Computer Engineering Department

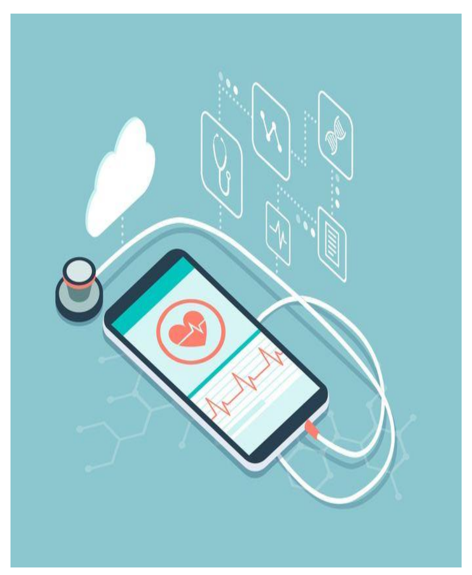
Adaptive and Heuristic Artificial Intelligence (AI) IoT Edge for Providing Proactive Healthcare to Rural and High-Risk Patients

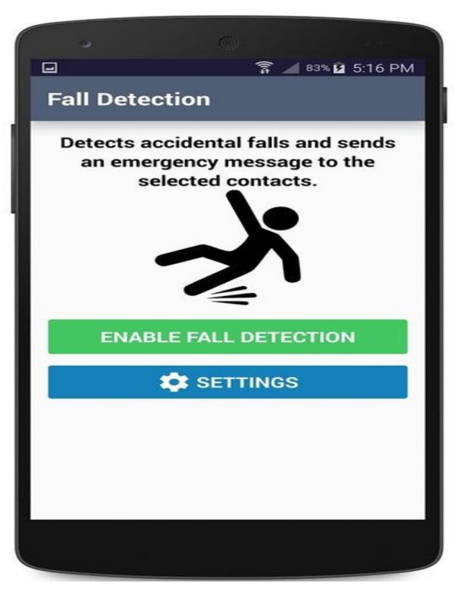
Project Advisor: Chandrasekar Vuppalapati

Jajoo, Priyanshi (MS Software Engineering)
Mahesh, Harshraj (MS Software Engineering)
Tajudeen, Thasleem Banu (MS Computer Engineering)
Kannankumarath Madom Krishnan, Parvathy (MS Software Engineering)

Introduction

The healthcare systems are overburdened, and a countless number of patients are deprived of treatments because of the inaccessibility to healthcare.There are myriad possibilities for AI technologies to be used in healthcare, starting from at-home diagnosis to preventive treatments. The proposed solution consisting of a mobile and web application aims to provide proactive healthcare to rural and high-risk patients. Patients take health tests using the mobile application that serves as an IoT device. It provides fast and reliable diagnosis and treatment recommendations after analyzing the data collected from the patient. The system provides recommendations for respiratory issues and falls.





Respiratory issues are analyzed by processing lung sounds recorded using a stethoscope and detecting crackles and wheezes in them. Falls are analyzed by checking the data from sensors on the mobile phone.

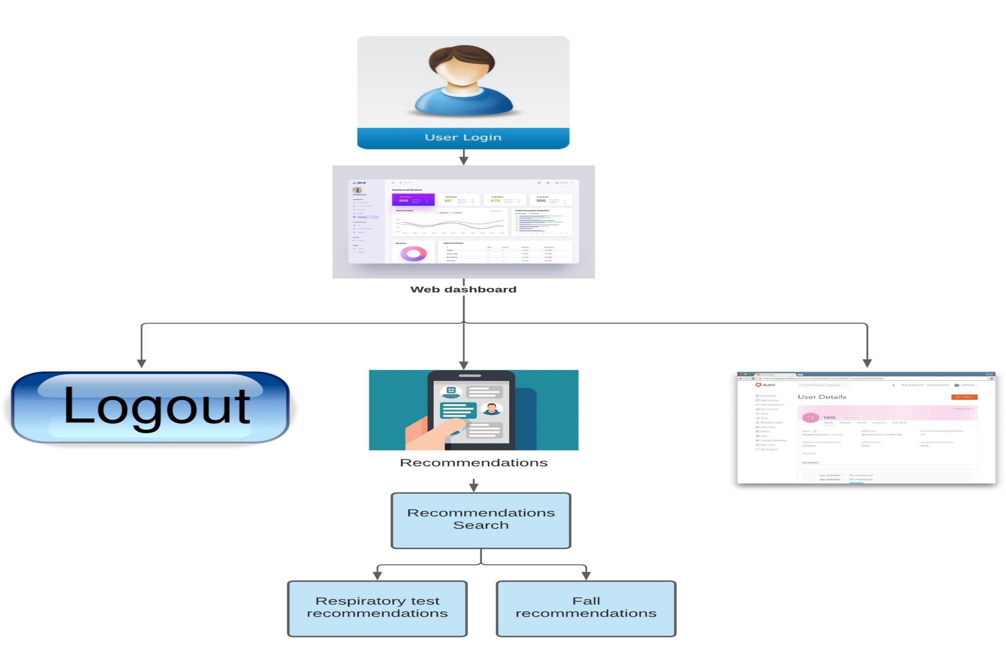
Methodology

System Architecture

The system consists of 3 subsystems namely mobile application, web application, and machine learning models.

Web Application

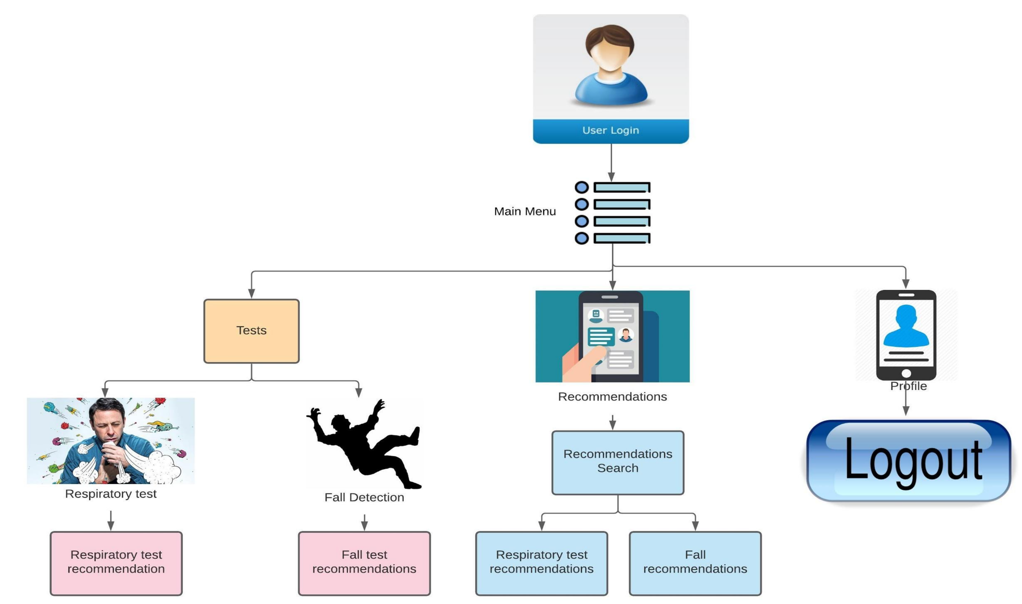
A single page application (SPA) is built using ReactJS, an open-source javascript library to build user interfaces. In this web application, the users will be able to login and view their profile and dashboard.



Methodology

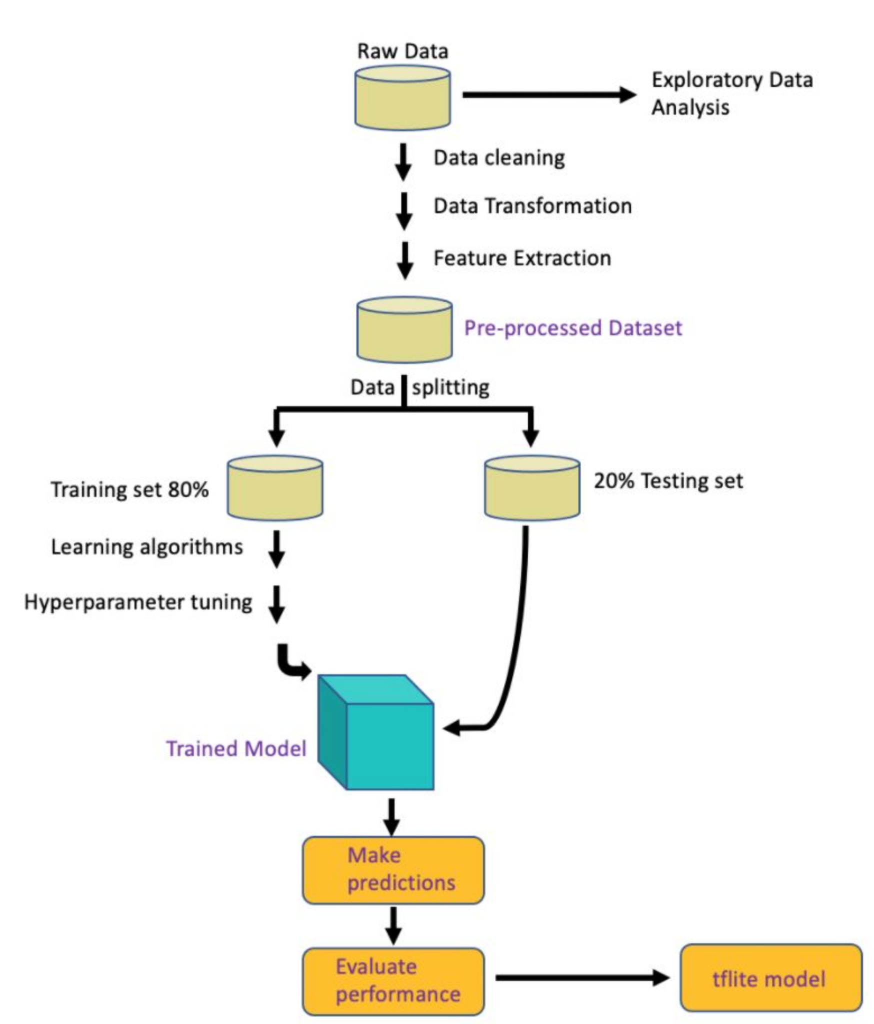
Mobile Application

The mobile app will host a neural network locally to predict users' conditions and suggest required actions or precautions to be taken by users. The app will be built using the native mobile platform, Android, and the neural network will be hosted within the mobile application using the Tensorflow Lite model, that is generated after compressing an existing Tensorflow model. Chaquopy python sdk for Android is used for preprocessing the data.



Machine Learning Models

Building an effective machine learning model that can be interpreted to make predictions on unseen data involves the following process. Data collection is the process of gathering relevant data that can serve the purpose of the proposed architecture; i.e., prediction of abnormal lung sounds and detection of fall events. Data is trained and tested using different machine learning algorithms and the one with high classification accuracy, high sensitivity, and specificity on the test data is chosen for deployment. This step typically involves data cleaning, data transformation, data reduction, and sampling. To accomplish the proposed design two different machine learning algorithms are trained. One for predicting respiratory illness and another one for fall detection.

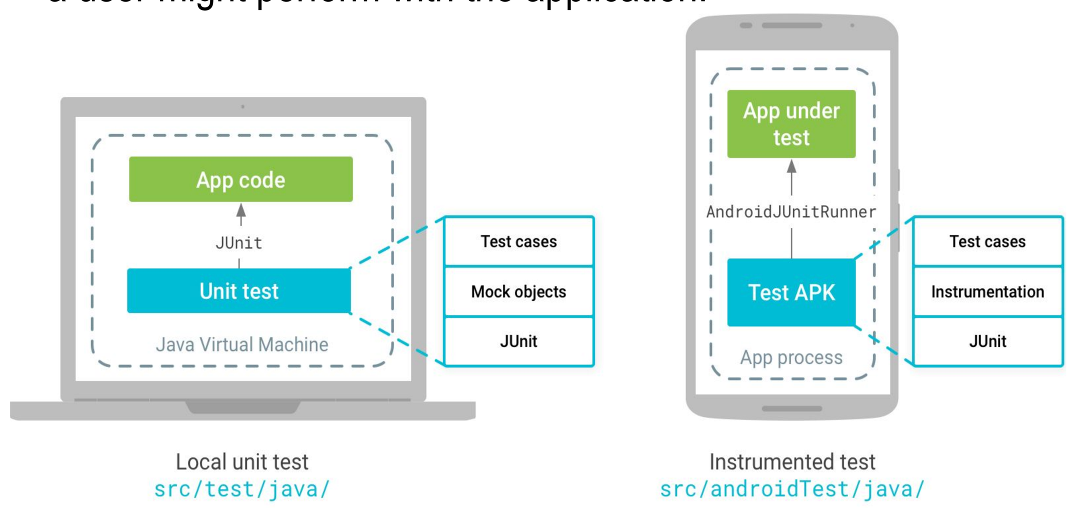


In both the settings, the preprocessed data is split into training and testing sets with an 80/20 ratio. The best models chosen for prediction are converted to tflite using a TensorFlow Lite converter and deployed in an Android application using Android Studio.

Analysis and Results

Functional and UI Testing

In this project, JUnit is used for instrumentation testing in which android and web application-specific functionalities such as activities, fragments, and services will be tested. Espresso testing framework is used for android application UI testing as it allows us to simulate every possible action a user might perform with the application.



Performance of Machine Learning Models

This project requires two machine learning models. One is for crackles/wheezes detection and the other is for fall detection and classification.

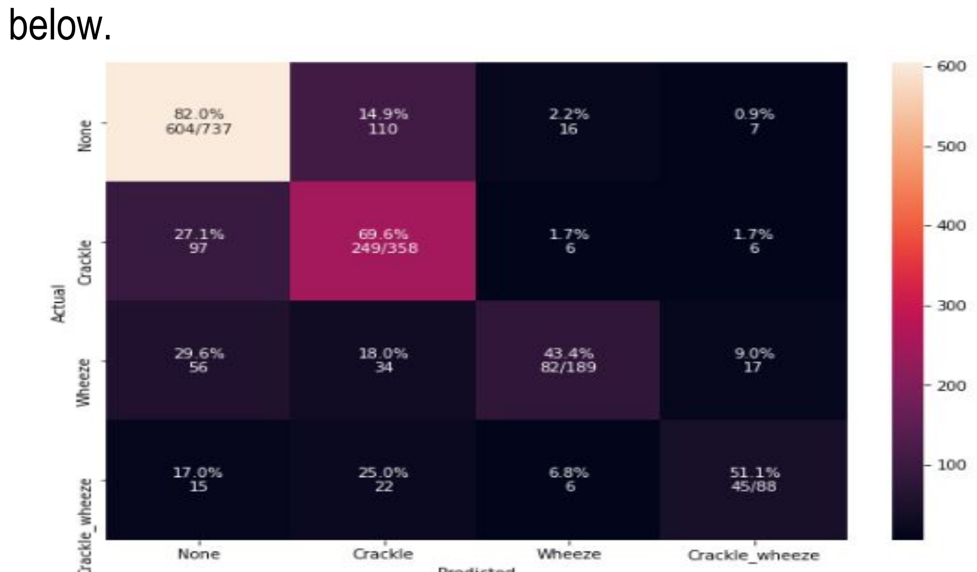
Respiratory Illness Detection

The following metrics for evaluating the classification results were defined by the authors of the ICBHI challenge dataset.

- Sensitivity (SE) = (Cc + Ww + Bb)/(C + W + B)
- Specificity (SP) = Nn/N
- Average score(AS) = (SE + SP)/2
- Harmonic score (HS) = (2 ∗ SE ∗ SP)/(SE + SP)

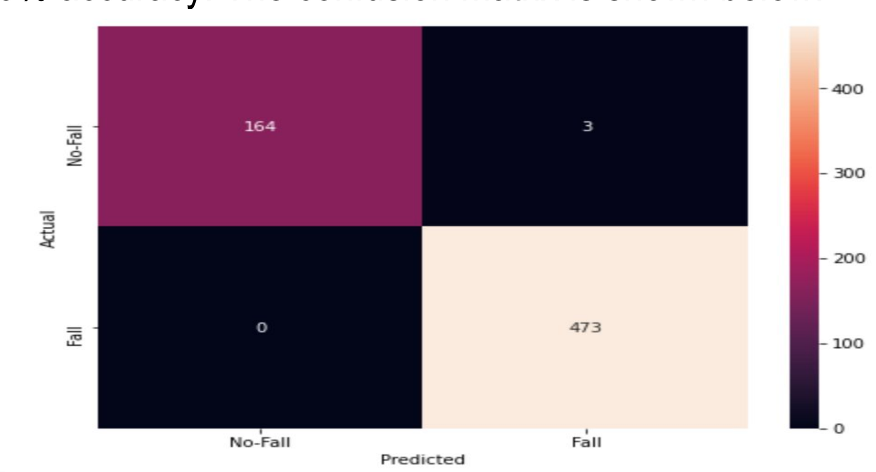
Model	Specificity	Sensitivity	Average Score	Harmonic Score
1D Separable Conv	0.74	0.52	0.63	0.61
2D CNN	0.80	0.52	0.66	0.63
BiLSTM	0.82	0.60	0.71	0.70

Overall LSTM model performed better and was chosen for deployment. The confusion matrix of LTSM model is shown below.



Fall Detection

fall data is trained using three machine learning models KNN, SVM, and Keras sequential model. This project used Keras sequential model for fall detection as it provided 99% accuracy. The confusion matrix is shown below.



Summary/Conclusions

This project proposes and builds an AI-enable IoT edge for rural and high-risk patients particularly narrating the issues with respiratory illness and elderly falls. The project consists of multiple modules to bring together this solution.

A machine learning pipeline to train and build models that can be deployed on an IoT edge device like a smartphone. An android app that runs machine learning models locally to predict the users' health. A web application to let the users visualize and analyze their health statistics.

Key References

[1] D. Perna and A. Tagarelli, "Deep auscultation: Predicting respiratory anomalies and diseases via recurrent neural networks," in 2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS), 2019.

[2] McClure, Hughes, Ren, et al, "The population approach to falls injury prevention in older people: findings of a two community trial", *BMC Public Health*, Vol.10, no.79, 2010. Available [Online]: https://bmcpublichealth.biomedcentral.com/articles/10.1186/1471-2458-10-79

[3] bmi.hmu.gr, "The MobiFall and MobiAct datasets", *bmi.hmu.gr*. Available [Online]: https://bmi.hmu.gr/the-mobifall-and-mobiact-datasets-2/

[4] J. Guo, and B. Li, "The Application of Medical Artificial Intelligence Technology in Rural Areas of Developing Countries", *Health Equity*, Vol. 2, no. 1, pp. 174-181, August 2018.

Acknowledgements

The authors are deeply grateful to Professor Chandrasekar Vuppalapati and Professor Dan Harkey for their indispensable comments and assistance in the development of this project.