Overview of TensorFlow Lite for Micro (TFLM) on HiFi 4 Processor

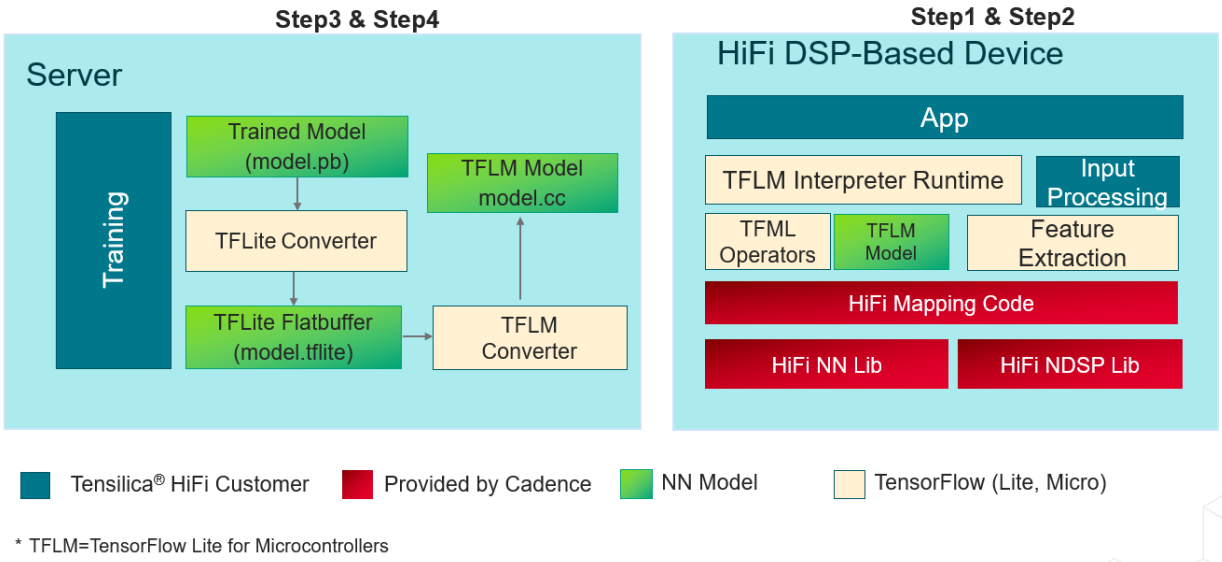


Figure 1: An End to End tool chain, from training to deployment on HiFi 4

TensorFlow Lite for Micro (TFLM) is a complete end to end tool chain that allows a developer to create trained models, convert them for HiFi DSP implementation, and actually run on the HiFi processor. Figure 1 above shows this entire flow. All modules provided in the "Server" box are provided by Google; in box "HiFi DSP-based device", theTFLM Interpreter Runtime (interpreter.cc) is also developed by Google. Cadence provides the libraries, HiFi NNLib, and HiFi NDSPLIB and has integrated them using the mapping code into interpreter.cc.

To learn more about TFLM, please refer to

https://github.com/tensorflow/tensorflow/tree/master/tensorflow/lite/micro

In figure 1, Steps 1 & 2 refer to running an example speech recognizer on HiFi 4 ISS (instruction set simulator). The example speech recognizer is called "micro_speech" that currently recognizes the words(labels) Yes, No, Unknown and Silence. Unknown is an "out of vocabulary" word which means any word that is not Yes or No. Silence is the ambient noise. Please see Figure2 for details on the front end pre processing and neural net used.

There is already a pre-trained model, called model.cc file, that has been given as a reference for beginners. The model.cc file contains the neural network used, along with the weights, for Yes, No, Unknown and Silence. When you follow the steps in Step1 and Step2, and see "All tests passed" then you know that you have successfully installed and executed the example speech recognizer on HiFi 4 DSP.

Step 3 details how to re-create the given model.cc file. Once you have re-created the model.cc file, you may run Step 2 to confirm that it works.

Step 4 shows how to create a *new* model.cc file using words other than Yes, No. The speech dataset is given for `yes,no,up,down,left,right,on,off,stop,go.` If you wish to add other words from the above list, please see step4.

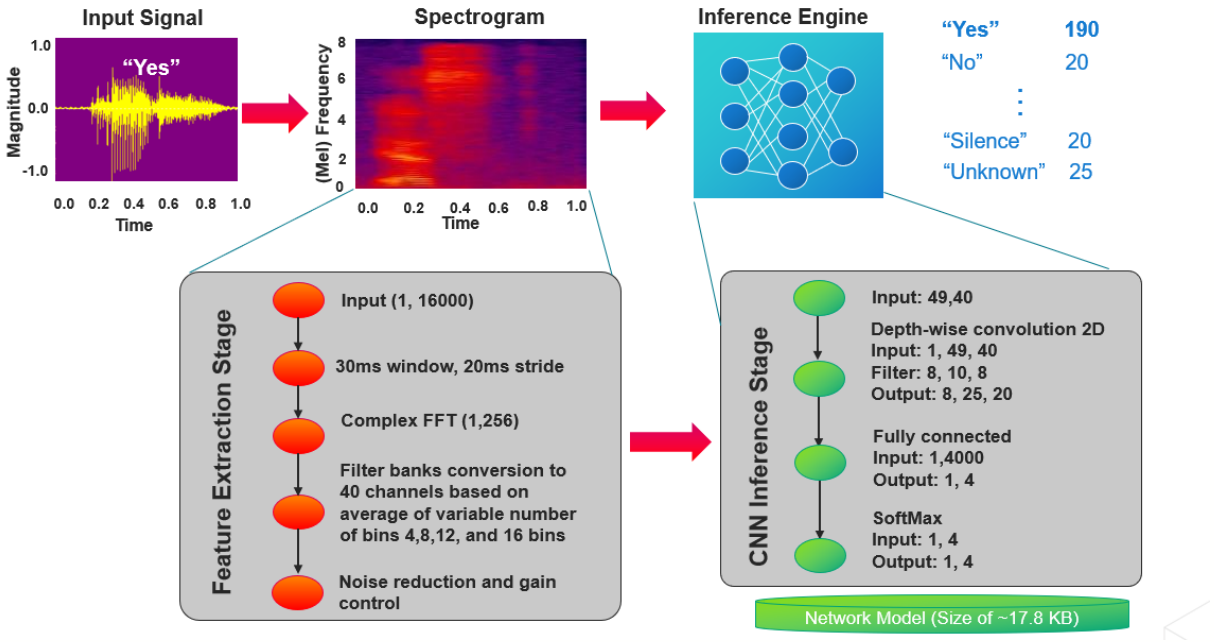


Figure 2: Neural Net used in micro_speech example

Pre-requisite

All the software described below is validated on a Linux PC with Ubuntu 16.04 and Red hat 6.1

Step1: Get TensorFlow and HiFi 4 source code to run Yes/No Recognizer

$ git clone https://github.com/tensorflow/tensorflow.git

$ cd tensorflow

Step2: Build & Run for HiFi 4 Processor with pre generated trained model

Model.cc is a model that has already been trained using the open sourced Google dataset for speech. Follow below steps to build and run micro speech test bench

2.1 Setup Xtensa tools

$ setenv XTENSA_CORE AE_HiFi4_LE5_FP_XC(Must select xclib HiFi 4 config, in order to compile C++11 source code as part of TFLM)
$ setenv XTENSA_SYSTEM ~/xtensa/XtDevTools/install/tools/RI-2019.2-linux/XtensaTools/config

$ env |grep XTENSA

XTENSA_SYSTEM=~/xtensa/XtDevTools/install/tools/RI-2019.2-linux/XtensaTools/config

XTENSA_CORE=AE_HiFi4_LE5_FP_XC

2.2 Run below command to **compile and run** micro speech test bench

$ make -f tensorflow/lite/micro/tools/make/Makefile test_micro_speech_test TARGET=xtensa_hifi TARGET_ARCH=hifi4

which should conclude with `~~~ALL TESTS PASSED~~~`.

Step3: How to generate given Pre-trained model(model.cc)

model.cc is C++ compliable model file, that contains 4 labels namely yes, no, unknown and silence.

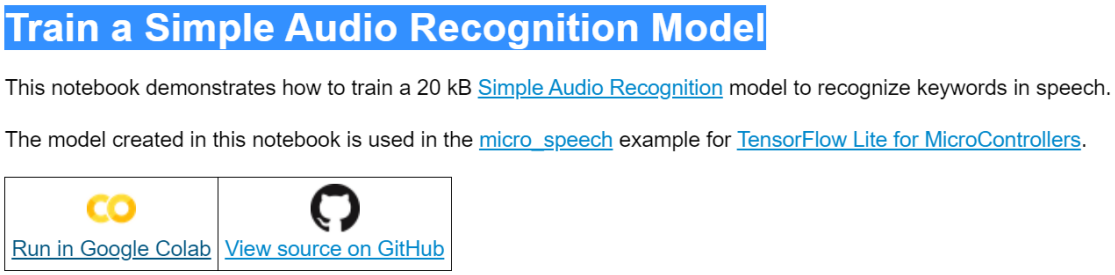
To learn more about how to create this file, please go through the following link.

https://github.com/tensorflow/tensorflow/tree/master/tensorflow/lite/micro/examples/micro_speech/train

To generate model.cc file for yes, no, unknown and silence, please go through following link.

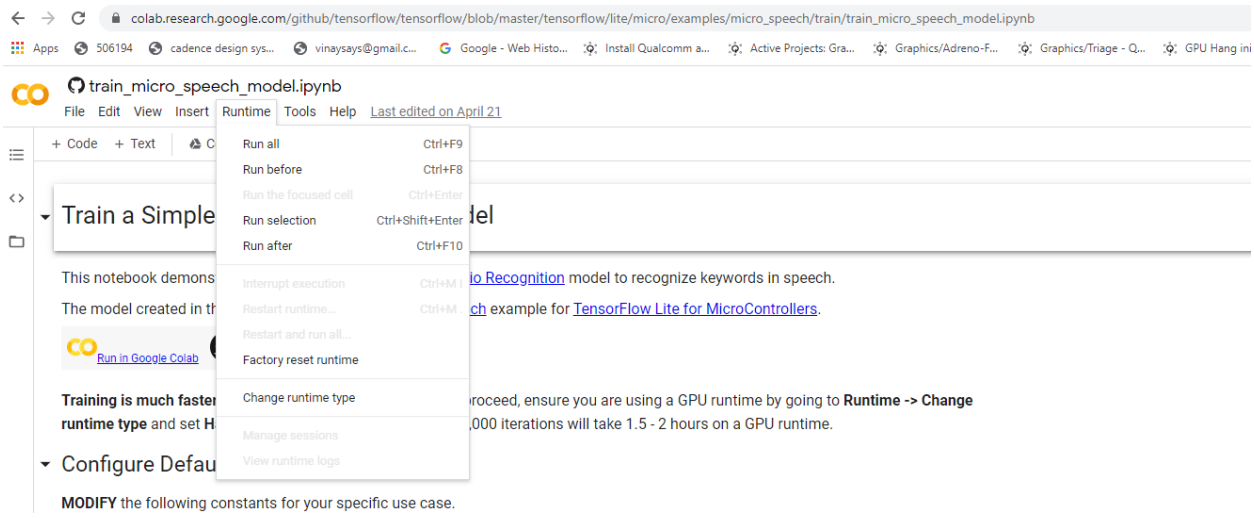
https://github.com/tensorflow/tensorflow/blob/master/tensorflow/lite/micro/examples/micro_speech/train/train_micro_speech_model.ipynb

Click on "Run in Google Colab"



And it opens link like below
https://colab.research.google.com/github/tensorflow/tensorflow/blob/master/tensorflow/lite/micro/examples/micro_speech/train/train_micro_speech_model.ipynb

Go to Runtime->Run All (Google sign-in is required, enter your google credentials to train model using Google Colab)



To validate generated model.cc file, please follow procedure given in step 2

Step4: How to generate new trained model with other vocabulary words (model.cc)

The given speech database contains `yes,no,up,down,left,right,on,off,stop,go`

If you wish to create model.cc with words other than yes and no, for example from `WANTED_WORDS = "yes,no"` to `WANTED_WORDS = "up,down"` and you can choose any two words and please refer to the following link for details.

https://colab.research.google.com/github/tensorflow/tensorflow/blob/master/tensorflow/lite/micro/examples/micro_speech/train/train_micro_speech_model.ipynb

runtime type and set **Hardware accelerator: GPU**. Training 15,000 iterations will take 1.5 - 2 hours on a GPU

Configure Defaults

MODIFY the following constants for your specific use case.

```
# A comma-delimited list of the words you want to train for.
# The options are: yes,no,up,down,left,right,on,off,stop,go
# All the other words will be used to train an "unknown" label and silent
# audio data with no spoken words will be used to train a "silence" label.
WANTED_WORDS = "yes,no"

# The number of steps and learning rates can be specified as comma-separated
```