Auscultation-Based Pulmonary Disease Detection and Classification Using Deep Neural Networks

Yusra Shaikh, Areej Fatemah Meghji*, Zobiya Jumani, and Mehak Jatoi
Department of Software Engineering, Mehran University of Engineering and Technology, Jamshoro, Pakistan
*Correspondence: areej.fatemah@faculty.muet.edu.pk

Citation | Shaikh. Y, Meghji. A. F, Jumani. Z, Jatoi. M, "Auscultation-Based Pulmonary Disease Detection and Classification Using Deep Neural Networks", IJIST, Vol. 07 Issue. 04 pp 2630-2645, November 2025
Received| September 01, 2025 **Revised|** September 26, 2025 **Accepted|** September 28, 2025 **Published|** November 01, 2025.

Pulmonary diseases like Pneumonia, Bronchiectasis, and Chronic Obstructive Pulmonary Disease cause a large number of deaths worldwide. For such diseases to be treated and managed effectively, an early and accurate diagnosis is essential. In this work, we propose a deep learning model based on Recurrent Neural Networks (RNN) that can detect three different pulmonary diseases, as well as healthy lung sounds, using only auscultation recordings. The model was trained using the ICBHI dataset, which contains 920 recordings from 126 people and covers more than 6,800 respiratory cycles. To uniform the data, the audios are padded to equal length. To tackle class imbalance in the dataset, augmentation techniques of Gaussian noise injection, time-shifting, and time stretching are used. We employ a simplified version of the Gated Recurrent Unit (GRU)-based RNN architecture to deal with the padded sequences, along with a dropout layer to avoid overfitting. The model is trained using the Adamax optimizer with categorical cross-entropy loss, along with a model checkpoint to ensure learning consistency. Apart from the evaluation of model accuracy, we also evaluated the F1-score, accuracy, and loss graphs to ensure the competitive performance of our approach. Out of the six different experiments, with different data variations and two different model architectures, the outperforming model exhibited an accuracy of 98.53%, a precision of 98.57%, a recall of 98.53%, and an F1-score of 98.52%.

Keywords: Pulmonary Disease; Auscultations; Deep Learning; Recurrent Neural Network; Data Augmentation; Gated Recurrent Unit

Introduction:

Pulmonary disease refers to any condition that affects the lungs and compromises respiratory function. Numerous pulmonary disorders exist, including asthma, Chronic Obstructive Pulmonary Disease (COPD), pneumonia, bronchiectasis, and respiratory tract infections [1]. Recent studies have identified pulmonary diseases as the third leading cause of death worldwide, accounting for nearly 4 million fatalities in 2019, with approximately 80% of these deaths attributed to COPD [2]. Pneumonia also poses a major public health challenge, as emphasized by UNICEF's 2017 global statistical report, which recorded its widespread prevalence worldwide [3]. In response to the increasing incidence of lung diseases, considerable global efforts have been made to reduce mortality rates. Assessment of lung health is done using pulmonary function tests, chest X-rays, and computed tomographic (CT) scans [4]. It is important to understand that these facilities and skilled staff are not always available or, if available, are expensive and time-consuming for most underprivileged areas. In contrast, auscultation provides a non-invasive, low-cost, portable alternative that enables doctors to use a standard stethoscope to analyze patients' lung sounds in order to detect diseases such as pneumonia, COPD, and asthma. However, since auscultation heavily relies on the physician's subjective interpretation, there is a pressing need for a more consistent and objective diagnostic approach.

With the rapid advancements in telemedicine and Artificial Intelligence (AI), it has become possible to detect subtle patterns in respiratory sounds, enabling more accurate and efficient clinical diagnosis. Accordingly, models can be built using AI that can detect respiratory diseases at an early stage, prior to the worsening of the patient's health, thus enabling timely treatment to save lives [4]. A core approach in this field is deep learning, which is inspired by the structure and architecture of the human brain. Several layers of artificial neurons make up deep learning models that derive complex and sophisticated features from input data [5]. These models are good at recognizing patterns, which form the core of a disease diagnosis from medical inputs such as auscultation recordings, CT scans, or X-rays. These models eliminate the need for manual feature extraction by automatically learning relevant features from raw data. This capability makes deep learning particularly well-suited for medical applications, where data complexity is substantial and diagnostic precision is essential.

Recurrent Neural Networks (RNNs) are a class of deep learning models that are especially well-suited to handling time-series and sequential data [6]. In contrast to conventional neural networks, Recurrent Neural Networks (RNNs) are designed with feedback loops that allow them to store information from earlier inputs, giving them the ability to process and learn from sequential data. This temporal memory is useful for the analysis of lung sounds, which are sequential and vary dynamically during the respiratory cycle. Therefore, RNNs can capture the rhythmic and often periodic structure of breathing sounds, as wheezes, crackles, and other respiratory conditions manifest themselves in this way. The RNN model proposed in this study classifies features into four classes, namely COPD, bronchiectasis, pneumonia, and healthy, using a classification technique. Classification is a supervised machine learning technique used to assign class labels to given inputs [7]. In this process, a model is initially trained using a labeled dataset, and its performance is later assessed on an unlabeled testing dataset. Once the model demonstrates satisfactory accuracy and reliability, it is deployed to make predictions on new, unseen data.

This research aims to develop a model, based on deep learning, which is able to detect and classify respiratory diseases from lung sound recordings accurately using the Int. Conf. on Biomedical Health Informatics (ICBHI) dataset [8]. The dataset consists of annotated auscultation recordings of various pulmonary conditions. This study undertakes six different experiments with multiple pre-processing techniques and adjusted RNN architectures in order

to determine the best combination of model structure with variation in data for proficient training.

Novelty:

Auscultation-based research emphasizes spectrogram-based features combined with convolutional or hybrid neural network architectures, while relatively little focus is given to alternative feature extraction and sequential modeling techniques. In this study, we aim to bridge these gaps by utilizing Mel-Frequency Cepstral Coefficients (MFCCs) instead of spectrograms, enabling a more effective representation of the spectral and temporal characteristics of respiratory signals. Also, as RNNs are best suited for sequential data and the temporal relationships found in lung sound recordings, the experiments focus on different model architectures for RNN as well as various data variation strategies, such as segmentation and the use of fully padded sequences, with the use of the masking layer to handle these sequences efficiently instead of relying on using convolutional or hybrid deep learning frameworks.

The subsequent sections detail the development of our proposed system. The Literature Review summarizes relevant studies, highlighting key approaches, findings, and limitations. The Methodology section explains the training dataset, its variations, preprocessing techniques, feature extraction methods, model architectures, and performance evaluation metrics. This is followed by Results and Discussion, where results from six experiments are analyzed and compared. Finally, the Conclusion presents the main insights drawn from the study and outlines directions for future work.

Literature Review:

This section establishes the foundation for understanding the research domain, existing methodologies, and current research gaps, thereby supporting the development of an effective solution. It presents a comprehensive review of five recent and relevant studies in the field.

Tariq et al. [9] proposed a CNN-based approach to classify seven respiratory sound classes from the ICBHI 2017 database. Mel spectrograms were generated using the Python library Librosa. To address the limitation of insufficient audio recordings for effective CNN training, the authors employed data augmentation techniques such as time stretching, pitch shifting, and dynamic range compression. They built a Custom 2D CNN (3 convolutional layers, 2 FC layers) along with ReLU (Leaky Rectified Linear Unit) activation and Softmax output. This configuration captured spatial patterns in spectrogram images efficiently. After comparing the results of different techniques applied to the dataset, they achieved the highest 97% accuracy using a 70/30 split by applying augmentation techniques on normalized data.

Basu and Rana [10] proposed a six-class deep neural network architecture by training it on Mel Frequency Cepstral Coefficients (MFCC) extracted using the ICBHI 2017 lungs sound dataset. The neural network is composed of five layers: a Gated Recurrent Unit (GRU), Leaky ReLU activation, a Dense layer, a Dropout layer, and an Add layer. Before training the model, data augmentation techniques are applied to enhance the representation of minority class recordings. The model was trained on 1000 iterations, and it achieved an accuracy of 95.67%±0.77%. Petmezas et al. [11] introduce a hybrid approach that integrates CNN and LSTM architectures. Utilizing the ICBHI 2017 dataset, their model employs a CNN to reduce input dimensionality and extract relevant features, while the LSTM component captures and retains the temporal patterns present in each input sequence. Further, they handle data imbalance and reduce prediction errors by implementing the focal loss (FL) function. The accuracy of the model, which was trained using spectrograms, was 73.69% by split of 60/40, 76.39% by interpatient 10-fold cross-validation, and 74.57% by leave-one-out cross-validation.

Zhang et al. [12] conducted a comparative study of four models: CNN, LSTM, CNN-LSTM, and CNN-BLSTM, and presented an effective approach for pulmonary disease

classification. The best-performing algorithm has been LSTM, achieving an overall accuracy of 98.82% owing to its capability to capture sequential patterns in time-series audio data. They implemented an imbalanced-learn toolbox to deal with class imbalances in the ICBHI 2017 dataset. A six-class LSTM model has been trained using five audio features: MFCCs, Chroma, Tonal Centroids, Mel Spectrogram, and Spectral Contrast. The dataset has been divided into 80% for training and 20% for testing.

Nawaz et al. [13] introduced a novel 1D CNN architecture integrated with a denoising autoencoder. They trained their model on a combination of three datasets: ICBHI 2017, KAUH dataset, and some self-collected samples, making this approach an eight-class solution. They extracted Mel Spectrograms for training and implemented an autoencoder to clear the noise from the spectrograms. Their model consists of five convolutional layers, each followed by a MaxPooling1D layer. They have achieved an accuracy of 92.7% on the combined dataset, with 99.9% on the ICBHI data, 99.85% on the KAUH data, and 95.5% on self-collected samples.

Although numerous studies have contributed to advancements in automated lung sound analysis, many continue to rely on similar experimental setups and well-structured datasets. The existing research emphasizes spectrogram-based features combined with convolutional or hybrid neural network architectures, while relatively little focus is given to alternative feature extraction and sequential modeling techniques. In this study, we aim to bridge these gaps by utilizing Mel-Frequency Cepstral Coefficients (MFCCs) instead of spectrograms, enabling a more effective representation of the spectral and temporal characteristics of respiratory signals [14]. In addition to that, instead of using convolutional or hybrid deep learning frameworks that are common in the literature, we focus on RNNs. RNNs are best suited for sequential data and the temporal relationships found in lung sound recordings [6]. We conducted numerous experiments on different model architectures and included data variation strategies, such as segmentation and the use of fully padded sequences, with the use of the masking layer to handle these sequences efficiently. Through these contributions, our study broadens the scope of existing research by investigating alternative yet effective design strategies for automated lung disease detection.

Research Methodology:

The research methodology followed in this research has been outlined in Figure 1.

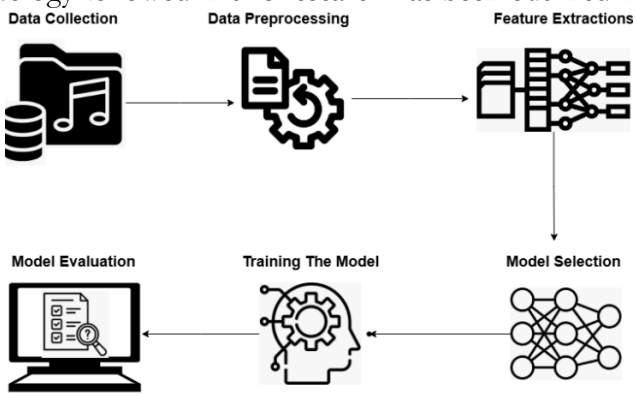

Figure 1. Research Methodology

Data Collection: Our model was trained using the publicly available ICBHI dataset [15], which included 920 annotated respiratory sound recordings obtained from 126 patients. The recordings were captured using various types of stethoscopes, with durations ranging from 10 to 90 seconds. In total, the dataset contains approximately 5.5 hours of audio, including samples with crackles, wheezes, and combinations of both, as well as recordings without any adventitious respiratory sounds. The dataset includes recordings from participants across various age groups, including children, adults, and the elderly, with adults and older individuals

contributing the majority of samples. It comprises audio data from seven pulmonary conditions (COPD, Pneumonia, Bronchiolitis, Asthma, Bronchiectasis, LRTI, and URTI) as well as from healthy subjects, resulting in a total of eight classes. Each patient is assigned a diagnostic label in a corresponding .csv file. Figure 2 illustrates the statistics for the number of audio recordings across the different classes in the dataset.

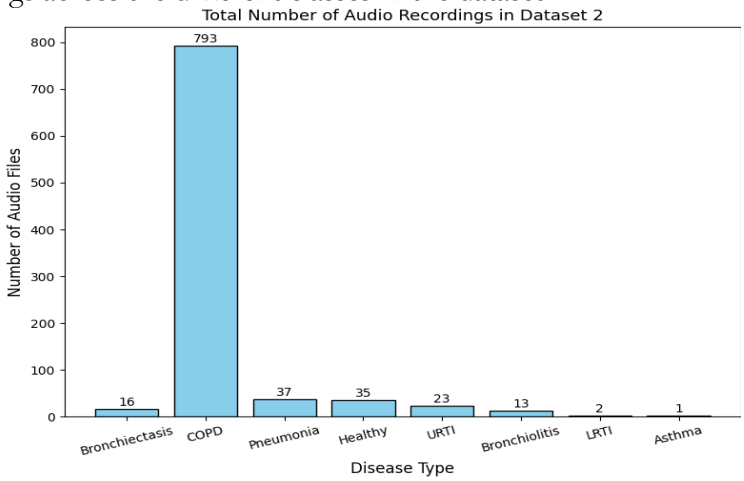

Figure 2. Total number of audios

Data Pre-processing: During the exploratory analysis of the dataset, we observed that it was not suitable for training in its original form. It was found that although the dataset contained 920 recordings, the class distribution was highly imbalanced, with categories such as Asthma, LRTI, URTI, and Bronchiolitis having only a limited number of samples. This difference can affect the performance of deep learning models through underfitting and overfitting [16]. Overfitting occurs when a model memorizes the majority class patterns and is unable to find the logic behind the prediction. The model also struggles to capture the underlying patterns associated with minority classes, such as rare disease cases, which ultimately results in poor generalization performance [17]. In underfitting, the model is unable to learn patterns in the dataset for the class distribution, leading to errors in disease detection. This is particularly important in healthcare, where minority classes may correspond to rare but clinically significant conditions. Several methods have been employed in the literature to address these challenges, such as resampling and data augmentation.

We excluded Asthma, LRTI, URTI, and Bronchiolitis to provide our model with a steadier learning process. This decision removed the severe shortage of instances, although there was still quite a noticeable class imbalance in the dataset, for which preprocessing before model training was warranted. To address class imbalance, only four recordings per patient were retained to undersample the COPD category. Data augmentation techniques such as time shifting, time stretching, and random noise injection were utilized to synthesize the minority class samples.

Random Noise: Adding random noise or Gaussian noise refers to adding low-level white noise to audio signals to increase the variability of the data [18]. In this work, we generated the Gaussian Noise by using NumPy in Python and mixed it with the original audio data. The augmented audio signal would therefore contain the original data with a kind of faint Gaussian noise.

Time Shift: Using this technique, we shifted the audio signal to the right, along the time axis, but kept the content and its duration intact [19]. This aims at emulating the temporal variations and delays that occur during the capture of real-world audio. We implement this in Python using the roll function from NumPy.

Time Stretch: Time stretch augmentation is used to change the speed of the lung sound recordings without by having them modify their pitch [20]. This technique introduces the

model to changes in a person's breathing rate and helps the model generalize better. We used the Python time_stretch function from the librosa.effects module.

In this variation, the preprocessing was performed using the complete audio recordings available in the dataset. However, an alternative approach was also explored, in which the original dataset was segmented into smaller audio clips to increase the number of samples. Segmentation is the process of dividing continuous lung auscultation recordings into shorter audio segments, each containing at least one complete respiratory cycle [21]. From this approach, two new datasets were created, one with segments of 10 seconds and the other with segments of 20 seconds, along with the previously mentioned augmentation techniques to further expand the dataset and improve diversity. For the original dataset or dataset 1, each audio recording was converted to a 40-dimensional MFCC feature sequence. The recordings maintained their full duration. The sequences were then padded to a fixed maximum length to standardize the input shape across dataset 1. In dataset 2, the audio recordings were divided into non-overlapping, 10-second segments. MFCC features were extracted from each segment in an attempt to enable the network to learn from shorter, consistently sized samples. Finally, in dataset 3, longer 20-second segments were used to determine if extended temporal context improves performance. Figure 3 illustrates the effect of preprocessing on all three dataset variations. The wave plots for each disease label and healthy lung sounds are shown in Figure 4. From Figure 4, we can observe the distinct patterns of how each disease label affects lung sound characteristics. The healthy waveform is smooth with smaller oscillations when compared to pneumonia, which has frequent and intense fluctuations. In contrast, COPD has less intense fluctuations but still irregularities. Bronchiectasis has significant oscillations, but the intensity is lower than pneumonia.

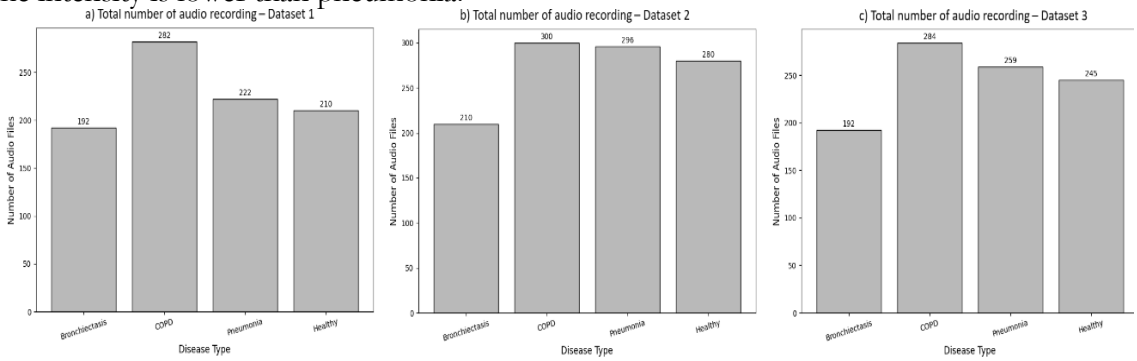

Figure 3. Number of Recordings in the Datasets after Preprocessing

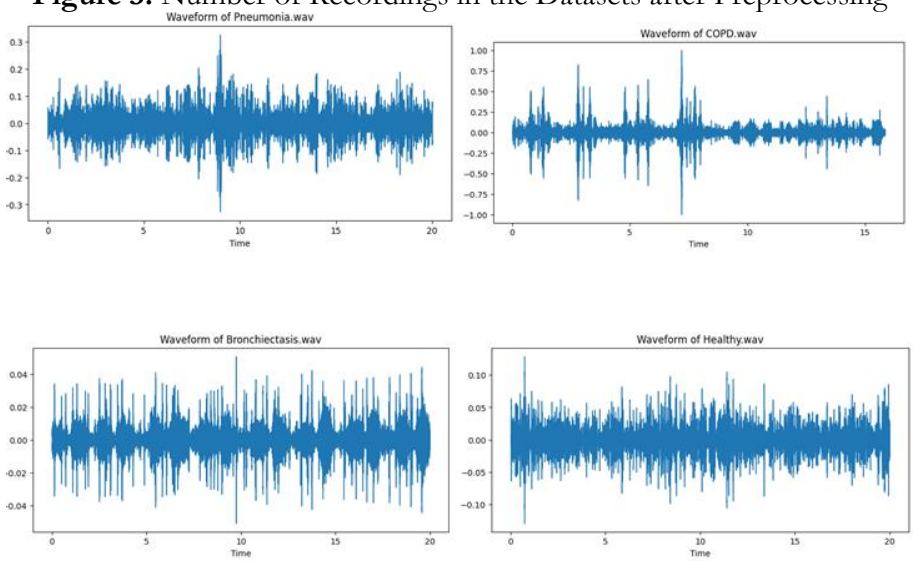

Figure 4. Wave Plots for each Disease Label

Feature Extraction:

To extract meaningful patterns from respiratory audio signals, we obtained 40 Mel Frequency Cepstral Coefficients (MFCCs) from each audio file. MFCCs effectively capture unique sound qualities in each recording, such as pitch, tone, and rhythm [14]. We retained the entire sequence instead of averaging the time-variant MFCCs, as preserving temporal information is crucial for accurately recognizing breathing patterns. The duration of raw audio files ranged from 10 to 90 seconds, resulting in MFCC arrays with different time steps. In order to resolve this anomaly and prepare a proper 3D input for the model, we employed the Pad_Sequence function [22]. This technique ensures that all audio samples have an equal number of time steps and that during training, the padded values are ignored. In this case, each audio file had a category label, such as COPD, Pneumonia, Bronchiectasis, or Healthy. These string labels were changed to binary one-hot encoding vectors, in which the disease is represented as a vector with a single '1' indicating the true class and '0's elsewhere.

Model Architecture:

RNNs are considered deep learning models that have been developed to deal with sequential data. These are effective in modeling temporal dependencies, usually involving time-series or audio signals [6]. In contrast to conventional neural networks, RNNs are designed with recurrent connections that enable them to remember information from earlier inputs, allowing the model to recognize temporal dependencies. When applied to respiratory audio, RNNs can learn the timing and duration of characteristic sounds such as wheezes and crackles, which are vital for identifying respiratory disorders. This capacity to interpret sequential data makes RNNs a powerful tool for analyzing and classifying health-related audio signals.

For our deep neural network architecture, we chose the following five layers: the masking layer, GRU layer, Leaky ReLU layer, Dense layers, and Dropout layers.

Masking Layer: Masking Layer is responsible for ignoring the padded values in audio sequences [23]. This helps the model not to be confused by artificially added silence, but rather focus its attention on real data.

Gated Recurrent Unit: GRU is a type of RNN that is used for processing sequential data such as audio [24]. It can remember useful information over time, which helps in learning temporal patterns.

Leaky Rectified Linear Unit: This activation function enables a small gradient in the case of negative input for resolving the dead neuron problem. The goal here is to enhance model learning capability through the addition of Leaky ReLU after each GRU and Dense layer [25].

Dense Layer: This layer connects all neurons from the previous layer to the next layer [26]. It is also used as the final layer with a softmax activation function to change the raw outputs into a probability distribution, allowing the model to select the most probable class as the prediction.

Dropout Layer: Dropout is used to prevent overfitting by dropping off a percentage of neurons randomly during training [27]. We used a dropout rate of 0.5.

To analyze the impact of architectural complexity on performance, we implemented two different variations of the proposed GRU-based model. The first variation, Model Architecture 1, features a more complex design consisting of six stacked GRU layers, each followed by a Leaky ReLU activation function. Additionally, a dropout layer is incorporated to prevent overfitting and enhance model generalization. Model Architecture 2 was designed as a comparatively simpler structure with three GRUs and leaky ReLU activation layers, and the application of a dropout layer at the end. The model summary for Model Architecture 2, which demonstrated the best performance among all configurations, is presented in Table 1.

Table 1. Details of Model Architecture-2

Layer (type)	Output Shape	Param#	Connected to
input_layer (InputLayer)	(None, None, 40)	0	-
not_equal_1 (NotEqual)	(None, None, 40)	0	input_layer[0][0]
masking_1 (Masking)	(None, None, 40)	0	input_layer[0][0]
any_1 (Any)	(None, None)	0	not_equal_1[0][0]
gru_5(GRU)	(None, None, 128)	65,280	masking_1[0][0], any_1[0][0]
leaky_re_lu_5 (LeakyReLU)	(None, None, 128)	0	gru_5[0][0]
gru_6 (GRU)	(None, None, 64)	37,248	leaky_re_lu_5[0][0], any_1[0][0]
leaky_re_lu_6 (LeakyReLU)	(None, None, 64)	0	gru_6[0][0]
gru_7 (GRU)	(None, 32)	9,408	leaky_re_lu_6[0][0], any_1[0][0]
dense_2 (Dense)	(None, 64)	2,112	leakygru_7[0][0]
leaky_re_lu_7 (LeakyReLU)	(None, 64)	0	dense_2[0][0]
dropout_2 (Dropout)	(None, 64)	0	leaky_re_lu_7[0][0]
dense_3 (Dense)	(None, 40)	260	dropout_2[0][0]
Total params: 342,926 (1.31 MB) Trainable params: 114,308 (446.52 KB) Non-trainable params: 0 (0.00 B) Optimizer params: 228,618 (893.04 KB)			

Model Training:

To assess the impact of the architectural design and pre-processing techniques applied thus far, a total of six experiments were carried out. These experiments integrated two different model architectures with three different pre-processing strategies, providing a foundation for comparison.

The dataset was pre-processed into three different input configurations to investigate how the segmentation and duration affect learning dynamics. The two model architectures we experimented with are as follows:

Model Architecture 1: The first configuration increased the network depth by adding additional GRU layers, thus creating a more complex model to test if greater representational Capacity might improve performance.

Model Architecture 2: The second model configuration involved three GRU layers, along with LeakyRelu, dense, and dropout layers were used in the outperforming model. This architecture represents a well-balanced design that effectively captures temporal dependencies while minimizing unnecessary complexity. It achieved the highest overall accuracy across all dataset variations. Figure 5 illustrates the workflow of the proposed RNN model.

For each experiment, we split the dataset into a testing and a training set with a 30/70 ratio, with 50% of the test set reserved as a validation set. The training of the model was performed with 32 batch size and 200 epochs. The model utilizes MFCC features derived from respiratory audio signals as its input. These features are first fed into a Masking layer to ignore padded values during the learning process. Then the masked input is processed through a deep

sequence of GRU layers, each followed by a Leaky ReLU activation function to improve learning and gradient flow. The resulting features are then passed through a Dense layer to reduce dimensionality, followed by a Dropout layer with a rate of 0.5 to reduce overfitting. Finally, a Dense layer with softmax activation and five neurons produces the probability distribution across four classes related to respiratory diseases, thus enabling accurate classification.

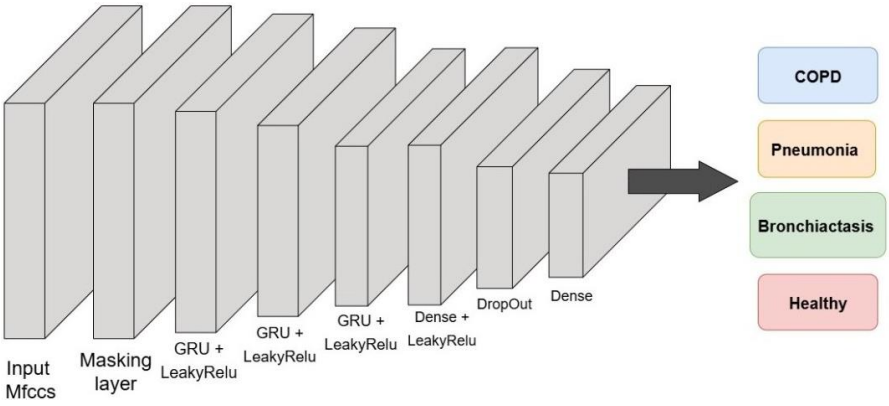

Figure 5. Flow of the proposed RNN architecture

Model Evaluation:

The experiments were run using Google Colab [28] with GPU acceleration, allowing for efficient training over 200 epochs, and evaluated using the following metrics:

Accuracy: Accuracy is the proportion of correct classifications out of the total made classifications [29] and is represented through eq. 1).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad 1)$$

Precision: Precision is a way of measuring and analyzing the model's positive classifications. It is measured using the equation. 2).

$$Precision = \frac{TP}{TP+FP} \quad 2)$$

Recall: Recall is the proportion of all actual positives that were classified correctly as positives [29]. Recall can be measured using Eq. 3.

$$Recall = \frac{FP}{TP+FN} \quad 3)$$

F1-Score: F1-Score is the harmonic mean of precision and recall, useful for imbalanced data [29]. It is measured using Eq. 4.

$$F1 - Score = 2 * \frac{Precision*Recall}{Precision+Recall} \quad 4)$$

Results and Discussion:

Experiment 01:

In this experiment, Model Architecture 1 was trained on Dataset variation 1, having full sequences of audio recordings. It achieved an accuracy of 95.59%, with a precision of 95.70%, a recall of 95.59%, and an F1-score of 95.58%, demonstrating strong performance in the experiment. The class-wise performance report of the model is presented in Table 2.

Table 2. Class-wise report of the first experiment.

Class	Precision	Recall	F1-Score
Bronchiectasis	1.00	1.00	1.00
COPD	0.97	0.90	0.94
Healthy	0.94	0.97	0.95
Pneumonia	0.91	0.97	0.94

Table 2 illustrates that bronchiectasis achieved perfect precision and recall. This is likely due to a smaller, less diverse subset of samples, leading to possible overfitting on that

class. COPD showed lower recall (0.90) compared to its precision (0.97), suggesting that the model misclassified several COPD instances as other diseases. Healthy and Pneumonia classes showed relatively balanced results, though the minor gap between their precision and recall indicates room for improvement.

Figure 6 shows the training and validation accuracy and loss curves. The training loss showed a steady decline, whereas the validation loss remained consistently high, indicating that the model suffered from overfitting. Overall, the model gave better classification results but shows a clear gap between training and validation loss, indicating poor performance.

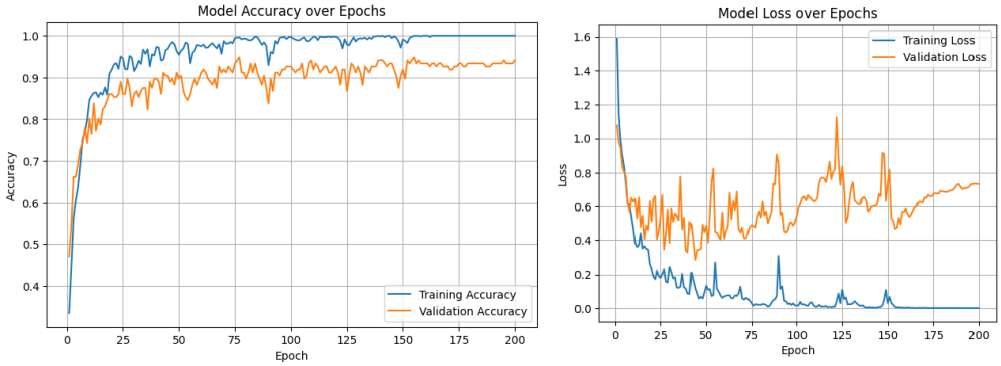

Figure 6. Training and validation accuracy and loss graph for Experiment 01

Experiment 02:

In this experiment, Model Architecture 1 was trained using Dataset Variation 2, which consisted of 10-second segmented audio recordings. The model achieved an overall accuracy of 95.51%, with a precision of 95.61%, a recall of 95.51%, and an F1-score of 95.52%, which are comparable to the first experiment's results. Class-wise report for this model is presented in Table 3.

Table 3. Class-wise report of the second experiment.

Class	Precision	Recall	F1-Score
Bronchiectasis	1.00	0.96	0.98
COPD	0.93	0.96	0.95
Healthy	0.93	0.98	0.95
Pneumonia	0.98	0.93	0.95

The class-wise report indicates that the Bronchiectasis class achieved perfect precision with some errors in recall, indicating the model's strong confidence in predictions but occasional failure to detect all instances. The COPD and Healthy classes showed improved recall (0.96 and 0.98, respectively) compared to Experiment 1, suggesting that segmentation may have helped the model recognize a wider range of acoustic variations for these conditions. However, Pneumonia displayed the opposite trend - precision improved (0.98) but recall dropped (0.93), implying that although predictions for pneumonia were more accurate, some positive samples were still missed.

The training and validation accuracy and loss curves are presented in Figure 7. Training accuracy improved steadily, while validation accuracy leveled off at a slightly lower value. Training loss decreased to about 8.4%, but validation loss settled at a higher value of about 35.6%, indicating mild overfitting. Overall, there is a very slight change in model accuracy compared to experiment 1, but the overfitting ratio visible in the loss graph has improved.

Experiment 03:

In the third experiment, Dataset 3 was used to train Model Architecture 1, and the overall results achieved were an accuracy of 93%, with a precision of 0.94, a recall of 0.93, and an F1-score of 0.93, showing a slight decline in numbers as compared to the previous experiments. Table 4 presents the class-wise report for this experiment.

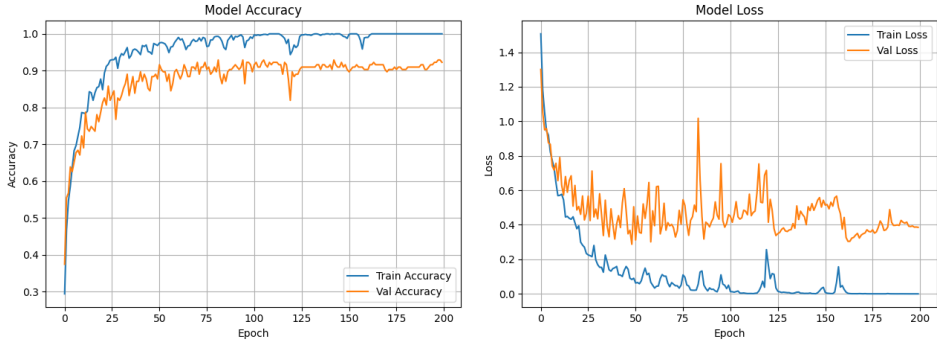

Figure 7. Experiment 02 training and validation accuracy and loss graph

Table 4. Class-wise report of the third experiment

Class	Precision	Recall	F1-Score
Bronchiectasis	0.97	1.00	0.98
COPD	1.00	0.88	0.94
Healthy	0.93	0.89	0.91
Pneumonia	0.98	1.00	0.93

The class-wise performance indicates mixed results. Bronchiectasis maintained strong results, reflecting consistent identification for this class. However, COPD exhibited an imbalance between precision (1.00) and recall (0.88), suggesting that the model failed to detect several true COPD samples. The Healthy class and Pneumonia also showed reduced performance as compared to previous approaches. The training and validation graphs are shown in Figure 8. The training accuracy increased steadily, while the validation accuracy plateaued at a lower level. Similarly, the training loss continued to decrease, but the validation loss remained relatively high. This indicates a consistent pattern of overfitting, as the gap between the training and validation curves appears at regular intervals.

Figure 8. Training and validation accuracy and loss graph - Experiment 03

Experiment 04:

After the previous three experiments, we shifted to Model Architecture 2 to analyze the behaviour. First, the Dataset variation 3 was used to train the new model arrangement. The model achieved an overall accuracy of 94.56%, precision of 94.82%, recall of 94.56%, and an F1-score of 94.52%, which shows a slight increase in performance as compared to Experiment 3, where the same Dataset 3 was used. The class-wise report is shown in Table 5.

Table 5. Class-wise report of the fourth experiment.

Class	Precision	Recall	F1-Score
Bronchiectasis	0.94	1.00	0.97
COPD	1.00	0.91	0.95
Healthy	0.94	0.89	0.91
Pneumonia	0.91	1.00	0.95

The class-wise evaluation also reflects consistent behaviour. Bronchiectasis exhibited perfect recall but a lower precision (0.94), while COPD and Pneumonia maintained balanced

precision–recall relationships. The Healthy class, however, showed lower recall (0.89), implying occasional misclassification as abnormal lung sounds. Although the numerical results are comparable to those obtained from the complex model, the training and validation curves in Figure 9 showed a great improvement with reduced gaps between training and validation for both loss and accuracy, indicating a better approach to handle overfitting.

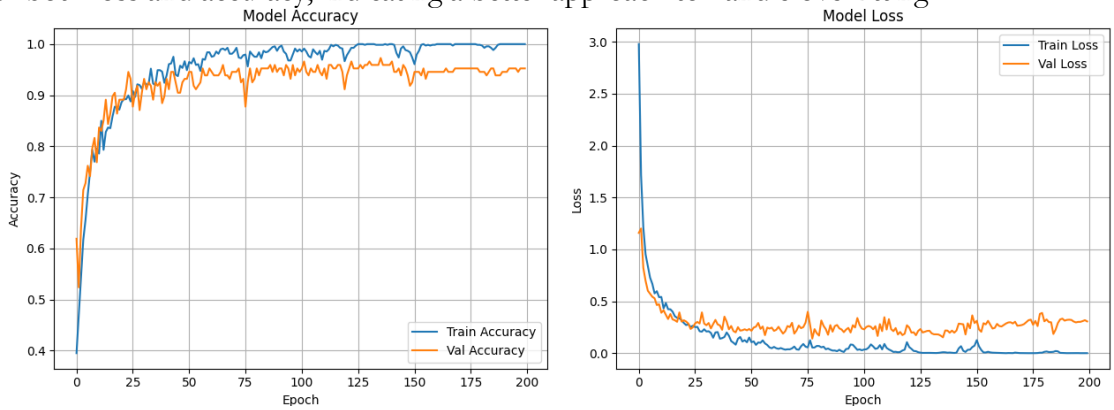

Figure 9. Training and validation accuracy and loss curve - Experiment 04

Experiment 05:

After an improved behaviour in experiment 4, Dataset 2 was used to train Model Architecture 2. The model got an overall accuracy of 0.91, with 0.91 precision, a recall of 0.91, and an F1-score of 0.91, marking a noticeable decline compared to previous experiments. These observations, along with results from other approaches using segmented data, suggest that segmentation may have led to the loss of critical information necessary for accurate classification. Class-wise report also presents similar behaviour in Table 6.

Table 6. Class-wise report of the fifth experiment.

Class	Precision	Recall	F1-Score
Bronchiectasis	0.89	1.00	0.94
COPD	0.95	0.91	0.93
Healthy	0.90	0.88	0.89
Pneumonia	0.91	0.91	0.91

Figure 10 shows the accuracy and loss curves. A similar pattern of reduced overfitting is observed as in experiment 4. This shows that the model is dealing well with overfitting. However, at some final epochs, an increasing trend of the gap can be seen for both loss and accuracy between training and validation. Overall, in comparison to experiment 2, this model has shown great improvement in accuracy and loss graphs.

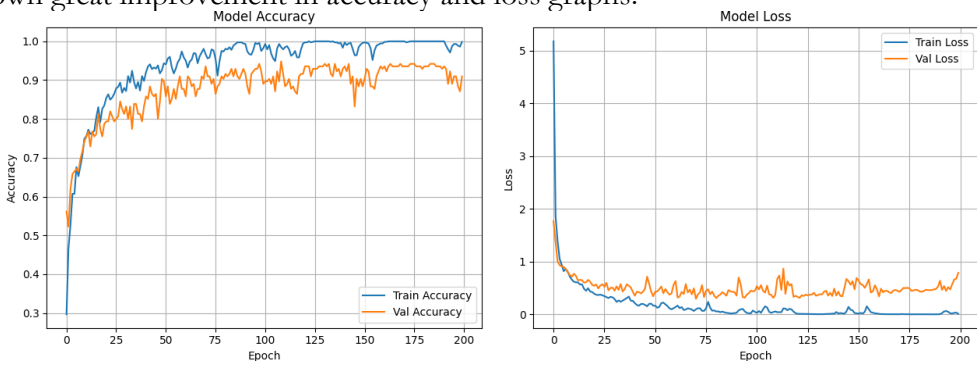

Figure 10. Experiment 05: training and validation accuracy and loss graph

Experiment 06:

For this final experiment, Model Architecture 2 was trained on Dataset 1, the full-length sequences. This configuration produced the best overall performance among all experiments, achieving an accuracy of 98.53%, precision of 98.57%, recall of 98.53%, and an

F1-score of 98.52%. The class-wise results shown in Table 7 further highlight the effectiveness of this configuration. Bronchiectasis achieved perfect scores across all metrics (1.00 each), while COPD, Healthy, and Pneumonia also showed near-perfect numbers, reflecting the model's strong discriminative power across all respiratory sound categories.

Table 7. Class-wise report of the sixth experiment.

Class	Precision	Recall	F1-Score
Bronchiectasis	1.00	1.00	1.00
COPD	1.00	0.95	0.98
Healthy	0.97	1.00	0.98
Pneumonia	0.97	1.00	0.99

The accuracy and loss graphs as presented in Figure 11 showed consistent results, with training and validation accuracy being close and a little difference in loss, showing good generalization. This approach not only improved in evaluation parameters but also presented a great difference in model loss graphs, indicating that full sequences of audio provided a greater chance to learn features from auscultation recordings.

Figure 11. Training and validation accuracy and loss for Experiment 06

Overall, the six experiments performed on different dataset variations and model architectures provided a chance to analyze different behaviours on evaluation parameters along with generalization ability from Model loss and accuracy graphs. The results of the sixth approach suggest that the model is capturing fine-grained temporal features while avoiding overfitting, making it the best approach with an accuracy of 98.53%.

Discussion:

The results of all experimental configurations undertaken in this research are summarized in Table 8. We explored different dataset variations and model architectures to assess their effect on performance. From these results, it is evident that complex model architectures tend to yield lower accuracy, while simpler architectures achieve higher accuracy and more stable convergence. This suggests that the effectiveness of a model depends not only on its architecture but also on the size and quality of the dataset. When the amount of training data is limited, a simpler design helps to reduce overfitting and improve generalization.

Table 8. Summary of Model Performance Across Different Configurations

Experiments	Preprocessing	Model Architecture	Accuracy	Training/validation loss curves
Experiment 01	Full audio sequences	5 GRU layers	95.59%	Poor convergence
Experiment 02	10 sec segmentation	5 GRU layers	95.51%	Poor convergence
Experiment 03	20 sec segmentation	5 GRU layers	93%	Poor convergence
Experiment 04	20 sec segmentation	3 GRU layers	94.56%	Good convergence
Experiment 05	10 sec segmentation	3 GRU layers	91%	Good convergence
Experiment 06	Full audio sequences	**3 GRU layers**	**98.53%**	**Good convergence**

Table 9 compares the performance of the model proposed in this research with other existing studies using the ICBHI 2017 dataset. The results show that our RNN-based model

performs as well as, or even better than, several existing advanced methods. For instance, Tariq et al. [9] achieved 97% accuracy with a 2D CNN, while Zhang et al. [12] reported 98.82% accuracy using an LSTM-based model. Our model achieved 98.53% accuracy, outperforming other deep learning approaches proposed in [10], which achieved an accuracy of 95.67% and [11], which achieved an accuracy of 76.39%, while maintaining a simpler structure and requiring fewer computational resources.

Table 9. Comparison of the Proposed RNN-based GRU Model with Other Research Works

Ref#	Preprocessing Techniques	Feature Extraction	Model Architecture	Accuracy (%)
[9]	Normalization, Augmentation (time stretch, pitch shift, dynamic range compression)	Mel Spectrograms	Custom 2D CNN (3 Conv + 2 FC layers, ReLU + Softmax)	97%
[10]	Undersampling and Augmentation for minority classes	Mel Frequency Cepstral Coefficients	GRU + Leaky ReLU + Dense + Dropout + Add layer	95.67%
[11]	Focal Loss to handle imbalance	Spectrograms	Hybrid CNN-LSTM	76.39%
[12]	Imbalance learn toolbox to deal with class imbalance	MFCCs, Chroma, Tonal Centroids, Mel Spectrogram, Spectral Contrast	LSTM	98.82%
[13]	Normalization, Augmentation, Denoising Autoencoder	Mel Spectrograms	1D CNN + Autoencoder	99.9%
Current Research	**Undersampling, Augmentation (Gaussian noise, time-shift, time-stretch), Padded sequences**	**Mel Frequency Cepstral Coefficients**	**Masking Layer, GRU, Leaky ReLU, Dense, Dropout**	**98.53%**

All studies compared in Table 9 utilized the ICBHI 2017 dataset [8] and relied on Mel Spectrogram-based feature extraction. In contrast, we used MFCCs because they effectively capture distinct sound features such as pitch, tone, and rhythm [14]. Additionally, rather than adopting complex or hybrid architectures, we implemented a GRU-based RNN model, which provided an effective balance between simplicity, efficiency, and accuracy. These findings confirm that robust preprocessing, MFCC feature extraction, and a lightweight GRU architecture together can achieve strong performance in pulmonary disease classification tasks using respiratory sound data.

Conclusion: In this research, an RNN-based approach was developed to classify auscultation sounds into four categories. The proposed model demonstrated an effective balance between accuracy and generalization, achieving a notable accuracy of 98.53% using the full-sequence audio data variation. Our findings highlight that accuracy alone cannot fully represent model performance. Other factors, such as stability and generalization capability, are equally critical when working with limited medical datasets. That is why model loss and accuracy curves were also analyzed to assess training behaviour and consistency to select an approach that performed optimally in both aspects.

In the future, we plan to expand this work by collecting auscultation recordings from nearby hospitals. This will further evaluate and enhance the model's generalizability in practical clinical settings. Ultimately, with this research, we aim to contribute towards reducing diagnostic costs and time, enabling timely interventions, and improving patient care through reliable, AI-assisted pulmonary disease detection.

Author's Contribution: Yusra Shaikh: Writing - original draft, methodology, experimentation, visualization, literature review, result reporting. Areej Fatemah Meghji: methodology, writing - editing and review, validation, and result reporting. Zobiya Jumani: methodology, writing - original draft, validation, result reporting. Mehak Jatoi: methodology, writing - original draft, experimentation, validation, result reporting.

Conflict of Interest: The authors declare they have no conflict of interest in publishing this manuscript in IJIST.

References:

[1] "Lung disease: MedlinePlus Medical Encyclopedia." Accessed: Nov. 01, 2025. [Online]. Available: https://medlineplus.gov/ency/article/000066.htm

[2] "Chronic respiratory disease is third leading cause of death globally with air pollution killing 1.3 million people | Institute for Health Metrics and Evaluation." Accessed: Nov. 01, 2025. [Online]. Available: https://www.healthdata.org/news-events/newsroom/news-releases/chronic-respiratory-disease-third-leading-cause-death-globally

[3] "The State of the World's Children 2017: Statistical tables - UNICEF DATA." Accessed: Nov. 01, 2025. [Online]. Available: https://data.unicef.org/resources/state-worlds-children-2017-statistical-tables/

[4] K. Q. Dong-Min Huang, Jia Huang, "Deep learning-based lung sound analysis for intelligent stethoscope," *Mil. Med. Res.*, vol. 10, no. 44, 2023, doi: https://doi.org/10.1186/s40779-023-00479-3.

[5] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nat. 2015 5217553*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[6] S. S. K. & I. B. Susmita Das, Amara Tariq, Thiago Santos, "Recurrent Neural Networks (RNNs): Architectures, Training Tricks, and Introduction to Influential Research," *Mach. Learn. Brain Disord.*, pp. 117–138, 2023, doi: https://doi.org/10.1007/978-1-0716-3195-9_4.

[7] "Classification in Machine Learning: A Guide for Beginners | DataCamp." Accessed: Nov. 01, 2025. [Online]. Available: https://www.datacamp.com/blog/classification-machine-learning

[8] "Respiratory Sound Database." Accessed: Nov. 01, 2025. [Online]. Available: https://www.kaggle.com/datasets/vbookshelf/respiratory-sound-database

[9] Z. Tariq, S. K. Shah, and Y. Lee, "Lung Disease Classification using Deep Convolutional Neural Network," *Proc. - 2019 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2019*, pp. 732–735, Nov. 2019, doi: 10.1109/BIBM47256.2019.8983071.

[10] V. Basu and S. Rana, "Respiratory diseases recognition through respiratory sound with the help of deep neural network," *4th Int. Conf. Comput. Intell. Networks, CINE 2020*, Feb. 2020, doi: 10.1109/CINE48825.2020.234388.

[11] G. A. C. Georgios Petmezas, "Automated Lung Sound Classification Using a Hybrid CNN-LSTM Network and Focal Loss Function," *Sensors*, vol. 22, no. 3, p. 1232, 2022, doi: https://doi.org/10.3390/s22031232.

[12] A. S. Pinzhi Zhang, "Pulmonary disease detection and classification in patient respiratory audio files using long short-term memory neural networks," *Front. Med.*, vol. 10, 2023, doi: https://doi.org/10.3389/fmed.2023.1269784.

[13] M. H. Nawaz, J. Ahmad, M. Haroon, M. Haseeb, and A. Salman, "Real-Time Deep

Learning for Lung Disease Classification: A Step Forward," *2024 Int. Conf. Front. Inf. Technol. FIT 2024*, 2024, doi: 10.1109/FIT63703.2024.10838437.

[14] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *J. Acoust. Soc. Am.*, vol. 87, no. 4, pp. 1738–1752, Apr. 1990, doi: 10.1121/1.399423.

[15] B. M. Rocha *et al.*, "A Respiratory Sound Database for the Development of Automated Classification," *IFMBE Proc.*, vol. 66, pp. 33–37, 2018, doi: 10.1007/978-981-10-7419-6_6.

[16] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009, doi: 10.1109/TKDE.2008.239.

[17] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2011, doi: 10.1613/jair.953.

[18] A. R.-C. Dulva Hina, "Impact Evaluation of Sound Dataset Augmentation and Synthetic Generation upon Classification Accuracy," *J. Sens. Actuator Netw*, vol. 14, no. 5, p. 91, 2025, doi: https://doi.org/10.3390/jsan14050091.

[19] J. Y. Qiurui Sun, "Advances and Challenges in Respiratory Sound Analysis: A Technique Review Based on the ICBHI2017 Database," *Preprints*, 2025, [Online]. Available: https://www.preprints.org/manuscript/202506.1527

[20] "Time Stretching And Pitch Shifting of Audio Signals – An Overview | Stephan Bernsee's Blog." Accessed: Nov. 04, 2025. [Online]. Available: https://blogs.zynaptiq.com/bernsee/time-pitch-overview/

[21] R. B. Constantin Constantinescu, "Lung Sounds Anomaly Detection with Respiratory Cycle Segmentation," *BRAIN. Broad Res. Artif. Intell. Neurosci.*, vol. 15, no. 3, pp. 188–196, 2024, [Online]. Available: https://brain.edusoft.ro/index.php/brain/article/view/1597

[22] "Timeseries data loading." Accessed: Nov. 01, 2025. [Online]. Available: https://keras.io/api/data_loading/timeseries/

[23] "Masking layer." Accessed: Nov. 01, 2025. [Online]. Available: https://keras.io/api/layers/core_layers/masking/

[24] E. H. I. E. Amr Mohamed El Koshiry, "Detecting cyberbullying using deep learning techniques: a pre-trained glove and focal loss technique," *PeerJ Comput. Sci.*, vol. 10, p. e1961, 2024, [Online]. Available: https://peerj.com/articles/cs-1961/#fig-6

[25] A. K. Dubey and V. Jain, "Comparative Study of Convolution Neural Network's Relu and Leaky-Relu Activation Functions," *Lect. Notes Electr. Eng.*, vol. 553, pp. 873–880, 2019, doi: 10.1007/978-981-13-6772-4_76.

[26] A. Gulli, A. Kapoor, S. Pal, O'Reilly for Higher Education (Firm), and an O. M. C. Safari, "Deep Learning with TensorFlow 2 and Keras - Second Edition," p. 646.

[27] S. V. Alex Labach, Hojjat Salehinejad, "Survey of Dropout Methods for Deep Neural Networks," *arXiv:1904.13310*, vol. 4, 2019, doi: https://doi.org/10.48550/arXiv.1904.13310.

[28] Google Colab, "Google Colaboratory: Python in the Cloud", [Online]. Available: https://colab.research.google.com

[29] M. O. Raza, A. F. Meghji, N. A. Mahoto, M. S. Al Reshan, M. S. Abosaq, H. A. Sulaiman, A. Shaikh "Reading Between the Lines: Machine Learning Ensemble and Deep Learning for Implied Threat Detection in Textual Data," *Int. J. Comput. Intell. Syst.*, vol. 17, no. 183, 2024, doi: https://doi.org/10.1007/s44196-024-00580-y.