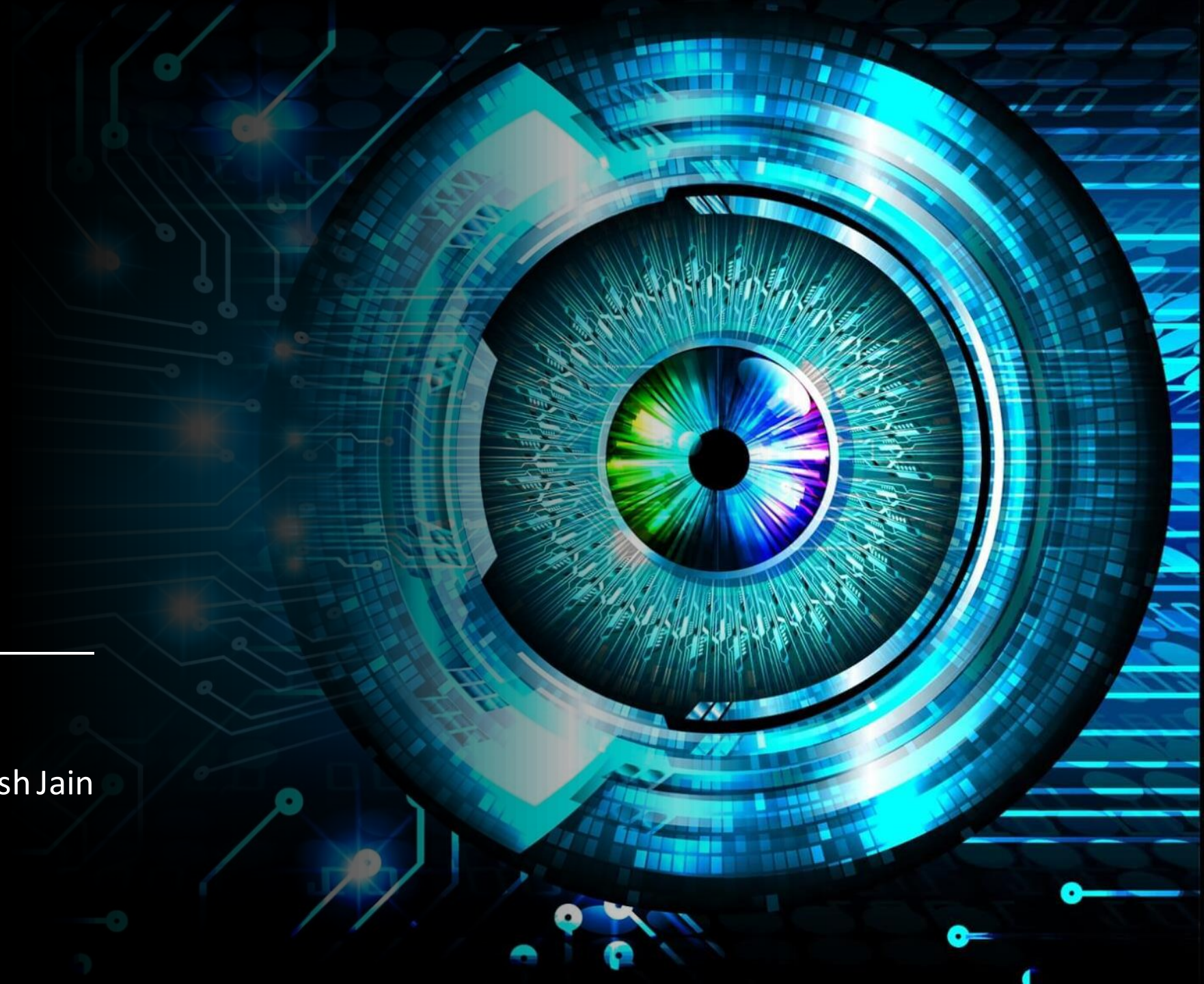
Mosaic PS-1

By Team DMD

Dhruv Sharma, Mitra P. ,Divyansh Jain

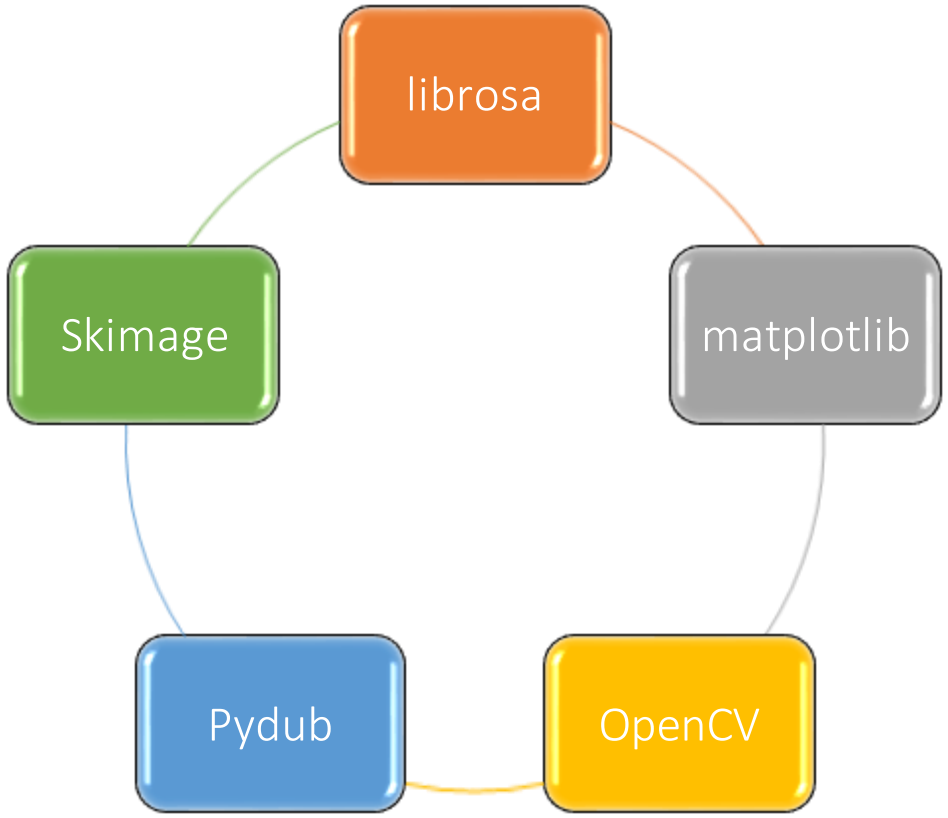


What We Know and Have:

Scientists are currently seeking to develop a computer vision-based solution for accurately diagnosing lung conditions, particularly wheezes and crackles, in patients.

Our task is to predict the presence or absence of crackles and wheezes based on audio and text files that describe the conditions of patients.

Pre-processing the data



Librosa

- Loading and processing audio files

matplotlib

- Plotting the spectrograms

Opencv

- Manipulating the images

Pydub

- Splitting the audio files based on the breathing cycles

Skimage

- Storing the spectrograms obtained to folder

Creating dataset

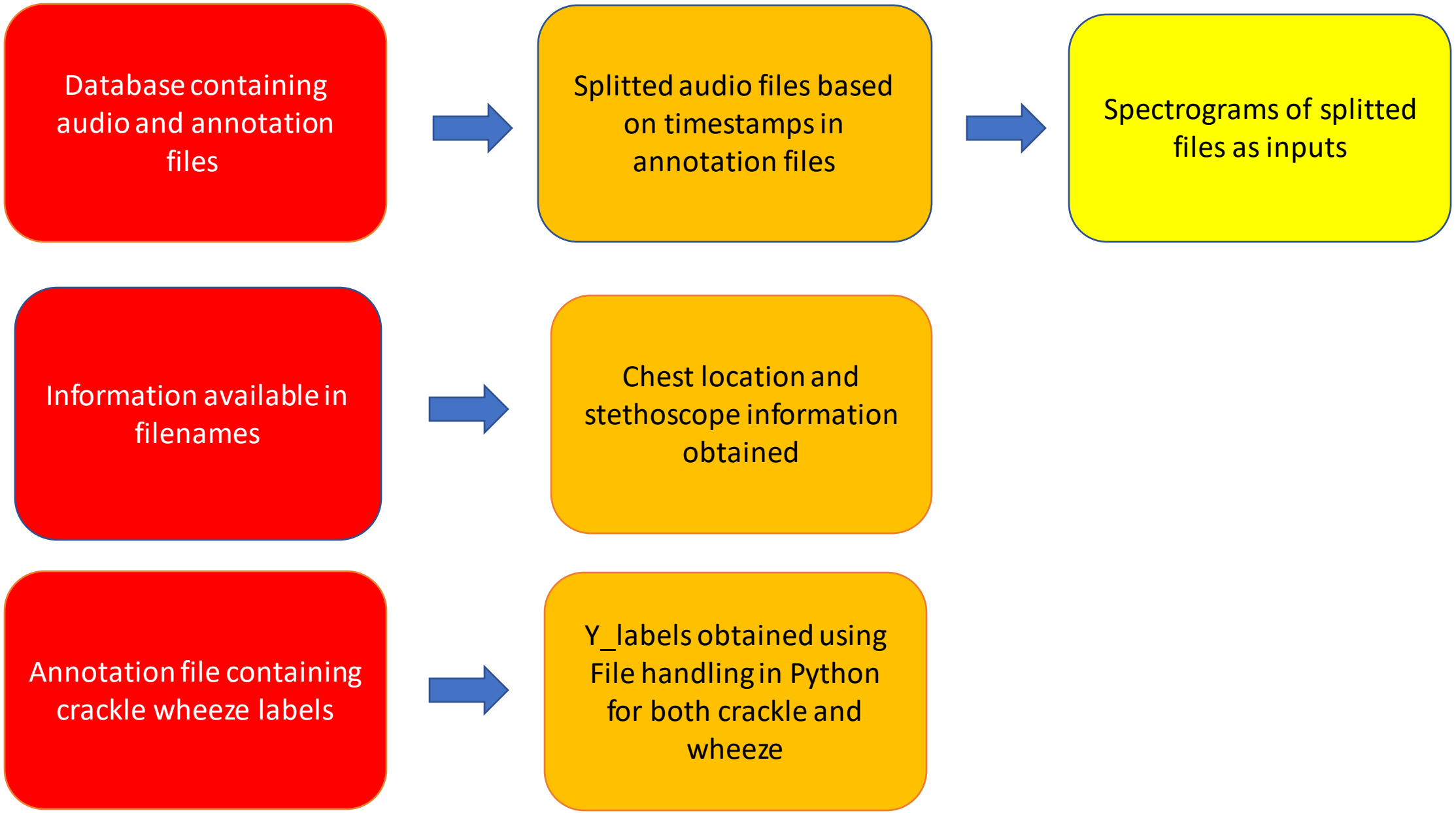
In the given database, we had .wav audio files and .txt annotation files. Now, to create the dataset, we first split the audio files based on the breathing cycle timestamps we have in annotations.

Using the Librosa library, we obtain the mel spectrograms of the parted audio files, and label them as numpy arrays of 2 elements – 0 or 1, accordingly as wheeze or crackle is present or not.

Chest location and stethoscope information was obtained from the filenames and included as features in the dataset.

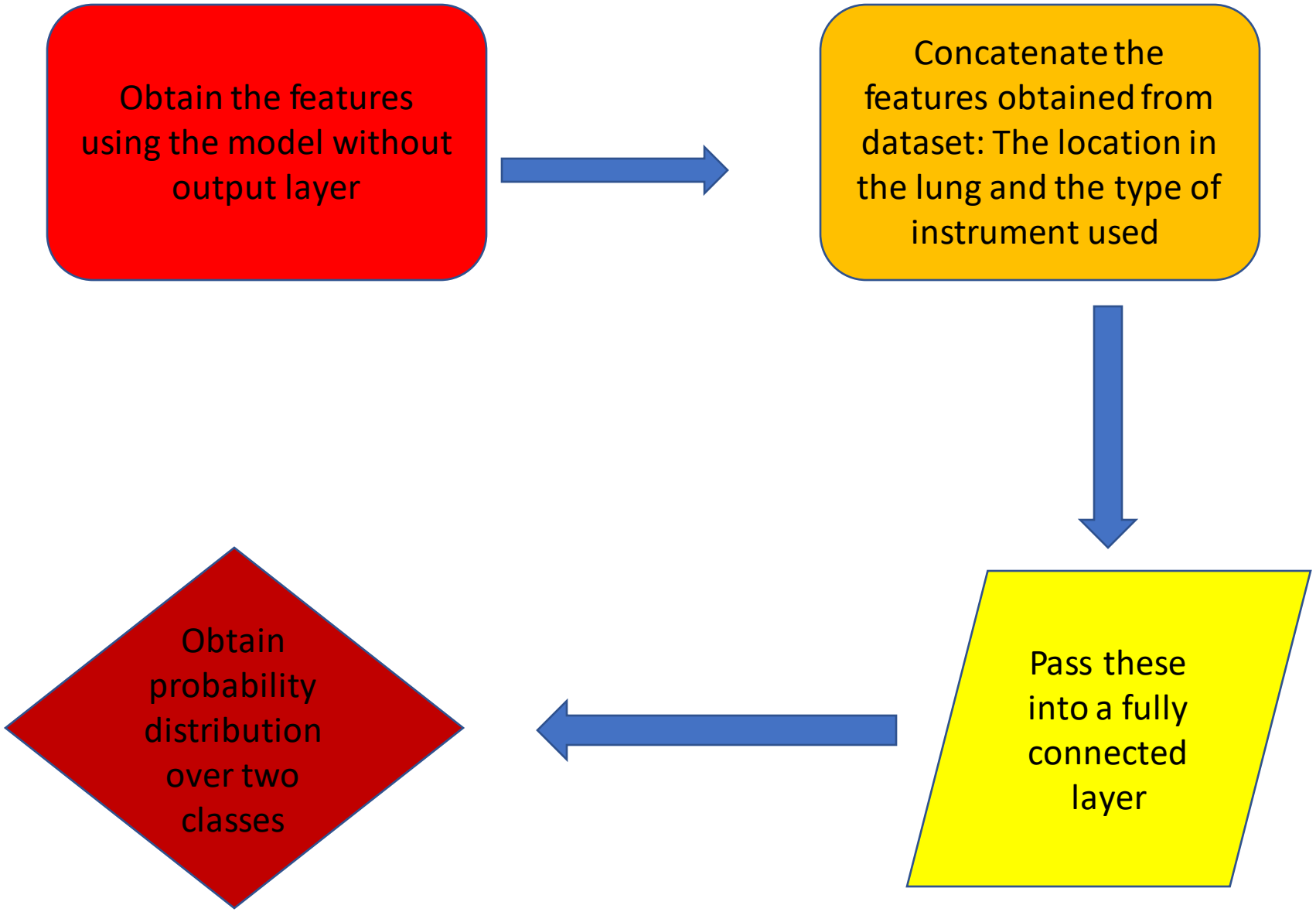
Then using the sklearn's train_test_split() function we have split the dataset into train and test sets in 80:20 ratio.

The final dataset has spectrograms, chest location features as input and the numpy array of crackle, wheeze probability as corresponding labels .


Database containing audio and annotation files
Splitted audio files based on timestamps in annotation files
Spectrograms of splitted files as inputs
Information available in filenames
Chest location and stethoscope information obtained
Annotation file containing crackle wheeze labels
Y_labels obtained using File handling in Python for both crackle and wheeze

Model description

- The model is a convolutional neural network implemented using Keras.
- It takes two inputs: a 128x128 RGB image and a scalar hard_feats(the features we obtain from the dataset: the location in the lung and the instrument).
- The architecture consists of four identical convolutional blocks, each followed by a max pooling layer.
- The output of the convolutional blocks is flattened and fed into a fully connected layer with 32 neurons.
- The output of this layer is combined with the scalar hard_feats input and fed into another fully connected layer with 2 neurons and sigmoid activation.
- The final output is a probability distribution over two classes.
- Prediction is made by setting classification threshold as 0.5 probability.


Obtain the features using the model without output layer
Concatenate the features obtained from dataset: The location in the lung and the type of instrument used
Pass these into a fully connected layer
Obtain probability distribution over two classes

Test-Set Results

```
BinaryCrossentropy Loss =  0.5059808492660522
```

Used as a parameter to be minimized by the Stochastic Gradient Descent algorithm, as the predictions are binary in nature – 0 for absence and 1 for presence of crackle or wheeze

```
Area under ROC curve =  0.7714003920555115
```

Found this to be a better parameter for evaluating multi label classification models than accuracy. The model is better as the value of AUC approaches 1

```
Accuracy =  82.79481530189514 %
```

Accuracy of the models predictions on the test dataset created from a random 20% of the given dataset

Problem we faced and how we dealt with it

Problem:

Not sure how to use the information provided in the dataset

Solution:

Extracted the features from the CNN without output layer and concatenated this information(location and instrument) to get all the features

THANK YOU!