

フレームワーク TensorFlowの準備

大沢 健太郎, 谷本 和俊

オープンソースのマイコン用機械学習フレームワーク TensorFlow Lite for Microcontrollers (以下, TFLM) を, Picoで動かす手順について解説します。TFLMのビルド環境にはセットアップが簡単なラズベリー・パイ4を利用します。

Pico用のTensorFlow リポジトリがある

Pico用のTFLMは、既にGitHub上に公開されています。ただし、Googleが運営しているTensorFlowのリポジトリではなく、ラズベリーパイ財団が運営している公式リポジトリ (<https://github.com/raspberrypi/pico-tflmicro>) にあります。

READMEを見ると、このリポジトリは自動生成されたもので、リード・オンリとあり、issueやpull requestは、TensorFlow側のリポジトリにファイルされるとあります。最新版を反映するには、TensorFlowのリポジトリにあるスクリプトgenerate.py (tensorflow/lite/micro/tools/project/generate.py) を使って生成するとの記載がありますが、現在はcreate_tflm_tree.py (tensorflow/lite/micro/tools/project_generation/create_tflm_tree.py) に置き換わっています。確認したところREADMEに記載されたオプション指定では動作しませんでしたので、今回はラズベリー・パイ側のリポジトリをそのまま利用します。

Pico用のTFLMには、表1に記載したhello_world, micro_speech, magic_wand, person_detectionの4つのサンプル・アプリケーションがあります。これらのうち、hello_worldだけはPicoのLEDで動作が確認できるように実装されています。hello_world以外のサンプルは、Picoにセンサが搭載されていないため、マイク、加速度、カメラなどのセンサからデータを取得するコードは自分で記述する必要があります。まずはビルド環境構築の確認の意味で、Pico単体で動作確認が可能なhello_worldを動かしてみます。

表1 TFLMのサンプル・アプリケーション一覧

サンプル名	内容
hello_world	入力値に対してsin波(サイン関数)の予測値を返す。PicoのLEDに対して予測値をPWM出力することでPicoのみで動作確認が可能
micro_speech	センサにマイク入力を使用したシンプルなスピーチ(キーワード)認識
magic_wand	加速度センサを使用したジェスチャ推定
person_detection	カメラ(画像入力)を使用した特定の人検出。make時に指定するサンプル名はperson_detection_int8

ビルド環境として利用する ラズパイ4のセットアップ

ビルド環境にラズベリー・パイ4(または3)を利用する場合、Pico用の公式SDK環境をセットアップし、TFLMのリポジトリをクローンするだけで環境を立ち上げることが可能です。以下の手順は、ラズベリー・パイ4の/home/piで実行していきます。

● Pico用の公式SDK環境セットアップ

Picoの公式ドキュメント⁽¹⁾の通りにスクリプトをダウンロードし、実行するだけでセットアップが完了します。

```
$ wget https://raw.githubusercontent.com/raspberrypi/pico-setup/master/pico_setup.sh
$ chmod +x pico_setup.sh
$ ./pico_setup.sh
```

● TFLMのクローン

ラズベリー・パイ公式のリポジトリから、Pico用のリポジトリをクローンします。

```
$ git clone https://github.com/raspberrypi/pico-tflmicro.git
```

Pico用の環境セットアップ・スクリプトが用意されているため、簡単に環境構築できます。ラズベリー・パイ4上でスクリプトを実行して20分程度(ラズベ