



Pytorch実践

-環境構築&インストール編-

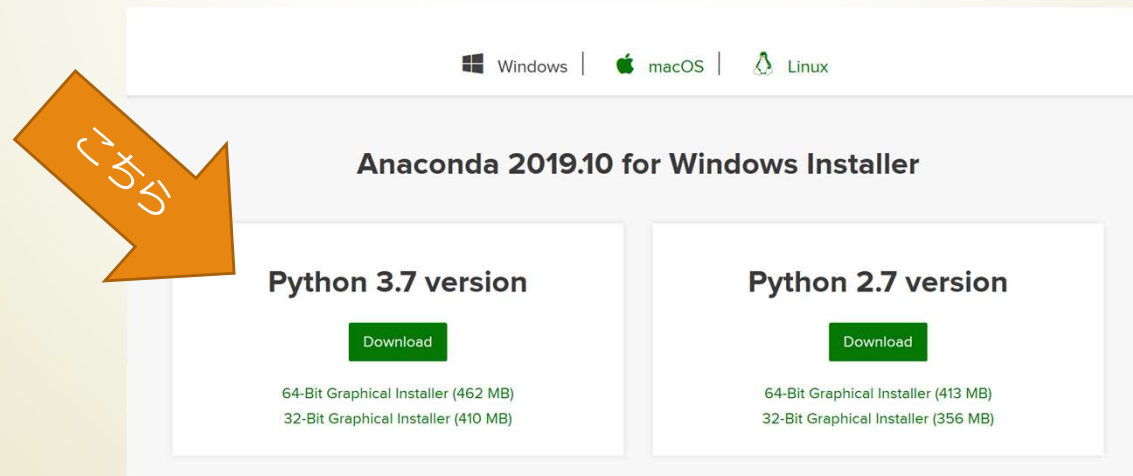
作成者: 上智大学 小松里奈

はじめに

- 本資料では、以下のハードウェア環境で深層学習ライブラリ"Pytorch"の環境構築を行っています
 - OS: Windows10 (64 bit)
 - GPUあり: NVIDIA製品
- 環境構築は、どのOSもおおかた以下の流れで実行します
 - 1. Pythonのインストール
 - 2. PythonでGPUを使えるように環境構築
 - CUDA & cuDNNのインストール
 - 3. Pytorchのインストール
- 使用したい深層学習ライブラリによっては、環境構築に必要な項目のバージョンが異なることもあるのであらかじめ調べておくこと！

1. Pythonのインストール

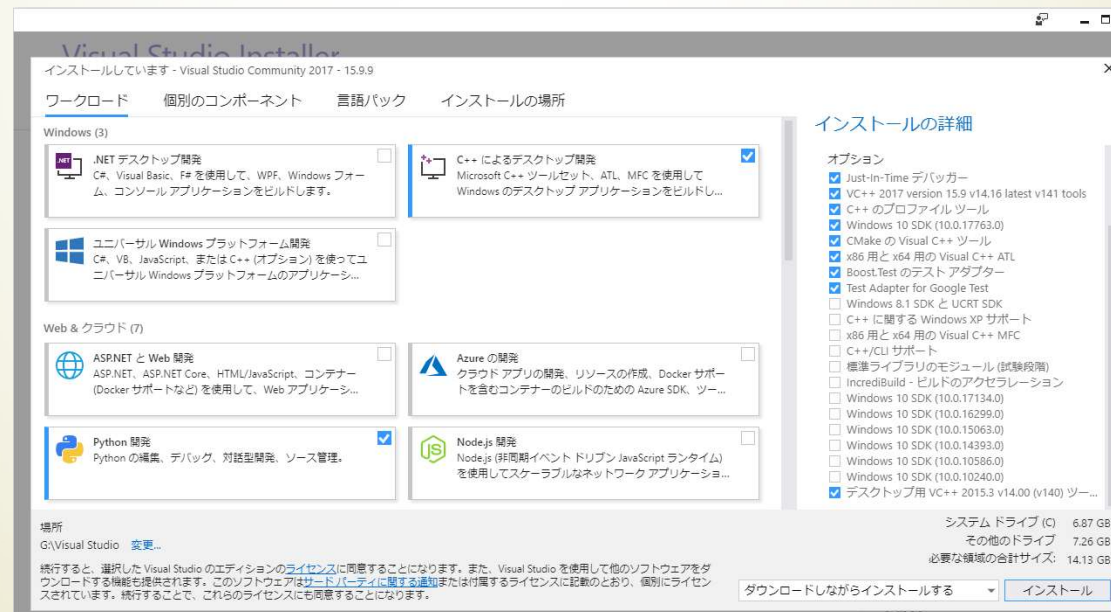
- ▶ Anacondaを通じてPython3をインストール
 - ▶ Anacondaは科学計算向けのパッケージが詰まっているオープンソースディストリビューション
 - ▶ 後々のPythonパッケージのインストールに必要なコマンドも入っているので便利
- ▶ <https://www.anaconda.com/distribution/> からAnacondaをインストール



2. PythonでGPUを使えるように環境構築(For Windows) - CUDAのインストール

➡ 2.1.1 Visual Studioのインストール

- ➡ インストールするパッケージにてPythonも選択
- ➡ ※画像はVisual Studio2017 Communityのインストールの様子です。Visual Studio 2019でも環境構築は行えます(2020.02.04時点)



2. PythonでGPUを使えるように環境構築(For Windows)

- CUDAのインストール

➡ 2.1.2 GPUドライバのチェック

➡ GPUドライババージョンは

<https://www.nvidia.co.jp/Download/index.aspx?lang=jp>
からチェックします

➡ Windowsの場合、GPUドライバ名はデバイスマネージャーから見られます



サイト
にて入力



検索を
クリック



2. PythonでGPUを使えるように環境構築(For Windows)

- CUDAのインストール

2.1.3 ドライバとCUDAの対応チェック

- ➡ <https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html> のTable1から対応できるCUDAをチェックします

GEFORCE GAME READY DRIVER

バージョン: **442.19** ^{WHQL}
リリース日: 2020.2.3
オペレーティングシステム: Windows 10 64-bit
言語: Japanese
ファイルサイズ: 529.54 MB

ダウンロード

CUDA TOOLKIT DOCUMENTATION

Table 1. CUDA Toolkit and Compatible Driver Versions

CUDA Toolkit	Linux x86_64 Driver Version	Windows x86_64 Driver Version
CUDA 10.2.89	>= 440.33	>= 441.22
CUDA 10.1 (10.1.105 general release, and updates)	>= 418.39	>= 418.96
CUDA 10.0.130	>= 410.48	>= 411.31
CUDA 9.2 (9.2.148 Update 1)	>= 396.37	>= 398.26
CUDA 9.2 (9.2.88)	>= 396.26	
CUDA 9.1 (9.1.85)	>= 390.46	
CUDA 9.0 (9.0.76)	>= 384.81	
CUDA 8.0 (8.0.61 GA2)	>= 375.26	
CUDA 8.0 (8.0.44)	>= 367.48	>= 369.30
CUDA 7.5 (7.5.16)	>= 352.31	>= 353.66
CUDA 7.0 (7.0.28)	>= 346.46	>= 347.62

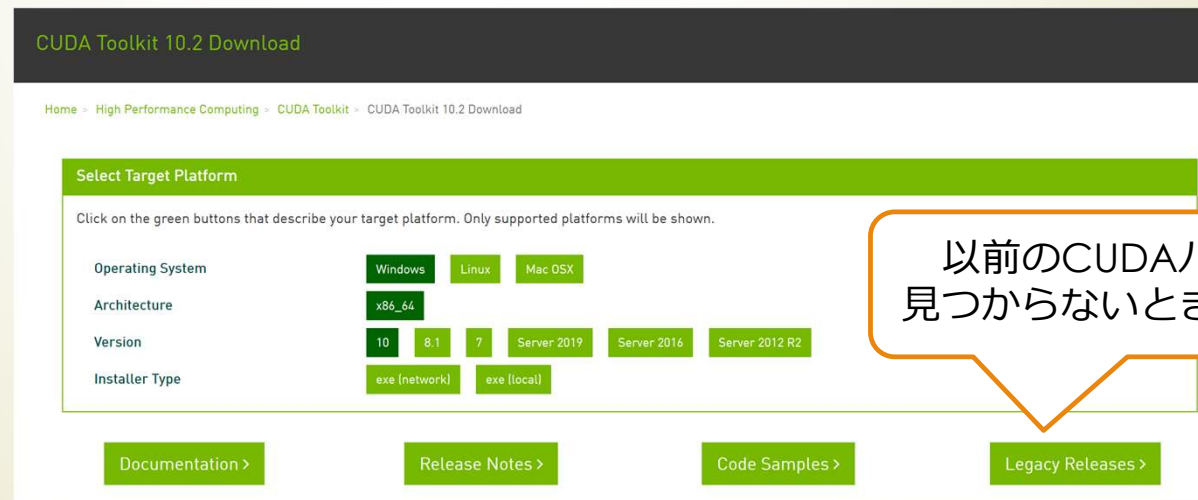
CUDA10でもOK!

2. PythonでGPUを使えるように環境構築(For Windows)

- CUDAのインストール

2.1.4 CUDAのインストール

- 注意 :深層学習ライブラリによっては対応できないCUDAバージョンもあるので併せてチェックすること！ ※PytorchはCUDA10でも対応
- <https://developer.nvidia.com/cuda-downloads> から、該当項目をクリックしてダウンロード
 - exe(local)、exe(network)どちらでも大丈夫
- インストーラーを起動し、案内通りに実行



2. PythonでGPUを使えるように環境構築(For Windows)

- cuDNNのインストール

➡ 2.2.1 NVIDIA Developerの会員登録

- ➡ <https://developer.nvidia.com/rdp/form/cudnn-download-survey> に移動。会員登録を済ませていない場合、"Join Us"から会員登録
- ➡ 登録を終えたら"Login"

NVIDIA Developer Program Membership Required

The file or page you have requested requires membership in the NVIDIA Developer Program. Please either log in or join the program to benefits of the NVIDIA Developer Program here.

Login

Join now

2. PythonでGPUを使えるように環境構築(For Windows)

- cuDNNのインストール

- 2.2.2 インストールしたCUDAのバージョンに合わせてcuDNNライブラリをダウンロード
 - Windows10の場合であれば、"cuDNN Library for Windows10"を選択しzipファイルをダウンロード

以前のcuDNN
バージョン
の場合はここから

cuDNN Download

NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.

☒ I Agree To the Terms of the [cuDNN Software License Agreement](#)

Note: Please refer to the [Installation Guide](#) for release prerequisites, including supported GPU architectures and compute capabilities, before downloading.

For more information, refer to the cuDNN Developer Guide, Installation Guide and Release Notes on the [Deep Learning SDK Documentation](#) web page.

[Download cuDNN v7.6.5 \[November 18th, 2019\], for CUDA 10.2](#)

[Download cuDNN v7.6.5 \[November 5th, 2019\], for CUDA 10.1](#)

[Download cuDNN v7.6.5 \[November 5th, 2019\], for CUDA 10.0](#)

[Download cuDNN v7.6.5 \[November 5th, 2019\], for CUDA 9.2](#)

[Download cuDNN v7.6.5 \[November 5th, 2019\], for CUDA 9.0](#)

[Archived cuDNN Releases](#)

2. PythonでGPUを使えるように環境構築(For Windows)

- cuDNNのインストール

2.2.3 ダウンロードしたファイルらの移動

- 解凍したら、cuDNNライブラリの中にある各フォルダの中身を

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\ (CUDAのバージョン) ※デフォルトでCUDAをダウンロードした場合

の対応したフォルダに上書き保存

ダウンロードしたcuDNNライブラリ

名前	更新日時
bin	2020/02/04 14:...
include	2020/02/04 14:...
lib	2020/02/04 14:...
NVIDIA_SLAT_cuDNN_Support.txt	2020/02/04 14:...

ファイル

フォルダの中にあるファイル
を選択して移動
上書き保存

C:\Program Files\NVIDIA GPU Computing
Toolkit\CUDA\ (CUDAのバージョン)

名前	更新日時
bin	2018/01/16 17:...
doc	2018/01/16 16:...
extras	2018/01/16 16:...
include	2018/01/16 17:...
jre	2018/01/16 16:...
lib	2018/01/16 16:...
libnvvp	2018/01/16 16:...
nvml	2018/01/16 16:...
nvvm	2018/01/16 16:...
src	2018/01/16 16:...
tools	2018/01/16 16:...
CUDA_Toolkit_Release_Notes.txt	2017/11/04 12:...
EULA.txt	2017/11/04 12:...

3. Pytorchのインストール

※すでにtorch, torchvisionをインストールしていたら、uninstall コマンドでモジュールを削除しておくこと

➡ 3.1 <https://pytorch.org/> からpytorchインストールのためのインストールコマンドの取得

➡ 作者はpipコマンドを選択しインストールしました

該当項目が見つからない場合はここから

The screenshot shows the PyTorch website's 'QUICK START LOCALLY' section. It provides instructions on how to install PyTorch based on user preferences. The page includes a table of options for PyTorch Build, Your OS, Package, Language, and CUDA. The 'Stable (1.4)' build is selected, and the 'Windows' OS is chosen. The 'Conda' package manager is selected, and the 'Python' language is chosen. The '10.1' CUDA version is selected. The resulting command to run is: `conda install pytorch torchvision cudatoolkit=10.1 -c pytorch`.

PyTorch Build	Stable (1.4)	Preview (Nightly)		
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python	C++ / Java		
CUDA	9.2	10.1	None	

Run this Command:

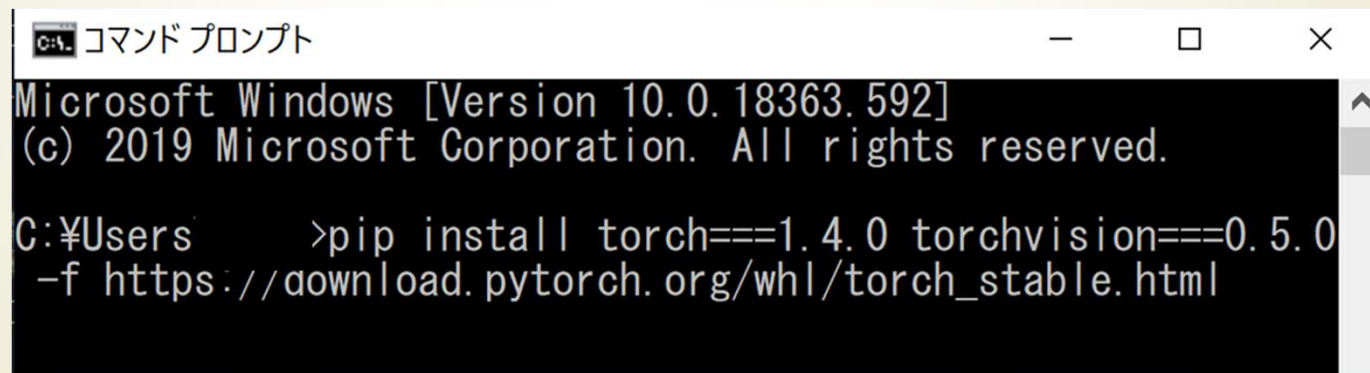
```
conda install pytorch torchvision cudatoolkit=10.1 -c pytorch
```

Previous versions of PyTorch

項目の内容に合わせて、インストールコマンドが表示！

3. Pytorchのインストール

- ▶ 3.2 コマンドプロンプトを起動しインストールコマンドをCopy&Paste、そしてEnter
 - ▶ PytorchがインストールできればOK!



```
コマンド プロンプト
Microsoft Windows [Version 10.0.18363.592]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\      >pip install torch==1.4.0 torchvision==0.5.0
-f https://download.pytorch.org/whl/torch_stable.html
```

最後に

- コマンドプロンプトにて"python"をコマンドし起動
- 以下のコードで、「pythonにpytorchをインストールできているか」「(GPU環境の場合)CUDAが認識されているかをチェック」

"import torch"
でエラーがでないか

プロンプト - python

```
>>> import torch
>>> torch.cuda.is_available()
True
>>>
```

"torch.cuda.is_available()"
でTrueが表示されるか
(GPU環境の場合)

アドバイス

- ここまでの手順の途中でエラーが出て行き詰った場合は、**ブラウザで検索を掛けてみるのもあり**
 - まずはヒューマンエラーがないかチェック
 - Ex. cuDNNライブラリのファイルを正しく移動させていない
インストールする際のOSの選択を間違えていた
 - 環境構築は行き詰るもの
 - 備忘録として対処法を記録したり、インターネットで質問する人も多い！
 - stackoverflowやgithub issueなどに同じ内容のエラーで引っかかっている人がいないかを調べてみる！