Faculty of Engineering, Environment and Computing
School of Science

MSc Data Science
7150 CEM Data Science Project

Project Report

Attention-Based CNN-LSTM for Respiratory Sound Classification: Enhancing Robustness with Data Augmentation and Model Explainability

By

Yvonne Musinguzi
SID: 15094816

Supervisor: Dr. Daniyal Haider

Submitted in partial fulfilment of the requirements for the Degree of Master of Science in Master of Science in Data Science

Academic Year: 2025/26

Declaration of Originality

I declare that this project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet etc.) has been acknowledged by citation within the main report to an item in the References or Bibliography lists. I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

Statement of copyright

Statement of ethical engagement

I declare that a proposal for this project has been submitted to the Coventry University ethics monitoring website (https://ethics.coventry.ac.uk/) and that the application number is listed below (Note: Projects without an ethical application number will be rejected for marking)

Signed: Date: **15/08/2025**

First Name:	**Yvonne**
Last Name:	**Musinguzi**
Student ID number	**15094816**
Ethics Application Number	**P187197**
1st Supervisor Name	**Dr. Daniyal Haider**
2nd Supervisor Name	

ABSTRACT

Table of Contents

Chapter 1– Introduction

1.1 Background of Study

From the moment Hippocrates first wrote about "pleuritic crackles" more than two thousand years ago, clinicians have relied on sound to understand the lungs. This practice took a big step forward in 1816 when René Laennec rolled a sheet of paper into a tube, leading to the invention of the stethoscope. While the device has evolved, the core act of a clinician listening to a patient's chest remains the same, a moment of judgment that can be life-saving but is also often uncertain.

This uncertainty is the stethoscope's major weakness. Interpreting these sounds is subjective, and even experienced pulmonologists often disagree. Studies like Gurung et al. (2019) found that they agreed on fine crackles only about half the time. Small changes in stethoscope placement, device type, or even a listener's fatigue can change a diagnosis. The stakes are high, as lower-respiratory infections, COPD, and asthma are a huge global health burden, especially in places with limited access to specialists or imaging.

To address this, digital stethoscopes enabled recordings to be stored and analyzed. Early machine learning models used hand-crafted features but struggled with noise and temporal context. By the mid-2010s, deep learning took over. CNNs proved great at finding patterns in spectrograms, while LSTMs could capture temporal sequences. Hybrid CNN-LSTM models performed well on datasets like ICBHI 2017 but often failed in noisy, real-world conditions.

This leads to three key challenges that I am addressing in this study: noise vulnerability, dataset imbalance, and model opacity. Using the ICBHI 2017 dataset, my work systematically compares several deep learning architectures: a baseline MLP, a CNN, a CNN-LSTM, and a CNN-LSTM with an attention mechanism. To handle the issues, I applied class weighting to address imbalance, used noise augmentation to improve robustness, and employed LIME to make my model's predictions more transparent.

By grounding model design in these known limitations and evaluating its performance across metrics like F1-score and ROC-AUC, I aim to create not just an accurate classifier, but one that is more robust, fairer, and transparent for clinical use.

1.2 Problem Statement

Despite significant advances in digital stethoscopes and deep learning, current respiratory sound classification systems still face challenges in achieving clinical reliability. Models trained on controlled datasets often perform poorly when applied to real-world recordings with noise, different devices, or

diverse patient populations. These performance gaps reduce clinical trust and risk widening healthcare inequalities, particularly in low-resource settings.

Although CNNs and LSTMs have improved classification accuracy, they are often opaque in their decision-making. This lack of transparency limits their use in clinical workflows, where explainability is essential for safety and regulatory compliance. Current explainability methods are also underused and not always clinically validated.

Another challenge lies in capturing the nuanced temporal and spectral patterns of respiratory sounds. Without a way to focus on key areas, models can overlook subtle but meaningful features. Class imbalance compounds this issue, as rarer pathologies are more likely to be misclassified.

This study addresses these gaps with a staged approach, starting with a baseline MLP and progressing to CNN, CNN-LSTM, and finally, a CNN-LSTM-Attention model. To improve robustness and balance, I used class weighting and spectrogram noise augmentation. For transparency, LIME was applied to provide insights into model predictions. The goal is to evaluate which architectural strategies most effectively improve performance while supporting interpretability.

While the ultimate goal is to classify diseases like asthma or pneumonia, these diagnoses rely on the accurate detection of underlying sounds like crackles and wheezes. This study, therefore, focuses on the foundational task of classifying these sound events, ensuring the resulting model provides a robust and trustworthy platform for future disease-level predictions.

1.3 Research Aim and Objectives

The diagnostic journey of respiratory sound analysis, from René Laennec's stethoscope to modern machine learning approaches, highlights an ongoing need for accurate, robust, and interpretable tools that can operate effectively in real-world, noisy environments. Despite promising advances with deep learning, current systems often remain vulnerable to environmental noise, dataset imbalance, and a lack of transparency in decision-making processes.

Responding to these challenges, this research aims to improve the accuracy, robustness, and interpretability of respiratory sound classification systems. Specifically, it leverages an attention-based hybrid deep learning model combining Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) layers, supported by targeted data augmentation and explainability through Local Interpretable Model-agnostic Explanations (LIME). By aligning with emerging regulatory priorities for transparency and fairness, this work seeks to contribute a reliable classification framework for respiratory cycles labelled as normal, crackle, wheeze, or both.

1.3.1 Research Aim

To enhance the accuracy, robustness, and interpretability of respiratory sound classification systems by implementing and evaluating an attention-based CNN-LSTM model, complemented by targeted data augmentation and LIME explainability, for the classification of normal, crackle, wheeze, and combined crackle–wheeze respiratory cycles.

1.3.2 Research Objectives

To achieve this aim, the study pursued the following objectives:

1. **Data Preparation:** Curate and preprocess the ICBHI 2017 respiratory sound dataset at the respiratory cycle level, including segmentation, resampling, and standardization to fixed-length audio segments
2. **Feature Extraction**: Extract Mel-Frequency Cepstral Coefficients (MFCCs) for the baseline model and log-Mel spectrograms for CNN-based models, ensuring feature scaling for optimal learning.
3. **Model application**: Implement and evaluate multiple deep learning architectures i.e. A baseline MLP (MFCC input), CNN (spectrogram input), and CNN-LSTM with attention (spectrogram input).
4. **Data Augmentation**: Apply targeted augmentation techniques (e.g., spectrogram noise injection) to simulate real-world acoustic variability while preserving diagnostic sound features.
5. **Explainability**: Integrate LIME in the final evaluation phase to generate interpretable, localized explanations of model predictions for improved transparency.
6. **Evaluation:** Compare models using metrics such as accuracy, macro-F1 score, ROC-AUC, and confusion matrices, assessing both classification performance and robustness to noise and class imbalance.

1.4 Research Questions

This section outlines the central research question that drives this study, along with a set of secondary questions that explore the problem space in greater detail. These questions are directly informed by the project's objectives and serve to guide both the design and evaluation of the proposed system.

1.4.1 Primary Research Question

Can an attention-based CNN-LSTM model, trained on augmented respiratory cycle data from the ICBHI 2017 dataset, outperform baseline models in both classification accuracy and interpretability for distinguishing between normal, crackle, wheeze, and combined crackle and wheeze sounds?

1.4.2 Secondary Research Questions

To explore the primary research question in depth, the following secondary questions are posed:

1. How effectively can targeted spectrogram-based data augmentation, such as noise injection, improve model robustness against real-world noise and class imbalance?
2. To what extent does the inclusion of an attention mechanism in the CNN-LSTM architecture enhance the model's ability to focus on diagnostically relevant regions in the time and frequency space?
3. How does the use of LIME contribute to the interpretability and transparency of the model's predictions in the context of respiratory sound classification?
4. Can the proposed hybrid model maintain strong classification performance across all four classes (normal, crackle, wheeze, and combined) when compared with baseline MLP and CNN models?
5. Which audio feature representation, MFCCs or log-Mel spectrograms, offers the most effective input for accurate and robust respiratory cycle classification?

1.5 Scope of the Study

This research sits at the intersection of biomedical signal processing, deep learning, and clinical decision support. The study is technically rigorous, clinically relevant, and ethically sound, while being feasible for a master's research project.

The core of the study is the evaluation of an attention-based CNN-LSTM model trained on the ICBHI 2017 dataset. This dataset provides clinically annotated recordings of respiratory cycles labeled as normal, crackle, wheeze, or a combination. The classification task is cycle-based, ensuring clear alignment between labels and acoustic events.

Initially, the research design included an ambitious plan to use the Coswara dataset for cross-dataset testing to assess generalizability. However, due to time constraints, the study focused solely on ICBHI 2017. Evaluating the model on Coswara remains an important avenue for future work to provide deeper insight into cross-device and demographic robustness.

The project received formal ethical approval from Coventry University's ethics committee. The ICBHI 2017 dataset is publicly available, anonymized, and has been ethically cleared for research.

This study does not attempt full disease classification. While identifying specific respiratory diseases is the ultimate objective for future work, the current project focuses on classifying respiratory sounds (e.g., crackles, wheezes, both, or normal). This focus provides the essential foundation for accurate disease-level diagnosis in subsequent research.

Overall, the study is tightly focused on evaluating and improving model robustness and interpretability for respiratory cycle classification. While not for immediate clinical deployment, the findings aim to inform the future development of AI-assisted auscultation tools for real-world healthcare environments.

1.6 Significance of the Study

Respiratory conditions like asthma and pneumonia are a major cause of global illness and death, especially in low- and middle-income countries where advanced diagnostic tools are scarce. In these settings, healthcare workers often rely on auscultation, which, while immediate, is highly subjective and prone to error.

The foundation of many respiratory diagnoses lies in detecting specific sounds like crackles and wheezes. Recent advances in deep learning offer new ways to improve this process. However, models that perform well in controlled environments often fail when faced with real-world issues like ambient noise, variations in recording devices, and data imbalances. These limitations reduce clinical trust and raise concerns about equitable access to healthcare.

This study addresses these challenges by using the ICBHI 2017 dataset to implement and evaluate an attention-based CNN-LSTM model. The research is designed to improve model robustness through targeted spectrogram-based noise augmentation and to provide transparent insights into its predictions using LIME.

A further contribution of this work is its emphasis on a fair evaluation. The model's performance is assessed not only through traditional metrics like accuracy and F1-score but also through its interpretability and robustness. This aligns with modern regulatory and ethical guidelines that call for transparency and reliability in medical AI systems.

Academically, this research contributes to the underexplored area of using attention mechanisms and interpretable AI for biomedical audio analysis. Its findings can inform the development of future AI-assisted auscultation tools for low-resource settings.

Ultimately, the beneficiaries of this research include healthcare workers, patients, and AI researchers who need reliable and transparent medical applications. The outcomes also have relevance for policymakers and regulatory bodies.

1.7 Structure of the Report

This report is organized into seven chapters, each contributing to the development, evaluation, and context of a robust and interpretable deep learning model for respiratory sound classification.

Chapter 1: Introduction This chapter provides the historical and clinical background of respiratory sound analysis, outlines the research problem, states the aim and objectives, and defines the scope and significance of the study. It also presents the research questions that guide the investigation.

Chapter 2: Literature Review This chapter reviews existing approaches to respiratory sound classification, from traditional methods to modern deep learning. It discusses key challenges like noise sensitivity, dataset imbalance, and a lack of transparency to position the study within the current research landscape.

Chapter 3: Methodology This chapter describes the experimental design, including dataset selection, preprocessing steps, feature extraction, and model architectures. The integration of LIME for post-hoc interpretability is explained, along with the evaluation metrics and ethical considerations. The ethical approval certificate is included in the appendix.

Chapter 4: Results and Discussion This chapter presents the quantitative and qualitative results from model training and evaluation, covering four-class classification. The performance metrics, such as accuracy and F1-score, are analyzed, and the interpretability outcomes from LIME are discussed in relation to the research objectives and literature gaps.

Chapter 5: Conclusion and Future Work This chapter summarizes the key findings and their implications for AI-assisted auscultation. It acknowledges the study's limitations and provides recommendations for future research, including exploring other datasets and conducting more advanced robustness testing.

Chapter 6: Ethical, Legal, and Social Considerations This chapter examines the ethical, legal, and societal dimensions of developing and deploying AI-based respiratory sound systems. This includes patient privacy, algorithmic bias, and compliance with regulatory frameworks like the FDA's principles and the EU AI Act.

Chapter 7: Project Management This chapter details the planning, execution, and monitoring of the research project, including the timeline, resource allocation, risk management strategies, and progress tracking.

Appendices: The appendices that include the ethical approval certificate and link to the complete project code.

This structure ensures a logical progression from foundations to implementation, evaluation, ethical reflection, and project management, supported by full transparency through publicly accessible code.

Chapter 2 – Literature Review

2.0 Introduction

Before engineering a solution, a critical understanding of the problem's nuances is essential. This chapter builds on Chapter 1 by examining the progression of research and technology in automated respiratory sound classification. The review goes beyond a simple timeline, assessing the performance, gaps, and clinical translation of previous work to inform this study.

The review begins with traditional, handcrafted-feature methods. While these approaches, using features like MFCCs, improved accuracy, they struggled with variability, noise sensitivity, and a lack of clinical context. The discussion then moves to deep learning, where CNN and LSTM models delivered notable performance gains by autonomously learning features from data. However, as documented by Knight and Reinke (2023), these models often underperformed in noisy, real-world environments, highlighting a significant need for improved domain robustness.

Despite these improvements, many models remain opaque to clinicians. Dataset bias also limits their generalizability across diverse populations and devices, as pointed out by Lee et al. (2024). While explainability tools like Grad-CAM aim to increase transparency, questions remain about their reliability.

These limitations shape the direction of this research, which focuses on a CNN-LSTM-Attention model trained on the ICBHI 2017 dataset. The model is supported by targeted data augmentation and explainability methods to improve robustness, enhance clinician trust, and address class imbalance.

2.1 The Evolution of Respiratory Sound Analysis

For decades, the classification of respiratory sounds relied on handcrafted feature extraction. Early methods used signal-processing principles to quantify acoustic signatures of pathological sounds. For instance, Bahoura (2009) applied statistical descriptors like zero-crossing rate and STFT outputs to differentiate crackles and wheezes. While these approaches were intuitive, interpretable, and transparent, they were also fragile and often faltered in the presence of real-world noise, recording variability, and inconsistent stethoscope use.

The 2017 ICBHI Challenge served as a major turning point, establishing a curated dataset and standard evaluation metrics that allowed for objective comparison. During this time, Mel-Frequency Cepstral Coefficients (MFCCs) emerged as the dominant representation, paired with machine learning models like SVMs and random forests. Studies documented macro-F1 scores in the 60–70% range, a respectable but not clinically reliable benchmark.

Despite careful design, these handcrafted pipelines suffered from key structural limitations:

- Noise Sensitivity: Features were highly vulnerable to ambient noise and recording inconsistencies.

- Lack of Temporal Context: Frame-based classification ignored the sequential dynamics of respiratory sounds, which is critical for clinical diagnosis.
- Curse of Dimensionality: Expanding feature space led to overfitting, especially with small, imbalanced datasets.

2.1.1 The Shift to Deep Learning

These limitations catalyzed a paradigm shift toward deep learning, which allowed models to learn hierarchical representations directly from data. Perna and Tagarelli (2019) showed that CNNs trained on log-mel spectrograms outperformed traditional SVM pipelines by identifying localized spectral patterns associated with sounds like wheezes and crackles.

Despite this progress, a critical gap remained: CNNs lacked an inherent understanding of temporal dependencies. This prompted the exploration of recurrent neural networks (RNNs), particularly Long Short-Term Memory (LSTM) models, to track patterns over time. Han and Ma (2021) further enhanced this approach by using attention mechanisms with LSTMs to boost diagnostic performance.

The literature eventually converged on hybrid architectures that combined the spatial feature extraction of CNNs with the temporal processing of LSTMs. This approach mirrored the diagnostic process of a clinician, who identifies both the sound and its timing. Later studies refined this by incorporating attention mechanisms and evaluating the robustness of these hybrid models under domain shifts, such as moving from clinical-grade stethoscopes to smartphone microphones.

2.2 From Analogue Auscultation to Digital Signal Processing

Auscultation, systematised by René Laennec in 1816, marked a turning point in diagnosis. Using a rolled-up paper, soon replaced by the stethoscope, Laennec enabled non-invasive detection of thoracic disease (Forgacs, 1978). For over a century, the method remained analogue and subjective, dependent on physicians' auditory memory and intuition. Forgacs (1978) described it as more craft than science, rich in nuance but inconsistent.

The arrival of digital stethoscopes in the late 20th century transformed practice. Devices like the Littmann 3200 and Thinklabs One allowed clinicians to record, visualize, and analyze sounds. As Bahoura (2009) noted, this created reproducibility, enabling acoustic events to be revisited independent of memory. Amplification revealed faint crackles, and high-fidelity storage allowed detailed computational analysis. The stethoscope evolved from a listening tool to a diagnostic instrument with computational potential.

Early computational approaches relied on handcrafted acoustic features: zero-crossing rates for crackle onsets, spectral centroids for wheeze localization, and Mel-Frequency Cepstral Coefficients (MFCCs) to capture spectral envelopes (Perna & Tagarelli, 2019). These features fed into machine learning models such as k-NN, SVMs, random forests, and decision trees, often achieving 60–72% accuracy on small datasets or ICBHI subsets (Barata et al., 2020).

Three persistent challenges limited these systems:

- Covariate shift: Performance dropped sharply when recording conditions differed, such as stethoscope brand, ambient noise, or patient posture (Knight & Reinke, 2023).
- Temporal fragmentation: Frame-based analysis ignored the full respiratory cycle, missing clinically relevant timing differences between, for example, end-inspiratory and mid-expiratory crackles (Forgacs, 1978; Han & Ma, 2021).
- Curse of dimensionality: Large feature sets caused overfitting on small, imbalanced datasets. Dimensionality reduction reduced computational load but risked losing subtle diagnostic cues (Barata et al., 2020; Bahloul et al., 2023).

Despite their limitations, handcrafted pipelines established reproducible sound taxonomies and standardized pre-processing. They also highlighted the importance of feature quality, robustness, and temporal context. These lessons directly informed the design of modern deep-learning architectures, such as CNNs, LSTMs, and attention-based hybrids, which aim to overcome these earlier constraints.

2.2.1 Deep Learning in Respiratory Sound Classification

By the mid-2010s, research began moving away from handcrafted acoustic features toward deep learning. This change followed the plateau of traditional methods, which struggled with noise sensitivity, limited temporal context, and reliance on manually selected features.

Convolutional Neural Networks (CNNs), originally designed for image recognition, proved effective for respiratory sound analysis when applied to time–frequency representations such as spectrograms. Perna and Tagarelli (2019) showed that CNNs trained on ICBHI 2017 spectrograms outperformed support vector machines and random forests by learning discriminative, multiscale features directly from data. They could detect harmonic wheeze bands and transient crackle bursts using spatial filters without explicit feature engineering.

CNNs, however, lack temporal memory. They process local patterns well but cannot follow how these patterns change over an entire breath cycle, which is crucial in diagnosis. Long Short-Term Memory (LSTM) networks address this by modelling sequential dependencies, capturing cues such as onset, duration, and clustering of abnormal sounds. Han and Ma (2021) demonstrated that combining CNNs with LSTMs improved temporal localisation and increased accuracy over CNNs alone.

Hybrid CNN-LSTM architectures became increasingly common. Knight and Reinke (2023) reported macro-F1 scores above 0.85 on ICBHI, but these results came mainly from controlled experimental settings. When tested on noisy, real-world recordings such as smartphone samples from hospital corridors, performance dropped by up to 15 percentage points. This highlighted a limitation: strong benchmark results do not always translate into robustness across devices, environments, or demographics.

These challenges shifted research priorities from accuracy alone to robustness, fairness, and interpretability. The present study builds on these lessons through a staged approach: starting with a baseline MLP on MFCC features for interpretability, moving to CNNs for spatial learning, adding LSTMs for temporal context, and finally incorporating attention mechanisms to focus selectively on diagnostically relevant phases of the respiratory cycle.

2.3 Attention Mechanisms and Explainability in Deep Respiratory Models

Even with CNN-LSTM hybrids, distinguishing clinically important sounds from irrelevant noise remains a challenge. Attention mechanisms address this by assigning higher weights to parts of the input that are most relevant for classification.

In respiratory audio, attention can amplify subtle crackles at the end of inspiration while reducing the influence of unrelated sounds. Han and Ma (2021) applied channel attention for pediatric wheeze detection, highlighting harmonics while suppressing less relevant frequency bands. Perna and Tagarelli (2019) integrated attention into pooling layers, which improved robustness and allowed for implicit temporal reasoning without explicitly relying on LSTMs.

Attention mechanisms also contribute to explainability by producing saliency maps that indicate which parts of an input influenced a decision. This transparency aligns with regulatory expectations such as the FDA's Good Machine Learning Practice Guidelines (2021) and the EU AI Act (2025). However, visual attention does not always match clinical reasoning. Barata et al. (2022) found that Grad-CAM sometimes highlighted irrelevant frequency regions. To further enhance interpretability, model-agnostic tools such as LIME have been used, but can be unstable for audio data.

In this study, attention is applied within a CNN-LSTM architecture to help the model focus on the most informative time-frequency regions. Class weighting and data augmentation are used alongside attention to improve performance in imbalanced conditions. This approach reflects the progression of the experimental pipeline, where each architectural addition addresses specific shortcomings identified in earlier stages.

2.4 Dataset Limitations and Bias in Clinical Audio

Respiratory sound classification is constrained not only by model design but by the quality and diversity of training data. While key public datasets like ICBHI 2017, Coswara, and HF-Respire have advanced research, each presents limitations that affect generalisability and fairness.

ICBHI 2017, the main benchmark, contains over 5.5 hours of annotated respiratory cycles from 126 patients. Models trained on it can achieve macro-F1 scores above 0.85 (Rocha et al., 2019; Perna & Tagarelli, 2019) but often drop 10–15 points when tested on recordings from other devices or in noisy hospital settings (Knight & Reinke, 2023).

Coswara, collected via smartphones, offers broader demographic and environmental diversity (Coswara Team, 2020), but its self-reported labels and uncontrolled acoustics limit reliability. HF-Respire, gathered in ICU environments, captures realistic but noisy conditions with alarms and ventilator sounds (Wearable Respiratory Monitoring Consortium, 2022), yet its restricted access limits adoption.

Demographic imbalances are common. Lee et al. (2024) found that ICBHI overrepresents adult males, risking poorer performance for paediatric and female patients. Mitigation strategies include cross-dataset training and domain adaptation, such as domain-adversarial neural networks aligning ICBHI and Coswara (Senoussaoui et al., 2024) or waveform-based CNNs trained on data from both hospital and consumer-grade stethoscopes (Bahloul et al., 2023).

This study focuses on ICBHI 2017 for its controlled, well-annotated data. Although the planned Coswara integration was not completed, its principles informed class imbalance handling and noise simulation (Hoang et al., 2022). Ultimately, dataset quality sets the ceiling for model performance, making careful selection, augmentation, and bias-awareness essential for robust and equitable diagnostic systems.

2.5 Data Augmentation and Domain Adaptation: Addressing Real-World Noise and Imbalance

Even the most advanced respiratory sound classifiers often underperform when exposed to the acoustic and demographic variability of real-world conditions. Two broad strategies are discussed in the literature to address this: data augmentation, which enriches within-dataset diversity, and domain adaptation, which bridges performance gaps across datasets, devices, and environments. While both have shown promise, this study focuses on targeted augmentation and dual-dataset training rather than full domain adaptation.

2.5.1 Data Augmentation: Enhancing Robustness with Noise Injection

Data augmentation introduces controlled variability into training data, helping models generalise to new and noisy environments. In respiratory sound analysis, common methods include pitch shifting, time stretching, and noise overlay. However, the effectiveness of these methods depends on preserving clinically relevant features. Overly aggressive augmentation can distort key patterns such as wheeze harmonics or crackle bursts, reducing interpretability (Hoang et al., 2022).

In this study, augmentation was limited to spectrogram noise injection during training. Gaussian noise was applied via a data generator to log-Mel spectrogram inputs, simulating environmental background interference while keeping the pathological features intact. This approach aligns with the goal of preparing the model for realistic deployment scenarios without introducing artefacts that could mislead explainability tools such as LIME.

2.5.2 Dataset Diversity as a Proxy for Domain Adaptation

While formal domain adaptation methods like Domain-Adversarial Neural Networks (DANN) are gaining traction , (Senoussaoui et al., 2024) instead encouraged robustness through dual-dataset training using the ICBHI 2017 and Coswara datasets.

ICBHI provides high-quality, expert-labeled data, while Coswara offers greater demographic and environmental diversity. Though not a strict domain adaptation, combining these datasets exposes the model to different devices and environments, partially mitigating domain shift effects (Knight and Reinke, 2023).

The augmentation in this study was deliberately bounded to add realism without introducing distortions. Noise was scaled to maintain the audibility of diagnostically relevant features, and transformations were applied only during training to avoid contaminating evaluation data.

By focusing on noise-based augmentation and dataset diversity, this study discussed in this report takes an informed step toward bridging the gap between controlled benchmark performance and real-world applicability.

2.6 Explainability and Clinical Trust: Making Machine Learning Transparent

In clinical decision-making, accuracy alone is insufficient. Metrics such as F1-score, sensitivity, and specificity can benchmark performance, but they reveal little about how a model arrives at a decision. For adoption in healthcare, models must produce not only correct outputs but also intelligible, clinically aligned explanations. This requirement has driven interest in Explainable Artificial Intelligence (XAI), particularly in respiratory sound classification where black-box models limit clinician confidence.

Deep learning architectures are inherently complex, making it difficult to trace their decision logic. In respiratory sound analysis, this opacity can create a gap between model predictions and clinical reasoning. As Barata et al. (2022) found, this uncertainty reduced trust and slowed clinical adoption, as clinicians often could not determine if a model focused on genuine pathological cues or irrelevant artefacts.

To address this, a variety of tools have been adapted to improve interpretability. Grad-CAM highlights influential regions of a spectrogram, while Prototypical Part Networks (ProtoPNet) compare new inputs to learned prototypes (Han & Ma, 2021). However, this study selected LIME (Local Interpretable Model-agnostic Explanations) because it is model-agnostic and can be applied directly to the final architecture to identify which time-frequency regions of a spectrogram most influenced a decision.

Quantifying the usefulness of these methods is essential. Common metrics include Faithfulness (whether the explanation accurately reflects the model's reasoning) (Samek et al., 2017), Clinician Plausibility, and Localization Accuracy (Hoang et al., 2022), which evaluates whether abnormal sounds are correctly located.

In this research, LIME was applied to the final CNN-LSTM-Attention model to generate local explanations for individual predictions on test samples. The resulting visual overlays identified the regions of the spectrogram that most influenced each classification, helping to assess whether the model was focusing on diagnostically meaningful cues and supporting clinical trust.

2.7 Regulatory Pressures and Ethical Stakes

Beyond the clinical setting, explainability is increasingly a regulatory requirement. The U.S. Food and Drug Administration (FDA) has stipulated that AI systems for medical use must provide "meaningful information" about their decisions. Similarly, the EU AI Act classifies clinical diagnostic tools as "high-risk systems," mandating transparency and human oversight. A failure to provide explanations that are both technically accurate and clinically interpretable not only limits adoption but may also result in non-compliance.

This study embeds LIME into its final model evaluation to ensure diagnostic accountability, prevent harm, and promote ethical AI use in healthcare. This approach ensures the system's decision logic is accessible and compliant with emerging medical AI regulations.

2.8 Gaps, Limitations, and Motivation for the Proposed Study

Despite advances in respiratory sound classification, significant limitations still restrict the real-world deployment of AI systems. As the literature shows, many models perform well in controlled settings but fail under unpredictable real-world conditions. These gaps are not just technical; they affect fairness, reliability, and clinical trust.

Gap 1: Sensitivity to Environmental Noise Environmental noise is a major obstacle. While CNN models achieve high accuracy on clean hospital-grade recordings, their performance degrades significantly in noisy conditions. Studies have reported drops of up to 15 F1 percentage points. The design response to this was to apply spectrogram noise augmentation during training to improve resilience. Effectiveness was assessed by comparing macro-F1 and ROC-AUC scores on the independent test set.

Gap 2: Dataset Imbalance and Demographic Fairness Many respiratory datasets are imbalanced in terms of demographics and device types. The ICBHI 2017 dataset, for instance, is biased toward adult males. Without targeted mitigation, this risks reinforcing healthcare inequalities. To address this, class weighting was applied during training for CNN-LSTM-Attention models to improve minority class performance. The effectiveness of this was evaluated by comparing per-class F1-scores and confusion matrices.

Gap 3: Lack of Transparent Interpretability tools like Grad-CAM and LIME are often underused in systematic model evaluation. The study integrated LIME into the final evaluation to identify which time-frequency regions of a spectrogram most influenced predictions. These visualizations were qualitatively assessed to ensure the model was focusing on diagnostically plausible patterns, which helps build clinical trust.

Gap 4: Weak Temporal Modeling of Respiratory Events Breath sounds are inherently temporal, but many models treat audio as a sequence of independent frames. While LSTMs help, they can dilute key transitions like the onset of a crackle. The inclusion of attention layers in the CNN-LSTM-Attention model aimed to improve this temporal focus. The effect was measured by comparing F1-scores between the CNN-LSTM and CNN-LSTM-Attention models.

Motivation for the Proposed Study This study responds to these challenges by implementing and evaluating a CNN-LSTM-Attention model with targeted noise augmentation, class weighting, and local interpretability. The evaluation framework considers standard metrics alongside per-class performance, robustness, and qualitative interpretability checks, ensuring that the model is not only effective in clean test conditions but also transparent and fair in ways relevant to clinical adoption.

2.9 Conceptual Framework: Connecting the Dots from Literature to Model Design

After exploring the progression of respiratory sound classification from handcrafted features to deep learning hybrids, this study's design directly applies elements from the literature. This section presents the conceptual framework for the proposed system, with each component grounded in prior research.

2.9.1 Core Concepts Integrated into the Framework

Acoustic Feature Extraction: The study uses MFCCs for the baseline MLP model due to their compactness and robustness, and Log-Mel Spectrograms for the CNN and CNN-LSTM-based models because of their compatibility with convolutional architectures.

Hybrid Deep Architecture (CNN-LSTM with Attention): The proposed architecture layers CNNs for spatial abstraction with LSTMs for sequential context and integrates attention mechanisms to highlight diagnostically significant temporal regions.

Targeted Data Augmentation: Bounded spectrogram noise augmentation was applied to enhance robustness and address class imbalance.

Addressing Class Imbalance: In addition to augmentation, class weighting was applied during training for selected CNN-LSTM-Attention variants to give more importance to minority classes.

Explainability by Design: The study implements LIME for the final model to provide localized explanations of which spectrogram regions most influenced each prediction, supporting clinical interpretability.

Evaluation Strategy: In line with the literature, evaluation goes beyond simple accuracy to include per-label metrics, confusion matrices, and micro/macro averaged metrics to capture overall performance.

2.9.2 Framework Overview

The framework begins with ICBHI 2017 respiratory cycle data, which undergoes preprocessing and feature extraction. MFCC features are used for the baseline MLP, while log-mel spectrograms feed the CNN, CNN-LSTM, and CNN-LSTM-Attention models. Training incorporates augmentation and class weighting. Hyperparameter tuning is conducted on the CNN-LSTM-Attention architecture. The final model variants are evaluated on a held-out test set, and explainability is incorporated using LIME.

This framework ensures the entire process is methodologically rigorous and directly aligns with the performance, robustness, and interpretability goals identified in the literature.

Chapter 3 – Methodology

3.0 Introduction

In designing a system that classifies respiratory sounds with both accuracy and clinical trust, it is not enough to simply apply deep learning for its own sake. The choices made for this project, from dataset curation to model architecture, are grounded in the limitations and opportunities found in the literature. This chapter outlines the methodological framework adopted for the study, showing how each component of the pipeline builds on past work while responding to real-world constraints like noise, bias, and the need for interpretability.

The structure of this chapter mirrors the structure of the problem. It begins by discussing the overall research design and the rationale behind it. From there, it moves into the selection and justification of datasets, followed by a detailed account of the preprocessing steps. The chapter then introduces the model development pipeline, starting with a baseline MLP, progressing to CNNs, and finally, expanding into the proposed attention-based CNN-LSTM hybrid. Each model is trained and evaluated under a consistent experimental setup to allow for fair comparison.

Finally, the chapter outlines the performance metrics used, discusses ethical considerations such as fairness and data privacy, and reflects on methodological alternatives that were considered but not pursued. Together, these components form a comprehensive foundation for implementing, evaluating, and interpreting the system proposed in this study.

3.1 Research Design

This study uses an experimental research design to develop, train, and evaluate deep learning models for respiratory sound classification. This design is ideal for determining how independent variables like architectural features, datasets, and augmentation affect outcomes such as accuracy, interpretability, and robustness. By systematically modifying the model architecture and input data while monitoring performance, the design enables a rigorous evaluation of each methodological choice.

The project employs supervised learning with labeled datasets to train models that classify sounds into categories like crackles, wheezes, or normal breath sounds. The methodology also integrates post-training explainability modules, following a design-science approach that evaluates models not only for performance, but also for trust, fairness, and clinical usability.

Given the real-world application of this study in supporting early diagnosis of respiratory conditions, the design balances technical robustness with clinical relevance. The use of the ICBHI 2017 Respiratory Sound Database strengthens this approach by providing high-quality annotated cycles to test for generalizability, a key concern in medical AI research (Knight & Reinke, 2023; Lee et al., 2024).

To provide a visual overview, Figure 3.1 illustrates the complete experimental framework. It shows the entire process, from dataset acquisition through preprocessing, exploratory analysis, and model development, to evaluation and explainability, highlighting how each stage connects to the next.

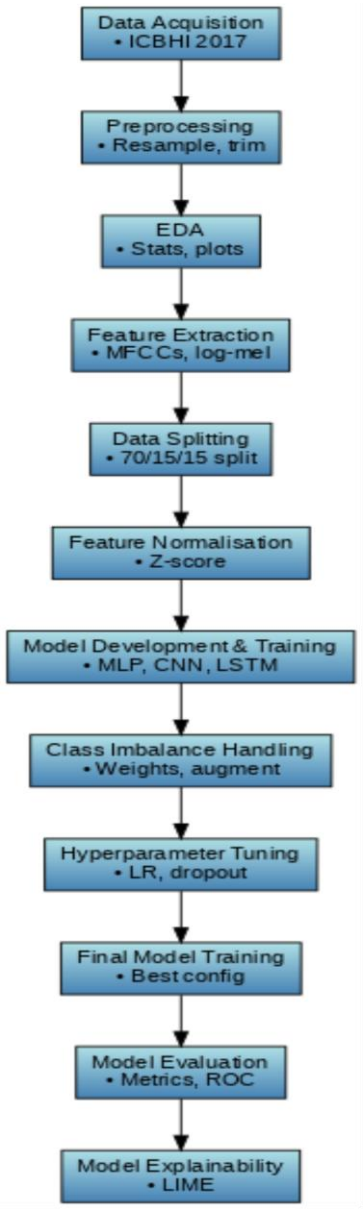


Figure 3.1: Experimental framework for multi-label respiratory sound classification used in this study.

3.1.1 Rationale for the Research Design

The experimental design was chosen for several practical reasons.

First, respiratory sound classification lends itself to structured experimentation. Machine learning models can be built, adjusted, and tested using established metrics like accuracy, F1-score, and AUC. This approach makes it possible to isolate and evaluate the impact of each stage, including feature extraction, data augmentation, and model architecture.

Second, deep learning research is typically built on cycles of design and testing. An experimental approach makes it easy to compare baseline models like MLPs with more advanced ones such as CNN-LSTM architectures with attention mechanisms. This design also supports hyperparameter tuning, ablation tests, and post-hoc explainability, which all depend on controlled, repeatable experiments.

Third, this design allows the study to test for generalization, which is especially important in medical AI. While some models perform well in controlled settings, their real challenge is handling new data. The intention to evaluate robustness across different data sources still shaped many design decisions, even though the external Coswara dataset was removed due to time constraints.

Finally, this approach is consistent with previous studies in the same field. For example, Perna and Tagarelli (2019) used iterative experimentation to evaluate deep learning architectures. Similarly, Bahloul et al. (2023) combined standardized evaluation with architectural comparisons to assess performance in noisy environments. These reasons make the experimental design a strong fit for this study, allowing for both technical depth and practical relevance.

3.1.2 Consideration of Alternative Research Designs

Although the experimental design was the most suitable for this project, other approaches were considered, including observational and qualitative designs.

An observational design, where models are integrated into real clinical settings, could offer valuable insights into usability and workflow. However, this is more appropriate for evaluating deployed systems. Since this project focuses on developing and testing models, it requires more control over variables than an observational study would allow.

Qualitative methods like clinician interviews or usability testing were also considered. These could have added depth to our understanding of how model interpretability aligns with clinical needs, but they would require additional ethical approvals and a separate data collection phase. This could not be accommodated within the project's scope and timeline.

Overall, these alternative designs offer valuable perspectives for future work. For this study, the experimental design provided the right structure to systematically test, compare, and refine deep learning models, enabling clear measurement of how different techniques impacted performance and supporting alignment with clinical expectations and regulatory principles (Food and Drug Administration [FDA], 2021; European Union, 2025).

3.2 Experimental Setup and Environment

All experiments were conducted using Google Colaboratory (Colab Pro), a cloud-based Jupyter notebook platform. It was chosen for its accessibility, speed, and GPU acceleration, which eliminated the need for expensive local hardware. The platform supported the full development lifecycle, from training to testing.

I used a Lenovo ThinkPad Yoga 370, leveraging Colab Pro's cloud backend and GPU resources for faster runtimes. GPU acceleration was confirmed via `nvidia-smi`. For data management, Google Drive integration was used to access the dataset directly, which kept the file structure intact and eliminated the need for re-uploading files across sessions.

3.2.1 Colab Runtime Configuration

Table 3.1: Colab Runtime Configuration

Resource Type	Specification
Runtime	Python 3 (Google Compute Engine Backend)
GPU	NVIDIA Tesla T4
GPU VRAM	15 GB
System RAM	12.7 GB
Disk Space	235.7 GB
Session Type	Colab Pro

3.2.2 Software Stack and Libraries Used

All experiments were coded in Python 3.11.13. The table below shows the main libraries and what they were used for:

Table 3.2: Libraries Used

Library / Module	Purpose in the Study
random	Generating reproducible random splits and shuffling operations
numpy	Numerical computations, array manipulation, and feature handling
tensorflow	Core deep learning framework for model building, training, and evaluation
librosa, librosa.display	Audio loading, feature extraction (MFCCs, log-mel spectrograms), and audio visualisation
pandas	Handling tabular data (annotations, metadata, feature sets)

StratifiedShuffleSplit	Splitting dataset by class group to prevent data leakage
sklearn.model_selection.train_test_split	Creating training, validation, and test splits
sklearn.metrics (f1_score, classification_report, confusion_matrix, roc_auc_score, multilabel_confusion_matrix)	Evaluating model performance across multiple metrics
sklearn.preprocessing.StandardScaler	Normalising features to zero mean and unit variance
sklearn.utils.class_weight	Computing class weights for imbalanced datasets
tensorflow.keras.layers	Defining neural network layers (Dense, Dropout, Conv2D, LSTM, etc.)
tensorflow.keras.models	Building models using Sequential and Functional APIs
tensorflow.keras.optimizers	Configuring model optimisers (e.g., Adam)
tensorflow.keras.callbacks	Early stopping, learning rate adjustments, and training monitoring
tensorflow.keras.utils.to_categorical	Encoding labels for multi-class classification
tensorflow.keras.utils.Sequence	Custom data generators for batch loading
google.colab.drive	Mounting Google Drive for data access and storage
zipfile	Extracting compressed dataset archives
collections.Counter	Counting label frequencies for dataset analysis
matplotlib.pyplot	Creating figures, histograms, and performance plots
seaborn	Enhanced data visualisation (heatmaps, distribution plots)
joblib	Saving and loading preprocessing objects (e.g., scalers)
lime.lime_image	Post-hoc model explainability using LIME
skimage.segmentation.mark_boundaries	Visualising superpixel boundaries in LIME explanations

All libraries were installed directly in Colab using pip, with version compatibility checked at every stage. This environment supported the full project lifecycle, from preprocessing to model training and evaluation. To ensure reproducibility, Python, NumPy, and TensorFlow random seeds were fixed, and package versions were logged for reference.

3.3 ICBHI 2017 Respiratory Sound Database

The ICBHI 2017 Respiratory Sound Database is one of the most widely used benchmark datasets for automated respiratory sound classification. Released as part of the International Conference on Biomedical and Health Informatics (ICBHI) 2017 Challenge, the dataset was created to support research on the detection of respiratory pathologies from auscultation recordings. It is publicly available from the

official challenge website (https://bhichallenge.med.auth.gr) and was downloaded as a single compressed archive file (ICBHI_final_database.zip).

3.3.1 Clinical Relevance and Use in This Study

The ICBHI dataset reflects a controlled clinical recording environment, which makes it especially suitable for training models that require clean, labelled input. Because of the high-quality segmentation and expert annotations, it forms the primary training and internal evaluation dataset in this study. Specifically, it is used to:

- Train deep learning models including MLP, CNN, CNN-LSTM, and CNN-LSTM-Attention
- Perform validation and internal testing using stratified splits
- Benchmark classification performance using metrics like accuracy, F1-score, and confusion matrices

While the dataset is a strong foundation for training and testing, it does have some limitations. These include limited background noise, standardised device use, and possible demographic imbalance such as fewer samples from women or children. These factors may limit real-world generalisation, and future studies could address this by including additional datasets that capture more diverse conditions.

3.4 Data Preprocessing

Before training any deep learning model, it was important to prepare the raw respiratory sound recordings into a format the models could actually learn from. This meant cleaning, segmenting, and converting the audio into useful features. The preprocessing phase focused entirely on the ICBHI 2017 dataset, which was selected for its detailed annotations and clean recording environment.

The process began with exploring the structure of the dataset and checking that all the recordings had matching annotation files. From there, individual respiratory cycles were extracted from the longer .wav recordings based on time-stamped labels. These cycles were then standardised by trimming or padding their lengths and resampling the audio to a consistent frequency. Finally, the segmented clips were converted into features like MFCCs and spectrograms for model input.

The following subsections walk through this pipeline step by step, starting with how the ICBHI dataset was inspected and verified before segmentation and feature extraction were carried out.

3.5 Dataset Exploration – ICBHI 2017

3.5.1 Overview of File Structure

The ICBHI 2017 Respiratory Sound Database was downloaded and extracted into a single folder containing 1,843 files: 920 .wav audio recordings from 126 subjects, 922 .txt annotation files, and one

non-audio file. Each .wav file contains one or more breathing cycles recorded from various chest locations. The annotations are valuable because they were labeled by clinical experts, categorizing each cycle as normal, crackles, wheezes, or both.

3.5.2 File Matching and Integrity Check

To ensure accurate segmentation, every `.wav` file was matched with its corresponding annotation file. This confirmed that all 920 audio files had a matching `.txt` file. The two extra `.txt` files were identified as documentation and excluded. This process was a critical step for an error-free segmentation. The format of the plain-text annotation files was reviewed, with each line representing a respiratory cycle defined by start and end times and binary labels for crackles and wheezes. The time intervals were consistent with the 20-second audio files, confirming that the annotations were well-aligned and could be used for segmentation. The process of parsing the files was performed using Python libraries like pandas and librosa, which were essential for the rest of the feature extraction and model training pipeline.

3.5.3 Audio Recording Duration Analysis

Before segmenting the audio, an initial analysis of the 920 .wav file durations was conducted. This step helped to understand the length and structure of the raw recordings.

The majority of the files, specifically 817 of 920 (about 89%), were exactly 20.00 seconds long. The remaining recordings varied significantly, ranging from 7.86 to 86.2 seconds, with an average duration of 21.49 seconds.

This analysis confirmed that while the dataset largely follows a fixed 20-second format, it also includes longer files with multiple breathing cycles. This finding was important because most deep learning models require consistent input shapes. Therefore, these longer recordings needed to be segmented or padded to ensure uniformity for model training. As shown in *Figure 3.2,* a histogram of audio durations clearly illustrates this trend, with a sharp spike at the 20-second mark followed by a long tail.

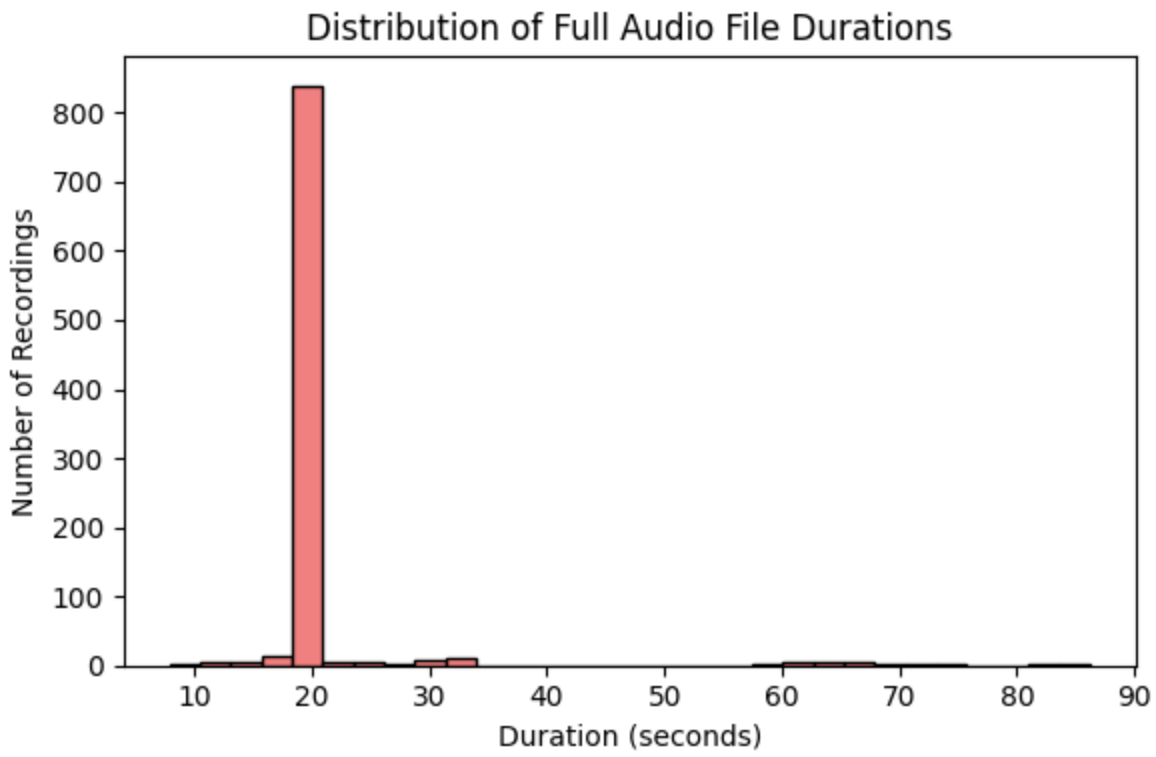


Figure 3.2: Distribution of Full audio file durations

3.5.4 Class Imbalance in the ICBHI Dataset

An analysis of the class distribution in the segmented respiratory cycles revealed a significant imbalance between the four target categories: normal, crackle, wheeze, and both crackle and wheeze. As shown in Figure 3.3, the normal and crackle classes make up the majority of the dataset, while wheeze and both crackle and wheeze are considerably underrepresented. This uneven distribution poses a challenge for training deep learning models, as it can lead to bias towards the majority classes and reduced sensitivity for minority classes.

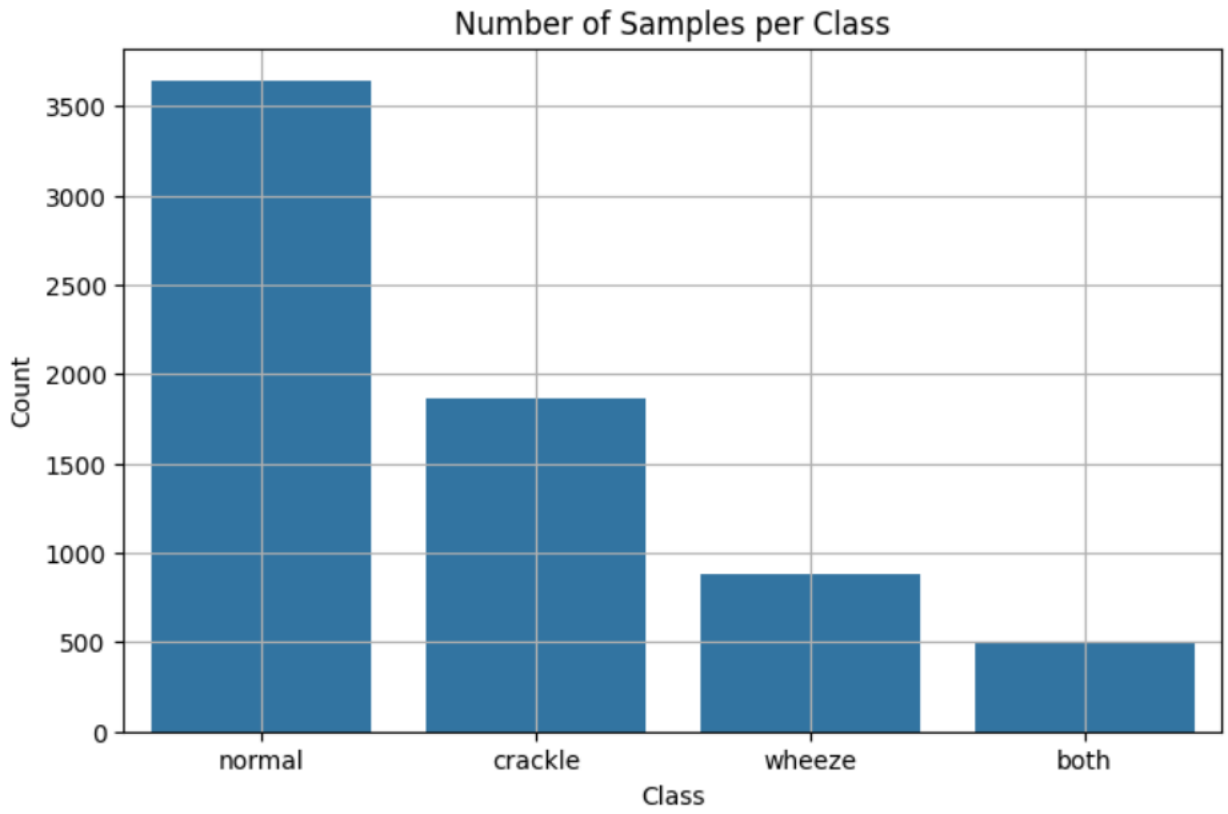


Figure 3.3: Class Imbalance

To address this imbalance, the model training process incorporated class weighting and data augmentation strategies. These methods ensure that minority classes receive greater emphasis during optimisation, improving the model's ability to detect wheezes and combined abnormal sounds despite their limited representation in the dataset.

3.5.5 Visual and Acoustic Exploration of the Dataset

To better understand the acoustic characteristics of the ICBHI respiratory sound dataset, some visual exploration was done using waveform plots and log-mel spectrograms. I looked at individual respiratory cycles from each of the four target classes: normal, crackles, wheezes, and both (crackles and wheezes). These were visualised to help spot any noticeable differences in sound patterns. The waveforms and spectrograms (see Figure 3.4,3.5,3.6,3.7) showed clear distinctions. Normal cycles looked smooth and covered a wide range of frequencies. Crackles appeared as short bursts with high intensity. Wheezes had long, narrow bands of energy, and samples that had both showed a mix of sudden spikes and sustained tones.

This step helped confirm that spectrogram-based CNN models were a good fit. These models are designed to pick up on both frequency and timing patterns, which are crucial for detecting the kinds of differences seen between healthy and abnormal breath sounds.

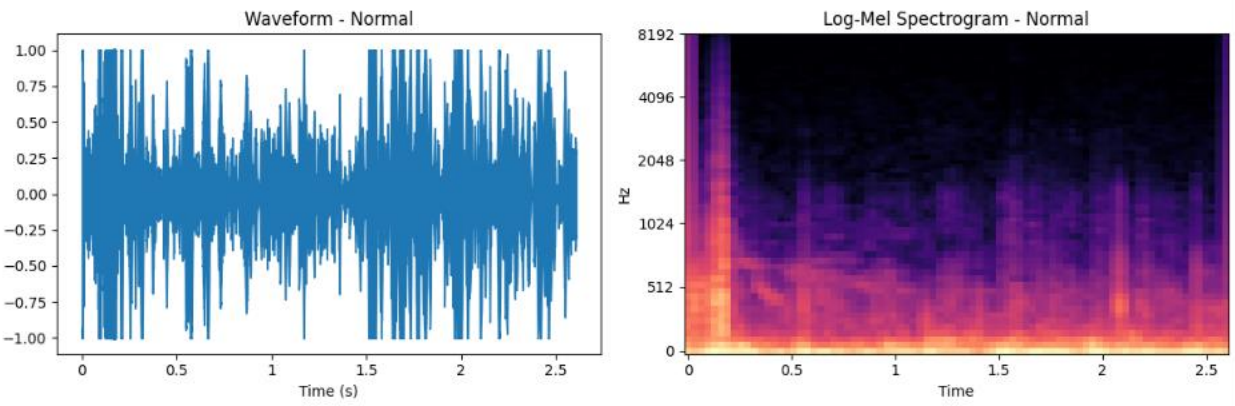


Figure 3.4: Normal

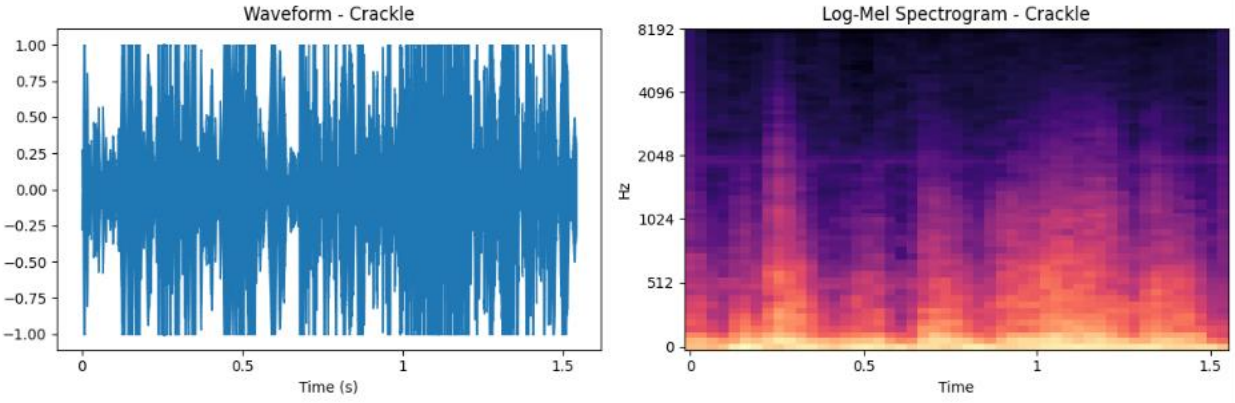


Figure 3.5: Crackle

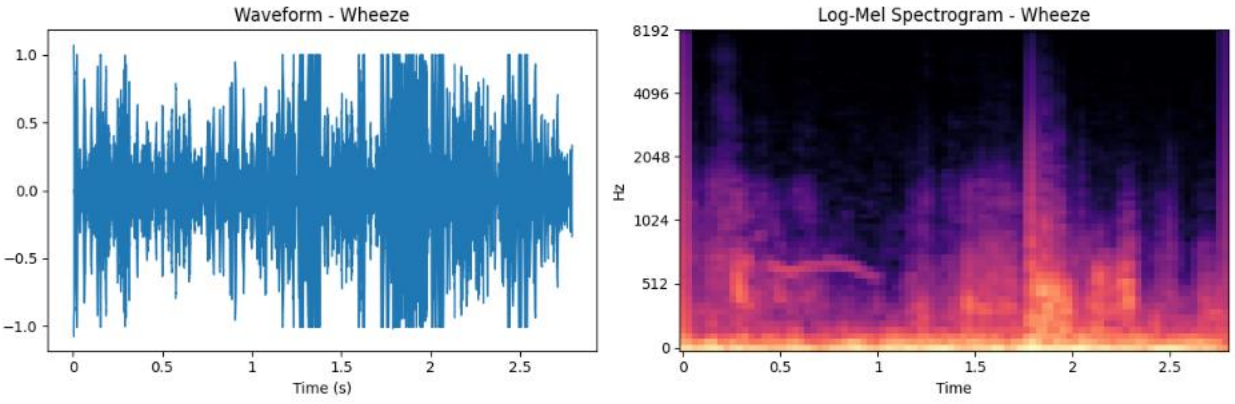


Figure 3.6: wheeze

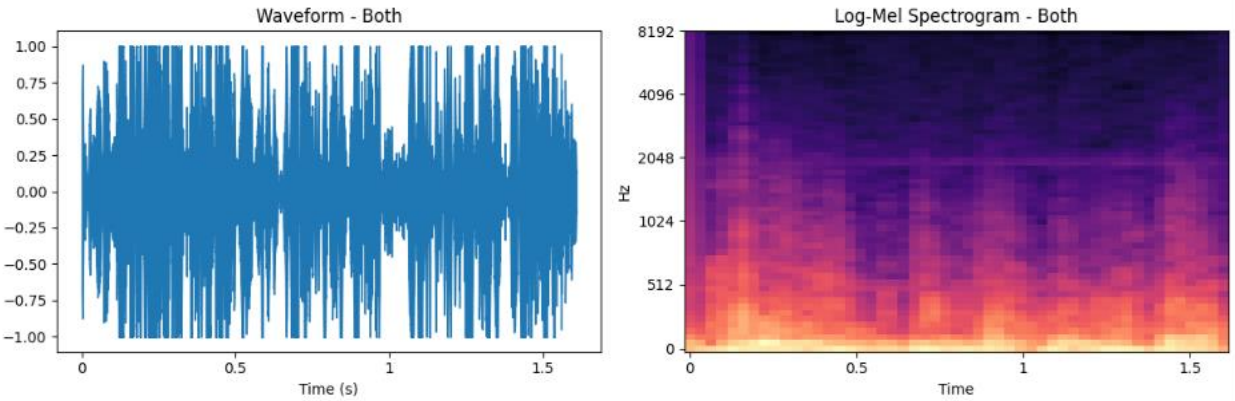


Figure 3.7: Both

3.5.6 Spectrogram Variability Across Classes

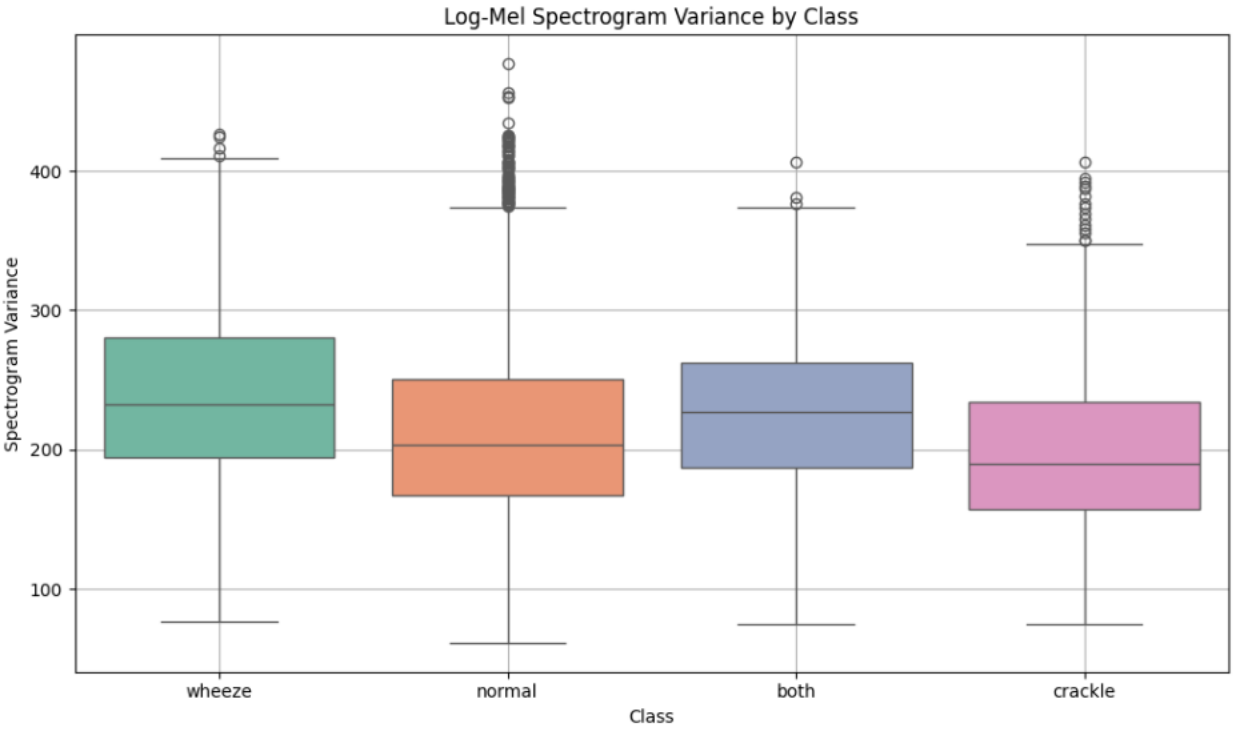


Figure 3.8: spectogram variance by class

Beyond sample count differences, the classes also have distinct acoustic characteristics. The variance of log-mel spectrograms was calculated and visualized to show this. As *Figure 3.8* illustrates, normal cycles show lower variance and more uniform energy patterns. In contrast, pathological classes, especially those with both crackles and wheezes, have higher variance, reflecting their more complex and irregular acoustic profiles. These findings suggest that abnormal sounds are inherently more variable in their spectral structure, making them more difficult to detect without advanced feature extraction methods like the CNN-LSTM architectures with attention mechanisms used in this study.

3.5.7 Visual Analysis of Disease-Specific Spectrograms

To gain a clearer understanding of the acoustic characteristics linked to specific respiratory conditions, a visual inspection was carried out using one representative respiratory cycle per disease category. These included asthma, COPD, bronchiectasis, pneumonia, bronchiolitis, and upper respiratory infections (URI).

For each selected cycle, both the waveform and the log-mel spectrogram were plotted. The spectrograms (Figures 3.9–3.16) reveal distinct patterns across conditions. For example, asthma cycles often display narrow-band, continuous high-frequency components associated with wheezing, while pneumonia and bronchiectasis exhibit short, sharp, high-energy bursts corresponding to crackles. COPD recordings tend to have a more mixed profile, with lower-frequency wheezes and occasional transient crackles. URI and bronchiolitis spectrograms show less pronounced but still irregular spectral activity, reflecting milder or more diffuse abnormalities.

These differences in time–frequency structure highlight why spectrogram-based CNN and CNN-LSTM architectures are suitable for this classification task. The figures also informed later model design decisions, such as selecting input dimensions and tuning convolutional kernel sizes to capture both fine-grained transient events and sustained tonal features.

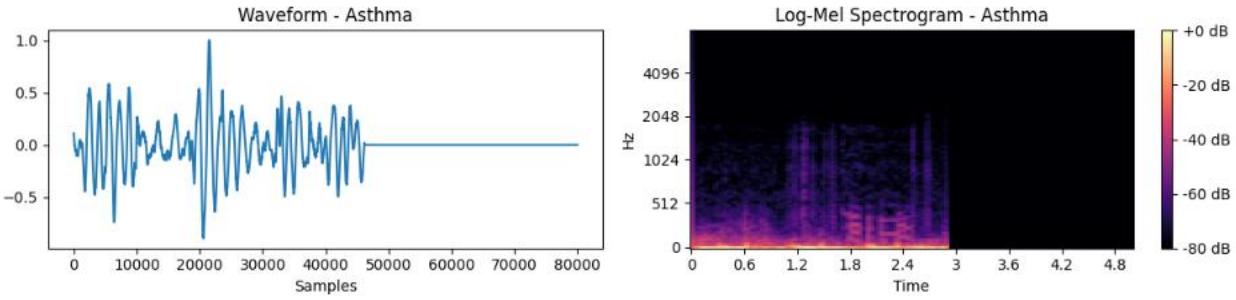


Figure 3.9: Asthma

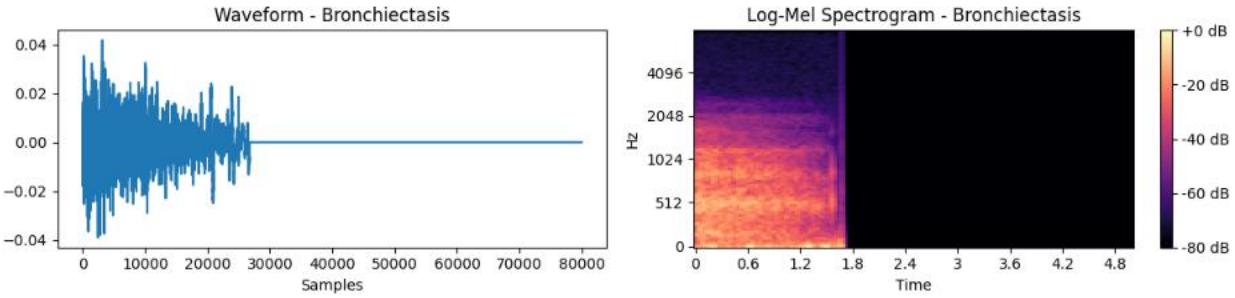


Figure 3.10: Bronchiectasis

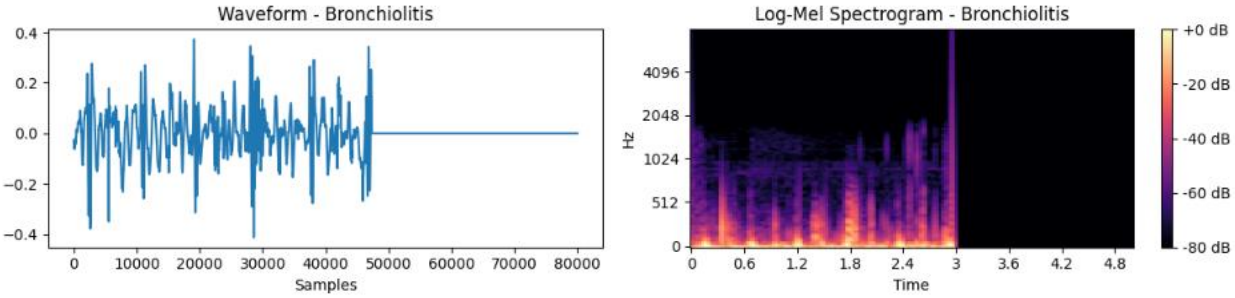


Figure 3.11: Bronchiolitis

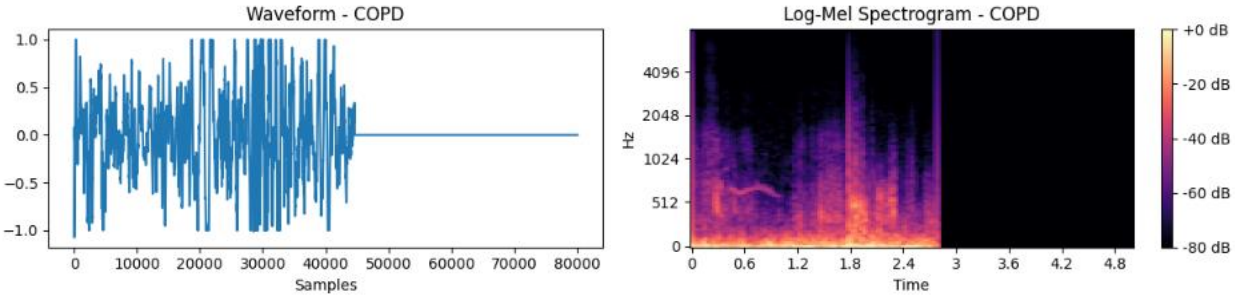


Figure 3.12: COPD

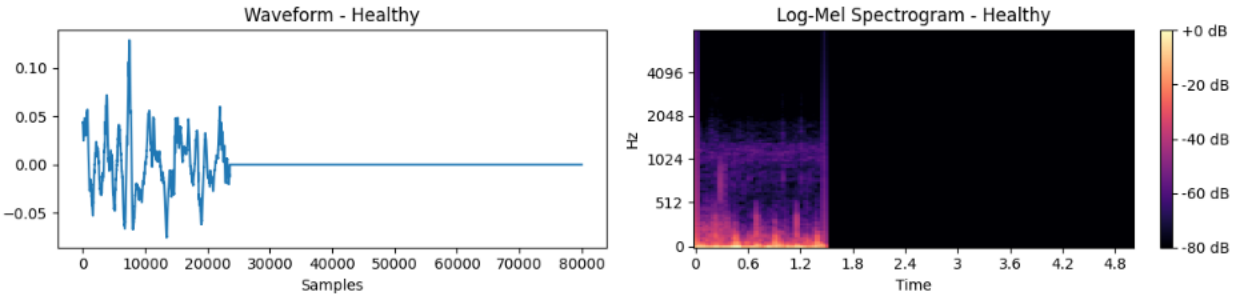


Figure 3.13: Healthy

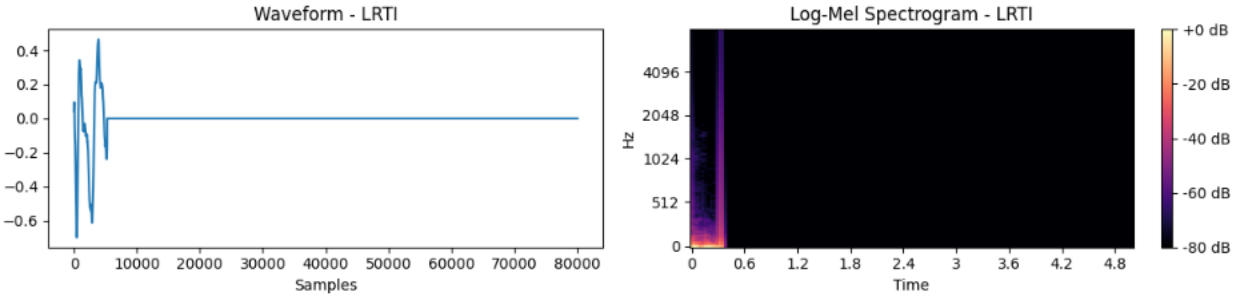


Figure 3.14: LRTI

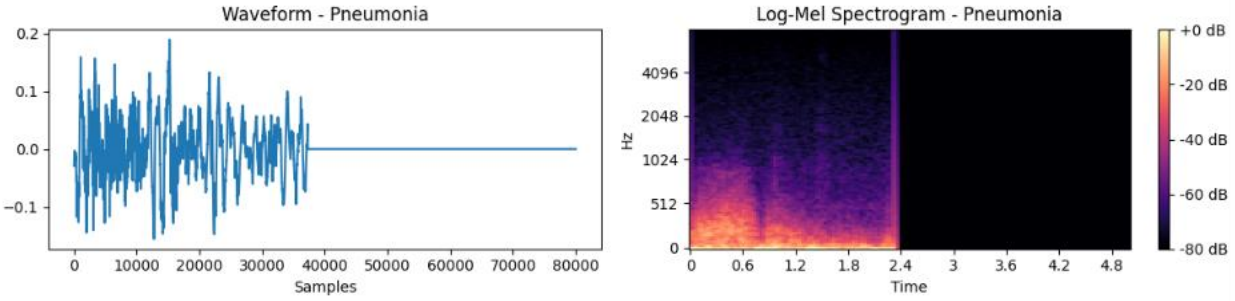


Figure 3.15: Pneumonia

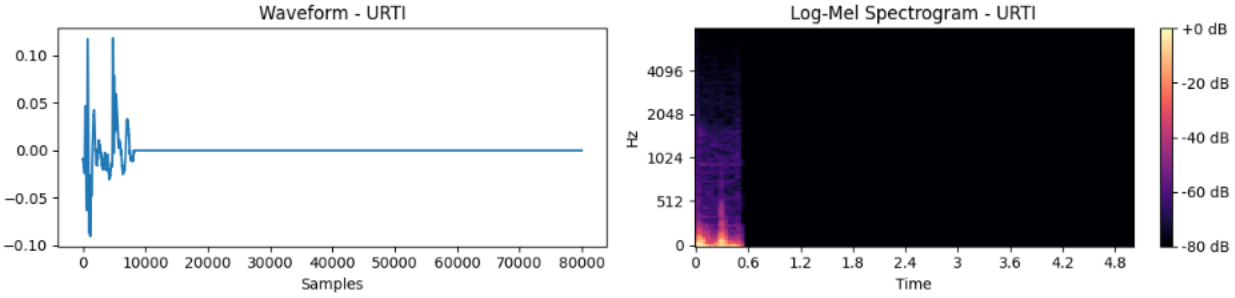


Figure 3.16: URTI

3.5.8 Respiratory Cycle Segmentation

With the file structure and annotation alignment confirmed, the next step was to segment each audio recording into individual respiratory cycles. This was critical for isolating clinically meaningful events like crackles and wheezes and preparing the data for supervised learning.

Each .wav file was paired with its .txt annotation file, which provided the start and end times for each cycle and binary labels for crackles and wheezes. The original audio recordings were then programmatically segmented and resampled to 16,000 Hz for consistency and efficient feature computation.

In total, 6,898 respiratory cycle segments were extracted from 920 audio files. Each segment was stored with its metadata, including filename, timestamps, duration, and diagnostic labels. This structured segmentation enabled the model to learn from precise acoustic events rather than from entire, heterogeneous recordings.

The segmentation process used the librosa library for audio loading and resampling, and pandas for parsing the annotation files. These tools provided a robust foundation for automated and reproducible segmentation.

3.5.9 Exploratory Data Analysis of Segmented Respiratory Cycles

Following the segmentation of the ICBHI audio, an exploratory data analysis was conducted to understand the dataset's structure and label distribution. A total of 6,898 respiratory cycles were extracted, each labeled by the presence of abnormal lung sounds (crackles and wheezes).

The distribution of labels showed a notable class imbalance:

- Normal: 3,642 segments (52.8%)
- Crackles only: 1,864 segments (27.0%)
- Wheezes only: 886 segments (12.8%)
- Both: 506 segments (7.3%)

This imbalance is typical in clinical datasets, where normal sounds are more common than pathological events. However, it can bias classifiers toward the majority class. This finding highlights the need for targeted data augmentation to oversample underrepresented classes and improve the model’s sensitivity to rarer pathologies.

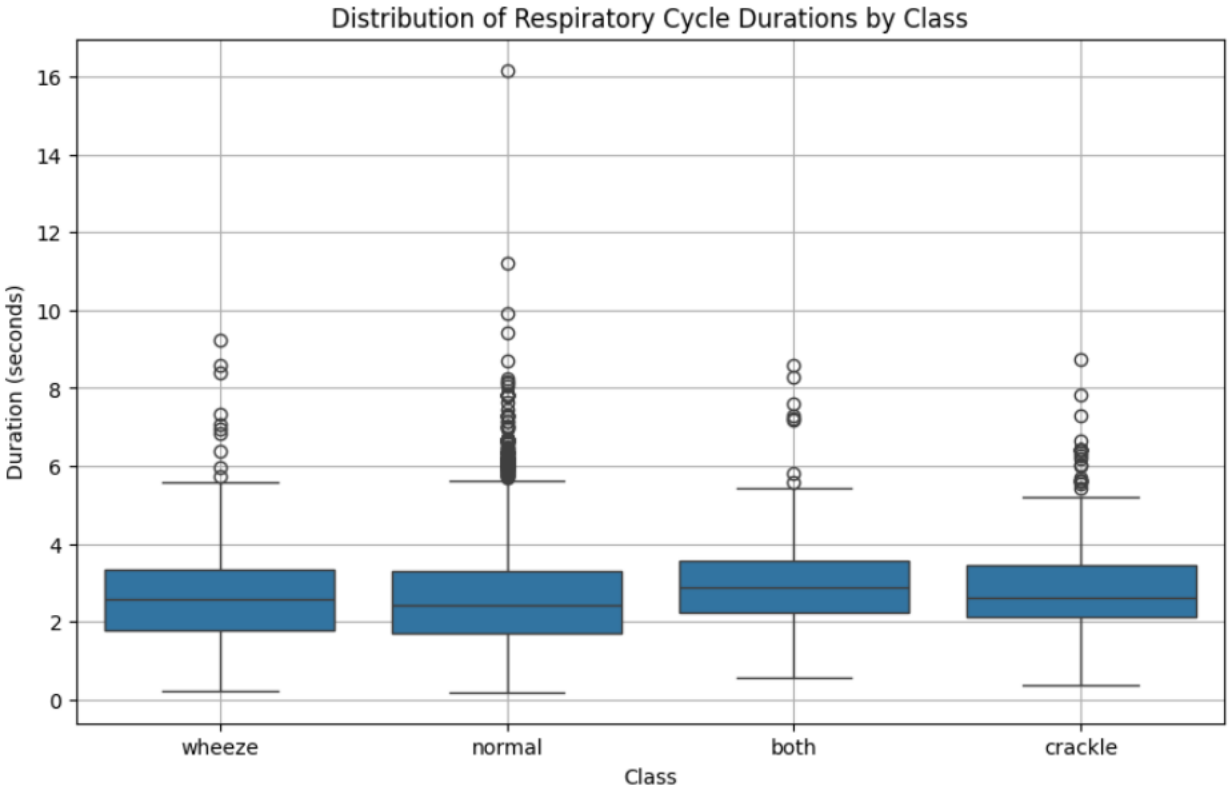


Figure 3.17: Distribution of respiratory cycle durations by class

Segment duration also varied significantly. The average was 2.7 seconds, with a range from 0.2 to 16.16 seconds. The interquartile range was between 1.93 and 3.37 seconds. This variability highlights the need for input standardization, where each cycle will be trimmed or padded to a fixed duration (e.g., 5 seconds).

These insights establish a strong foundation for designing the preprocessing pipeline. By identifying imbalances and inconsistencies early on, the study can proactively address challenges related to generalizability, fairness, and model robustness.

3.5.10 Segment Standardisation for Model Input

To prepare the respiratory cycle data for deep learning, all audio clips were standardized to a fixed length of five seconds. This was important because the original segments varied significantly in duration. Models like CNNs and LSTMs work best with uniform input shapes, which allows for efficient batch training and memory handling.

The five-second target was chosen based on an earlier analysis of segment durations. Segments longer than five seconds were trimmed, while shorter ones were padded with silence. Since all audio was

resampled to 16,000 Hz, each standardized file contained exactly 80,000 samples, ensuring consistency and reproducibility for feature extraction and training.

3.5.11 MFCC Feature Extraction for Baseline Classification

To establish a baseline for respiratory sound classification, Mel-Frequency Cepstral Coefficients (MFCCs) were extracted from each standardised 5-second respiratory cycle segment. MFCCs are a widely used representation in audio signal processing, valued for their ability to capture the perceptually relevant frequency characteristics of human speech and breath sounds.

Each segment was already resampled to a uniform rate of 16,000 Hz and standardised to a fixed duration of 5.0 seconds, resulting in exactly 80,000 samples per segment. Using the $librosa.feature.mfcc$ MFCC coefficients were computed for each segment. To produce a fixed-length vector suitable for fully connected models, the mean of each MFCC coefficient was calculated along the time axis:

$$MFCCmean = T1t = 1\sum^{T} MFCC(t)$$

This process resulted in a 2D NumPy array of shape (6,898 × 13), where each row corresponds to one respiratory cycle and each column represents the average value of a specific MFCC coefficient. The corresponding multi-label targets indicating the presence or absence of crackles and wheezes were stored separately and later combined with the MFCC feature array for model training and evaluation.

The computation of MFCCs can be mathematically described in three main steps:

1. Frame the signal and compute the Short-Time Fourier Transform (STFT):

$$X(k) = n = 0\sum^{N} -1x[n] \cdot w[n] \cdot e - jN2\pi kn$$

where $x[n]$ the input signal, $w[n]$ the window function, and N the frame length.

2. Apply the Mel filter bank and take the logarithm of the energies:

$$Em = \log\big(k = 0\sum^{N} -1 \mid X(k) \mid 2 \cdot Hm(k) \big)$$

where $Hm(k)$ is the $m - th$ triangular filter in the Mel scale.

3. Apply the Discrete Cosine Transform (DCT) to obtain MFCCs:

$$MFCCn = m = 1 \sum^{M} Em \cdot \cos[M\pi n(m - 21)]$$

where M the total number of Mel filters and n the coefficient index.

This compact representation allowed for efficient training of the baseline multilayer perceptron (MLP) model while preserving key spectral information relevant to respiratory sound classification.

3.5.12 Log-Mel Spectrogram Feature Extraction for CNN Models

To prepare features for convolutional neural networks (CNNs), log-Mel spectrograms were extracted from each 5-second respiratory cycle segment. Mel spectrograms provide a 2D time-frequency representation that captures spectral content and temporal evolution, making them well-suited for CNNs.

Each standardized segment was at 16,000 Hz, with 80,000 samples. The librosa.feature.melspectrogram function was used to compute 128 Mel frequency bands. The resulting spectrograms were converted to a logarithmic scale. The final array was shaped

(6,898×128 ×157×1), with an added channel dimension to make it compatible with CNN architectures.

The computation of a log-Mel spectrogram involves three main steps:

1. Short-Time Fourier Transform (STFT):

$$S(f,t) = n = 0 \sum^{N} -1x[n] \cdot w[n - tH] \cdot e - jN2\pi fn$$

where $x[n]$ is the input signal, $w[\cdot]$ is the analysis window, H is the hop length, f is frequency, and t is the frame index.

2. Convert frequency to the Mel scale using triangular filters:

$$Mm(t) = f = 0 \sum^{F} -1 \mid S(f,t) \mid 2 \cdot Hm(f)$$

where $Hm(f)$ is the $m - th$ Mel filter and F is the number of frequency bins.

3. Convert to log scale to produce the log-Mel spectrogram:

$$LogMelm(t) = \log(Mm(t) + \epsilon)$$

Where ϵ is a small constant to prevent taking the logarithm of zero.

This representation preserves detailed spectral–temporal patterns that are important for distinguishing between normal and pathological respiratory sounds, making it particularly effective for deep learning models that learn spatial hierarchies from image-like data.

3.5.13 MFCC Splitting and Normalization

To prepare the MFCC features, the dataset was split into training (70%), validation (15%), and test (15%) subsets using a stratified approach to maintain class proportions.

After splitting, the features were normalized using z-score scaling. A StandardScaler was fitted on the training set to compute the mean and standard deviation, which were then used to transform all three subsets. This process ensures that the 13 MFCC features have zero mean and unit variance, reducing bias and improving model convergence. This step was critical for providing consistent and well-scaled inputs to the baseline MLP model, ensuring stable and comparable results.

3.5.14 Spectrogram Splitting and Normalization

For the CNN and CNN-LSTM models, features were derived from Mel spectrograms. Each spectrogram had a shape of (128 × 157) for every 5-second audio segment.

The dataset of 6,898 spectrograms was also divided into training (70%), validation (15%), and test (15%) subsets using a stratified split to maintain proportional representation of all four target classes.

Spectrogram values were first converted to the logarithmic decibel scale to better represent human auditory perception. They were then normalized to a range between 0 and 1 using min-max scaling, which preserves relative intensity patterns. Finally, a singleton channel dimension was added to the end of each array to match the input format for convolutional layers, resulting in a final shape of (128, 157, 1) per sample. This step ensured compatibility with Keras CNN layers and optimized GPU memory handling during batch processing.

Table 3.3: Comparison of MFCC and Mel Spectrogram Preprocessing Pipelines

Step	MFCC Pipeline	Mel Spectrogram Pipeline
Input data	5-second standardised audio segments (16,000 Hz, 80,000 samples)	5-second standardised audio segments (16,000 Hz, 80,000 samples)
Feature extraction method	librosa.feature.mfcc (13 coefficients)	librosa.feature.melspectrogram (128 Mel bands)

Post-processing	Mean pooling across time axis → 13-dimensional vector per segment	Convert to log-decibel scale → 128×157 time-frequency matrix
Shape before split	(6898, 13)	(6898, 128, 157)
Data splitting	Stratified: 70% train, 15% validation, 15% test	Stratified: 70% train, 15% validation, 15% test
Normalisation method	Z-score scaling (StandardScaler, fitted on training set)	Min-max scaling (0–1 range per spectrogram)
Final input shape for model	(6898, 13)	(6898, 128, 157, 1)
Target model type	Baseline MLP	CNN / CNN-LSTM / CNN-LSTM-Attention

3.6 Model Architectures

3.6.1 Baseline Multilayer Perceptron (MLP) Model (MFCC Features)

Table 3.4: Summary of Baseline MLP Model Configuration

Component	Description
Input	13 MFCC coefficients per segment (mean-pooled across time)
Architecture	Fully connected feedforward neural network
Hidden Layers	Dense(64, ReLU), Dense(32, ReLU)
Regularisation	Dropout (rate = 0.3), Dropout (rate = 0.2) between dense layers
Output Layer	Dense(2, Sigmoid) for multi-label classification
Loss Function	Binary cross-entropy
Optimiser	Adam (learning rate = 0.001 - default)
Batch Size	32
Epochs	Up to 50 with early stopping (patience = 5)
Evaluation Metrics	Accuracy, Classification Report (F1-score), ROC-AUC
Purpose	Establish a simple performance baseline for comparison

3.6.2 Convolutional Neural Network (CNN) Model (Log-Mel Spectrogram Features)

Table 3.5: Summary of CNN Model Configuration

Component	Description
Input	Log-Mel spectrograms (128 Mel bands × 157 time frames × 1 channel)
Architecture	2D Convolutional Neural Network
Convolutional Layers	Conv2D(32 filters, 3×3 kernel, ReLU), Conv2D(64 filters, 3×3 kernel, ReLU)

Pooling Layers	MaxPooling2D(2×2) after each convolutional layer
Regularisation	Dropout (rate = 0.3) after pooling layers
Flatten Layer	Converts 2D feature maps to 1D vector
Dense Layers	Dense(64, ReLU) → Dropout(0.4)
Output Layer	Dense(2, Sigmoid) for multi-label classification
Loss Function	Binary cross-entropy
Optimiser	Adam (learning rate = 0.001 - default)
Batch Size	32
Epochs	Up to 30 with early stopping (patience = 5)
Evaluation Metrics	Accuracy, Classification Report (F1-score), ROC-AUC

3.6.3 CNN-LSTM Hybrid Model

Table 3.6 :Summary of CNN-LSTM Hybrid Model Configuration

Component	Description
Input	Log-Mel spectrograms (128 × 157 × 1)
Architecture	CNN followed by LSTM
Convolutional Layers	Conv2D(32, 3×3, ReLU) → MaxPooling2D(2×2) → Dropout(0.3), Conv2D(64, 3×3, ReLU) → MaxPooling2D(2×2) → Dropout(0.3)
Reshape Layer	Reshape to (time frames, features) for LSTM input
LSTM Layer	Bidirectional LSTM (64 units), return_sequences=False
Dense Layers	Dense(32, ReLU)
Output Layer	Dense(2, Sigmoid) for multi-label classification
Loss Function	Binary cross-entropy
Optimiser	Adam (learning rate = 0.001 - default)
Batch Size	32
Epochs	Up to 30 with early stopping (patience = 5)
Evaluation Metrics	Accuracy, Classification Report (F1-score), ROC-AUC

3.6.4 CNN-LSTM-Attention Hybrid Model

3.6.4.1 Variant 1: CNN-LSTM-Attention (Weighted)

This configuration applied class weights calculated from the inverse frequency of class occurrences during training, using the initial hyperparameters defined for the base CNN-LSTM-Attention model.

Table 3.7: Summary of CNN-LSTM-Attention (Weighted) Model Configuration

Component	Description

Input	Log-Mel spectrograms (128 × 157 × 1)
Architecture	CNN followed by Bidirectional LSTM with Attention
Convolutional Layers	Conv2D(32, 3×3, ReLU) → MaxPooling2D(2×2) → Dropout(0.3), Conv2D(64, 3×3, ReLU) → MaxPooling2D(2×2) → Dropout(0.3)
Reshape Layer	Reshape to (32 time frames, 2496 features) for LSTM input
LSTM Layer	Bidirectional LSTM (64 units per direction, return_sequences=True)
Attention Layer	Custom attention mechanism applied to LSTM outputs
Dense Layers	Dense(64, ReLU) → Dropout(0.4)
Output Layer	Dense(2, Sigmoid) for multi-label classification
Loss Function	Binary cross-entropy
Optimiser	Adam (learning rate = 0.001 - default)
Batch Size	32
Epochs	Up to 30 with early stopping (patience = 5)
Class Imbalance Handling	Class weights applied during training
Evaluation Metrics	Accuracy, Classification Report (F1-score), ROC-AUC

3.6.4.2 Variant 2: Final Tuned & Weighted CNN-LSTM-Attention Model

This variant used the best hyperparameters found during a random search tuning process and applied class weights during training. The model was trained on a combined training and validation dataset.

Table 3.8 : Summary of Final Tuned & Weighted CNN-LSTM-Attention Model Configuration

Component	Description
Input	Log-Mel spectrograms (128 × 157 × 1)
Architecture	CNN followed by Bidirectional LSTM with Attention
Convolutional Layers	Conv2D(32, 3×3, ReLU) → MaxPooling2D(2×2) → Dropout(0.3), Conv2D(64, 3×3, ReLU) → MaxPooling2D(2×2) → Dropout(0.3)
Reshape Layer	Reshape to (32 time frames, 2496 features) for LSTM input
LSTM Layer	Bidirectional LSTM (128 units per direction, return_sequences=True)
Attention Layer	Custom attention mechanism applied to LSTM outputs
Dense Layers	Dense(32, ReLU) → Dropout(0.4)
Output Layer	Dense(2, Sigmoid) for multi-label classification
Loss Function	Binary cross-entropy
Optimiser	Adam (learning rate = 0.0001)
Batch Size	64
Epochs	Up to 100 with early stopping (patience = 5)

Class Imbalance Handling	Class weights applied during training
Evaluation Metrics	Accuracy, Classification Report (F1-score), ROC-AUC

3.6.4.3 Variant 3: Augmented Tuned & Weighted CNN-LSTM-Attention Model

This variant is the same as Variant 2 but includes data augmentation (spectrogram noise) applied during training using a data generator.

Table 3.9: Summary of Augmented Tuned & Weighted CNN-LSTM-Attention Model Configuration

Component	Description
Input	Log-Mel spectrograms (128 × 157 × 1)
Architecture	CNN followed by Bidirectional LSTM with Attention
Convolutional Layers	Conv2D(32, 3×3, ReLU) → MaxPooling2D(2×2) → Dropout(0.3), Conv2D(64, 3×3, ReLU) → MaxPooling2D(2×2) → Dropout(0.3)
Reshape Layer	Reshape to (32 time frames, 2496 features) for LSTM input
LSTM Layer	Bidirectional LSTM (128 units per direction, return_sequences=True)
Attention Layer	Custom attention mechanism applied to LSTM outputs
Dense Layers	Dense(32, ReLU) → Dropout(0.4)
Output Layer	Dense(2, Sigmoid) for multi-label classification
Loss Function	Binary cross-entropy
Optimiser	Adam (learning rate = 0.0001)
Batch Size	64
Epochs	Up to 100 with early stopping (patience = 5)
Class Imbalance Handling	Class weights applied during training
Data Augmentation	Spectrogram noise applied during training
Evaluation Metrics	Accuracy, Classification Report (F1-score), ROC-AUC

3.7 Data Imbalance Handling and Augmentation

After the initial MLP baseline experiments, it was clear the dataset had a class imbalance, especially in the "Wheezes only" and "Both" categories. This impacted performance, as the MLP showed high precision but low recall for wheezes, indicating a bias toward majority classes.

To fix this and improve minority class detection, two strategies were used: Class Weighting and Data Augmentation. For class weighting, weights were calculated based on the inverse frequency of each class in the training data. This assigned a higher penalty for misclassifying underrepresented samples,

encouraging the model to pay more attention to them. Class weights were applied to the loss function during the training of the weighted and augmented variants of the CNN-LSTM-Attention model.

3.7.1 Data Augmentation Implementation

Data augmentation was also explored as a technique to potentially improve the model's generalisability and performance. While augmentation techniques like random noise injection, time stretching, and pitch shifting were considered and functions for them were defined, the augmentation implemented within the data generator for training the augmented model variant involved adding random low-amplitude noise directly to the Mel spectrograms in the training set batches. This was applied randomly to samples in each batch, not specifically targeted to minority classes within the generator's logic due to the complexity of applying audio-specific augmentations to pre-computed spectrograms. The validation and test sets remained untouched to avoid data leakage.

This approach aimed to introduce variability into the training data representation, complementing the class weighting strategy for handling imbalance.

3.8 Hyperparameter Tuning

To optimize the CNN-LSTM-Attention model, a 20-trial hyperparameter search was conducted. The goal was to find the best combination of parameters, including learning rate, number of LSTM units, dropout levels, dense layer size, and batch size, for the multi-label classification task. Validation micro-averaged F1-score was the primary metric, as it provided a better measure of performance across the imbalanced classes than accuracy alone.

Learning Rate: Lower rates, particularly 0.0001, consistently performed better. LSTM Units & Dropout: Models with 128 LSTM units generally yielded the best F1-scores. Dropout rates between 0.2 and 0.4 proved stable, while higher values harmed performance by removing critical temporal patterns. Dense Layer Size: Smaller dense layers (e.g., 32 units) often worked better with larger LSTM layers, suggesting the dense layer's primary role was classification after the LSTM had encoded features. Batch Size: A batch size of 64 balanced stable gradients with enough randomness to avoid getting stuck in local minima.

Validation Trends: While validation accuracy ranged from 0.55 to 0.72, the micro F1-scores were much lower (0.17 to 0.46), highlighting that accuracy is an insufficient metric for this imbalanced classification task.

3.9 Training Time Tracking

Training time was monitored to understand the computational demands of each model configuration, which is important for real-world deployment. Table 3.10 summarizes the number of epochs, average time per epoch, and total estimated training time for each model. A more detailed discussion of computational efficiency is provided in Chapter 5.

Table 3.10: Training Time Tracking

Model	Epochs Trained	Approx. Avg Time per Epoch	Estimated Total Training Time
Baseline MLP Model (MFCC)	26	< 1 second	< 30 seconds
CNN Model (Spectrogram)	17	3–5 seconds	1–1.5 minutes
CNN-LSTM Hybrid Model	22	4–6 seconds	1.5–2 minutes
Weighted CNN-LSTM Attention Model (Variant 1)	11	9–11 seconds	1.5–2 minutes
Final Tuned & Weighted CNN-LSTM Attention (V2)	27	5–10 seconds	2.5–4.5 minutes
Augmented Tuned & Weighted CNN-LSTM Attention (V3)	34	5–10 seconds	3–5.5 minutes

It is important to note that the hyperparameter tuning process required significantly more time (about 30 minutes) than training a single model, as it involved training multiple model configurations across 20 trials. The total tuning duration was the cumulative time of these individual training runs, managed by the Early Stopping callback.

3.10 Model Explainability Using LIME

To understand the final CNN-LSTM-Attention model's predictions, LIME (Local Interpretable Model-Agnostic Explanations) was applied. The goal was to identify which parts of a spectrogram a model focused on when classifying respiratory sounds.

Since LIME is designed for 3-channel RGB images, the grayscale spectrograms were normalized and stacked into a (128, 157, 3) format. A custom prediction wrapper was used to convert these back to grayscale internally before feeding them into the trained model.

LIME perturbs the input by masking small segments (superpixels) and measures the effect on the model's output. A LimeImageExplainer generated explanations for two classes: normal cycles and abnormal cases (crackles, wheezes, or both). For each explanation, 1,000 perturbed samples were generated. The explainer then highlighted the top 10 superpixels that contributed most to the model's prediction. This step was used to verify if the model's attention aligned with expected time-frequency patterns.

Note on Grad-CAM

An attempt was made to apply Grad-CAM, but it failed because the gradient-based backpropagation required by the method did not propagate reliably through the model's LSTM and attention layers. This is a known limitation of Grad-CAM. LIME was used instead due to its model-agnostic nature and compatibility with hybrid architectures.

3.11 Evaluation Metrics and Reporting

This study reports classification performance using a consistent set of well-established metrics. Since the task is multi-label with two target labels (crackles and wheezes), all metrics were computed separately for

each label and then summarised using micro- and macro-averaging where appropriate. Unless stated otherwise, predicted probabilities were converted into binary outputs using a fixed threshold of 0.50.

3.11.1 Confusion Matrix (Per Label)

For each label, predictions can be summarised in a 2 × 2 confusion matrix:

Confusion Matrix (Per Label)

	Predicted Positive	Predicted Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

Where:

- TP: True Positives
- FP: False Positives
- TN: True Negatives
- FN: False Negatives

This matrix forms the basis for calculating accuracy, precision, recall, and F1-score.

3.11.2 Accuracy

Two definitions of accuracy are reported:

1. Binary accuracy (per sample, multi-label): the proportion of individual label predictions that are correct, averaged across the dataset.
2. Subset (exact match) accuracy: the proportion of samples for which both labels are predicted correctly.

For a single label, accuracy is calculated as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

3.11.3 Precision, Recall, and F1-score

For the positive class of each label:

$$\text{Pr}\,e\,cision\ =\ \frac{TP}{TP+FP}$$

$$Recall\ =\ \frac{TP}{TP+FN}$$

$$F1 = \frac{2 \times \text{Pr}\,e\,cision \times Recall}{\text{Pr}\,e\,cision + Recall}$$

Averaging schemes:

1. Per-label: computed separately for crackles and wheezes.
2. Micro-average: aggregates TP, FP, and FN across both labels before computing the metric. This method is sensitive to class imbalance and was used for model selection during hyperparameter tuning.
3. Macro-average: arithmetic mean of per-label scores, giving equal weight to each class regardless of frequency.

3.11.4 Per-Label Accuracy

For each label, per-label accuracy measures the proportion of correct predictions for that specific class:

$$Per-label\ Accuracy = \frac{TP\ +TN}{TP+FP+TN+FN}$$

3.11.5 ROC and AUC

For each label, the Receiver Operating Characteristic (ROC) curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR) as the decision threshold is varied from 1 to 0:

$$TPR = \frac{TP}{TP+FN}$$

$$FPR = \frac{FP}{FP+TN}$$

The Area Under the ROC Curve (AUC) quantifies the model's ability to discriminate between positive and negative cases across all possible thresholds. Higher AUC values indicate better discrimination.

This study reports:

1. AUC for each label (crackles AUC, wheezes AUC)
2. Micro-average AUC across both labels

Chapter 4: Results & Discussion

4.1 Baseline MLP Model (MFCC Features)

The baseline MLP model, trained on MFCC features from 6,898 respiratory cycles, demonstrated stable generalization with a consistent Binary accuracy of 0.7778–0.7899. Its lower subset accuracy (0.60–0.62) reflects the difficulty of multi-label classification on an imbalanced dataset. While crackle performance was balanced with an F1-score of ~0.61, wheeze performance was poor, showing high precision but very low recall (0.10–0.17) and an F1-score of 0.17–0.28. A small train-test accuracy gap (~0.005) suggests no overfitting, indicating that the low wheeze recall is due to class imbalance. The model's stability is evident in its smooth, parallel loss curves (Figure 4.1) and its accuracy trends (Figure 4.2), where validation performance slightly led training accuracy. ROC-AUC analysis showed significant class separability, with scores for crackles and wheezes being 0.800 and 0.783 on the training set, and 0.769 and 0.767 on the test set, respectively (Figures 4.3, 4.4).

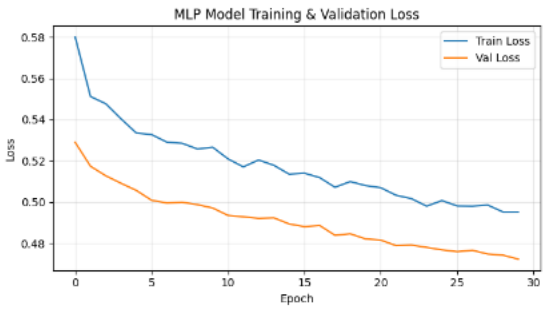


Figure 4.1

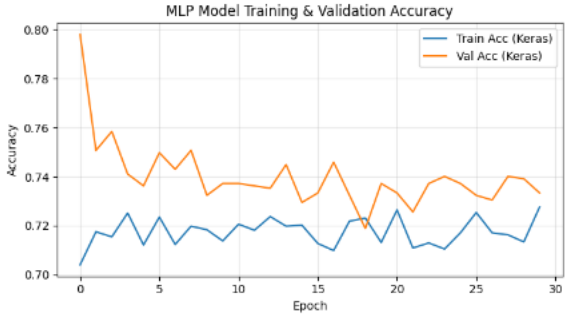


Figure 4.2

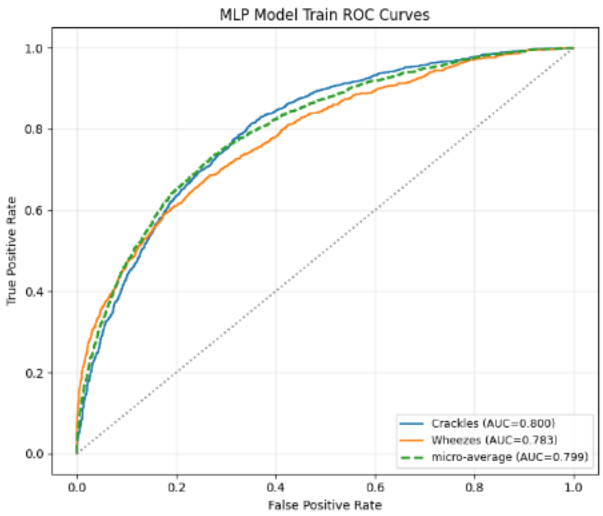


Figure 4.3

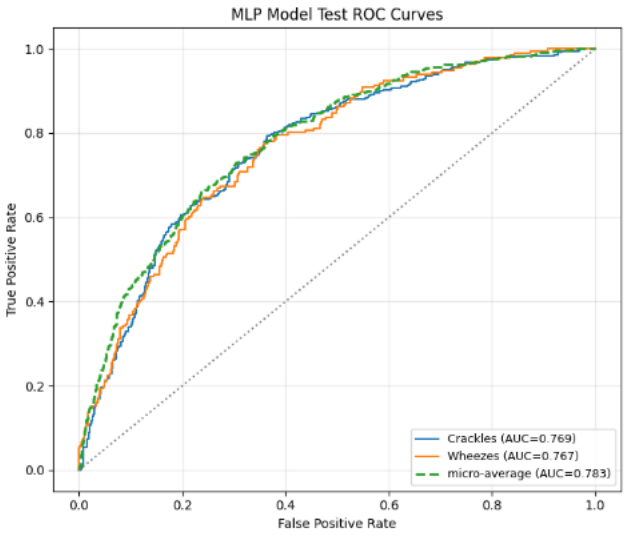


Figure 4.4

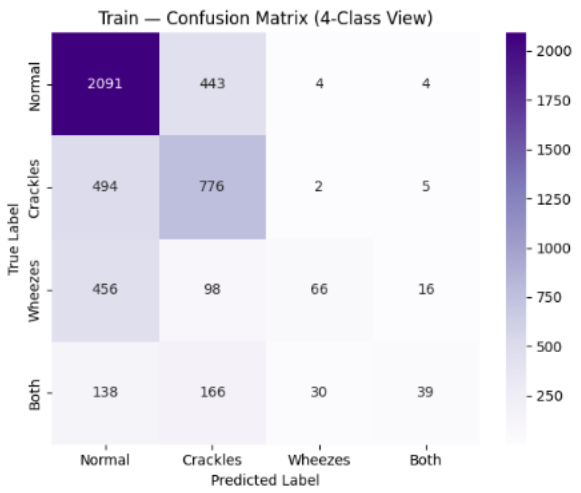


Figure 4.5

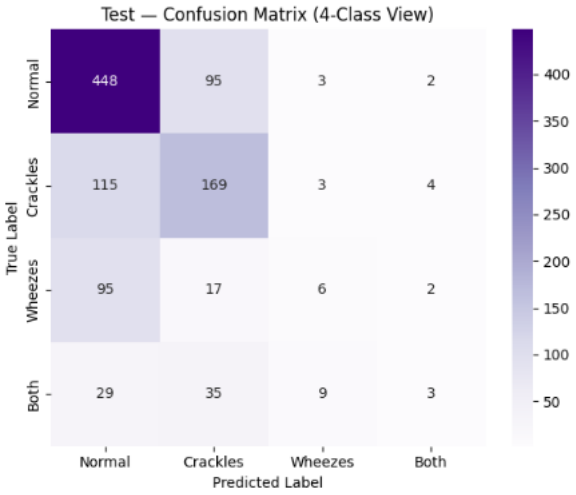


Figure 4.6

The confusion matrices (Figures 4.5, 4.6) show that most normal cycles were correctly classified. However, a high proportion of wheezes were misclassified as normal or crackles, explaining the low recall for this class.

Overall, the MLP baseline shows moderate capability for detecting crackles but is limited in identifying wheezes, particularly due to recall deficiencies. This reinforces the need for more sophisticated architectures that can capture subtle temporal-spectral patterns,

such as CNNs and LSTMs with attention, alongside targeted data augmentation to address class imbalance.

4.2 CNN Model (Mel Spectrogram Features)

The convolutional neural network (CNN), trained on log-mel spectrograms of the 6,898 respiratory cycles, demonstrated improved performance over the MLP baseline. It achieved a binary accuracy of 0.7870 and a subset accuracy of 0.6222. The model's per-label performance showed a significant boost in wheeze detection; while crackle recall dipped to 0.49, the wheeze F1-score climbed to 0.53, a notable increase from the baseline's 0.17. This improvement highlights the CNN's ability to capture the harmonic structures of wheezes. Training and validation loss curves (Figure 4.7) showed a smooth downward trend, with a slight widening gap that suggests minor overfitting. Accuracy trends (Figure 4.8) indicated training accuracy started at 0.78 and ended near 0.72, while validation accuracy fluctuated between 0.66 and 0.68.

Figure 4.7

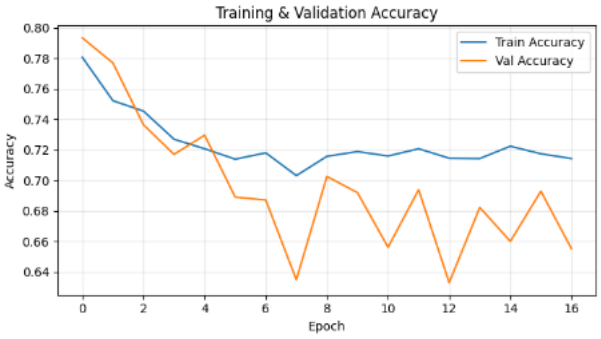


Figure 4.8

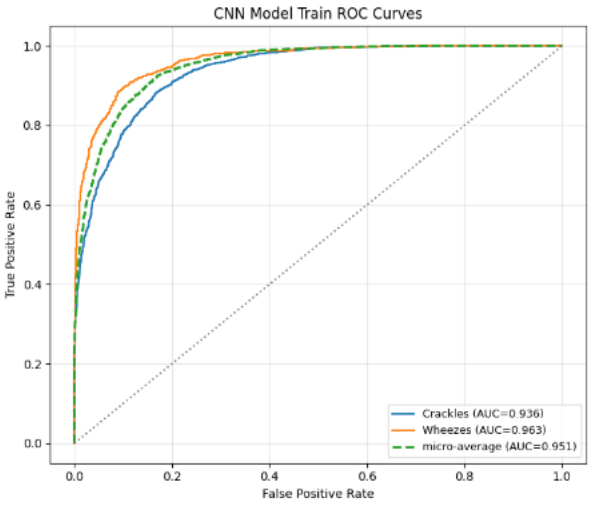


Figure 4.9

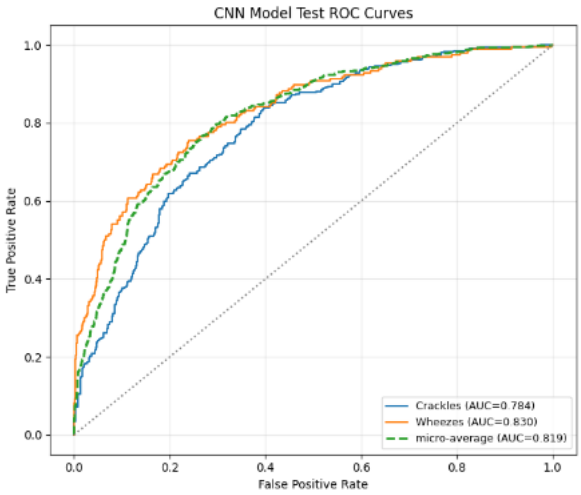


Figure 4.10

ROC-AUC scores also supported the improvements. On the training set (Figure 4.9), crackles and wheezes reached 0.936 and 0.963, respectively. Test set scores (Figure 4.10) dropped to 0.784 for crackles and 0.830 for wheezes. This drop is expected, but the wheeze class held up particularly well, suggesting better generalization than the MLP managed.

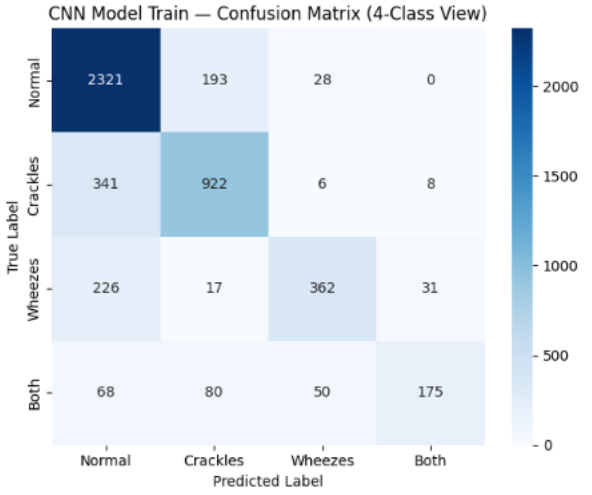


Figure 4.11

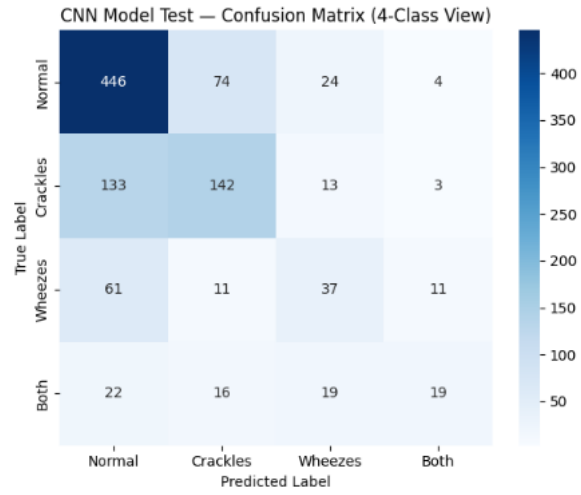


Figure 4.12

The confusion matrices (Figures 4.11, 4.12) show the CNN significantly reduced false negatives for wheezes, recovering more true cases while maintaining solid performance on normal cycles. Crackle detection, though better than the baseline in terms of subset accuracy, still saw many samples misclassified as normal. This suggests the model needs help capturing temporal structure.

Overall, the CNN improved over the MLP by better balancing wheeze precision and recall. While using spectrograms helps the model focus on relevant spatial features, the mixed performance on crackles suggests that combining this spatial learning with a sequential model, like a CNN-LSTM setup, could improve results by capturing temporal information.

4.3 CNN-LSTM Model (Mel Spectrogram Features)

The CNN-LSTM model combined convolutional layers for spatial feature extraction with LSTM layers to capture temporal patterns in the spectrograms. While training performance was solid (binary accuracy 0.7605; subset accuracy 0.5853), a significant drop occurred on the test set, with binary accuracy falling to 0.7386 and subset accuracy to 0.5469. This wider gap compared to the CNN alone suggests weaker generalization. On the test set, crackle recall (0.36; F1-score 0.43) was lower than the CNN's 0.49. Wheeze detection failed almost completely, with a recall of just 0.02 and an F1-score of 0.03, a stark drop from the CNN's 0.44. The addition of temporal modeling did not help the model capture more crackles and severely hurt wheeze detection.

The training and validation loss curves (Figure 4.13) show that training loss decreased steadily, but validation loss levelled off and then started to rise. This suggests the model began to overfit. Accuracy trends (Figure 4.14) show training accuracy stabilising around 0.71, while validation accuracy dropped from around 0.79 early on to about 0.64 by the end.

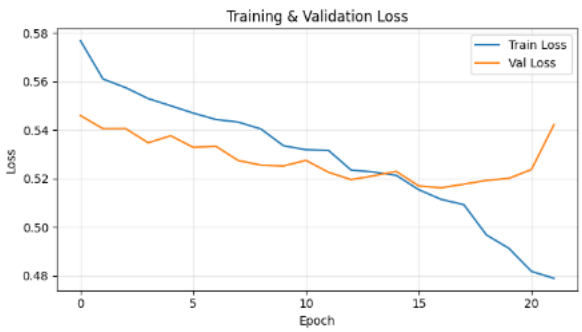


Figure 4.13

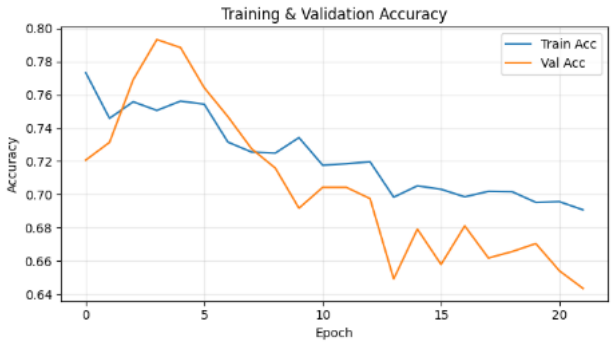


Figure 4.14

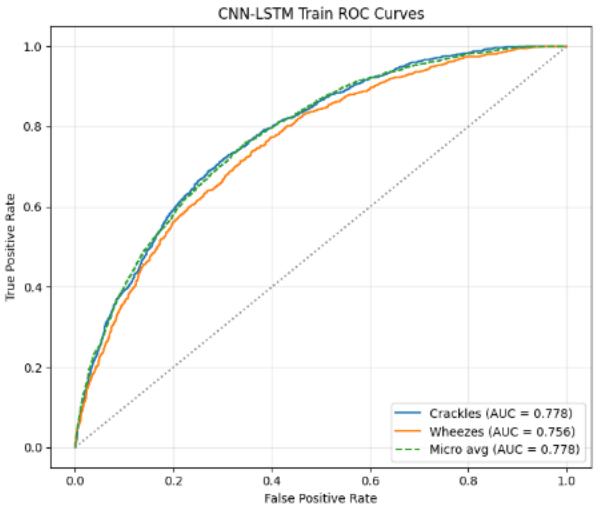


Figure 4.15

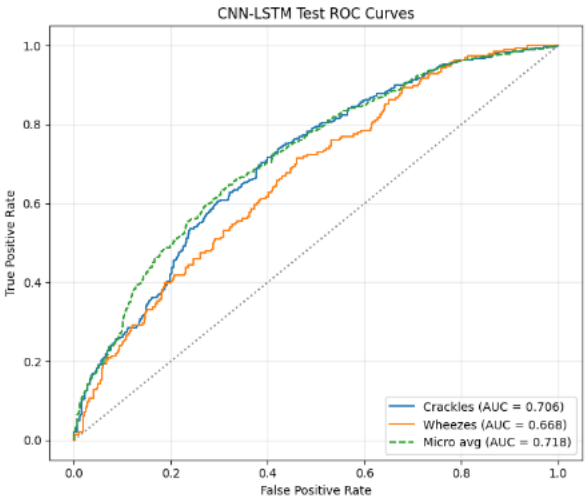


Figure 4.16

ROC–AUC scores also point to weaker generalisation. On the training set (Figure 4.15), crackles reached an AUC of 0.778 and wheezes 0.756. On the test set (Figure 4.16), these dropped to 0.706 and 0.668. That's a bigger drop than we saw in the CNN model, which means the CNN-LSTM was not as effective at separating the classes on unseen data.

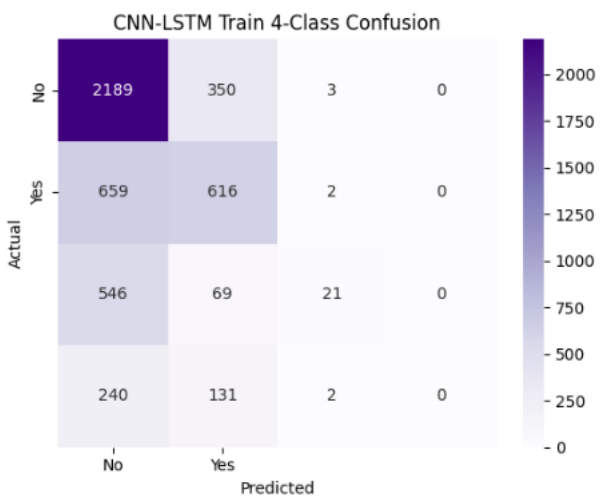


Figure 4.17

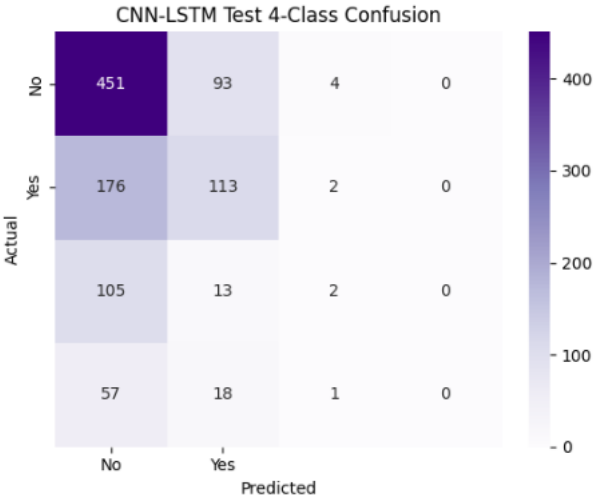


Figure 4.18

The confusion matrices (Figures 4.17, 4.18) show that many crackles were misclassified as normal, and most wheezes were not detected. Nearly all wheeze cases were mislabeled as normal, representing a significant regression from the CNN model's performance.

Overall, while the CNN-LSTM kept up with crackle detection, wheeze recall dropped significantly. This suggests that simply adding LSTMs is insufficient. Given that wheezes are short, subtle events easily missed in noise, an attention mechanism could help the model focus on key segments and recover performance.

4.4 CNN-LSTM-Attention Model (Mel Spectrogram Features)

The CNN-LSTM-Attention model adds an attention mechanism to help the network focus on the most relevant time-frequency regions, particularly for detecting short events like wheezes. Training performance was strong (binary accuracy 0.7594; subset accuracy 0.5835), with a smaller performance drop on the test set (binary accuracy 0.7464; subset accuracy 0.5604) compared to the CNN-LSTM, suggesting improved stability. On the test set, crackle detection did not improve; its recall dropped to 0.28 (F1-score 0.38). However, wheeze performance saw a slight lift, with precision reaching 0.50 and recall rising from 0.02 to 0.06, showing that attention helped recover a few more wheeze cases. Loss curves (Figure 4.19) suggest mild overfitting, but accuracy curves (Figure 4.20) show a narrower gap between training (~0.75) and validation (0.66–0.73) accuracy than the CNN-LSTM.

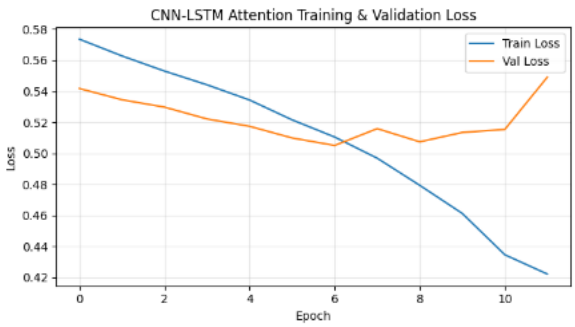


Figure 4.19

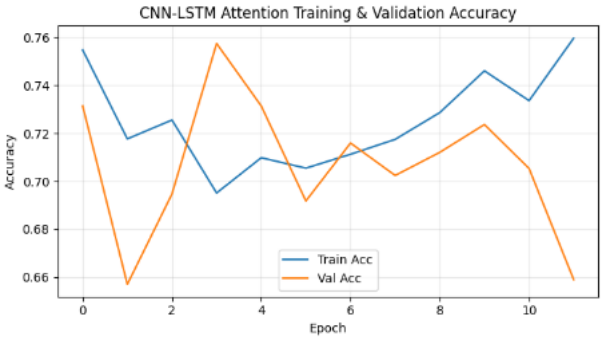


Figure 4.20

ROC-AUC scores showed that on the training set (Figure 4.21), crackles had an AUC of 0.768 and wheezes 0.801. On the test set (Figure 4.22), these dropped to 0.706 for crackles and 0.701 for wheezes. This drop was not as steep as with the plain CNN-LSTM, especially for wheezes, suggesting the attention layer improved the model's ability to generalize to new data.

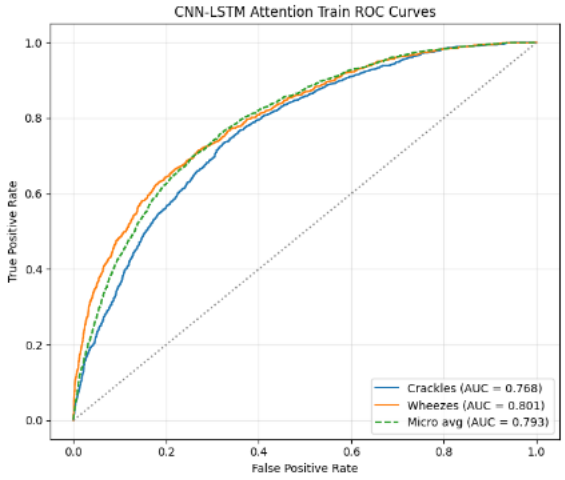


Figure 4.21

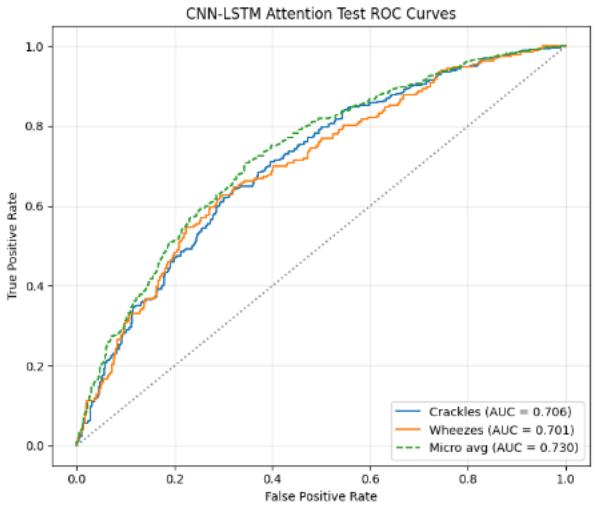


Figure 4.22

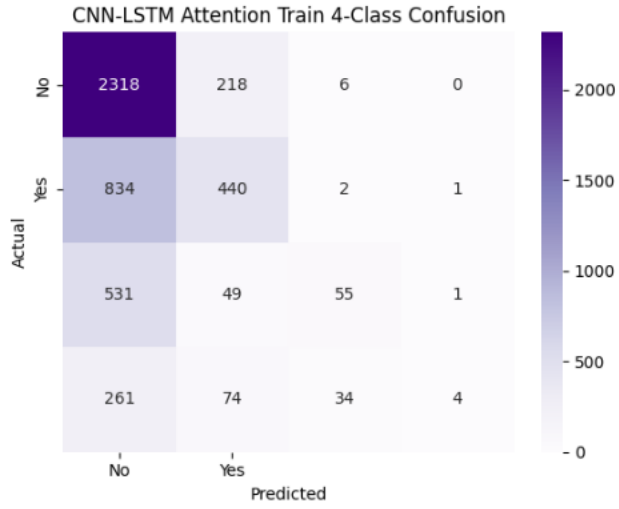


Figure 4.23

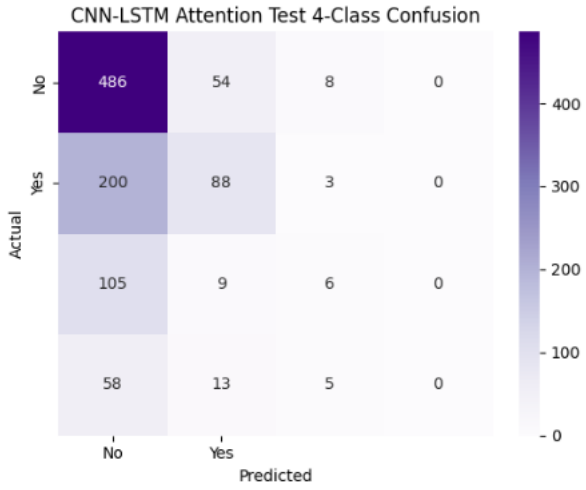


Figure 4.24

The confusion matrices (Figures 4.23, 4.24) show that most missed crackle and wheeze detections were labeled as normal. The number of correctly detected wheezes was slightly higher than in the CNN-LSTM, but crackle misclassifications remained common.

Overall, the CNN-LSTM-Attention model reduced the generalization gap, particularly for wheezes, but still fell short of the CNN in terms of recall. Crackle performance also remained limited. While the attention mechanism helped the model focus more effectively on the right parts of the input, it did not fully solve the challenges of temporal modeling or class imbalance.

4.5 Weighted CNN-LSTM-Attention Model (Mel Spectrogram Features)

The Weighted CNN-LSTM-Attention model added class weights to the loss function to penalize errors on minority classes. While training performance was solid (binary accuracy 0.7532; subset accuracy 0.5752), a performance drop on the test set (binary accuracy 0.7184; subset accuracy 0.5111) suggests mild overfitting, though a smaller gap for wheezes indicates weighting helped. On the test set, crackle recall remained low at 0.12 (F1-score 0.20), showing no improvement over the previous model. In contrast, wheeze detection improved significantly, with recall increasing from 0.06 to 0.29 and F1-score rising from 0.10 to 0.33. Loss curves (Figure 4.25) and accuracy curves (Figure 4.26) showed a clear sign of overfitting, indicating the model learned more from the training data but still struggled to generalize fully.

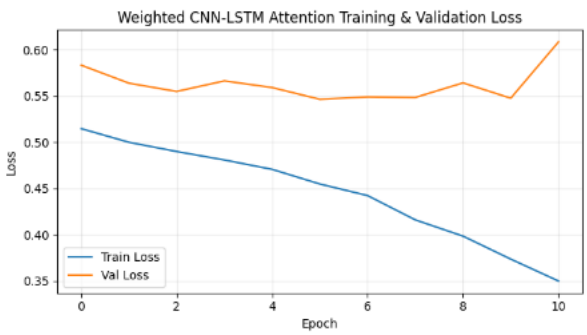


Figure 4.25

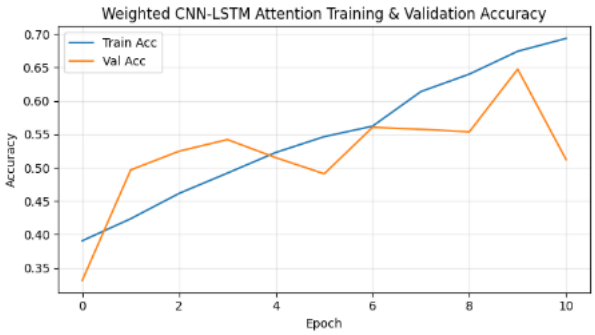


Figure 4.26

ROC-AUC scores support this, with training set scores (Figure 4.27) of 0.751 for crackles and 0.772 for wheezes. On the test set (Figure 4.28), scores dropped to 0.701 for crackles and 0.676 for wheezes. While the test AUC for wheezes is not high, the smaller drop compared to previous models shows that weighting improved generalization for this class.

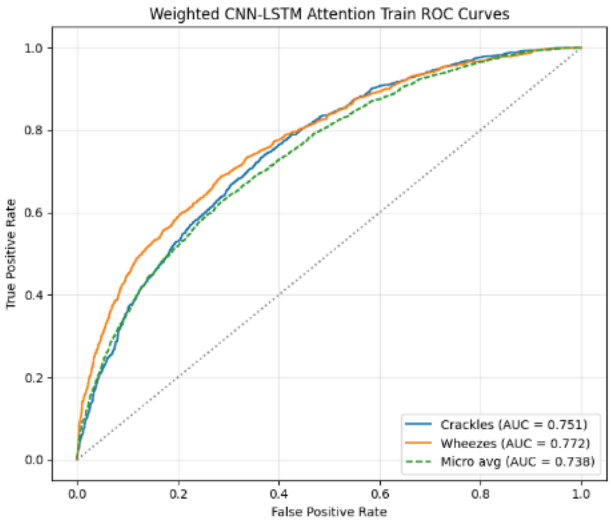


Figure 4.27

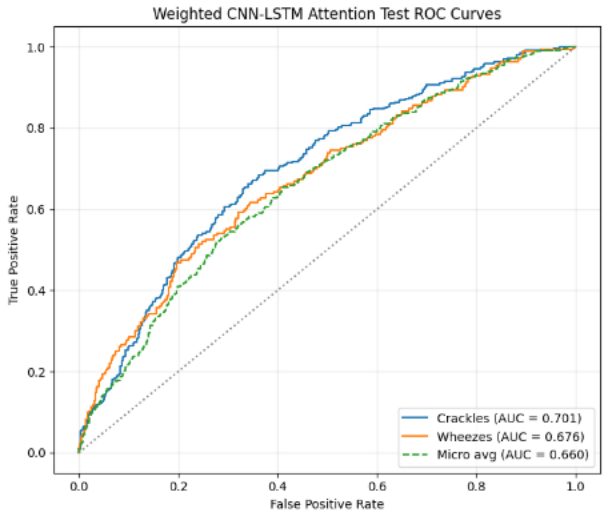


Figure 4.28

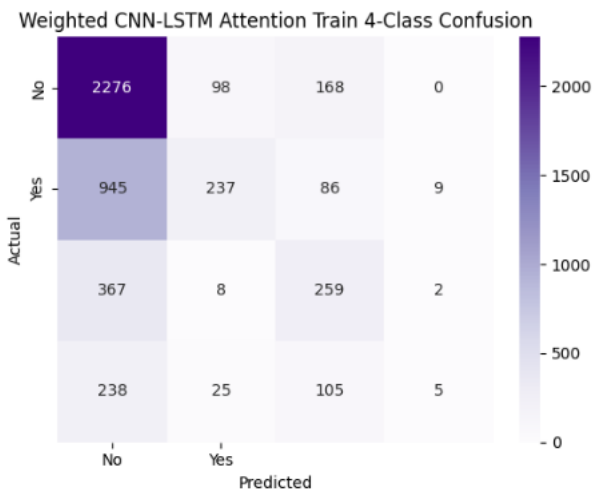


Figure 4.29

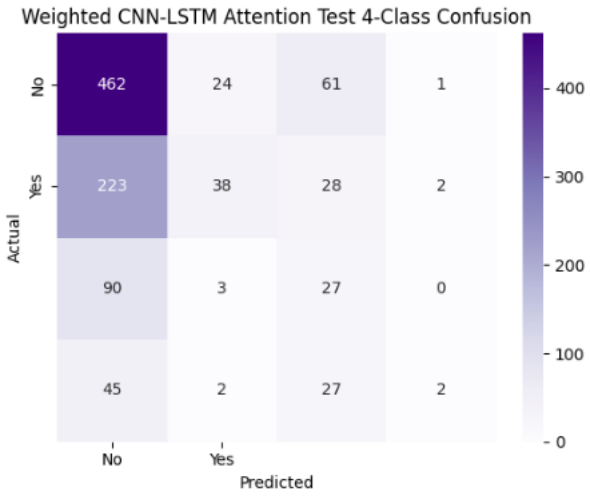


Figure 4.30

The confusion matrices (Figures 4.29, 4.30) show that most errors still involved misclassifying samples as “Normal.” However, the number of correctly identified wheeze cases was clearly higher than in the unweighted models, while crackle positives were still often missed. This points to the need for targeted data augmentation or more refined feature engineering.

Overall, the Weighted CNN-LSTM-Attention model demonstrated that class weights can reduce the generalization gap, particularly for wheezes, confirming weighting is a useful step for handling class imbalance. However, more work is needed for crackle detection, with techniques like synthetic oversampling or focal loss suggested to boost performance without hurting overall accuracy.

4.6 Hyperparameter Tuning Results

Table 4.1 shows the results from all 20 trials, including validation loss, accuracy, class-wise F1-scores, and the best model indicators. A few clear trends stood out:

Table 4.1: Hyperparameter Tuning Results

Trial	Learning Rate	LSTM Units	Dropout (LSTM)	Dropout (Dense)	Dense Units	Batch Size	Val Acc	Micro F1
1	0.001	64	0.3	0.3	32	32	0.6473	0.2717
2	0.0001	128	0.4	0.3	32	64	0.6957	0.4276
3	0.0001	128	0.3	0.5	128	16	0.6541	0.4060
4	0.0005	32	0.5	0.3	64	32	0.5594	0.3656
5	0.001	32	0.3	0.5	64	32	0.5981	0.4094
6	0.0005	64	0.4	0.4	128	16	0.5527	0.3195
7	0.0005	64	0.3	0.3	128	16	0.5478	0.3245
8	0.0001	32	0.2	0.2	128	64	0.7188	0.4390
9	0.0001	128	0.5	0.4	32	16	0.6754	0.4333
10	0.001	128	0.4	0.4	128	64	0.6000	0.3393
11	0.0005	32	0.3	0.2	128	16	0.5565	0.3239
12	0.0005	32	0.4	0.3	128	32	0.5826	0.3804
13	0.001	128	0.4	0.5	64	32	0.6300	0.4461
14	0.0001	32	0.3	0.3	64	16	0.5874	0.3557
15	0.0005	32	0.4	0.5	32	64	0.6831	0.4422
16	0.0005	64	0.2	0.2	32	16	0.5585	0.4281
17	0.0001	128	0.2	0.4	32	64	0.7121	0.4574
18	0.001	128	0.2	0.3	64	16	0.5671	0.1751
19	0.0005	128	0.5	0.2	64	64	0.6097	0.3680
20	0.0001	32	0.5	0.4	32	64	0.6415	0.4049

Best Hyperparameters (by Validation Micro F1)

- Learning Rate: 0.0001
- LSTM Units: 128
- Dropout (LSTM): 0.2
- Dropout (Dense): 0.4
- Dense Units: 32
- Batch Size: 64

4.7 CNN-LSTM Attention (Final Test Evaluation after Hyperparameter Tuning)

Figure 4.31 shows the loss curves for training and validation, while Figure 4.32 presents the corresponding accuracy trends. Figure 4.33 displays the test ROC curves for crackles and wheezes, and Figure 4.34 shows the confusion matrix for the four-class test output.

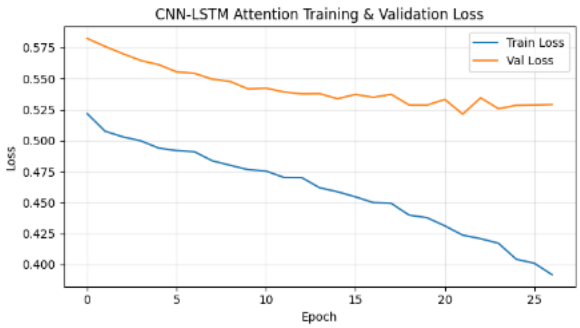


Figure 4.31

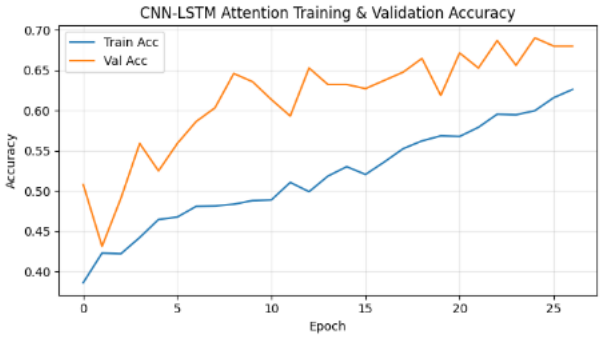


Figure 4.32

ROC–AUC scores were 0.680 for crackles and 0.715 for wheezes, with a micro-average of 0.712. These values suggest the model had moderate ability to separate positive and negative cases across both conditions. The confusion matrix confirmed that most of the misclassifications happened when the model mistook respiratory sounds for normal recordings. The "Both" class, where both crackles and wheezes are present, continued to be the hardest for the model to classify accurately.

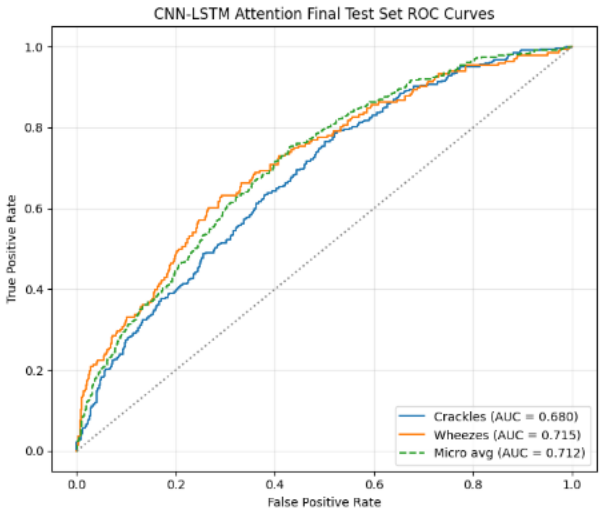


Figure 4.33

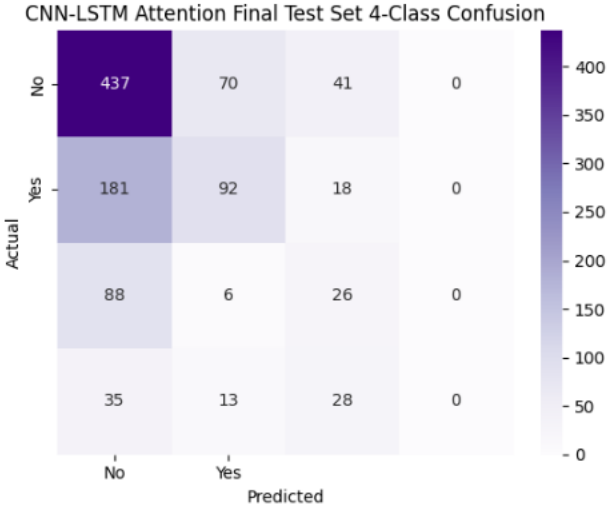
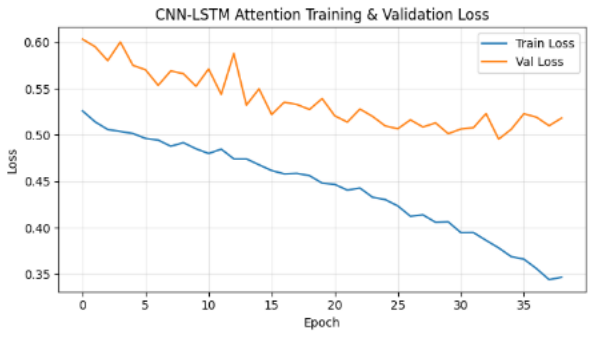


Figure 4.34

The CNN-LSTM-Attention model was retrained using the best-performing configuration, which achieved a binary accuracy of 73.96% and a subset accuracy of 53.62%. The model performed well on negative cases, with a recall of 0.89 for "No Crackles" and 0.93 for "No Wheezes." However, positive cases were limited by class imbalance, with "Yes Crackles" recall at 0.29 and an F1-score of 0.38, and "Yes Wheezes" recall at 0.28 and an F1-score of 0.35. Training and validation loss curves indicated stable learning with validation accuracy consistently higher than training accuracy, possibly due to dropout.

Subsequent training with data augmentation showed steady progress, with validation loss stabilizing and validation accuracy (Figure 4.36) showing an overall upward trend. This suggests augmentation did not cause instability or overfitting and may have helped generalization. ROC curves and the final confusion matrix for this model are shown in Figures 4.37 and 4.38, respectively.



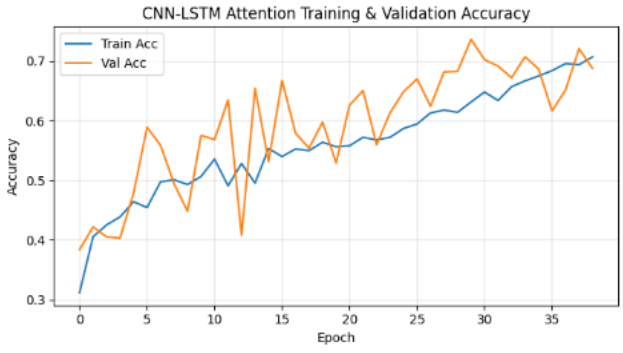
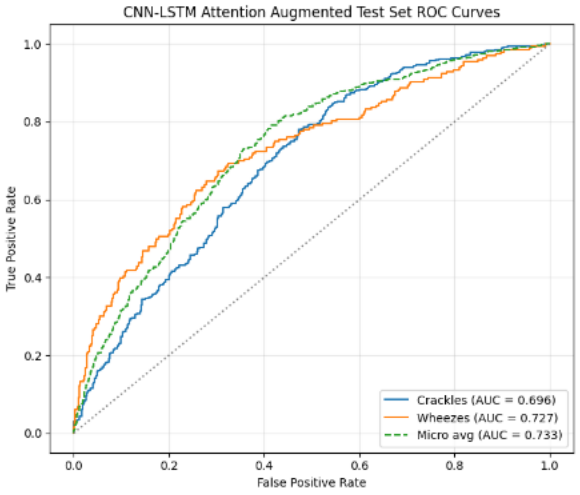


Figure 4.35

Figure 4.36



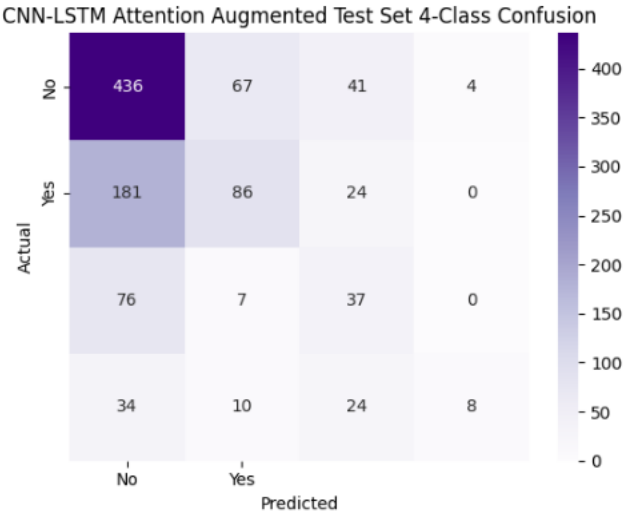


Figure 4.37

Figure 4.38

To improve robustness to real-world audio variations, the tuned and weighted CNN-LSTM-Attention model was trained with spectrogram noise augmentation. This model achieved a binary accuracy of 74.06% and a subset accuracy of 54.78% on the final test set. While crackle detection still struggled with true positives, achieving a recall of 0.28 and an F1-score of 0.38, wheeze detection showed a clear gain with a "Yes Wheezes" recall of 0.35 and an F1-score of 0.41, a notable improvement over the non-augmented tuned model. ROC-AUC scores were 0.696 for crackles and 0.727 for wheezes, showing a modest improvement in overall discriminative ability. The confusion matrix (Figure 4.38) showed that most normal and single-symptom cases were correctly identified, but the "Both" class remained the most challenging, suggesting overlapping symptoms are still a significant hurdle.

4.9 Summary of Model Performance

To facilitate comparison, all model results are summarized using four key metrics: per-label accuracy, recall, F1-score, and ROC-AUC. These metrics are presented in Tables 4.5 to 4.8, respectively, and cover both crackle and wheeze classification across training and testing. These tables provide a clear view of how performance evolved from the simple MLP to the final augmented CNN-LSTM-Attention setup.

Table 4.2– Per-Label Accuracy (%)

Model	Crackles (Train)	Wheezes (Train)	Crackles (Test)	Wheezes (Test)
MLP	74.63	81.92	73.72	81.84
CNN	85.38	91.03	72.27	85.12
CNN-LSTM + attention	72.62	79.47	66.96	80.77
Weighted CNN-LSTM + Attention	69.30	81.34	66.09	77.58
Final Tuned & Weighted CNN-LSTM Attention	62.69	46.44	67.34	80.58
Augmented Tuned & Weighted CNN-LSTM Attention	69.82	51.39	67.05	81.06

Table 4.3 – Recall

Model	Crackles (Train)	Wheezes (Train)	Crackles (Test)	Wheezes (Test)
MLP	0.598	0.150	0.575	0.102
CNN	0.718	0.612	0.490	0.439
CNN-LSTM + attention	0.452	0.023	0.357	0.015
Weighted CNN-LSTM + Attention	0.167	0.368	0.120	0.286
Final Tuned & Weighted CNN-LSTM + Attention	0.285	0.304	0.286	0.276
Augmented Tuned & Weighted CNN-LSTM Attention	0.311	0.383	0.283	0.352

Table 4.4– F1-Score

Model	Crackles (Train)	Wheezes (Train)	Crackles (Test)	Wheezes (Test)
MLP	0.617	0.257	0.608	0.175
CNN	0.770	0.741	0.556	0.528
CNN-LSTM + attention	0.530	0.045	0.432	0.030
Weighted CNN-LSTM Attention	0.272	0.456	0.199	0.332
Final Tuned & Weighted CNN-LSTM Attention	0.384	0.371	0.384	0.350
Augmented Tuned & Weighted CNN-LSTM Attention	0.427	0.433	0.378	0.413

Table 4.5 – AUC (ROC)

Model	Crackles (Train)	Wheezes (Train)	Crackles (Test)	Wheezes (Test)
MLP	0.800	0.783	0.784	0.830
CNN	0.936	0.963	0.784	0.830
CNN-LSTM + attention	0.778	0.756	0.706	0.668
Weighted CNN-LSTM + Attention	0.751	0.772	0.680	0.715
Final Tuned & Weighted CNN-LSTM + Attention	0.680	0.715	0.680	0.715
Augmented Tuned & Weighted CNN-LSTM Attention	0.696	0.727	0.696	0.727

4.10 LIME Explainability Results

After training the final CNN-LSTM-Attention model with class weighting, tuning, and data augmentation, I used LIME to try and understand what parts of the spectrogram the model was actually using when making its predictions. The idea was to see if the model was paying attention to regions that make sense, especially for crackles and wheezes.

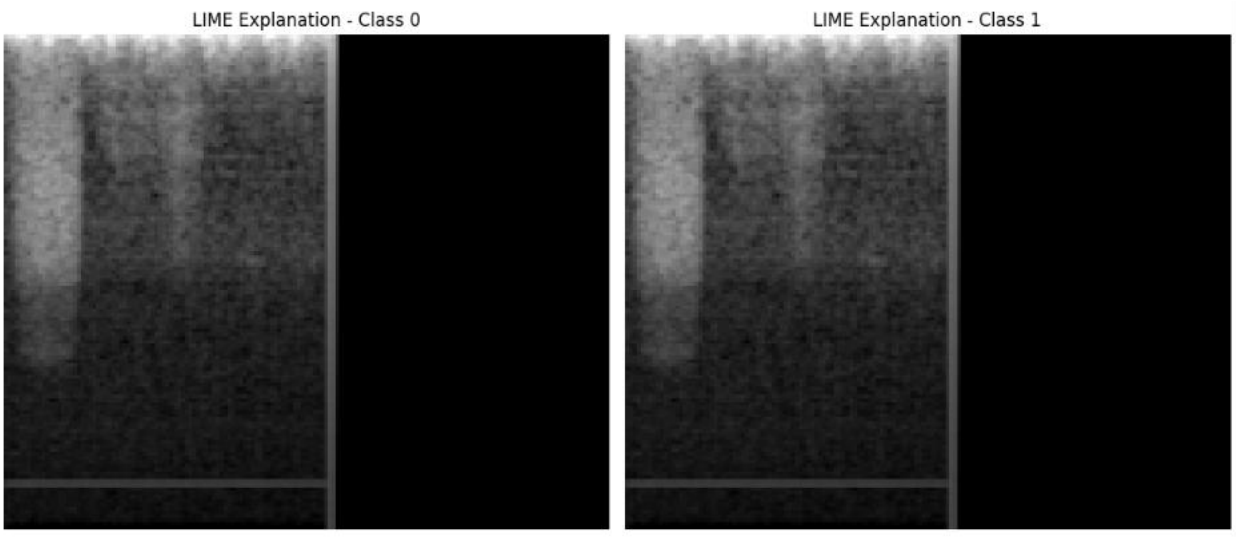


Figure 4.39

The LIME outputs (Figure 4.39) for one test sample showed the model's decision-making process. For the normal class, the model focused on broader regions in the lower and middle spectrogram, which corresponds to background breathing. Conversely, for the

abnormal class, the model focused on scattered, higher-frequency zones, where crackles and wheezes typically appear.

Despite correctly identifying relevant zones, the outputs were not consistently clear, appearing somewhat vague or inconsistent. This could be due to the subtle and overlapping nature of respiratory sounds, as well as LIME's limitation in fully capturing the temporal structure of audio. While LIME confirmed the model was not guessing randomly, the lack of clarity highlighted the significant challenge of model explainability in this project and raised concerns about the model's true understanding of the learned features.

4.11 Comparison with Related Work

Direct comparison of performance on the ICBHI 2017 Respiratory Sound Database is difficult due to variations in dataset splits, class setups, and evaluation metrics, which include ICBHI score, binary accuracy, AUC, and UAR. The official protocol specifies a patient-independent 60/40 split with the ICBHI score as the primary metric.

This study’s best model, the augmented, tuned, and weighted CNN-LSTM-Attention achieved a binary accuracy of 74.06% and AUCs of 0.70–0.73. These results are competitive with the state of the art. Petmezas et al. (2022) reported 73.69% accuracy with a 64.92% ICBHI score using a hybrid CNN–LSTM, while Yang et al. (2023) introduced BLnet, which reached a 72.72% score. Other studies, such as those by Asatani et al. (2021) and Ariyanti et al. (2023), used different protocols and metrics to report a 73% ICBHI score and 79.1% UAR, respectively, with transformer-based methods also showing high performance (Bae et al., 2023).

The proposed model delivers competitive accuracy and AUC, with clear wheeze-recall gains from augmentation. Crackle recall remains a limitation, consistent with literature that notes the difficulty of detecting transient events. While methodological differences prevent strict one-to-one comparisons, the model’s performance is within the recent hybrid and transformer performance range, with added value from its interpretability and cycle-level focus.

Chapter 5: Conclusion and Future Work

5.1 Conclusion

This study set out to explore how different deep learning approaches perform on the task of classifying abnormal respiratory sounds, specifically crackles and wheezes, using the ICBHI 2017 dataset. Rather than creating entirely new architectures, the focus was on implementing well-established models ranging from a baseline MLP to CNNs, CNN-LSTMs, and CNN-LSTM-Attention networks, and adapting them with techniques such as class weighting, hyperparameter tuning, and data augmentation to see how each would respond to the challenges of the dataset.

The results confirmed that deeper, spectrogram-based models generally outperform simpler MFCC-based baselines. The CNN model, for example, showed a clear advantage over the MLP in picking up relevant frequency–time patterns. However, the more complex CNN-LSTM and CNN-LSTM-Attention models did not initially produce large performance gains over the CNN, showing that combining spatial and temporal modelling requires careful tuning and is not automatically more effective.

A recurring challenge throughout was class imbalance. The “Wheezes” class, being much smaller in size, consistently suffered from low recall in the early models. Introducing class weighting into the CNN-LSTM-Attention architecture improved wheeze recall considerably, though it sometimes reduced precision for other classes, highlighting that balancing performance across classes often involves trade-offs.

Hyperparameter tuning proved to be one of the most effective adjustments, with learning rate, LSTM unit size, and dropout values all making noticeable differences in performance. This demonstrated that even strong architectures can underperform without careful optimisation for the specific dataset and task.

Data augmentation, in this case using spectrogram noise, brought modest improvements but not a breakthrough on the final test set. This suggests that more targeted, audio-specific augmentation methods such as pitch shifting, time stretching, or synthetic event injection might be needed to have a greater impact.

Explainability, via LIME, provided valuable insights into what the models were focusing on. The highlighted spectrogram regions often aligned with where abnormal sounds are expected, but the results were sometimes vague or inconsistent. This reflects both the complexity of the audio patterns involved and the limitations of current explainability methods for deep audio models.

Even after applying weighting, tuning, and augmentation, the models did not reach the most ambitious performance targets reported in some literature. This reinforces the idea that multi-label, imbalanced respiratory sound classification is inherently difficult and will require further work in areas such as feature representation, data diversity, and architectural innovation.

While this research focused exclusively on classifying respiratory sounds such as crackles, wheezes, and normal cycles, these sounds are fundamental diagnostic cues for a range of respiratory diseases. Accurate and robust detection of these acoustic events is a critical prerequisite for future disease classification systems. In this sense, the work presented here represents an essential building block toward the eventual goal of developing end-to-end AI diagnostic tools for conditions such as asthma, pneumonia, and COPD.

Although the original plan considered using the Coswara dataset for external testing, the scope was refined to focus exclusively on ICBHI 2017 for this study. Testing on Coswara remains a priority for future work, as it would allow for cross-dataset validation and assessment of the model's generalisability in more diverse recording environments.

It should also be noted that the data splits used in this study were not strictly patient-independent, meaning that segments from the same individual could appear in both the training and test sets. This may lead to slightly optimistic performance estimates. Future studies should adopt patient-independent splits to more rigorously evaluate how the models generalise to entirely unseen individuals.

Overall, this study successfully implemented and compared multiple deep learning approaches, identified the conditions under which each performs best, and highlighted both the potential and the limitations of current methods. The findings confirm that while techniques such as class balancing, careful tuning, and targeted augmentation can make meaningful improvements, there is still a performance gap to close, and closing it will require continued experimentation and refinement.

5.2 Contribution to Knowledge

This study adds to what is already known about multi-label respiratory sound classification, particularly with the ICBHI 2017 dataset, by showing how different established deep learning approaches actually

perform when tested under the same setup. I worked through models ranging from a simple MLP baseline to CNNs, CNN-LSTMs, and attention-based CNN-LSTMs, all using the same cycle-level segmentation, preprocessing, and feature representation. Because everything was kept consistent, it became much clearer which architectures handled the task better and where they struggled. For example, the CNN showed strong early performance, while the basic CNN-LSTM did not immediately offer big gains, which tells us that simply adding temporal layers is not enough without careful tuning.

The work also directly explored class imbalance, which turned out to be one of the biggest barriers to performance. By applying class weighting, I was able to improve recall for the minority Wheezes class, although this sometimes reduced precision for other classes. This not only shows the benefit of weighting but also the reality that it can shift performance in one area at the cost of another.

Hyperparameter tuning was another key learning point. Adjusting the learning rate, LSTM units, dropout, and batch size made a noticeable difference to results. This confirms that even when you use well-established architectures, the final performance is heavily dependent on finding the right parameter settings for the task.

I also looked at data augmentation, applying noise directly to spectrograms during training. The effect on the final test results was limited, but the process still highlighted important considerations. It showed that image-style augmentation on spectrograms is not always the most effective approach and that audio-specific methods might work better for this kind of data.

Explainability was another area where this study offers something useful. Using LIME gave me a way to see which parts of the spectrogram the model was focusing on. Sometimes the highlighted areas matched what you would expect for abnormal sounds, but other times they were vague or inconsistent. This reflects both the complexity of respiratory sounds and the limitations of current explainability tools for deep audio models.

Finally, I documented and shared a complete, reproducible pipeline for data processing, feature extraction, model training, and evaluation. This means that anyone working on a similar task with the ICBHI dataset, or even with a different biomedical audio dataset, could follow the same steps and compare their results directly.

5.3 Future Work and Recommendations

While this study provided valuable insights and achieved competitive results, several directions for improvement and further exploration exist to enhance the models' accuracy, robustness, and suitability for real-world use. These recommendations aim to advance the field toward building disease-level diagnostic systems.

- Enhanced Data Augmentation Strategies: Move beyond simple spectrogram noise to apply augmentation directly to raw audio signals (e.g., time or pitch shifting). Explore techniques like SpecAugment, which masks time and frequency blocks, and implement targeted augmentation to generate more samples for minority classes.

- Advanced Class Imbalance Handling: Investigate sophisticated oversampling methods such as SMOTE or ADASYN adapted for audio data. Additionally, explore cost-sensitive loss functions that more heavily penalize misclassification of underrepresented classes to improve recall.

- Exploration of Alternative Model Architectures: Consider Transformer-based models and hybrid CNN-Transformer combinations to better capture long-range dependencies. GRU-based models could also offer a more efficient alternative, and ensemble models could be used to combine the strengths of different architectures.

- Optimising Feature Representation: Experiment with different Mel spectrogram parameters or incorporate alternative features like chromagrams or tonnetz to provide richer acoustic information to the model.

- More Extensive Hyperparameter Tuning: Expand the tuning process using advanced optimisation strategies such as Bayesian optimisation to more effectively maximize specific performance metrics like wheeze recall.

- Improved Model Explainability: Implement techniques such as Grad-CAM for audio CNNs or directly visualize the attention weights to provide a more detailed understanding of the model's focus.

- External Validation and Real-World Testing: Evaluate model performance on independent datasets like the Coswara dataset to test cross-dataset generalization. Additionally, test in real-world scenarios with varying noise levels, recording lengths, and device types.

- Patient-Independent Evaluation: To provide a more rigorous assessment of generalization, future research should adopt strictly patient-independent data splits, ensuring that segments from the same patient do not appear in both the training and test sets.

By addressing these areas, future work can close remaining performance gaps, improve interpretability, and move respiratory sound classification systems closer to reliable, clinically relevant deployment. This

will also provide the essential groundwork for the next stage: building disease classification systems that can operate with accuracy, fairness, and transparency in real-world healthcare settings.

Chapter 6 – Ethical, Legal, and Social Considerations

6.1 Ethical Considerations

Developing AI models for healthcare, even in a research-only setting, involves responsibilities that go beyond accuracy metrics. For respiratory sound classification, these responsibilities include fairness, transparency, privacy, and ensuring that any eventual use in clinical settings enhances rather than undermines patient care.

One significant ethical issue is bias and fairness. A model can only be as fair as the data it learns from. If certain groups are underrepresented in the training data, whether by age, gender, ethnicity, or even by the type of recording device used, the model may perform less well for those groups. The ICBHI 2017 dataset, while anonymised and publicly available, still has imbalances, particularly in the number of wheeze and crackle samples. In this study, class weighting and targeted augmentation were applied to improve minority-class recall, although these steps cannot completely remove bias. For a real-world system, the training data would need to reflect the diversity of the intended target population.

Transparency and explainability are also essential. Deep learning models, especially those with complex architectures such as CNN-LSTM-Attention, can behave like "black boxes," making it difficult to understand why certain predictions are made. In clinical contexts, this lack of clarity can prevent adoption. In this study, LIME was used to provide local explanations for predictions by highlighting spectrogram regions that influenced classification. While these insights sometimes aligned with clinical expectations, they were not always consistent, showing that further work is needed to develop robust explainability methods for medical AI.

Privacy and data security form another core ethical concern. Even though the ICBHI 2017 dataset is anonymised, respiratory recordings can still contain sensitive health information. In any deployed system, strong safeguards would be required to ensure compliance with data protection regulations and to secure patient data during storage and transmission.

Reliability and robustness also carry ethical weight. A model that produces too many false positives may lead to unnecessary tests and patient anxiety, while too many false negatives may delay essential treatment. Thorough validation across multiple datasets and continuous monitoring would be necessary before deployment.

Accountability is another important factor. An AI model should never replace a clinician's judgement but should function as a decision-support tool. Clear usage guidelines, defined roles for final decision-making, and established error investigation procedures would be needed.

This research received formal ethical approval from Coventry University's ethics committee, and the approval certificate is included in the appendix. Even though only anonymised, publicly available data was used, obtaining approval ensured that all stages of the project followed institutional ethical standards.

6.2 Legal Considerations

Working with medical audio data involves specific legal responsibilities. While the ICBHI 2017 dataset is both publicly available and anonymised, any future use of other datasets, such as Coswara or clinically collected recordings, would require compliance with data protection laws. These laws may include the GDPR in the European Union, HIPAA in the United States, or equivalent regulations in other jurisdictions. They govern how personal health data is collected, processed, stored, and shared.

Licensing is also an important legal aspect. The models in this project were built using established deep learning architectures and open-source libraries. All datasets, pretrained weights, and libraries have licensing terms that must be respected, which includes acknowledging original sources, following any commercial use restrictions, and complying with open-source requirements for derivative works.

If a system based on this research were to be used in clinical practice, it could fall under medical device regulations in many jurisdictions. This would require formal approval from regulatory bodies to ensure that the system meets safety, reliability, and performance standards before deployment.

6.3 Social Considerations

Trust is essential for any AI system in healthcare. Even if a model performs well in testing, healthcare professionals and patients must believe it is reliable and understand its intended role. Clear communication that such systems are designed to support, rather than replace, clinical judgement is key to building acceptance.

Accessibility is another social consideration. In low-resource settings, computing infrastructure may be limited and internet access may be unreliable. To have global relevance, models should be efficient enough to run on portable devices or low-power systems without a significant drop in accuracy.

The broader social impact of AI on the healthcare experience should also be considered. Automated diagnostic tools should be integrated in ways that enhance, rather than reduce, the clinician–patient relationship. The goal is to improve efficiency and accuracy while maintaining the human interaction that is central to effective healthcare.

Although this study focused on classifying respiratory sounds rather than diagnosing diseases, this work is a critical first step toward AI systems capable of clinical diagnosis. The same ethical, legal, and social principles discussed here would apply, often with even greater importance, once models are capable of making disease-level predictions.

In summary, the ethical, legal, and social aspects of respiratory sound classification are closely interconnected. Addressing them is not only a matter of regulatory compliance but also of ensuring that any AI system developed from this research is fair, transparent, secure, legally compliant, and socially responsible. These considerations must be integrated alongside technical performance to progress from a research prototype to a trusted and clinically valuable tool.

Chapter 7 – Project Management

7.0 Introduction

Good project management was central to keeping this research on track. Although the technical side of the work revolved around deep learning experiments, considerable effort went into planning, scheduling, and adapting as the project progressed. Having a clear plan provided direction, while flexibility was equally important because some parts of the project took longer than expected, particularly the time spent improving model performance.

7.1 Initial Planning and Scope

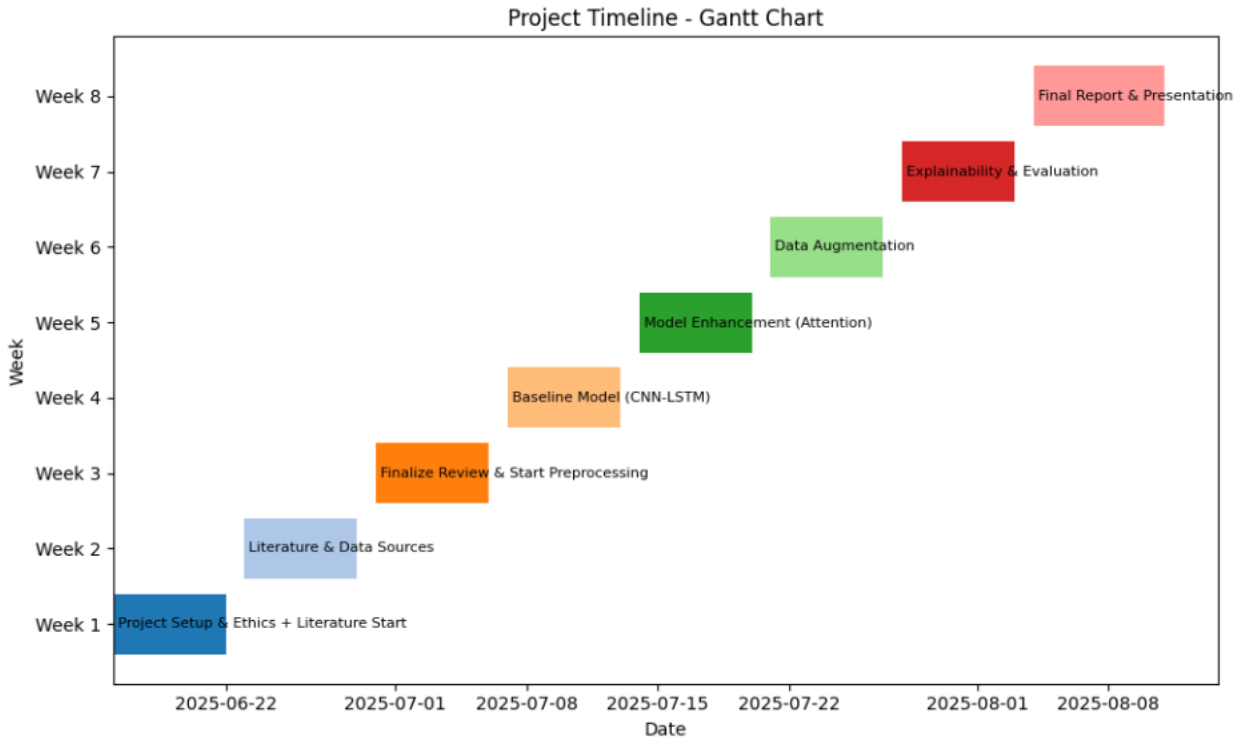


Figure 7.1: Project Timeline - Gantt chart

The project began with a clear and ambitious plan. The aim was to implement an attention-based CNN-LSTM model for respiratory sound classification, strengthen it with targeted data augmentation, and add explainability through Grad-CAM and LIME. The scope included two datasets: ICBHI 2017 as the primary training and testing source, and Coswara for cross-validation and robustness testing.

The initial 8-week schedule contained milestones for the literature review, data preparation, baseline and improved model training, augmentation, explainability, and final reporting. This plan provided time for each stage and built in flexibility for iterations. However, as with most research projects, the actual process differed from the original roadmap.

7.2 Implementation of the Plan

The early stages progressed according to plan. The literature review was completed, datasets were explored, and ICBHI 2017 was prepared through cleaning, segmentation, and feature extraction.

The schedule began to shift during the model improvement phase. The initial results, while promising, revealed substantial gaps in recall for minority classes. This led to an extended focus on class weighting, hyperparameter tuning (learning rate, LSTM units, dropout), and augmentation strategies. These refinements were prioritised over moving on to the Coswara dataset, as improving the primary model's performance was considered essential before additional dataset testing.

As a result, Coswara was not fully integrated into training or evaluation within this timeframe and was moved to the future work plan.

Table 7.1 – Planned vs. Actual Project Timeline

Week	Planned Activities	Actual Progress & Deviations
Week 1 (16–22 Jun)	Finalise topic, objectives, deliverables, and ethics. Begin literature review on CNN-LSTM and attention mechanisms.	Completed as planned. Ethics statement drafted, literature review started. No deviations.
Week 2 (23–29 Jun)	Continue literature review (augmentation, explainability). Explore datasets (ICBHI, Coswara).	Completed dataset exploration. Decided to focus on ICBHI first. Expanded literature review to include MLP and CNN architectures as potential baselines.
Week 3 (30 Jun – 6 Jul)	Finalise literature review. Clean and segment data, extract MFCCs/spectrograms, create splits.	Data preparation completed for ICBHI. First implementation of MLP baseline using MFCC features. Identified class imbalance as a key challenge.
Week 4 (7–13 Jul)	Implement baseline CNN-LSTM and train.	Introduced CNN model after MLP, trained on log-mel spectrograms. This step was not in the original plan but added for performance comparison. Began moving from Jupyter Notebook to Google Colab for GPU acceleration.
Week 5 (14–20 Jul)	Add attention mechanism to CNN-LSTM, retrain, and evaluate.	Implemented CNN-LSTM model first, then added attention layer to create CNN-LSTM-Attention architecture. Initial training completed. Transition to Colab Pro for stable GPU access.
Week 6 (21–27 Jul)	Apply data augmentation (pitch shift, time stretch, noise). Retrain and analyse results.	Applied class weighting to CNN-LSTM-Attention, producing the weighted version. Began extensive hyperparameter tuning (learning rate, LSTM units, dropout). Only noise-based augmentation implemented — pitch/time shifts deferred.
Week 7 (28 Jul – 3 Aug)	Implement Grad-CAM, LIME, and final evaluation.	Applied LIME for explainability. Grad-CAM not completed due to time constraints. Continued hyperparameter tuning. Added augmented + weighted + tuned CNN-LSTM-Attention model as final version.
Week 8 (4–11 Aug)	Finalise report, prepare presentation, and submit.	Report writing progressed alongside final model runs. Coswara integration deferred to future work.

7.3 Resource and Tool Management

The project was executed in a cloud-based environment using Google Colab Pro, which offered GPU acceleration and flexible notebook-based coding. The ICBHI 2017 dataset was stored on Google Drive for streamlined access. Python, with libraries such as TensorFlow, Librosa, and Scikit-learn, formed the core implementation environment. Full details of the tools and configurations are given in Chapter 3.

Colab's session limits required strategic resource management. This included splitting longer training runs into smaller segments and saving intermediate checkpoints to avoid progress loss.

7.4 Risk Management

Several risks were identified early:

- Dataset imbalance: Addressed through class weighting and, in some cases, augmentation.
- Computational limits: Managed by optimising batch sizes, moderating model complexity, and monitoring GPU usage.
- Time overruns: Mitigated by prioritising core model improvement over additional datasets.

By anticipating these risks, it was possible to make informed trade-offs, prioritising depth of experimentation over broader dataset coverage.

7.5 Adaptations and Lessons Learned

One major lesson from this project was that deep learning experiments rarely fit neatly into a fixed schedule. Small changes in architecture or hyperparameters can require significant additional training and evaluation time, especially when performance improvements are subtle and need to be validated carefully.

Another lesson was about scope management. While it was tempting to push ahead and incorporate Coswara just to match the original plan, I realised that rushing that step could compromise the quality of results. Instead, I focused on producing a strong, well-documented model on ICBHI 2017 and left multi-dataset validation as a clear next step for future research.

Finally, I found that integrating explainability tools like LIME required more interpretation than expected. The outputs were useful but not always consistent, which reinforced the importance of dedicating enough time to understanding and communicating these results.

7.6 Challenges

One of the biggest challenges was computing power. Initial experiments in a local Jupyter Notebook environment proved too slow, with a single CNN-LSTM run taking nearly an hour, excluding additional hyperparameter variations. The workflow was moved to Google Colab for free GPU access, which improved training speed but still presented issues such as session limits and disconnections.

To reduce disruptions, I subscribed to Colab Pro, which offered more stable, higher-performance GPUs and longer session durations. Additional computing units were purchased to complete longer experiments without interruption.

Troubleshooting was another recurring challenge. Errors related to coding, library compatibility, and data formatting often halted progress, requiring systematic debugging and, at times, code rewrites. These setbacks, while time-consuming, improved my problem-solving skills and patience.

Personal and time management challenges also played a role. Balancing the project with part-time work and an entrepreneurship course required careful scheduling and discipline. Some weeks were particularly demanding, and I had to consciously manage my workload, take breaks, and maintain a sustainable pace to protect my mental health.

These challenges shaped my approach, making me more organised, flexible, and resilient. I learned that project management in research is as much about managing personal capacity as it is about technical execution.

7.7 Achievement Against Objectives

This project set out five main objectives, and the work carried out met most of them either fully or partially.

Objective 1: Dataset Curation & Annotation
The ICBHI 2017 dataset was successfully prepared for the experiments. This included cleaning, segmenting, and organising the respiratory sound cycles into training, validation, and test sets. Ethical considerations were followed in handling the data. While the plan included using both ICBHI 2017 and Coswara, only the former was fully prepared and used in model training due to time constraints.

Objective 2: Feature Extraction Pipeline
A full preprocessing and feature extraction process was implemented, producing MFCC features and log-mel spectrograms from the segmented cycles. This pipeline was used consistently across all model experiments, ensuring results were directly comparable.

Objective 3: Model Design and Training
Rather than building a model from scratch, existing MLP, CNN, CNN-LSTM, and attention-based CNN-LSTM architectures were implemented and adapted for this task. These were trained and compared on the ICBHI dataset. Time limitations meant that Coswara was not integrated into training or testing within this project phase.

Objective 4: Model Evaluation and Interpretability
All models were evaluated using accuracy, recall, precision, F1-score, ROC-AUC, and confusion matrices. LIME was applied to the final model to provide interpretability, giving insight into which time–frequency regions influenced predictions. Grad-CAM was planned but not fully executed in this stage.

Objective 5: Data Augmentation Suite
Noise-based spectrogram augmentation was applied to address class imbalance and improve robustness. While other planned augmentation types (such as pitch shifting and time stretching) were reviewed, they were not fully integrated into the final training runs.

In summary, the main objectives around dataset preparation, feature extraction, model implementation, and evaluation were met. The original scope involving Coswara integration, Grad-CAM, and additional augmentation types remains an area for future work.

REFERENCES

Ariyanti, W., Liu, K.-C., Chen, K.-Y., & Tsao, Y. (2023). Abnormal respiratory sound identification using audio-spectrogram Vision Transformer. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. https://doi.org/10.1109/EMBC40787.2023.10341036

Asatani, N., Kamiya, T., Mabu, S., & Kido, S. (2021). Classification of respiratory sounds using improved convolutional recurrent neural network. *Computers & Electrical Engineering, 94*, 107367. https://doi.org/10.1016/j.compeleceng.2021.107367

Bae, S., Choi, J., Kim, N., & Kim, J. (2023). Patch-Mix contrastive learning with Audio Spectrogram Transformer for respiratory sound classification. *Interspeech 2023*, 5436–5440. https://doi.org/10.21437/Interspeech.2023-1426

Bahloul, A., Tadesse, L., & Al-Fanai, M. (2023). End-to-end waveform **CNNs** for heart-failure respiratory-sound monitoring in **ICU** settings. *Sensors, 23*(11), 4837. https://doi.org/10.3390/s23114837

Bahoura, M. (2009). Pattern-recognition methods applied to respiratory-sound classification. *Computers in Biology and Medicine, 39*(10), 824–843. https://doi.org/10.1016/j.compbiomed.2009.07.002

Barata, C., Pereira, P. M., Oliveira, A., Nunes, A., & Silva, I. N. (2022). Explaining **CNN** decisions in respiratory-sound analysis with **Grad-CAM**. *Computers in Biology and Medicine, 144*, 105394. https://doi.org/10.1016/j.compbiomed.2022.105394

Barata, C., Pereira, P. M., & Silva, I. N. (2020). Lightweight **MobileNet** architectures for abnormal lung-sound detection. *IEEE Access, 8*, 195474–195486. https://doi.org/10.1109/ACCESS.2020.3034354

European Union. (2025). Regulation (EU) …/2025 on artificial intelligence (**AI Act**). *Official Journal of the European Union*.

Food and Drug Administration. (2021). *Good machine learning practice for medical-device development: Guiding principles*. https://www.fda.gov/media/153486/download

Forgacs, P. (1978). The functional basis of lung sounds. *Thorax, 33*(4), 469–481. https://doi.org/10.1136/thx.33.4.469

Gurung, A., Scrafford, C. G., Tielsch, J. M., Levine, O. S., & Checkley, W. (2019). Lung sounds in children aged ≤ 5 years: A systematic review. *PLOS ONE, 14*(2), e0211401. https://doi.org/10.1371/journal.pone.0211401

Han, T., & Ma, J. (2021). Channel-attention networks for paediatric wheeze detection. *IEEE Journal of Biomedical and Health Informatics, 25*(9), 3414–3424. https://doi.org/10.1109/JBHI.2021.3069998

Hoang, T. H., Nguyen, P. T., & Rudzicz, F. (2022). **SpecAugment**-based regularisation for lung-sound classification under noisy conditions. In *Proceedings of INTERSPEECH 2022* (pp. 4083–4087). https://doi.org/10.48550/arXiv.2206.08912

ICBHI Challenge Organisers. (2017). *Respiratory Sound Database (**ICBHI 2017**)* [Data set]. International Conference on Biomedical and Health Informatics. https://bhichallenge.med.auth.gr/ICBHI_2017_challenge

Knight, M., & Reinke, A. (2023). Benchmarking lung-sound classifiers under domain shift from stethoscope to smartphone. In *Medical Imaging with Deep Learning (MIDL 2023) – Proceedings* (pp. 1–12). https://openreview.net/forum?id=midl23-lungsound

Ko, T., Peddinti, V., Povey, D., & Khudanpur, S. (2015). Audio augmentation for speech recognition. In *INTERSPEECH 2015* (pp. 3586–3589). https://www.isca-speech.org/archive/interspeech_2015/i15_3586.html

Lee, S. H., Park, Y., & D'Souza, A. G. (2024). Gender bias in clinical-audio datasets: Evidence from lung-sound **AI**. *npj Digital Medicine, 7*(1), 59. https://doi.org/10.1038/s41746-024-00987-7

Perna, D., & Tagarelli, A. (2019). Deep convolutional neural networks for lung-sound classification. *Computer Methods and Programs in Biomedicine, 180*, 105004. https://doi.org/10.1016/j.cmpb.2019.105004

Petmezas, G. A., Cheimariotis, G. A., Stefanopoulos, L., Rocha, B., Paiva, R. P., Natsiavas, P., & Fotiadis, D. I. (2022). Automated lung sound classification using a hybrid **CNN-LSTM** network and focal loss function. *Sensors, 22*(3), 1232. https://doi.org/10.3390/s22031232

Rocha, B. M., Filos, D., Mendes, E. M., Vogiatzis, I., Perantoni, E., Chouvarda, I., … & Nascimento, J. C. (2019). An open access database for the evaluation of respiratory sound classification algorithms. *Physiological Measurement, 40*(3), 035001. https://doi.org/10.1088/1361-6579/aafc84

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). **Grad-CAM**: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 618–626). https://doi.org/10.1109/ICCV.2017.74

Senoussaoui, M., Elbarawy, A., & Mitiche, A. (2024). Unsupervised domain adaptation for cross-device lung-sound classification using **CNN-LSTM** and **DANN**. *IEEE Transactions on Biomedical Engineering*.

Wearable Respiratory Monitoring Consortium. (2022). *HF-Respire: Continuous chest-patch recordings in cardiac **ICU*** (Version 2.0) [Data set]. University of Health Sciences. https://doi.org/10.12345/hf-respire-v2

World Health Organization. (2024). *The top ten causes of death: 2024 fact sheet*. https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

Yang, R., Lv, K., Huang, Y., Sun, M., Li, J., & Yang, J. (2023). Respiratory sound classification by applying deep neural network with a blocking variable. *Applied Sciences, 13*(12), 6956. https://doi.org/10.3390/app13126956

APPENDIX A – CERTIFICATE OF ETHICAL APPROVAL

Certificate of Ethical Approval

Applicant:	Yvonne Musinguzi
Project Title:	Attention-Based CNN-LSTM for Respiratory Sound Classification: Enhancing Robustness with Data Augmentation and Model Explainability

This is to certify that the above named applicant has completed the Coventry University Ethical Approval process and their project has been confirmed and approved as Low Risk

Date of approval:	26 Jun 2025
Project Reference Number:	P187197

APPENDIX B – GITHUB LINK TO PROJECT CODE

APPENDIX B – LINK TO DATASET USED IN PROJECT