

## Fall Detection Application on Android

This document explains a fall detection application that is developed in Android. The application detects when falls happen by using a deep learning model. The deep learning model was trained by using a simplified version of the UMAFall Dataset[1]. When a fall is detected, the application sends patient information and location data to the server and also has the ability to send SMS.

### Collecting Sensor Data on an Android Device

Most Android smartphones have built-in sensors (accelerometer and gyroscope) that measure motion and orientation. These sensors are capable of measuring and providing data with high precision and accuracy so they are useful to monitor three-dimensional device movement or positioning. For this work, built-in accelerometer and gyroscope data were used to predict the falling.

### Saving measured data to built-in SQLite database

Accelerometer and gyroscope data were saved in a built-in SQLite database.

### Deep Learning model and using it on Android programming

A simplified version of the UMAFall Dataset was used for training. Original UMAFall Dataset contains 746 csv files which belong to 19 participants, the size of the files is 373.88 GB. 3 individuals (Subject-2, Subject-4 and Subject-14) and 4 activities (Walking, BackwardFall, ForwardFall and LateralFall ) were chosen for the sake of simplicity. Details of selected subjects are below:

Subject	Gender, Age	Activities	Device
2	Female, 22	<ul style="list-style-type: none"><li>• Walking</li><li>• BackwardFal</li><li>• ForwardFall</li><li>• LateralFall</li></ul>	LG G4 and sensor tag
4	Male, 27	<ul style="list-style-type: none"><li>• Walking</li><li>• BackwardFal</li><li>• ForwardFall</li><li>• LateralFall</li></ul>	LG G4 and sensor tag
14	Female, 22	<ul style="list-style-type: none"><li>• Walking</li><li>• BackwardFal</li><li>• ForwardFall</li><li>• LateralFall</li></ul>	Samsung S5 and sensor tag

In the original UMAFall work, smartphone gyroscope data wasn't included so smartphone accelerometer and waist sensor tag was used for gyroscope data.

Processed data looks like below:

	X_ACC	Y_ACC	Z_ACC	X_GYRO	Y_GYRO	Z_GYRO	Label
0	0.863525	-0.067383	-0.447510	-2.421875	-0.773438	0.71875	1
1	0.863525	-0.067383	-0.447510	-2.734375	0.210938	0.18750	1
2	0.862549	-0.064941	-0.438965	-2.734375	0.210938	0.18750	1

Deep learning model is pretty simple and has only one hidden layer. Model trained 500 epoch and the batch size is 128. Model saved as TensorFlow Lite to be used in Android. Training code and the dataset can be seen in the Github repository[2]

Layer (type)	Output Shape	Param #
=====		
dense_6 (Dense)	(None, 64)	448
dense_7 (Dense)	(None, 1)	65
=====		
Total params: 513		
Trainable params: 513		
Non-trainable params: 0		

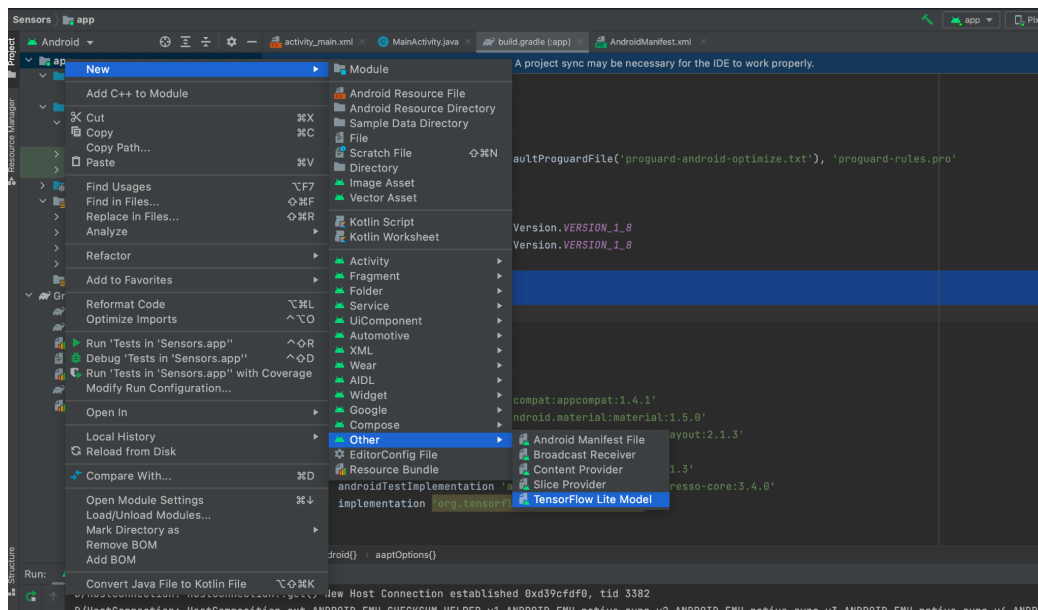
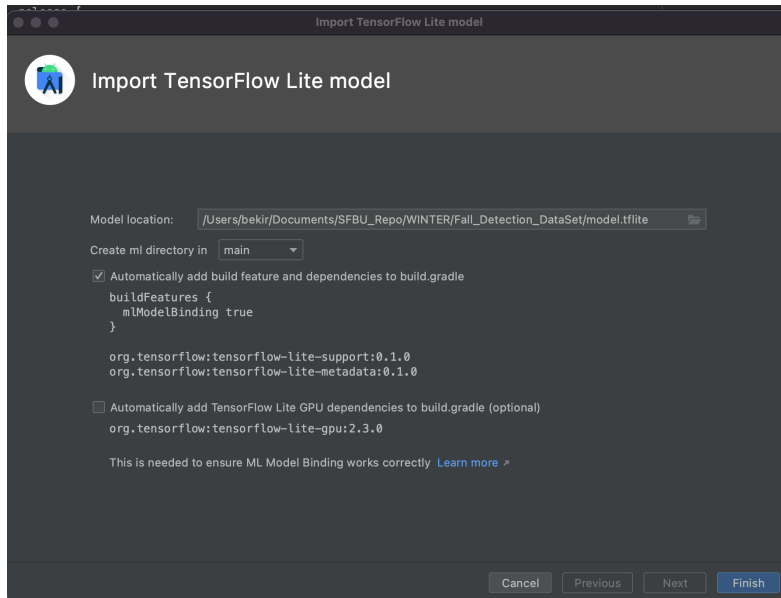
## Using The Model in Android Application

Trained model saved as TensorFlow Lite model to be used in Android. TensorFlow Lite is a set of tools that enables on-device machine learning by helping developers run their models on mobile, embedded, and edge devices[3].

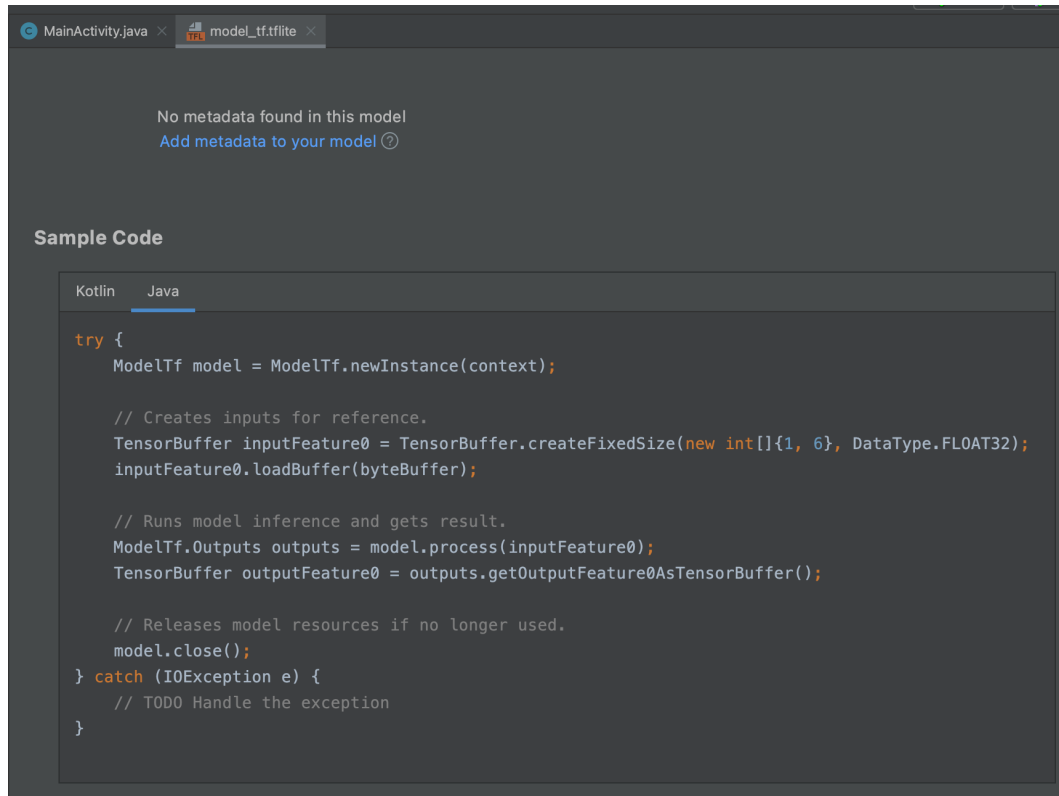
Here is the process of using the TFLite model in an Android application:

**Right-click on the app file the TFLite model or click on File, then New > Other > TensorFlow Lite Model**

Select the location of your TFLite file. After this process, Android Studio will configure the module's dependency on your behalf with ML Model binding and all dependencies automatically inserted into your Android module's build.gradle file. The process is simple as below:



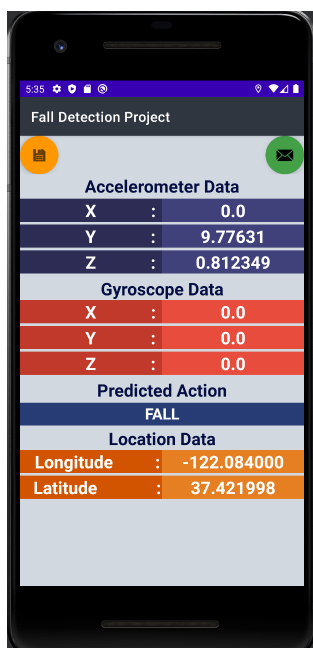
After adding the tflite model, we can start to get predictions from the model. Android Studio gives sample code like below to show how to use the model but we need to install tflite-support **`pip install tflite-support`** to get the sample code like below. The model gets a float array of 6 float numbers and gives a prediction.



## Sending data to server by using TCP programming and SMS function

When a fall is detected, the program sends the patient and location data to the server and an sms to a provided phone number. The Java server program can be seen in the Github repo.

## UI of the program



## Sources

- [1] [http://webpersonal.uma.es/de/ECASILARI/Fall\\_ADL\\_Traces/UMA\\_FALL\\_ADL\\_dataset.html](http://webpersonal.uma.es/de/ECASILARI/Fall_ADL_Traces/UMA_FALL_ADL_dataset.html)
- [2] <https://github.com/inputvector/Simple-Deep-Learning-Model-for-Fall-Detection>
- [3] <https://www.tensorflow.org/lite/guide/android>