimated				
True	W	N1	N2	N3	REM
W	0.84	0.11	0.04	0.00	0.02
N1	0.15	0.64	0.12	0.00	0.09
N2	0.05	0.16	0.67	0.08	0.05
N3	0.04	0.00	0.12	0.76	0.08
REM	0.05	0.24	0.06	0.02	0.63

Metric	**Accuracy**	**Macro F1**
Value	0.7052	0.7070

Class	**F1-score**
Wake	0.7850
N1	0.5953
N2	0.6634
N3	0.8172
REM	0.6738

Post-EA Model Evaluation Summary

The application of the evolutionary algorithm (EA) yielded mixed results, demonstrating that the optimization process did not lead to a consistent, across-the-board improvement over the base model. While performance on certain sleep stages and datasets saw modest gains, these were often offset by declines in others. For instance, the EA model showed a notable improvement in REM detection on the Sleep-EDF-x dataset (F1-score increase from 0.8545 to 0.8922) but a decrease in N1 sleep detection on the Sleep-EDF-20 dataset

(F1-score from 0.4557 to 0.3030). This suggests that the EA, under the defined fitness function and constraints, primarily led to a re-balancing of the model's performance characteristics rather than a fundamental enhancement of its overall generalizability.

Lessons Learned

During the development and experimentation phase of research we explored several strategies aimed at improving convergence, diversity, and efficiency. While these ideas were motivated by sound intuition, they ultimately proved ineffective in practice and were removed from our final pipeline.

Tournament of Champions

To address concerns about under-training within the EA, we introduced a *Tournament of Champions* strategy. Every N generations, we fully trained the top $M\%$ of individuals. From this set, the top 50% were retained as "champions." We then randomly selected an equal number of individuals from the remainder of the population and generated a new population through crossover and mutation. The rationale was to perform periodic "status checks" by training select models to convergence, ensuring the evolutionary search was progressing toward higher-quality solutions. In practice, this approach was prohibitively time-consuming, as fully training even a small set of individuals required substantial computation. Moreover, the "champions" often did not generalize well, and their inclusion failed to drive the population toward better performance. As a result, the approach was abandoned.

Uniqueness-Based Selection

To promote diversity, we incorporated a uniqueness metric into selection, scoring individuals based on both fitness and their genetic distance from selected models. However, grid search showed the fitness landscape lacked clear structure, with only minor variations. While uniqueness encouraged novel models, they often underperformed, slowing convergence without clear benefits. Ultimately, the method was removed, and its effectiveness remains uncertain.

Age-Layered Population Structure (ALPS)

Motivated by the hypothesis that premature convergence was limiting EA performance, we implemented the Age-Layered Population Structure (ALPS) approach (**?**). ALPS attempts to maintain diversity by stratifying individuals into layers based on age, preventing younger individuals from directly competing with older, more mature ones. This method represented the most significant time investment of all our experiments. However, it failed to yield consistent improvements. Moreover, we later determined that ALPS was addressing a problem we did not actually have. No evidence of premature convergence was observed during our experiments. The added complexity increased implementation overhead without tangible benefits, and the feature was eventually removed.

Discussion & Conclusion

In this work, we explored CNN-based ensembles for multimodal sleep stage classification and applied an EA to optimize kernel sizes of the single-signal models. Our results indicate that the base ensemble model consistently achieves strong performance across all datasets, confirming that carefully designed ensembles remain effective for this task. In contrast, the EA did not improve the ensemble model; post-EA results were equal to or worse than the base models. This outcome suggests that evolving kernel sizes alone is insufficient to enhance performance and highlights the limitations of a narrow search space in neuro-evolution.

Despite the limited impact of the EA, the study demonstrates the robustness of the ensemble approach and its potential applicability to wearable sleep monitoring devices. The model's strong baseline performance positions it as a competitive candidate among current sleep staging methods, although further benchmarking is needed to determine whether it represents one of the best-performing models in the field. Importantly, our experiments reveal the need for interpretability: understanding which aspects of the ensemble contribute to its success remains an open question, and future work should focus on methods to probe and explain model behavior beyond a black-box evaluation.

Beyond sleep staging, the approach offers a framework for multimodal modeling and prediction. Neuro-evolution could be extended with a broader search space or additional architectural parameters to support new devices or tasks, potentially reducing the reliance on manual preprocessing. Future research should aim to combine automated architecture search with interpretability and task-specific constraints to improve both performance and deepen understanding of what drives model effectiveness.

An alternative direction is to apply evolutionary optimization directly to the ensemble rather than to individual signal models. This could better capture cross-signal interactions, which are not addressed in the current per-modality optimization. However, fully training models within EAs would likely increase computational demands, potentially requiring smaller populations or resulting in substantial resource costs.

References

Alattar, M.; Govind, A.; and Mainali, S. 2024. Artificial Intelligence Models for the Automation of Standard Diagnostics in Sleep Medicine—A Systematic Review. *Bioengineering*, 11(3): 206. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute.

Arnardottir, E. S.; Islind, A. S.; and Óskarsdóttir, M. 2021. The Future of Sleep Measurements: A Review and Perspective. *Sleep Medicine Clinics*, 16(3): 447–464.

Elsken, T.; Metzen, J. H.; and Hutter, F. 2019. Neural Architecture Search: A Survey. ArXiv:1808.05377 [cs, stat].

Fiorillo, L.; Favaro, P.; and Faraci, F. D. 2021. DeepSleepNet-Lite: A Simplified Automatic Sleep Stage Scoring Model with Uncertainty Estimates.

Hardarson, E.; Islind, A. S.; Arnardottir, E. S.; and Óskarsdóttir, M. 2023. Error Propagation from Sleep Stage Classification to Derived Sleep Parameters in Machine Learning on Data from Wearables. *Current Sleep Medicine Reports*, 9(3): 140–151.

Kong, G.; Li, C.; Peng, H.; Han, Z.; and Qiao, H. 2023. EEG-Based Sleep Stage Classification via Neural Architecture Search. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31: 1075–1085. Conference Name: IEEE Transactions on Neural Systems and Rehabilitation Engineering.

Rala Cordeiro, J.; Raimundo, A.; Postolache, O.; and Sebastião, P. 2021. Neural Architecture Search for 1D CNNs-Different Approaches Tests and Measurements. *Sensors (Basel, Switzerland)*, 21(23): 7990.

Ren, P.; Xiao, Y.; Chang, X.; Huang, P.-y.; Li, Z.; Chen, X.; and Wang, X. 2022. A Comprehensive Survey of Neural Architecture Search: Challenges and Solutions. *ACM Computing Surveys*, 54(4): 1–34.

Supratak, A.; Dong, H.; Wu, C.; and 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. ArXiv:1703.04046 [stat].

Thomas Elsken; Jan Hendrik Metzen; and Frank Hutter. 2019. Neural Architecture Search: A Survey. *The Journal of Machine Learning Research*.

White, C.; Safari, M.; Sukthanker, R.; Ru, B.; Elsken, T.; Zela, A.; Dey, D.; and Hutter, F. 2023. Neural Architecture Search: Insights from 1000 Papers. ArXiv:2301.08727 [cs, stat].

Zhang, X.; Zhang, X.; Huang, Q.; Lv, Y.; and Chen, F. 2024. A review of automated sleep stage based on EEG signals. *Biocybernetics and Biomedical Engineering*, 44(3): 651–673.