

TFL Micro "Micro Speech" Example - Entire Workflow

Cristinel Ababei



MARQUETTE
UNIVERSITY

BE THE DIFFERENCE.

1

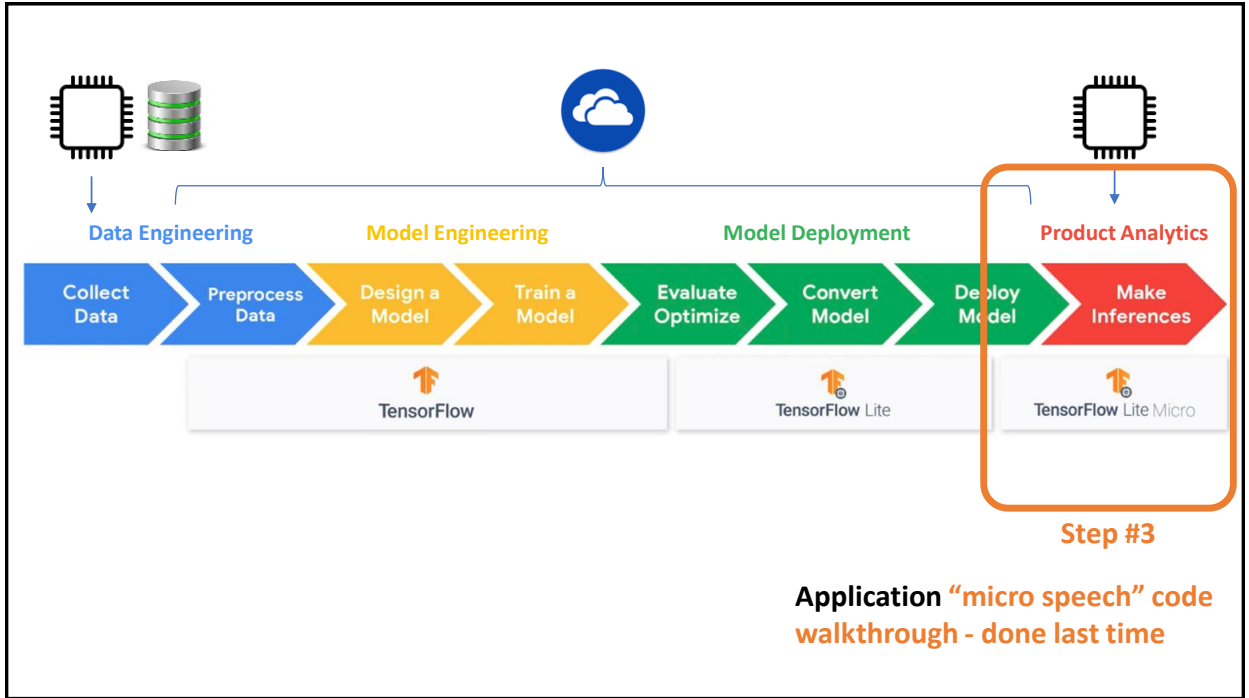
1

Workflow: Model and application development & testing

- **Step #1:** Model development, training, optimization (pruning, quantization), TFL-Micro model generation; Python; Done in Google Colab or locally on your computer.
- **Step #2:** Application development: Testing and Application Run on **development machine (or cloud)**; C/C++; Done on macOS (this is what textbook authors use) or Linux machine.
- **Step #3:** Application deployment to Microcontroller; C/C++; Arduino IDE; Done on local computer and executed on Arduino Nano 33 BLE Sense in our case.

2

2



3

Compile and run “micro speech” example in Arduino

```

File Edit Sketch Tools Help
New Ctrl+N
Open... Ctrl+O
Open Recent
Sketchbook
Examples
Close Ctrl+W
Save Ctrl+S
Save As... Ctrl+Shift+S
Page Setup Ctrl+Shift+P
Print Ctrl+P
Preferences Ctrl+Comma
Quit Ctrl+Q

#include "tensorflow/lite/micro/micro.h"
#include "tensorflow/lite/micro/micro_error_reporter.h"
#include "tensorflow/lite/micro/micro_interpreter.h"
#include "tensorflow/lite/micro/micro_model.h"
#include "tensorflow/lite/micro/micro_session.h"

// Globals, used for debugging only
namespace {
  tflite::ErrorReporter* error_reporter = nullptr;
  const tflite::Model* model = nullptr;
  tflite::MicroInterpreter* interpreter = nullptr;
  tflite::Tensor* input = nullptr;
  tflite::Tensor* output = nullptr;
  int32_t previous_time = 0;
}

// Create an area of memory to use for data
// This is (usually) a global variable
uint8_t memory[1024];

void setup() {
  Serial.begin(115200);
  Serial.println("Starting micro_speech");

  // Initialize TensorFlow Lite Micro
  error_reporter = tflite::ErrorReporter::CreateMicro(
    memory, sizeof(memory));
  model = tflite::Model::LoadFromFile("model.tflite");
  interpreter = tflite::MicroInterpreter::CreateWithoutResources(
    model, error_reporter, memory);
  input = interpreter->AllocateTensors();
  output = interpreter->AllocateTensors();

  // Initialize the audio provider
  AudioProvider audio_provider;
  FeatureExtractor feature_extractor(audio_provider);
  Model model(feature_extractor);
  CommandRecognizer command_recognizer(model);
  CommandResponder command_responder(command_recognizer);

  // Start the main loop
  while (true) {
    audio_provider.Record();
    feature_extractor.ExtractFeatures();
    model.Run();
    command_recognizer.RecognizeCommands();
    command_responder.Respond();
  }
}

void loop() {
  // Do nothing
}
  
```

```

COM3
Heard unknown (202) @571568ms
Heard unknown (204) @574480ms
Heard yes (203) @576000ms
Heard yes (214) @577520ms
Heard yes (210) @579168ms
Heard yes (202) @580880ms
Heard unknown (202) @584096ms
Heard unknown (204) @587392ms
Heard unknown (201) @589760ms
  
```

4

Library Updates

- **Arduino library Harvard_TinyMLx may be outdated**
 - Examples can still be compiled and will work
 - But, model training scripts (Python) and application development and testing code (C/C++) available on github may be more recent and not matching exactly the examples from Harvard_TinyMLx Library
- **To get the latest TFL Micro examples:**
 - Inside Arduino IDE: Tools→Manage Libraries... Search for tinyML, identify Harvard_TinyMLx and UNINSTALL it
 - Goto github: <https://github.com/tensorflow/tflite-micro-arduino-examples/tree/main> and Download it all as: [tflite-micro-arduino-examples-main.zip](#)
 - Place the above .zip file inside your Arduino libraries folder; in my case it is: **M:\arduino221\libraries\tflite-micro-arduino-examples-main.zip**
 - Inside Arduino IDE: Sketch→Include Library→Add .ZIP Library and select the above .zip file. This then should show up inside your libraries/ folder as: **libraries\Arduino_TensorFlowLite** where you will find the latest examples (hello_world, micro_speech, etc.).
 - These up-to-date examples should be in sync with the model training scripts (Python) and application development and testing code (C/C++) available on github at: <https://github.com/tensorflow/tflite-micro/tree/main/tensorflow/lite/micro/examples>

5

TensorFlow Lite Micro “Micro Speech” Model Running the example application!



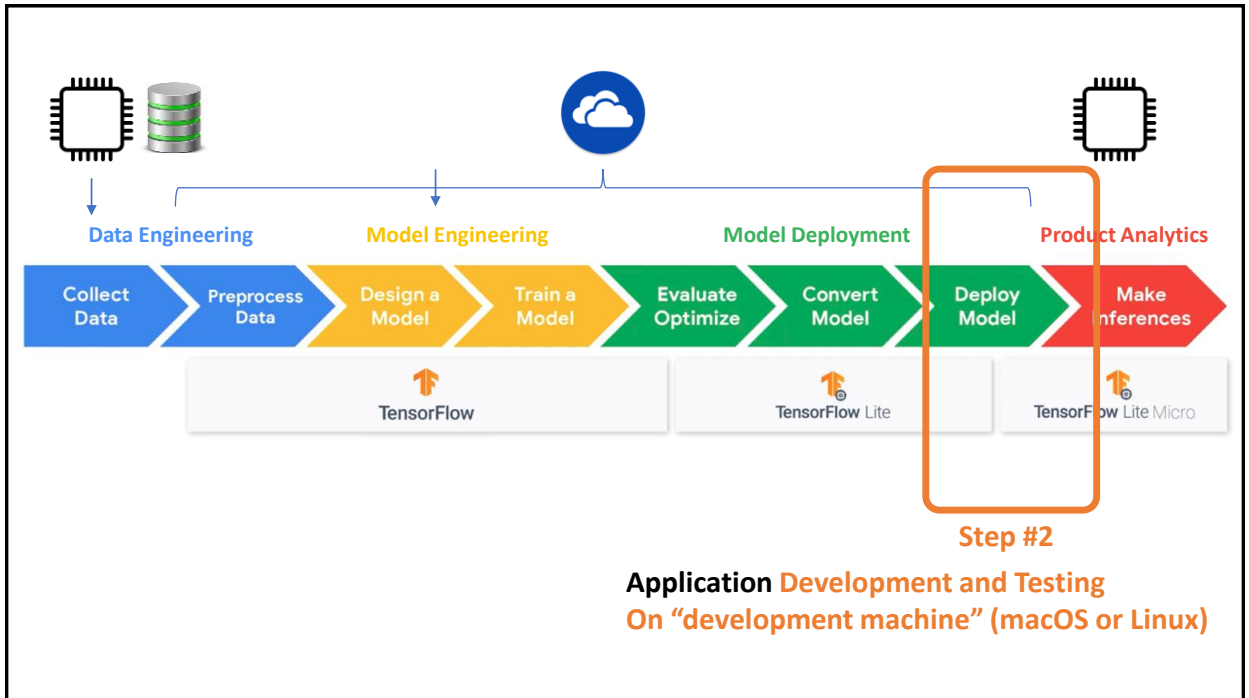
micro_speech.ino

C:\Users\Cristinel Ababei\Documents\Arduino\libraries\Harvard_TinyMLx\examples\micro_speech
OR: M:\arduino221\libraries\libraries\Harvard_TinyMLx\examples\micro_speech

Read at the same time the “Tutorial 11_2”, Section 6

6

6



7

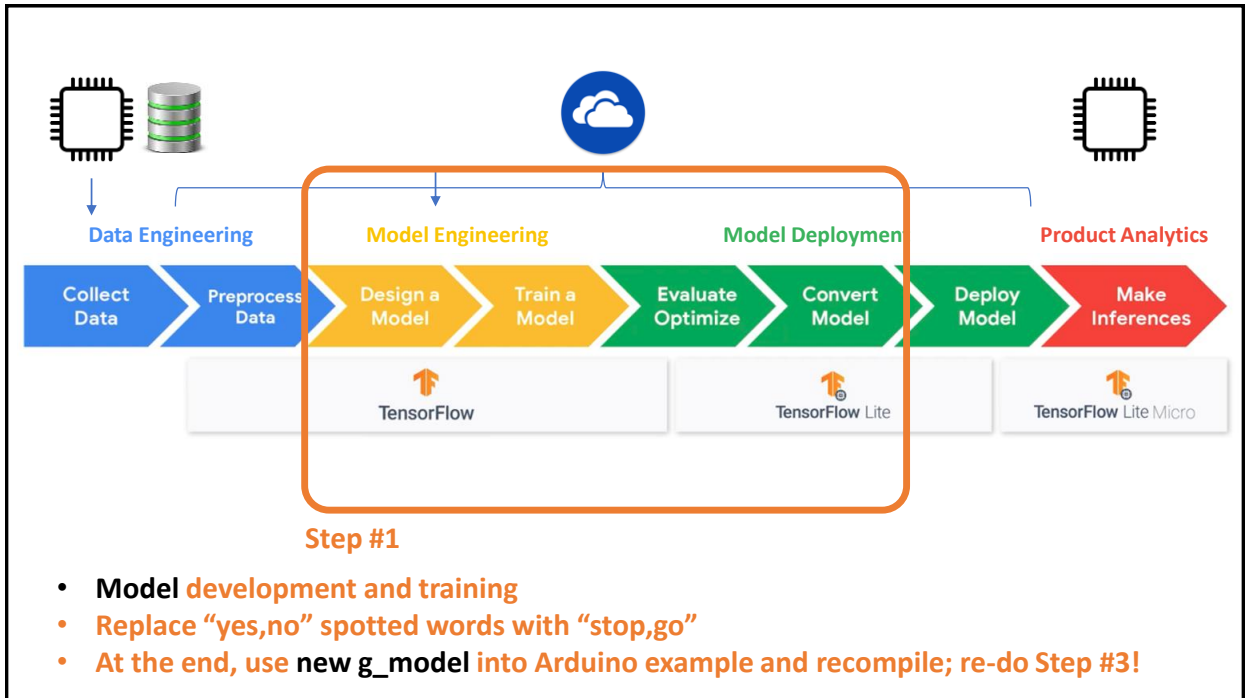
Application Development and Testing

Code Time!

Run Tests and Application on macOS or Linux
Read first “Tutorial 11_2”, Sections 4 & 5



8



9

Training a Simple Audio Recognition Model

Code Time!

`train_micro_speech_model.ipynb` (takes 1.5-2h)
 Read at the same time the “Tutorial 11_2”, Section 3



10

10

Credits

- A previous edition of this course was developed in collaboration with Dr. Susan C. Schneider of Marquette University.
- We are very grateful and thank all the following professors, researchers, and practitioners for jump-starting courses on TinyML and for sharing their teaching materials:

- Prof. Marcelo Rovai - TinyML - Machine Learning for Embedding Devices, UNIFEI
 - <https://github.com/Mjrovai/UNIFEI-IESTIO1-TinyML-2022.1>
- Prof. Vijay Janapa Reddi - CS249r: Tiny Machine Learning, Applied Machine Learning on Embedded IoT Devices, Harvard
 - <https://sites.google.com/g.harvard.edu/tinyml/home>
- Prof. Rahul Mangharam – ESE3600: Tiny Machine Learning, Univ. of Pennsylvania
 - <https://tinyml.seas.upenn.edu/#>
- Prof. Brian Plancher - Harvard CS249r: Tiny Machine Learning (TinyML), Barnard College, Columbia University
 - https://a2r-lab.org/courses/cs249r_tinyml/

11

11

References

- Additional references from where information and other teaching materials were gathered include:

- Applications & Deploy textbook: “TinyML” by Pete Warden, Daniel Situnayake
 - <https://www.oreilly.com/library/view/tinyml/9781492052036/>
- Deploy textbook “TinyML Cookbook” by Gian Marco Iodice
 - <https://github.com/PacktPublishing/TinyML-Cookbook>
- Jason Brownlee
 - <https://machinelearningmastery.com/>
- TinyMLedu
 - <https://tinyml.seas.harvard.edu/>
- Professional Certificate in Tiny Machine Learning (TinyML) – edX/Harvard
 - <https://www.edx.org/professional-certificate/harvardx-tiny-machine-learning>
- Introduction to Embedded Machine Learning - Coursera/Edge Impulse
 - <https://www.coursera.org/learn/introduction-to-embedded-machine-learning>
- Computer Vision with Embedded Machine Learning - Coursera/Edge Impulse
 - <https://www.coursera.org/learn/computer-vision-with-embedded-machine-learning>

12

12