

Googleのマイコン向けフレームワーク TensorFlow Lite for Microcontrollers

谷本 和俊

マイコン×人工知能のいま

● GoogleもArmも本気を出してきた

2019年末から2020年にかけてマイコンAIの世界で大きな2つの動きがありました。1つはTensorFlow Liteのマイコン正式対応です。これまでGitHub上で実験的な実装(experimental内での実装)が展開されていましたが、experimentalを脱し、正式にメンテナンスされていくことになりました。

もう1つはArmから機械学習の実行に最適化されたCortex-M55コアとNPU(Neural Processing Unit) Ethos-U55の発表です。同社からはCortex-M55を使用すると最大15倍、Ethos-U55を使用すると最大480倍の機械学習性能が得られると発表⁽¹⁾されています。

● エンド・ポイントAI化でIoTのIがIntelligenceになる

これまでAIと言えば、クラウド・ベースのAIか、GHzクラスの周波数で動作するCPUやGPUにGバイト超のメモリを持ったプラットフォームで実行するエンド・ポイントAIでした。これに加え、電池で動作するようなCPU周波数で、メモリ量にも制約のあるマイコン・クラスまでもがAIアプリケーションを実行可能となります。一部ではマイコンでのAIアプリケーションをエンド・ポイントAIと区別して表現し始めています。CES 2020で掲げられたIntelligence of Thingsが現実のものになり、IoTの「I」の意味がInternetからIntelligenceへと変わる節目になるかもしれません。

● 既に手に入るマイコン&開発経験で時代を先取り

原稿執筆時点では、これらの新しいCPUコアやNPUを搭載したデバイスはまだ手に入りません。そこで現状でも手に入る、Cortex-M4などをサポートするTensorFlow Lite for Microcontrollersを使ってエンド・ポイントAIアプリケーションを開発します。そ

の上で、特につまずきそうなポイントやカスタマイズ方法を紹介します。

スポーツ・センシングの世界でプレイヤーを対象とする場合、ウェアラブル機器に搭載されるようなマイコンでのAIアプリケーション開発を体験していただき、今後のエンド・ポイントAIアプリケーション開発への移行・着想に役立てていただければ幸いです。

本稿ではまず、TensorFlow Lite for Microcontrollersのサンプル・アプリケーションに対応しているマイコン・ボードSparkfun Edge(SparkFun製)を使い、プログラムのビルドと実行手順を紹介します。そしてマイコン上でのTensorFlow Liteの性能を確認します。次にオリジナルのアプリケーションとして、スクワットのカウントを題材に、オリジナル学習モデルの作り方を説明します。

マイコン向けフレームワーク TensorFlow Lite for Microcontrollers

● TensorFlow との違い

TensorFlowはGoogleが開発したオープンソースの機械学習ソフトウェア・ライブラリです。TensorFlowで学習したモデルによる推論処理を、モバイル機器や組み込みデバイスでローカル実行可能とするモデル変換の仕組みがTensorFlow Liteという位置付けになります。さらにメモリ資源に制約のあるマイコンで推論処理を実行可能にしたものが、TensorFlow Lite for Microcontrollersです。

このソフトウェア・ライブラリにはクラウドへのアクセスを前提としてサーバで学習、推論を実行するTensorFlowと、学習済みモデルの推論処理をAndroidやiOSなどのプラットフォーム上で実行するTensorFlow Liteがあります。それに加えて学習済みモデルをC言語のバイト配列に変換し、動的メモリ割り当てを必要とせずマイコン・クラスで実行可能にしたものがTensorFlow Lite for Microcontrollersという区分けになります。