A Deep Learning Approach to Respiratory Phase Detection from Audio Spectrograms

Machine Learning I · Spring 2025 · Kaggle Project

Dohyeop Lim 24101518 Songwon Won 24102401

Team: ST_ML2025_2

Department of Artificial Intelligence, SeoulTech

***Abstract*—We present a deep learning framework for non-invasive respiratory phase classification (inhalation/exhalation) from one-second audio clips. Our hybrid-feature approach uses nine spectral representations, including Mel-spectrograms and MFCCs, and 39 scalar features to train two distinct Convolutional Neural Networks (CNNs): a lightweight 2.43M parameter model and a larger 8.15M parameter VGG-inspired architecture. To improve generalization against real-world acoustic variability, the models are trained with CutMix and MixUp data augmentation. A final weighted ensemble of these models, with weights determined by validation performance, achieves a classification accuracy of 76.7% on the private test set.**

I. Introduction

A. Problem Definition

While breathing patterns offer valuable clues for detecting chronic disease, practical tools for home monitoring are missing. The existing options are simply not viable for widespread use. Hospital-grade devices are prohibitively expensive and complex, and manually reviewing audio data is an unscalable process unsuitable for real-time applications [1].

The most obvious alternative—listening to the breath—is simple and non-invasive. But it has a critical flaw. Inhales and exhales often sound remarkably alike, a similarity that confounds simple algorithms (Fig. 1). This problem is amplified by real-world conditions like background noise or differences between individuals, causing conventional acoustic methods to fail. A more sophisticated interpretive model is therefore necessary to reliably classify respiratory phases from sound alone [2].

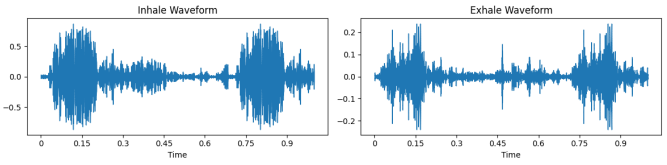

Fig. 1. Example waveforms showing inhale vs exhale patterns

B. Research Objectives

This study confronts the problem by designing a deep learning model specifically trained to differentiate between the subtle acoustic signatures of inhalation and exhalation. We feed a rich tapestry of acoustic information—spanning multiple spectrogram formats, frequency-domain vectors, and signal energy characteristics—into a multi-input Convolutional Neural Network(CNN).

C. Dataset Overview

The dataset provided for this study consists of 5,000 individual audio recordings of human respiratory sounds. The data is pre-partitioned into a training set of 4,000 samples and a test set of 1,000 samples. Each sample is a one-second, mono-channel audio file in the .wav format, and labled either an inhalation or an exhalation for the training set.

II. Method

This section details the feature extraction process, model architectures, and training procedures used for respiratory phase classification. The overall feature extraction pipeline is illustrated in Fig. 2.

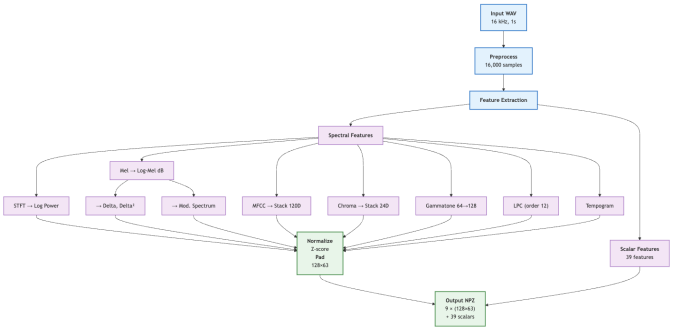

Fig. 2. Feature extraction pipeline diagram

A. Feature Extraction

The input to our system is a one-second audio clip sampled at 16 kHz. From each clip, we extracted two categories of features: spectral representations and scalar values. All features were pre-computed and stored as NumPy archives to accelerate the training pipeline.

1) Spectral Representations: We computed nine distinct time-frequency representations: STFT, Mel-spectrogram with its first (delta) and second (delta-delta) derivatives, Mel-Frequency Cepstral Coefficients (MFCC), chroma, gammatone, modulation spectrum, and tempogram. Visual comparisons for these features are provided in Figs. 3-9. All spectral features were reshaped or padded to a uniform dimension of 128 frequency bins by 63 time steps and were normalized using z-scoring.

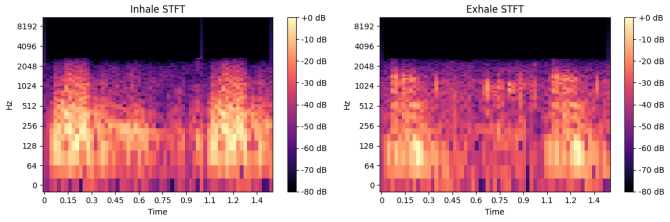

Fig. 3. STFT spectrograms for inhale/exhale

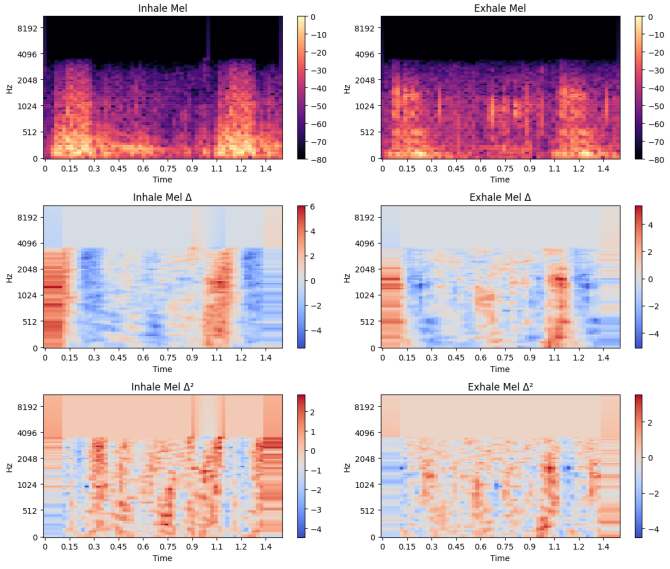

Fig. 4. Mel spectrograms with delta coefficients

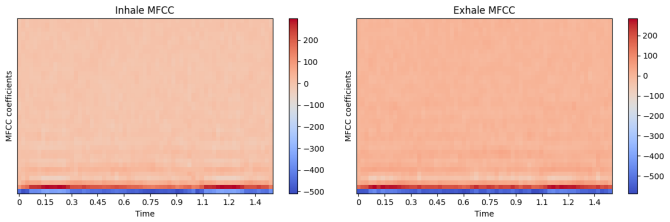

Fig. 5. MFCC comparison

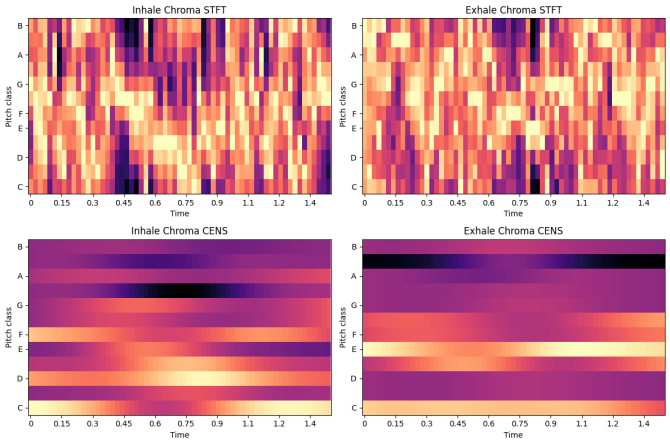

Fig. 6. Chroma STFT and CENS features

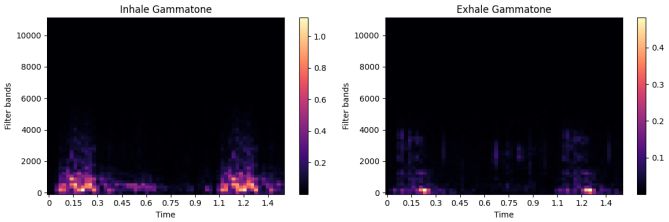

Fig. 7. Gammatone filterbank features

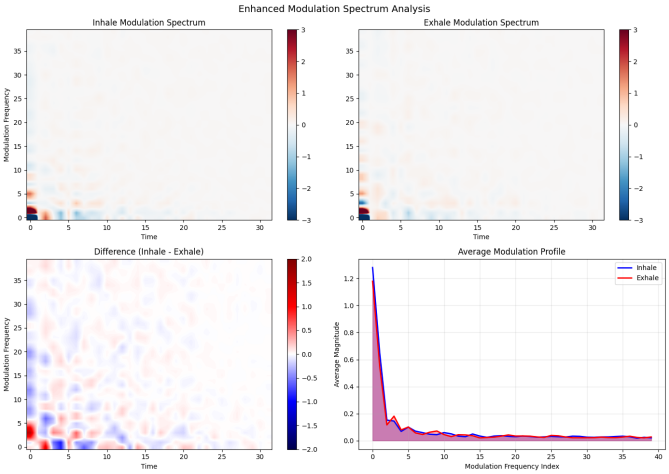

Fig. 8. Modulation spectrum analysis

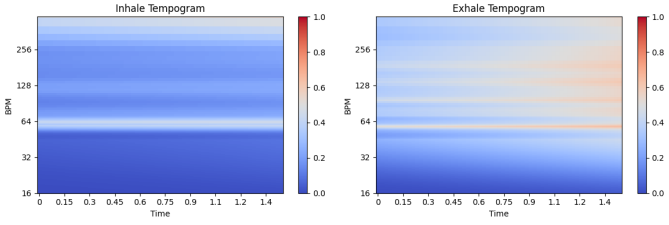

Fig. 9. Tempogram comparison

2) Scalar Features: We extracted 39 scalar features—spanning temporal metrics (e.g., RMS energy, ZCR) and spectral statistics (e.g., centroid, bandwidth)—to characterize the signal. Key feature comparisons between respiratory phases are plotted in Fig. 10.

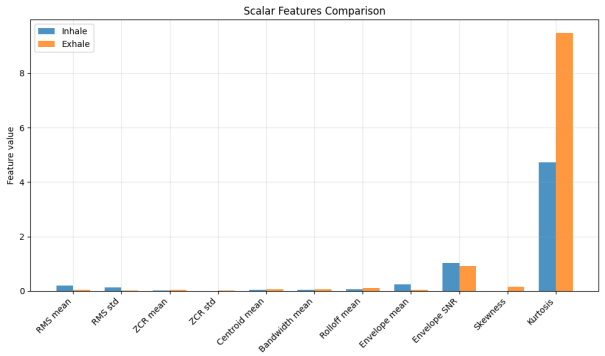

Fig. 10. Scalar features comparison

B. Model Architectures

We designed two Convolutional Neural Network (CNN) architectures to process the combined feature sets. Each model contains a convolutional pathway for spectral features and a Multi-Layer Perceptron (MLP) pathway for scalar features.

1) CNN8: This is a lightweight model with eight convolutional layers organized in four blocks ($32 \rightarrow 64 \rightarrow 128 \rightarrow 256$ channels). It uses ReLU activations, Batch Normalization, and max pooling. The scalar pathway is a two-layer MLP.

2) VGG-Inspired Model: This higher-capacity model contains 12 convolutional layers in four blocks ($64 \rightarrow 128 \rightarrow 256 \rightarrow 512$ channels) with GELU activations. A residual connection is included in the final block. The model contains 8.15M trainable parameters.

C. Training and Optimization

1) Optimizer: Models were trained with the *AdamW* optimizer using a weight decay, λ of 10^{-4} . The learning rate followed a cosine annealing schedule with a linear warmup. Initial learning rates were set to 4×10^{-4} for CNN8 and 1×10^{-3} for the VGG-inspired model. Gradient norms were clipped at a maximum value of 1.0.

2) Data Augmentation: After a four-epoch warmup, we applied stochastic data augmentation. The methods used were *CutMix*, with a probability of 0.6, and *MixUp*, with a probability of 0.4.

3) Training Configuration: We used a binary cross-entropy with logits loss function. The data was split into training (80%) and validation (20%) sets using a stratified method. Training ran for a maximum of 140 epochs with early stopping based on validation accuracy. Mixed-precision training was used to reduce computation time.

D. Ensemble Method

Final predictions for the test set were generated by an ensemble of the trained models. The sigmoid outputs of the models were combined using a weighted average, as shown in (1).

$$P_{\text{final}} = \sum_{i=1}^{N} \alpha_i \cdot \sigma(\text{logits}_i) \tag{1}$$

The weights, α_i , were derived from the softmax-normalized validation accuracy of each contributing model, giving greater influence to better-performing models.

III. Results

A. Model Performance

Table I summarizes the performance of our two base classifiers (CNN8 and VGG-inspired) alongside their final ensemble. Although the individual models reached 77.8% and 79.2% accuracy on the validation split, the ensemble delivered superior real-world results—achieving 78.6% on the Kaggle public test and 76.7% on the private holdout—outperforming the best single model by over two percentage points and demonstrating the clear benefit of combining their complementary strengths.

TABLE I
COMPARATIVE PERFORMANCE OF THE BASELINE CNN8, VGG-INSPIRED MODEL, AND THEIR FINAL ENSEMBLE.

Model	Validation					Test (Acc.)	
	Acc.	AUC	Prec.	Rec.	F1	Public	Private
CNN8	0.778	0.831	0.780	0.775	0.777	0.761	0.737
VGG-inspired	0.792	0.845	0.796	0.789	0.792	0.771	0.747
Ensemble	–	–	–	–	–	**0.786**	**0.767**

Also, data augmentation with CutMix and MixUp boosted our test-set accuracy from 72.0% (public) and 69.3% (private) to 78.6% and 76.7%, respectively—an improvement of over two percentage points versus the best individual model (Fig. 11).

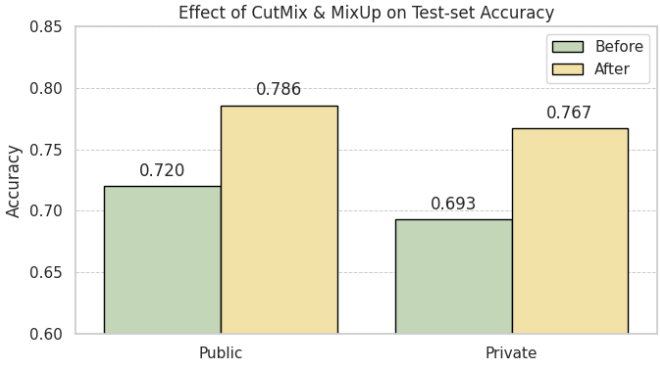

Fig. 11. Impact of CutMix and MixUp augmentation on Kaggle test-set accuracy. "Before" refers to the baseline (72.0% public / 69.3% private); "After" shows the augmented results (78.6% public / 76.7 % private).

IV. Discussion

A. Interpretation of Results

A key observation from our experiments is how well data augmentation methods from computer vision, specifically CutMix and MixUp, transferred to the domain of audio spectrograms. Their success here suggests a degree of cross-domain applicability, though our results also show the model benefits from a few warmup epochs to grasp basic patterns before augmentation begins. The value of the hybrid-feature design was also clear; adding the scalar feature vector consistently improved accuracy, pointing to complementary acoustic information that spectral images alone do not capture.

One of the main takeaways is the trade-off between model size and performance. The VGG-inspired network, despite having over three times the parameters, was only marginally better than the lightweight CNN8. The fact that our ensemble outperformed both models underscores that, for this problem, combining architecturally different models is a more effective strategy than simply increasing the scale of a single network.

B. Limitations

This study has two main limitations. First, the fixed one-second segmentation of audio clips prevents the models from learning temporal dependencies that span multiple respiratory cycles. Information about the overall breathing rhythm is lost.

Second, our feature engineering strategy, while effective, is fixed. An end-to-end approach that learns features directly from raw audio might discover more optimal, task-specific representations.

C. Future Work

Based on these findings, future research could proceed in several directions. To address the limitation of fixed-length inputs, recurrent or attention-based architectures (e.g., LSTMs, Transformers) could be employed to model long-range temporal context. An end-to-end learning framework should also be investigated as a potential alternative to hand-crafted features.

For clinical applicability, two extensions are proposed. First, model interpretability methods, such as gradient-based attribution, could be used to visualize which parts of a sound are most indicative of inhalation or exhalation. Second, the binary classification task could be expanded to a multi-class problem that includes other respiratory events, such as breath-holds and coughs, to create a more comprehensive monitoring tool.

V. Conclusion

We presented a lightweight reproducible pipeline for non-invasive detection of breathing phases from short audio clips. We extracted spectral and scalar features and used CutMix and MixUp to improve Kaggle test set accuracy from 72.0% public and 69.3% private to 78.6% and 76.7% respectively. A simple ensemble of CNN8 and a VGG inspired network added two percentage points and showed the value of model combination. The system requires no specialized hardware or manual review and suits real time home use scenarios. Future work will address classification of pathological sounds such as wheeze and evaluate end to end deep models on raw audio data to improve performance.

Appendix A
Code Availability

The source code for this project is publicly available at: https://github.com/dohyeoplim/breathing-phase-classifier

References

[1] D. Lee, "호흡 모니터링을 위한 저전력 고속 웨어러블 이산화탄소 센서 개발," KAIST News, Online, Mar. 2025, accessed: Jun. 19, 2025. [Online]. Available: https://times.kaist.ac.kr/news/articleView.html?idxno=22164

[2] S. Lee, "≪후생신보≫ ai로 '짹짹'거리는 숨소리 '천명음' 찾는다," Whosaeng News, Online, Feb. 2023, accessed: Jun. 19, 2025. [Online]. Available: https://www.whosaeng.com/141894