Recreation of "Bringing At-home Pediatric Sleep Apnea Testing Closer to Reality: A Multi-modal Transformer Approach (Fayaaz et al. 2023)"

Alobba, Francis (falobba2) , Wall, Camden (cnwall2)

Abstract

Link to video summary - https://mediaspace.illinois.edu/media/t/1_qp3mt4ck

Project code repository - https://github.com/cnwall99/SP25CS598DLH[1]

Introduction

The research is about Sleep Apnea testing. The paper[2] wants to address the lack of testing among children. It also addresses the difficulty of testing in terms of time and location since current testing is done in a laboratory as well as the need to test several signals. Past sleep apnea testing research and testing has been aimed specifically to adults but even among children this can lead to several health problems if left untreated. What the research paper is trying to develop is a transformer-based model that can detect sleep apnea events among children accurately using only a couple of polysomnography signals, specifically ECG and SpO2. ECG and SpO2 are the signals chosen to hone in on because they are the signals that are the easiest to test and collect data from away from the labs or clinics. The research tries to compare if testing just these two signals can have a similar sleep apnea detection rate against the entire polysomnography signals test.

The research contributes several topics within the research space. First, the research uses a transformer-based model customized for pediatric sleep apnea detection. They tested this model against previously used baselines such as CNNs, CNN+LSTM, and hybrid models and saw improvements in F1 and AUROC scores. Second, the research tested using only ECG and SpO2 and compare it to testing various combinations of signals, including all signals, that are typically included in polysomnography tests. The combination of ECG and SpO2 did have comparable detection rates compared to other signal combinations. Another thing the research finds is that EEG, which is one of the signals thought to be the most impactful for sleep apnea detection, might not have as big of an effect compared to ECG and SpO2. Because ECG and SpO2 could have comparable results to full polysomnography tests this can have a significant impact in terms of accessibility. Patients are able to save money and time. Instead of going to a sleep lab and be put on a waitlist, patients can instead test at home and save some stress of taking their children to get laboratory testing done as well as save money with lab costs.

We were not able to reproduce the data processing for both datasets utilized and had to generate our own preprocessed data to continue with the other parts of the project. We were not able to reproduce fully an identical model to the ones the researchers used but we have our own

attempt of implementing it which did not produce similar results. For the baselines, we successfully implemented two of the four baselines that the research introduced.

Methodology

This project utilized Python 3.12.3 and requires the package dependencies as outlined in Table 1 of Appendix A.

The original paper utilized two datasets, the Nationwide Children's Hospital Sleep DataBank (NCH), and the Childhood Adenotonsillectomy Trial (CHAT) datasets. These datasets supply the necessary polysomnography data as well as additional characteristic data for children aged 0-6 years. Access to this data is requested through the governing bodies which manage the datasets (National Sleep Research Resource [NSRR] and BioLINCC respectively). After data access was approved, the datasets were obtained through the NSRR's Ruby accessor. This provided two directories of data, with the NCH data consisting of annotation files, polysomnography files, and sensor input files (.ann, .edf, .tsv respectively). As the full NCH dataset is over 2TB in size, a decision to download a subset of the data (~ 50GB) was made in an effort to preserve processing time and compute resources. The CHAT dataset was similarly ordered, and again a subset of the data was downloaded to preserve resources (~80GB).

The authors maintained a repository for the developed codebase[3] which was used to aid in recreation in this project. The preprocessing code from this repository was used as a prompt and provided to the Large Language Model (LLM) GPT-4o mini (OpenAI, 2024) with additional input to modify the code to resolve errors with project setup. The data preprocessing involves loading the raw data, performing integrity checks to ensure all channels are present, calculating and filtering by the Apnea-Hypoapnea Index (AHI) (which was not present in the codebase, had to be developed, and will be contributed to the PyHealth library), parsing the event annotations to extract events (apnea, hypoapnea, and wakefulness) from the raw data, and segmenting the data into epochs (30 second, non-overlapping epochs). The channels are then parsed according to the model being ran, and the data is resampled to 64Hz. Epochs with overlapping wake events are filtered out, and the time overlapping with apnea and hypoapnea events are stored as the ground truth values. This processed data is then compressed to preserve memory, and passed to the data loader which performs batching and prepares the data to be fed to the model.

Table 1. NCH Preprocessing Output Description

Field Name	Shape	Type	Description
data	(samples, timesteps, channels)	float32	Multichannel PSG signal segments (e.g., ECG, SpO_2)
labels_apnea	(samples,)	int	Binary label: 1 = apnea, 0 = no apnea
labels_hypopnea	(samples,)	int	Binary label: 1 = hypopnea, 0 = no hypopnea

While the original publication utilized both the NCH and CHAT datasets, we were not able to recreate the data processing for the CHAT dataset, therefore only the NCH data was used.

The original study used a transformer-based architecture with five components including: data sources, segmentor, tokenizer, transformer, and multi-layer perceptron head. The data sources will include signals such as ECG and SpO2. The segmentor divides the signals into smaller fixed-length epochs. The tokenizer converts the segmented data into a set of time series that can be processed by the transformer. The transformer has a multi-head attention module and a position-wise fully connected network. The multi-head attention module computes attention scores between tokens which the position-wise fully connected network will apply non-linear transformations. The multi-layer perceptron head is the final layer which is a two-layer fully connected layer that will output a probability score for likelihood of apnea-hypopnea event. The baselines are from four studies on adult apnea detection: CNN, Fusion, CNN+LSTM, and Hybrid Transformer. They used 2 evaluation metrics: F1-score and AUROC. First, using both data sources they compared F1-score and AUROC of the baselines to their model. Second, the F1-score and AUROC of subsets of 1 and 2 PSG signals is compared to when all 6 signals is used. Finally, using NCH dataset the AUROC scores of the baselines is compared to their model across different age brackets from ages 0 to 18. The main hypothesis of the original paper is by using a transformer-based architecture, we can use the combination of ECG and SpO2 signals and have comparable sleep apnea event detection in children to fully lab-based PSG testing in adults. Their method is better than baselines because the transformer model's attention module handle temporal context and interaction between modalities better. The multi-modal approach is more resilient to noise and variability to PSG. Additional data such as comorbidities associated with sleep apnea or audio recordings might improve outcomes. Their hypothesis seems legitimate as the AUROC and F1-score of their model outperforms baseline models on the two data sources used. Also, the combination of ECG and SpO2, which are easier to collect at home, have comparable results to a full PSG test.

In this recreation of the original paper, the models utilized were able to be generated using the original codebase as input to the LLM. The architecture of these models (CNN, SEMSCNN,

CNN+LSTM, HYBRID) is as shown in Appendix A. The hybrid model consists of convolution layers which extract features from the processed sequence of signals,

$$y = \phi(W * x + b)$$

where ϕ is the activation function (ReLU, sigmoid, ..), W is the convolution kernel, and b is the bias, normalization layers which normalize the inputs per sample:

$$Norm(x) = \frac{(x - \mu)}{\sqrt{\sigma^2 + \epsilon}}$$

where μ is the mean of each feature, σ^2 is the variance per feature, and ϵ is a normalization constant, multi-head-attention layers which perform self-attention based on the trained parameters, and the dense, multi-layer-perceptron (MLP), to output the classification based on the previous model processing

$$\hat{y} = \sigma(Wx + b)$$

where $\hat{y}$ is the probability of output (apnea vs no-apnea).

Training

The training uses Adam optimizer with the following parameters:

$$\beta_1 = 0.9, \beta_2 = 0.999, \text{ and } \varepsilon = 10^{-7}$$

It uses a learning rate of 0.001. They used an L2 weight regularization with $\lambda = 0.001$ and a dropout rate of 0.25. A moderate regularization was used since sleep data have a high noise-to-signal ratio. The dropout rate is on the lower side since polysomnography signals contains several important data and we don't want these to be dropped frequently. They used a batch size of 256. They have early stop for training if there is no improvement in validation loss after 20 epochs.

The training we used have a max number of 100 epochs with early stopping. Since polysomnography signals have complex patterns, we want multiple iterations to learn the correlations between the signal channels. The stopping criteria is the validation loss. The average runtine for each epoch is around 15 minutes. The model uses TensorFlow and was implemented with GPU acceleration. Especially for the raw data which contained several EDFs for each sample a large storage is needed since a single EDF file are in the rages of 500mb to 1.5gb. Due to this the preprocessing step is very intensive so a sufficient RAM of 16GB is suggested as well.

The original research paper uses Binary Cross-Entropy as the loss function since detecting sleep apnea uses a binary classification of apnea and hypopnea or presence and absence in a given epoch. Binary Cross-Entropy is calculated by:

$$BCE(y, \hat{y}) = -[y * \log(\hat{y}) + (1 - y) * \log(1 - \hat{y})]$$

where y is the true label (0 or 1) and $\hat{y}$ is the predicted probability. It especially works well with physiological signals that can have both very weak or strong patterns because BCE is stable even

when predictions approach 0 or 1. It works well with the L2 regularization model that the research uses to avoid overfitting to noise.

Evaluation

The two main metrics we use are F1 score and Area Under Curve score. We also used other metrics such as accuracy, precision, recall, and specificity. F1 score is the harmonic mean of recall and precision that is calculated by F1 = 2 * (precision * recall) / (precision + recall). It is useful in cases where there is an imbalance in class distribution. We have 2 metrics for Area Under Curve/AUC. The first AUC metric is Area Under the Receiver Operating Characteristic/AUROC. It indicates the model's ability to discriminate across the possible thresholds of the research and like the F1 score is less sensitive to class imbalance. The second AUC metric is Area Under Precision-Recall Curve/AUPRC. It gives a summary of the trade-off between precision and recall and is useful where there is an imbalance dataset especially when there is more false negatives than false positives. Accuracy measures the amount of correctly classified instances out of the total and is calculated by Accuracy = 1. * (TP + TN) / (TP + TN + FP + FN); where T = True, F = False, P = Positive, and N = Negative. It is a good general measure that is easy to understand and can be used to compare with previous studies. Precision measures the amount of true positives out of all the positive predictions and is calculated by Precision = 1. * TP / (TP + FP). Recall measures the number of true positives that are correctly detected and is calculated by Recall = 1. * TP / (TP + FN). Because the primary purpose of the research is detecting apnea events which are true positives then this metric is particularly relevant. Specificity measures the number of true negatives that are correctly detected and is calculated by Specificity = 1. * TN / (TN + FP). It indicates how many non-apnea events are correctly identified as actual negatives which helps reduction of frequent false alarms.

Results

The hybrid model was evaluated for each combination of signals (up to n=2) as well as all available signals. It was determined that the recreated model architecture and training was insufficient to produce a model which was capable of effectively predicting the ground truth value of the input. The model seemed to overfit to the case of always predicting the majority class (no-apnea event ‘0’), which generally resulted in an F1 score of 0 and a variable ROC AUC score (area under the receiver operating characteristic curve). The results for each model evaluation is as shown in Table 2 of Appendix A. This failure to effectively train is likely due to the limited data set that was selected to reduce the computational resources. After the preprocess filtering, only 11 patient data streams were applicable, with ~96% of each stream resulting in windows with no apnea activity occurring. This results in an extreme bias in the model, proving accurate prediction nearly impossible. The signal set with the highest ROC AUC was EOG + EEG, with the ECG+SP02 evaluation having a ROC AUC of 65.93

In the original paper, it was shown that the newly generated model was capable of exceeding the F1 and ROC AUC score of each of the CNN, SEMCNN, and CNNLSTM model architectures across the combination (up to n=2) of patient signals. ECG + SP02 specifically

resulted in an F1 and ROC AUC of 80.7 and 88.4 respectively, with the highest score being the full signal set at 82.6 and 90.4 F1 and ROC AUC. This is significant as it shows that ECG + SP02 monitoring, which are significantly more simple and less invasive to operate in the home setting, still generate competitive results when compared to the full battery of polysomnography signal collection. This relation was loosely reflected in the results generated from this recreation, though this replicated data is largely useless as there is no true predictive power.

The LLM suggested 4 additional extensions or ablations: transfer learning, focal loss, ablation on transformer components, and temporal information enhancement. Transfer learning is valid since even though we are mainly focused on pediatric sleep apnea, there is limited samples of NCH and CHAT datasets compared to adult datasets like SHSS or MESA. Focal Loss is valid since sleep apnea events are less common and harder to classify than normal breathing which produces a class imbalance. Ablation on transformer components is valid because the researchers did not provide a reason why they chose these layers on this transformer architecture or which layers contributes to the performance more. It can help with optimization of the model size and which components are the most important for sleep apnea detection. Temporal information is valid because there are several sleep stages we encounter throughout the night and may lead to missing some relevant temporal dependencies. The only problem for this is there is already a limited data for pediatric sleep and the problems encountered with preprocessing will be further increased.

It is especially important to be specific about the number of results you want from the LLM. When we first asked for suggestions about additional extensions or ablations, the LLM gave us a list of more than 20. We had to add that we wanted the most relevant and feasible extension/ablation as well as give an explanation to some of their pros and cons. We went with Focal Loss as this looked to be a feasible extension to add as well as relevant to sleep apnea data.

Discussion

There are several implications of the experimental results. Using two signals, ECG and SpO2, the detection performance of sleep apnea is comparable to the full polysomnography signals testing. This is significant because ECG and SpO2 are easier to collect with commercial devices compared to the other polysomnography signals. Because of the easier access, this suggests that the easier tests can be done at home and not needed to be collected in a sleep lab. The transformer-based model performed better than the previous models such as CNN, SE-MSCNN, CNN+LSTM, and Hybrid across both NCH and CHAT data. The multi-modal approach is effective because it incorporates information from the different signal types. Also the transformer architecture is more flexible in handling the huge variability in signal quality. The model has a consistently robust performance among the different pediatric age ranges. Currently at-home testing is only suggested to be effective in adults and older children, so this research suggest that this approach can be done throughout the entire pediatric population. The results also show that the model sustains its performance even with added noise. This could imply that the testing can be used even with commercial products that usually have lower qualities of signals.

The original paper wasn't fully reproducible. The difficult factor that made it irreproducible was mainly the raw data and the models used. The raw data uses several hundreds of EDFs which is each around 500mb - 1.6gb in size. This caused several issues in the preprocessing step and we had to generate our own preprocessed data to continue with other parts of the research. We were also not able to reproduce the model the researchers used fully. Although we tried implementing the model they described, the results we gathered was not in line with what the researchers collected. Training and evaluation was the easier parts to reproduce as they were explained in more detail by the researchers in terms of both calculation and in their code.

A recommendation to the authors could be creating a readme file with a detailed setup instruction that includes dependencies, hardware requirements and expected runtime. This will help future researchers of the topic to save time in debugging and especially wrangling the raw data. Another recommendation is creating a configuration file instead of hard-coded paths in their GitHub. They can also include anonymized sample data themselves for testing and validation by future researchers. For future researchers a recommendation would be reporting the performance of the models across different demographic groups like gender. They can also use container technologies like Docker to make sure the research is reproducible in different environments.

Author Contributions

Workload separation was done as shown in Table 3 of Appendix A.

APPENDIX A

Table 1. Project Dependencies

absl-py==2.2.2 astunparse==1.6.3 bidict==0.23.1 biosppy==2.2.3 certifi==2025.1.31 charset-normalizer==3.4.1 colorama==0.4.6 contourpy==1.3.2 cycler==0.12.1 decorator==5.2.1 flatbuffers==25.2.10 fonttools==4.57.0 gast==0.6.0 google-pasta==0.2.0 grpcio==1.71.0 h5py==3.13.0 idna==3.10 Jinja2==3.1.6 joblib==1.4.2 keras==3.9.2 keras_contrib @ git+https://www.github.com/keras-team/keras-contrib.git@3fc5ef709e061416f4bc8a92ca3750c824b5d2b0 kiwisolver==1.4.8 python-dateutil==2.9.0.post0 pytz==2025.2 PyWavelets==1.8.0	lazy_loader==0.4 libclang==18.1.1 Markdown==3.8 markdown-it-py==3.0.0 MarkupSafe==3.0.2 matplotlib==3.10.1 mdurl==0.1.2 ml_dtypes==0.5.1 mne==1.9.0 mock==5.2.0 namex==0.0.8 numpy==2.1.3 opencv-python==4.11.0.86 opt_einsum==3.4.0 optree==0.15.0 packaging==25.0 pandas==2.2.3 PeakUtils==1.3.5 pillow==11.2.1 platformdirs==4.3.7 pooch==1.8.2 protobuf==5.29.4 Pygments==2.19.1 pyparsing==3.2.3	requests==2.32.3 rich==14.0.0 scikit-learn==1.6.1 scipy==1.15.2 setuptools==79.0.0 shortuuid==1.0.13 six==1.17.0 tensorboard==2.19.0 tensorboard-data-server==0.7.2 tensorflow==2.19.0 termcolor==3.0.1 threadpoolctl==3.6.0 tqdm==4.67.1 typing_extensions==4.13.2 tzdata==2025.2 urllib3==2.4.0 Werkzeug==3.1.3 wheel==0.45.1 wrapt==1.17.2

*Dependencies are stored as ‘requirements.txt’ in the project Github Repository

Figure 1. Model Architectures

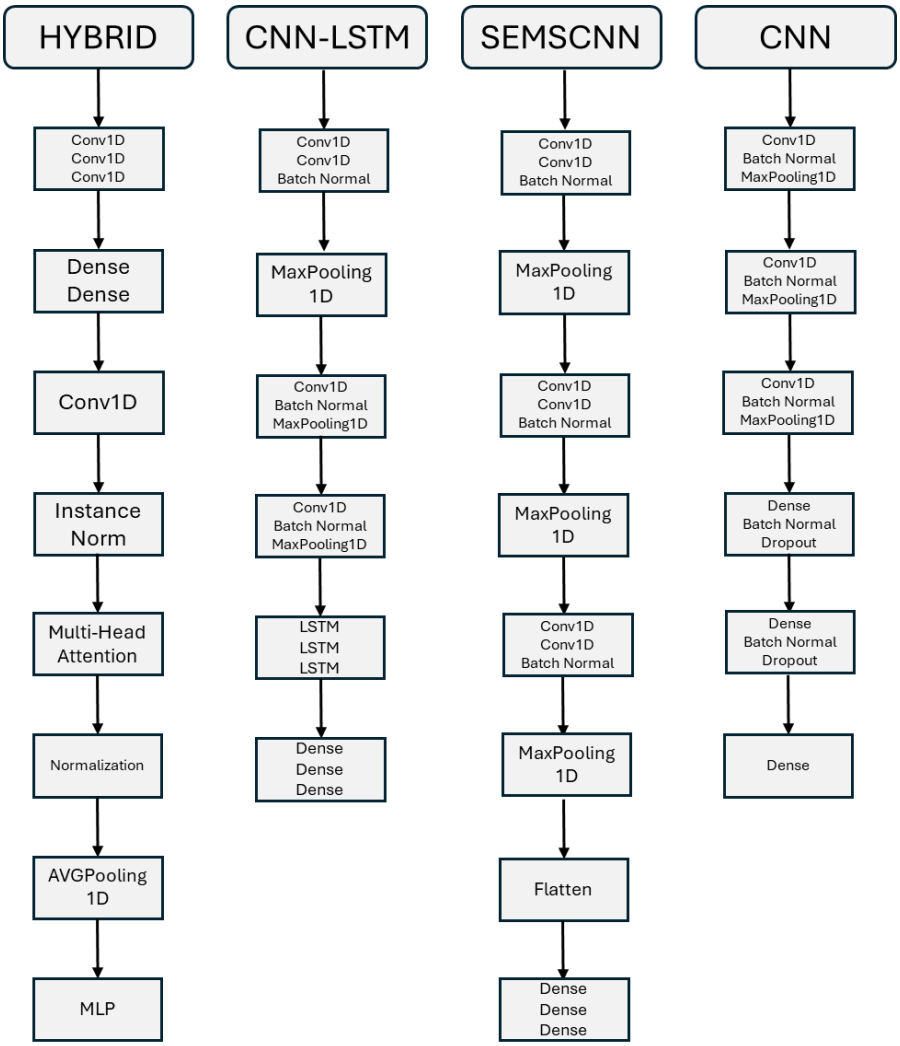

Table 2. Hybrid Model Evaluation F1 and ROC AUC Scores

EOG	EEG	RESP	SPO2	ECG	DEMO	CO2	F1	ROC AUC
X	X	X	X	X	X	X	0.00	66.35
X	X						0.00	66.73
X		X					0.00	66.49
X			X				0.00	66.53
X				X			0.00	66.43
X					X		0.00	64.55
X						X	0.00	66.13
X							0.00	66.53
	X	X					0.00	63.86
	X		X				0.00	65.86
	X			X			0.00	66.1
	X				X		0.00	66.29
	X					X	0.00	66.08
	X						0.00	52.05
		X	X				0.00	66.55
		X		X			0.00	66.28
		X			X		0.00	65.87
		X				X	0.00	66.42
		X					0.00	66.61
			X	X			0.00	65.93
			X		X		0.00	66.36
			X			X	0.00	66.31
			X				0.00	62.16
				X	X		0.00	66.37
				X		X	0.00	42.16
				X			0.00	46.91
					X	X	0.00	65.69
					X		0.00	65.95
						X	0.00	42.52

Table 3. Workload Separation

Author	Task / Section
Alobba, Francis (falobba2)	Introduction, Methodology – [Training & Evaluation], Results – [Additional Extension / Ablation], Discussion
Wall, Camden (cnwall2)	Methodology – [Data, Model], Results – [Tables, Figures, Discussion of Results], PyHealth Contribution (in GH repository)

REFERENCES

1. Alobba. Francis, Wall. Camden "SP25CS598DLH" , Github, 2025 [Online] Available: https://github.com/cnwall99/SP25CS598DLH
2. Fayyaz. Hamed, Strang. Abigail, Beheshti, Rahmatollah, "Bringing At-home Pediatric Sleep Apnea Testing Closer to Reality: A Multi-modal Transformer Approach" *Proc Mach Learn Res*. August, 2023 [Online] https://pmc.ncbi.nlm.nih.gov/articles/PMC10854997/pdf/nihms-1925296.pdf
3. Fayyaz. Hamed, "Pediatric-Apnea-Detection", GitHub, 2023 [Online] Available: https://github.com/healthylaife/Pediatric-Apnea-Detection
4. OpenAI. (2024). *GPT-4o mini model* [Large language model]. [Online] Available: https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/