Arm Workout Tracking and Recognition using Arduino Nano 33 BLE Sense **Manisha Rajkumar**
YouTube Link: https://youtu.be/Iv5NPEad0GE **UNI: mr4136**

MOTIVATION

To build smart watch that can also act as a fitness trainer that is not just be limited to calculating the steps and calories burnt but also record and analyze the different workouts you do in the gym. Although the existing fitness devices capture and record different physical activities, they do not capture the exact type of exercise in a workout routine. The project is designed to recognize the type of exercise in an Arm Workout using the accelerometer and gyroscope readings from the inbuilt LSM9DS1 built in the Arduino Nano 33 BLE Sense. The models in this project are trained to recognize the following set of exercises: Lateral Pulldown, Reverse Pulldown, Cable Curl, Dumbbell Bicep Curl, Pec fly, Triceps Kick-backs and the study is expanded to include Chest Press, Mid row, Reverse Fly, Shoulder Shrug, Triceps Extension, Side Raise.

SYSTEM DESIGN

Google's Tiny Motion Trainer is used to capture the data and create a rapid prototype model. The capture settings used are, threshold: 0.1, Samples per second: 20, Delay: 0.2 sec. The details of the entire implementation are shown below in the system architecture figure.

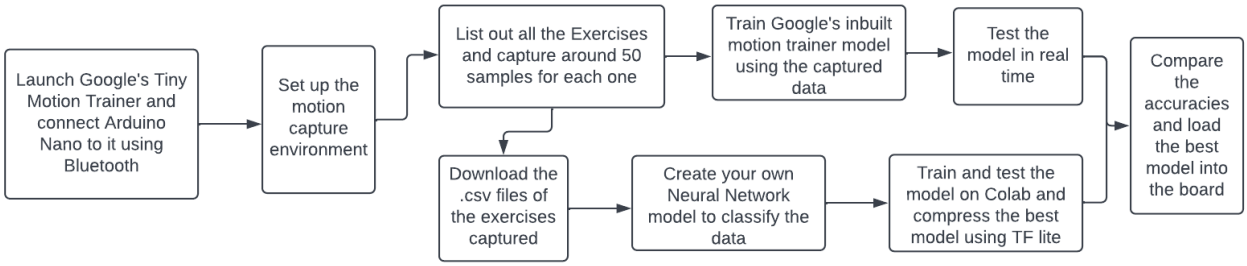

Fig.1. System Architecture

The CSV files of the captured exercise are used to train a custom fully connected neural network. The figure below shows the details of the neural network model used.

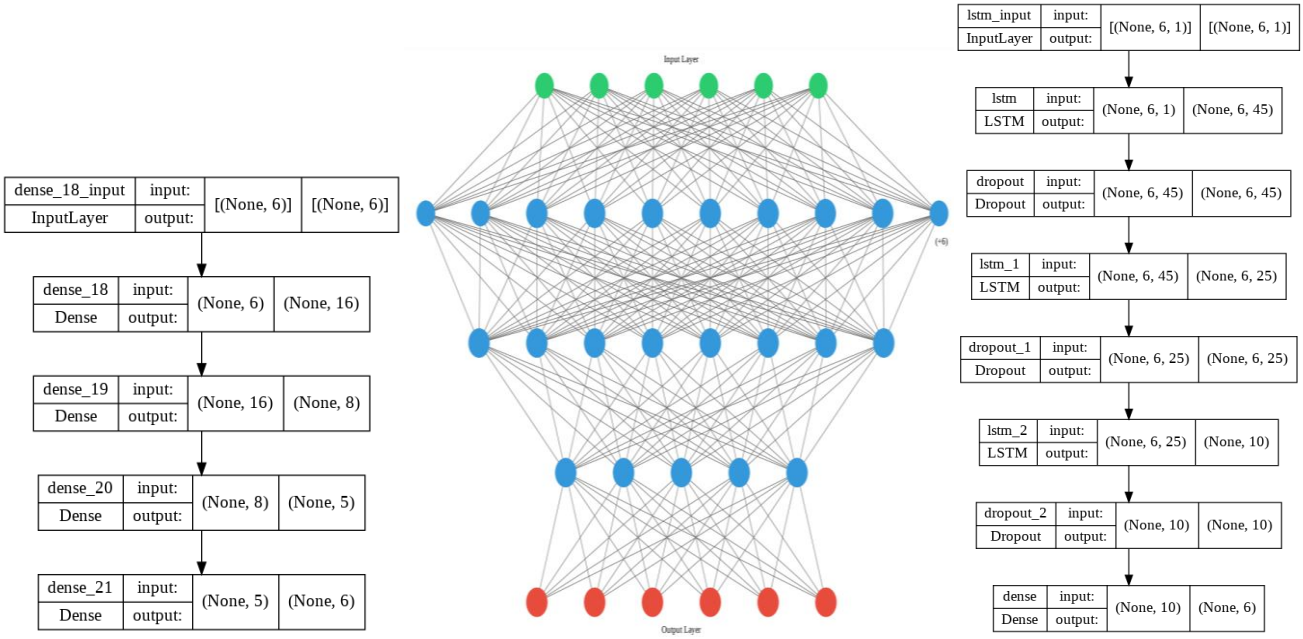

Fig.2. Neural Network Architecture and Visualization of Fully connected Network and LSTM Network

The trained model is then compared against the Tiny Motion Trainer model. The best model is compressed using TensorFlow Lite and loaded into the Arduino board.

RESULTS

Model	Training Set with 6 Exercises		Training Set with 10 Exercises	
	Training Accuracy	Validation Accuracy	Training Accuracy	Validation Accuracy
Tiny Motion Trainer	**98.96%**	89.58%	**97.92%**	90.63
Full Connected Network	97.19%	96.75%	91.62%	91.37%
LSTM Network	98.08%	**98.25%**	97.11%	**95.91%**

Table 1. Summary of the Training and Validation accuracies of all the models with 2 variations of the dataset

The custom neural network models outperform the built-in model by Google's tiny motion trainer. Although the training accuracy achieved by the Tiny Motion Trainer model is higher than the one achieved by the Full connected neural network (FCNN) and the LSTM Network, there is a significant gap between the training and validation accuracy. The validation accuracy of the LSTM Network is much higher than that of tiny motion trainer model. The test accuracies of the FCNN and the LSTM Network are 97.58% and 97.91% respectively for the model trained on 6 exercises and 91.21% and 96.75% respectively for the model trained on 10 exercises. It has been observed that the LSTM model scales better than the FCNN model.

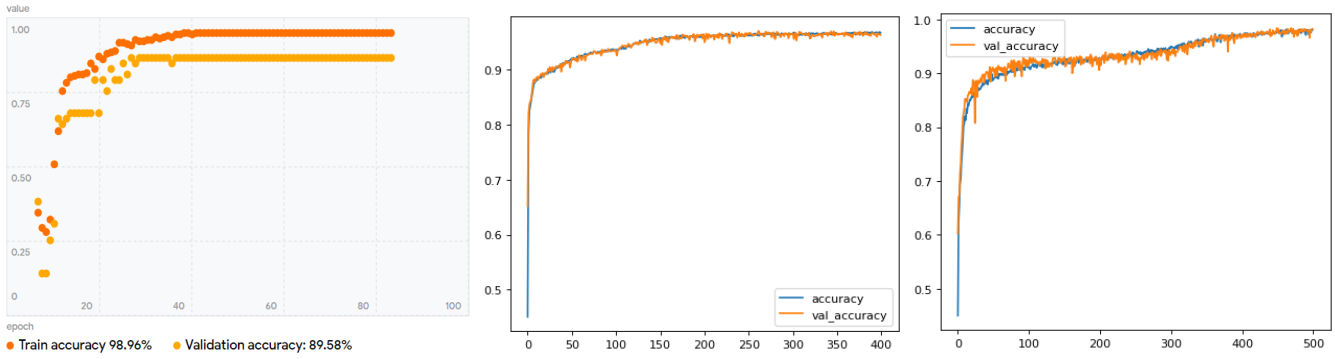

Fig.3. Training and Validation Accuracies Curves of Google's Tiny Motion Trainer Model, Fully Connecter Network and the LSTM Network respectively

The Details of the model size have been shown below. Depending on the hardware memory restrictions and the inference time desired, either of these custom models can be deployed on the edge device.

Model	Model Size
Tiny Motion Trainer	177 KB
Full Connected Network	21.6 KB
LSTM Network	566.5 KB

Table 2. Model Sizes of all the models in discussion

This concept of motion capture and determining the activity can be extended to track the number of reps and sets and to check if the exercise was performed correctly by including a set of the same of exercises performed incorrectly. This can be used in for other applications such as recording the type of shots taken in various sports such as cricket, badminton, and so on.