

ラズベリー・パイでAIを動かす

佐藤 聖

ソフト①…マルチコア用ニューラル・ネットワーク 高速計算パックNNPACK

● ラズベリー・パイの4コアCPUを生かせる

ラズベリー・パイのCPUを普通に使っただけでは機械学習の積和演算を並列処理するには限界があります。NNPACK⁽¹⁾はニューラル・ネットワーク計算用の高速化パッケージです。NNPACKはマルチコアCPU用のconvolutional層の高性能実装を提供することを目的としています。

Darknetも1つのCPUで処理するよりも、複数CPU処理の方が高速に動作します。Darknetのソースコードを、configureコマンドで実行する際に、NNPACKが利用できます。しかし、あらかじめ環境を整備しないと利用できません。

筆者の環境では、Raspbian Stretch with desktop and recommended softwareのバージョン(April 2019)で構築しています。microSDカードは、8Gバイト以上のものを利用するとよいでしょう。Raspbianを一通り環境設定やネットワーク設定済みです。PandasなどのPythonライブラリも設定済みです。

この状態からセットアップを開始する前提です。

● 構築ツールNinjaのインストール

NNPACKのビルドには、構築ツールNinjaをインストールします。また、Ninjaが必要とする関連構築ツール(PeachPy, confu, re2cなど)が必要になるので、最初にセットアップする必要があります。次に以下のコマンドでツール群をインストールします。

```
sudo pip install --upgrade git+  
https://github.com/Maratyszczca/  
PeachPy  
sudo pip install --upgrade git+  
https://github.com/Maratyszczca/  
confu  
sudo apt-get install re2c
```

次にNinjaをセットアップします。次のコマンドでソースコードをダウンロードして、releaseブラン

チに切り替えます。

```
git clone https://github.com/ninja-  
build/ninja.git  
cd ninja  
git checkout release
```

Ninjaをビルドします。

```
./configure.py --bootstrap
```

ビルドが終わったらlsコマンドでninjaファイルが表示されるか確認します。バイナリ・ファイルが見つければ、ビルドに成功していますので構築ツールのセットアップは完了です。

● Ninjaへのパス設定

これからNNPACKのビルドにNinjaを使いますが、パス設定されていないと使えませんので、

```
export NINJA_PATH=$PWD
```

を実行して一時的にパスを設定しておきます。Ninjaは、NNPACKのビルドにしか使用しないため、ここでは環境設定ファイルに書き込みません。

● NNPACKをGitHubから入手&ビルド

引き続きNNPACKをセットアップします。cdコマンドでユーザのホーム・ディレクトリに戻り、次のコマンドでGitHubのリポジトリからNNPACKのソースコードをダウンロードしてビルドします。

```
git clone https://github.com/  
shizukachan/NNPACK  
cd NNPACK  
confu setup  
python ./configure.py --backend  
auto
```

もしgitがなければ

```
sudo apt-get install git
```

でインストールできます。そしてNNPACKのインストールに必要な環境変数やライブラリが正しく設定、設置されているかなどをチェックします。処理が終わったら\$NINJA_PATH/ninjaコマンドでNNPACKのソースコードをビルドします。