_{歯科AI研究会} PyTorch環境構築

セットアップマニュアル

Windows11を中心に説明しますがMacでも可 ただし、MacではNVIDIA GPUは不可です (M1 MacのMPSは利用可能です) Intel Macは未検証です Python3.8以上が動作するPCが必要です

> Anaconda Navigation 2.6.2 (CUDA 11.8 + cuDNN 8.9.7_cuda11) python 3.8.19 PyTorch 2.3.1 Jupyter Notebook

(作成) 岐阜大学工学部電気電子・情報工学科 岐阜大学人工知能研究推進センター 東海国立大学機構健康医療ライフデザイン統合研究教育拠点 原 武史+chatGPT

2025年2月2日公開

1

はじめに

このマニュアルは、PyTorchを利用した深層学習の環境を構築するために、GPU対応のWindowsPCを構築し、その上にAnacondaで仮想環境を作り、プログラミングの環境を構築するために作成しました.

構築は

Phase 1	CUDAとcuDNNのインストール
---------	-------------------

- Phase 2 Anacondaのインストールとpython環境の構築
- Phase 3 PyTorch環境の構築
- Phase 4 基本的な動作確認

の4つからなります.

Windowsで, **対応のGPUを搭載している場合には, Phase1から設定**を行なってく ださい. すでにCUDA, cuDNNのインストールされている場合にはバージョンの確 認をお願いします.

Windowsを利用していても**GPUがない場合、もしくは、MacOSを利用の場合に は、Phase 2から設定**を行なってください、MacOSから操作する場合には、いわ ゆるNVIDIAの「GPU」は利用できません。

M1/M2チップ搭載のMacは, Phase 2から設定が可能です。十分にチェックできて いませんが, M1/M2 MacのMetal Performance Shader(MPS)も使えそうです。 M3はまだ持ってませんので未チェックです。



目 次

Phase 1 CUDAとcuDNNのインストール

GPU対応のWindowsPCを構築する 実行環境 Windows11 RTX3060 12GB CUDA 11.8 cuDNN 8.9.7

Phase 2	Anacondaのインストール	٤p	ython環境の構築	11
		_		

Anacondaをインストールして仮想環境を作る

実行環境

Anaconda 2.3.2くらいで

python 3.8を使う(python 3.9以降はライブラリで難あり)

Phase 3 PyTorch環境の構築	20
構築した仮想環境にPyTorchをインストールする	
GPU対応と非対応で少し操作が異なります	

実行環境

PyTorch 2.3.1

<u>Phase 4 基本的な動作確認</u>

30

4

基本的な動作確認をする

<u>補</u>	足	35
	どのGPUはどのバージョンのCUDAが必要か?	35
	Apple Metal Performance Shaders (MPS)の利用	36

DentalAI2025

Phase 1

CUDAとcuDNNのインストール

GPU対応のWindowsPCを構築する

CUDA11.8, cuDNN 8.9.7_cuda11を例に説明します お持ちのGPUの種類を調べましょう GPUのCompute Capabilityを確認してバージョンを決めてください (補足を参照)

> Linux/Windows対応のPyTorchの安定版(2.3.1)は CUDA 11.8を推奨のようです (7/20/2024現在)

<u>GPUを利用する場合:利用するGPUの判定が必要です・NVIDIA限定です</u>

Windows11を対象に説明します。Macは利用できません。 Compute Capability(CC)が3.5から対応です。まとめはp.35を参照。 https://developer.nvidia.com/cuda-gpus NVIDIA RTX2080はCC=7.5です。RTX30x0はCC=8.6です。 1080はCC=6.1です。1650TiはCC=7.5です。 Win10/11用のCUDA11.8とcuDNN 8.6.0が必要です。

GPU Step1: CUDAのダウンロード

以下からダウンロードします. 最新がありますが, 11.8にします. https://developer.nvidia.com/cuda-toolkit-archive CUDA Toolkit 11.8 を探します Windows, x86_64, 10もしくは11, localです. 3.0GBあります. ダウンロードします.

CUDA Toolkit 11.8 Downloads						
Salast Target Blatform						
Click on the green buttons that describe your target platform. Only supported platforms will be shown. By downloading and using the software, you agree to fully comply with the terms and conditions of the CUDA EULA.						
Operating System Linux Windows						
Architecture x86_64						
Version 10 11 Server 2016 Server 2019 Server 2022						
Installer Type exe (local) exe (network)						
Download Installer for Windows 11 x86_64						
The base installer is available for download below.						
> Base Installer Download (3.0 GB)						
Installation Instructions:						
 Double click cuda_11.8.0_522.06_windows.exe Follow on-screen prompts 						
The checksums for the installer and patches can be found in <u>Installer Checksums</u> . For further information, see the <u>Installation Guide for Microsoft Windows</u> and the <u>CUDA Quick Start Guide</u> .						

GPU Step2: CUDAのインストール

ダウンロードした実行ファイルを実行し,指示通りインストールします. 実行環境にもよりますが,ファイルが大きいためダブルクリックのあと 実行されるまでに無反応の時間が長いです.複数起動しがちです. ご注意ください. インストール作業自体には15分程度必要です.

<u>GPU Step3: cuDNNのダウンロード</u>

以下のページからダウンロードします. ダウンロードには、NVIDIAのアカウントが必要です. ログイン要求などされて、認証されるとダウンロードできます.

cuDNN Archive

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

Download cuDNN v8.9.7 (December 5th, 2023), for CUDA 12.x

Download cuDNN v8.9.7 (December 5th, 2023), for CUDA 11.x

Local Installers for Windows and Linux, Ubuntu(x86_64, armsbsa)

Local Installer for Windows (Zip)

https://developer.nvidia.com/rdp/cudnn-archive Download cuDNN v8.9.7 for CUDA 11.xを探します. その中に「Local Installer for Windows (Zip)」があります.

2000	cudnn-wind	ows-x	(86_64-8.	9.7.: ×	+				
÷	\rightarrow	\uparrow	С		> ダウンロ	コード >	cudnn-windows-x86_64-8.9.7.29_cuda	11-archive.zip >	
÷	新規作成 、		26	c (ò Ø	R	① ↑↓ 並べ替え ~ = 表示 ~	宿 すべて展開	
1							種類		
Į	◀ ギャラリー				💼 cudnn-windows-x86_64-8.9.7.29_cuda11-archive ファイル フォルダー				

ダウンロードしたら、ZIPファイルを展開します.

GPU Step4: cuDNNのインストール

展開すると以下のようなファイル構成が見えます.



「bin」「include」「lib」の3つのフォルダーが見えます.

GPU Step5: cuDNNのインストール:インストール先の確認

CドライブのProgram Filesの中を見ます. その中に「NVIDIA GPU Computing Tookkit」フォルダがあります. その中に「CUDA」フォルダがあります. その中に「v11.8」フォルダーがあります. これを<u>「インストール先フォルダー」</u>とします. インストール先フォルダーの中身は、以下のようになります.

	>	PC >	Windo	ows (C:)	>	Program	n Files >	NVIDI/	A GPU Co	omputing Toolkit	>	CUDA	>	v11.8	>
)	lõ		R	Ũ	∿	並べ替え	~ ≡	表示 ~							
1	4	名前		~			更新日時			種類	t	ナイズ			
		bin					2023/07/1	3 13:51		ファイル フォルダー					
		compute	sanitizer				2023/07/1	3 13:47		ファイル フォルダー					
		extras					2023/07/1	3 13:47		ファイル フォルダー					
		include					2023/07/1	3 13:51		ファイル フォルダー					
*		lib					2023/07/1	3 13:47		ファイル フォルダー					

これが見えない場合にはCUDAのインストールに失敗しています.

次のステップで,先にダウンロードしたcuDNNのファイルコピーします. コピーには管理者権限が必要になる場合があります. <u>GPU Step6: cuDNNのインストール:「bin」の中身をコピー</u>

ダウンロードしたファイルの「bin」の中にある 「cudnn64_8.dll」などすべてのファイルを インストール先フォルダーの「bin」の中にコピーします.



まだまだつづくよ~

<u>GPU Step7: cuDNNのインストール: 「include」の中身をコピー</u>

ダウンロードしたファイルの「include」の中にある

「cudnn.h」などを

インストール先フォルダーの「include」の中にコピーします.



まだまだつづくよ~

<u>GPU Step8: cuDNNのインストール: 「lib」の中身をコピー</u>

「lib」のコピーは少し異なります.

ダウンロードしたファイルの「lib」の中のファイルをコピーします ただし、コピー先は、CUDA/v11.8/lib の中の「x64」フォルダーです 「lib」ではなくその中の「x64」フォルダーがコピー先です 注意してコピーします



GPU Step9: 念のため再起動します

ライブラリ系が増えたので念のため再起動します.

以上でCUDAとcuDNNのインストールは終了です.

DentalAI2025

Phase 2

Anacondaのインストールとpython環境の構築

Anacondaをインストールして 仮想環境を作る

比較的新しいPC上のWindowsだと Python 3.9である必要があるようです しかしライブラリの都合上Python 3.8を利用します **Step 15以降の手順でPythonのバージョン変更が 必要な場合があります** PyTorch 2.3.1はPython3.8以上が必要です 3.9は未確認です Python3.8以降が動作しないPCは諦めてください <u>Step1:</u>ウェブサイトからダウンロードしてインストールします. Anacondaは有料化さましたが,個人の研究利用においては無償で利用できます. リンク先:https://www.anaconda.com/download/success



Step 2: 「Download」をクリックしすると、ダウンロードが始まります. この時点で、ダウンロードするPCの種類が選択されています. 上の図の例では、Windows用のAnacondaであり、 Python3.12を利用する環境のAnacondaがインストールされるようです. Macの場合には、Downloadのアイコンがアップルになります. Intel系とM1/M2系が選べます.



<u>Step 3:</u>ダウンロードされたファイルを起動します. インストーラが起動します. そのまま設定を変更せずインストールします.



<u>Step 4:</u>インストーラーを起動すると、次の画面が表示されます。 「Next >」などをクリックして、そのまま進めます。

<u>Step 5</u>: 「Install」をクリックすると、実際のインストール作業が開始されます. しばらく時間がかかります.

Step 6: インストールが終了すると最後の案内が表示され完了できます。

<u>Step 7</u>: 完了すると,メニューに「Anaconda」が表示されるようになります。折りたたまれているかもしれません.この中で「Anaconda Navigator (anaconda3)」をクリックします.



Step 8: 起動すると次のような画面が開きます. 場合によっては、アップデートを促す画面がでます.しつこいです.

	NAVIG		ite Now Connect
Home	All application	Notebooks	
Environments Learning	1		1
Community	PyChari The Python II	da Toolbox 0.15 16 Addition	
	combines the in notebooks with i assistance, Anacor I	Targed with a suite of rat, starting with the start Al chatbot.	
aconda Toolbax ercharged		Update Application Start There's a new version of Anaconda Navigator available. We strongly recommend yo	× nu to
al notebooks. It the Toolbox to Install.		An update. If you click yes, Anaconde Navigator will close and then the Anaconde Navigator Upd start.	ater will
Documentation	Anaconda	Do you wish to update to Anaceeda Navigator 2.6.1 now?	
Anaconda Blog	Cloud-hosted r Anaconda. Lai environment wi	Don't No, don't show again No, remind me later <u>Yes</u>	

出たらアップデートしましょう. 警告が重なって表示が見にくくなります. アップデートは,

・アップデートをする

・Anacondaを終了する

・アップデートする(くどい)

の順です。ウィンドウで終了します。

<u>Step 9:</u>起動すると次のような画面が開きます(2)

左側に「Environments」のタブがあるのでそれをクリックします. この

「Environments」がPCの中に作る計算環境を表します. つまり, 1台のPCの中に 複数の環境を作り実験ができます.

次に、新しい環境を作ります.下の「Create」をクリックします.

O Anaconda Navigator				-	- 0 X
File Help					1
	DA.NAVIGATOR	2			Connect ~
✿ Home	Search Environments	٩	Installed	Channels Update index Search Pa	ackages Q
Environments	base (root)	0	Name 🗸	T Description	Version ^
Learning			_anaconda_depends	O Simplifies package management and deployment of anaconda	2024.06
Community			abseil-cpp	O Abseil common libraries (c++)	↗ 2021110
			aiobotocore	O Async client for aws services using botocore and aiohttp	2.12.3
			diohttp	O Async http client/server framework (asyncio)	3.9.5
			aioitertools	O Asyncio version of the standard multiprocessing module	0.7.1
Anaconda Toolbox		Ì	🗹 aiosignal	O Aiosignal: a list of registered asynchronous callbacks	1.2.0
local notebooks. Click the Toolbox			alabaster	O Lightweight, configurable sphinx theme	0.7.16
tile to Install.			altair	O A declarative statistical visualization library for python	5.0.1
Read the Docs			anaconda-anon- usage	O Basic anonymous telemetry for conda clients	0.4.4
Anaconda Blog			anaconda-catalogs	O Client library to interface with anaconda cloud catalogs service	0.2.0
			anaconda-client	O Anaconda.org command line client library	1.12.3
y 👜 🗣 🏅	🕒 🎼 -5 i	t, 🗊	The second stands	Charles and Barris Economic de standards	× *
	Create Clear Import Ba	ckup Remove	509 packages available		
	and the second s				

Step 10:新しい環境の名前(Name)の入力欄と使うPythonのバージョンが表示 されます. Pythonの欄のタブを展開します. 表示内容でこのあとの作業が分岐します.

A Home	Search Environments		۹)	Installed	~	Channels	Update in
The Environments	base (root)	•	>	Name	· т р	escription	
Learning	•	Create new e	environment	🛛 anaconda denends	o si	molifies package X	managemen
Community		Name:	JAMIT2024	_ру38			aries (c++)
		Location:	C:\Users\takeshi	anaconda3\envs\JAMIT2024_py3	18		services us
		Packages:	Python	3.12.4	~)	rver framev
			R	2.7.18		1	he standard
1000				3.5.6			
Anaconda Toolbox				3.6.13		Create	gistered asy
local notebooks.				3.7.16		ghtweight, conf	igurable sphir
tile to Install.				3.8.19		daclarativa stat	istical visualiza
Read the Docs				3.9.19		dectaracive stat	Iscical visualiza
Documentation				3.10.14		sic anonymous	telemetry for
				3.11.9		ient library to ir	nterface with
Anaconda Blog				✓ 3.12.4		aconda.org con	nmand line cli
Y You 🖓	0 I A	Ť		•••	0.	increased mail	

場合によっては、たくさんのバージョンのPythonが表示されます. もし「3.8.19」が表示されれば、それを選択してください.

環境名(Name)は、JAMIT2024_py38(名前は自由)としています. この例のように、3.8が表示されない場合もあります. その場合、3.9系や3.10/11/12系を選びそのまま進みます. この後のステップでバージョンを下げます. <u>Step 11:</u>環境名は自由に命名できます.python3.8を選べた場合には,環境名を「DAI2025_py38」とします.Pythonは3.9.12など3.9系しか選べない状態だの場合は「DAI2025」としてみましょう.入力後「Create」をクリックします.

<u>Step 12</u>:新しく作成した環境名は、次の図のようにEnvironmentsに現れます. これで計算環境の大元ができました。緑色の三角をクリックすると、「Open Terminal」と「Open with Python」が見えます. 「Open Terminal」をクリック して、環境構築を行います.

O ANACON	DA.NAVIGATOF	2		
A Home	Search Environments	٩	Installed	v (
Environments	base (root)		Name	T Descrip
Learning	JAMIT2024		ca-certificates	O Certific
Community	JAMIT2024_py38	0	Open Terminal	O A porta
-	•		Open with Python	O Openss protoco
			Open with IPython	O Pypa re
			Open with Jupyter Notebook	O Genera
Anaconda Toolbox Supercharged			< setuptools	O Downlo
local notebooks.				_ Implem

<u>Step 13</u>:「Open Terminal」を選択すると、次の画面が開きます。環境名「DLS39」が書かれています。



Step 14: このターミナルの中にコマンドを投入して環境を構築します.まずは, 最新のバージョンにアップデートします.そのために,このターミナルの中に,次 のコマンドを入力します.

conda update -n base -c defaults conda

確認を求められるので「y」を入力してリターンキーを押します。

<u>Step 15</u>: このターミナルの中に「python」コマンドを投入してpythonを起動してみます. そうすると, pythonのバージョンが3.8.19であることがわかります. quit()で終了します.



以上でAnacondaのインストールと 仮想環境の構築は終了です.

注意(1)

比較的新しいPC上のWindowsだと Python 3.9である必要があるようです.

注意(2)

ちょっと古い(2014年頃)のPC上のWindowsだと Python 3.8は動かないようです。 その場合には、Step 15の画面で conda install python=3.8

を投入してください.

その結果,その環境のPythonのバージョンが3.8になります.

注意(3)

環境名は別の名前でもよいです。

ただし英文半角文字で数値を先頭にしないをお勧めします。

Anacondaは頻繁にバージョンアップされます. 環境名の削除がうまくいかない,削除しても同じ名前が 作れないような不具合がありました DentalAl2025

Phase 3

PyTorch環境の構築

構築した仮想環境に PyTorchをインストールする

注意

GPU対応と非対応で操作が少し異なります

<u>Step 1:</u>ターミナルを開きます. 前のステップでターミナルを開いていればそのままでもよいです.

Anaconda Navigator				
File Help				
🔘 ANACOI	NDA.NAVIGATOF	2		
•				
ft Home	Search Environments	٩	Installed	✓ Channels
Trvironments	base (root)		Name	✓ T Description
Learning	JAMIT2024		🗹 anyio	O High level comp loop implement
Community	JAMIT2024_py38	0	Open Terminal	O Alliance for ope
			Open with Python	O The secure argo
			Open with IPython	

Step 2: ターミナルにコマンドを入力します:色々なライブラリインストール
 GPU対応版と非対応版(CPU版), Mac版共通です。
 1行ずつターミナルに入力します。頻繁に確認(yで回答)が求めらます。

conda update -n base -c defaults conda conda install jupyter matplotlib imageio pandas conda install scikit-learn scikit-image statsmodels conda install cython tqdm seaborn conda install -c conda-forge openjpeg jpeg conda install -c anaconda libjpeg-turbo conda install -c anaconda libpng conda install -c conda-forge imgaug timm albumentations conda install -c conda-forge pydicom conda install -c anaconda pillow natsort conda install -c anaconda pillow natsort conda install conda-forge::opencv (もしくは conda install anaconda::opencv) conda install openpyxl

<u>Step 3</u>: ターミナルにコマンドを入力します: PyTorchインストール 前のステップでターミナルを開いていればそのままでもよいです GPU対応版と非対応版(CPU版)でコマンドが異なります.

<u> PyTorchは2.3.1をインストールします.</u>

状況に合わせてどれか(1)~(3)のいずれか1行を入力してください.

Windowsの人は(1)か(2)のいずれか.

- (1)はPhase1のようにGPUの設定をした人
- (2)はCPU環境のみで実行する人.

Macな人は(3)

 (3)はIntel系MacもしくはM1/M2 Macな人 MacOSは12.3以上であること MacでもM1/M2(MPS利用はApple Siliconのみ) Intel版ではCPU版として動作する可能性あり(未検証)

(1) Win+GPU版・Phase1で設定したGPU対応版・Windowsの場合:

conda install pytorch torchaudio pytorch-cuda=11.8 -c pytorch -c nvidia conda install torchvision=0.18.1 pytorch-cuda=11.8 -c pytorch -c nvidia

(2) Win+CPU版・Phase1でGPUの設定をしていないCPU版・Windows

conda install pytorch torchaudio cpuonly -c pytorch conda install torchvision=0.18.1 cpuonly -c pytorch

(3) Mac+MPS(Metal performance shaders対応・MacOS

conda install pytorch::pytorch torchvision torchaudio -c pytorch conda install -c conda-forge lapack conda install -c conda-forge libblas

<u>Step 4:</u>追加でインストール conda install conda-forge::vit-pytorch <u>Step 5</u>: すべてがインストールできると構築した環境の中に「Open with Jupyter Notebook」が現れます。それを選択します。

File Help				
🔵 ANACON	IDA.NAVIGATOF	2		
•				
1 Home	Search Environments	٩	Installed	~ Cł
Environments	base (root)		Name	✓ T Descrip
Learning	JAMIT2024		🗹 anyio	O High lev loop imp
Scommunity	JAMIT2024_py38	0	Open Terminal	O Alliance
			Open with Python	O The sec
			Open with IPython	O Low-lev
			Open with Jupyter Notebook	The ast
Anaconda Toolbox		_	<	generat

<u>Step 6</u>: 選択すると「Jupyter Notebook」が起動します. ブラウザ経由で起動し ます. Edge (Windows10/11) やSafari (Mac) で利用できます.

Step 7: 早速,入力してみましょう.右上の「New」をクリックすると,「Notebook」が表示されます. それを選択します.

💄 🔞 🗖 🗘 Home	X 🔵 Home	× 🤗 ViT02	× +			-	• >	×
\leftarrow C (i) localhost:8889/tree			A ^N 🏠	ф	¢ 🕀	%	🤻	2
💭 Jupyter								
File View Settings Help								
Files Running Select items to perform actions on them. /					- New	± Upload	C	
Name				L	E Console) >	lize	
🗆 🖿 anaconda3					E New Fil	e		
anaconda3_2024					ID New Fo	lder		
anaconda3_old					5 months ag	0	-	
anaconda3_old2					yesterda	У		

Step 8: スクリプトを入力する画面が表示されます.



Step 9: 緑の枠の中に次のスクリプトを入力してみましょう. 実行するときは、シフトキーを押しながらリターンを入力します. Windows/GPU版の場合はバージョンが表示されます Windows CPU版/Mac版の場合は長文のエラーが出ます

import torch
print(torch.__version__)
print(torch.cuda.is_available())
print(torch.cuda.device_count())
print(torch.cuda.current_device())
print(torch.cuda.get_device_name())
print(torch.cuda.get_device_capability())

```
[1]: import torch
 print(torch.__version__)
 print(torch.cuda.is_available())
 print(torch.cuda.device_count())
 print(torch.cuda.current_device())
 print(torch.cuda.get_device_name())
 print(torch.cuda.get_device_capability())
2.3.1
True
1
0
NVIDIA GeForce RTX 3060
(8, 6)
```

<u>(M1/M2なMacな人の場合は以下, Trueが表示されればMPS利用可能)</u>

import torch
print(torch.backends.mps.is_available())
print(torch.backends.mps.is_built())



<u>Step 10:</u>バージョンの確認

これで、インストールされたPyTorchのバージョンが表示されればOKです。GPUの 種類や、GPUを搭載していない場合には、結果が異なります。表示されなければ、 環境構築に問題があります。

Step 11: OpenCVなどインストールしたライブラリのバージョンも確認しましょう.
 以下を投入. バージョンらしき数字が表示されれば動作します.
 読み込みができない場合には,途中でエラーがでます.
 その際には一度再起動を試みてください.
 cv2のエラーがなかなかしつこいかもしれません.
 必要に応じて「Step 3」の操作を確認してください.

import matplotlib import imageio import pandas import sklearn In [1]: import matplotlib import cython import imageio import cv2 import pandas import sklearn import PIL import cython import pydicom import cv2 import PIL print(matplotlib.__version__) import pydicom print(imageio. version) print(matplotlib. version) print(pandas.__version__) print(imageio.__version) print(sklearn.__version__) print(pandas.__version__) print(cython. version) print(sklearn.__version__) print(cv2.__version__) print(cython.__version__) print(cv2.__version__) print(PIL. version) print (PIL.__version__) print(pydicom. version) print(pydicom.__version__) 3.7.1 2.26.0 1.5.3 1.2.2 0.29.35 4.6.0 9.3.0 2.4.0

Step 12: 作成したスクリプトは名前をつけて保存できます.以下の例は, 「Save as...」から「MyScriptO1」と名前をつけて保存しています.

💭 ju	pyter	Unt	itled (auto	saved
File	Edit	View	Insert	(
New	Notebook	(•	
Open				
Make	a Copy			١٥٣
Save	as			rsi
Rena	me			
Save	and Che	ckpoint	Ctrl-S	_
Reve	rt to Cheo	ckpoint	•	ərs
Print	Preview			3101
Dowr	nload as		•	
Trust	ed Noteb	ook		
Close	e and Half	t		

Save As	×
Enter a notebook path relative to notebook dir	
MyScript01	
	Cancel Save

Step 13:保存先は特に指定しないと、ホームディレクトリの直下になるようで す.以下の例は、ユーザ名「user」でそのホームディレクトリの内容です.一番下 に「MyScriptO1.ipynb」として保存されています. 拡張子は「.ipynb」で、自動的 に付加されます. あらかじめフォルダーを作っておいたほうが良いと思われます.



Step 14: 終了するときは, Close and Haltから行います. 複数のスクリプトが実行した状態のままになり, コンピュータのリソースを無駄に使う場合があるので, 確実に終了します. 実行中のままになっているスクリプトは「Running」と表示されます. 意図していないプログラムが動かないよう, うまく管理しましょう.

💭 ju	pyte	r MyS	Script01	(au
File	Edit	View	Insert	
New	Notebool	¢	•	
Open	l			
Make	a Copy			0.0
Save	as			rsi
Rena	me			
Save	and Che	ckpoint	Ctrl-S	-
Reve	rt to Che	ckpoint	•	ars
Print	Preview			sic
Dowr	nload as		•	
Trust	Noteboo	k		-
Close	e and Hal	t	_	

TestScripts	5日前
C Videos	4日前
MyScript01.ipynb	Running 10分前 1.31 k
🗌 ┛ Untitled.ipynb	12分前 1.31 k

Phase 4

基本的な動作確認

基本的な動作確認をする 少し練習です GPUが望ましいです (CPU版では未チェック@1/21/2025)

30

<u>Step 1:練習1</u>

保存済みのスクリプトを開く場合には、Jupyter Notebookで「Open」からファイ ルを指定します。もしくは、起動した画面からそのフォルダー/スクリプトにたど り着きます。ダウンロードしたファイルをダブルクリックしても動作しません.

動作確認用のスクリプトのZIPファイルをウェブページからダウンロードしてくださ い. ZIPファイルを解凍して、作業用フォルダーに置きましょう. 3つのフォルダー があります. checkO1の中にある「CheckScript1.ipynb」ファイルを開くと、ス クリプトが表示されます. シフトクリックで動作してください. スクリプトの前半 はインストールしたライブラリのバージョンが表示されます. importに失敗してい ないかエラーをよく確認しましょう.

このサンプルが最後まで実行できれば、最後には美味しそうな「エビの握り」が表示されます。ネタに困るとお寿司です。



<u>Step 2:練習2</u>

動作確認用のスクリプトの「CheckScript2.ipynb」ファイルを開くと、スクリプトが表示されます。シフトクリックで動作してください。 手書き文字認識:MNISTをCNNで解釈します。最後には学習曲線と正解率(ほぼ100%)が表示されます。



<u>Step 3:練習3</u>

動作確認用のスクリプトの「CheckScript3.ipynb」ファイルを開くと、スクリプトが表示されます.シフトクリックで動作してください.ファイルを開くと、スクリプトが表示されます.シフトクリックで動作してください.練習1で使った画像を、事前学習済みの「ResNet101」に入力して、1000分類から該当したカテゴリを出力します.最後には「えび」の画像が「112番 American_lobster」と近からず遠からずの分類結果が得られます.

In [16]:	1 print(out.shape)
t	orch.Size([1, 1000])
In [17]:	<pre>1 with open("./imagenet_classes.txt") as f: 2 labels = [line.strip() for line in f.readlines()]</pre>
In [18]:	1 labels
Out[18]: [<pre>['0, tench', '1, goldfish', '2, great_white_shark', '3, tiger_shark', '4, hammerhead', '5, electric_ray', '6, stingray', '7, cock', '8, hen', '9, ostrich', '10, brambling', '11, goldfinch', '12, house_finch', '12, house_finch', '13, junco', '14, indigo_bunting', '15, robin', '16, bulbul', '17, jay', '18, magpie', '10 = bickedee'</pre>
In [19]:	1 _, index = torch.max(out, 1)
In [20]:	1 labels[index[0]]
Out [20]: '	'122, American_lobster'

以上でセットアップは終了です

どのGPUはどのバージョンのCUDAが必要か?

導入するGPUのHardware Generationを調べそのCompute Capabilityを調べま す. それに応じたCUDAのバージョンを決定します. そして, そのCUDAのバー ジョンに対応したcuDNNをインストールします.

NVIDIAのアーキテクチャとCompute CapabilityおよびCUDAバージョン一覧
(7/20/2024版)

Hardware Generation	Compute Capability	CUDA 10.1	CUDA 11.0-3	CUDA 11.4	CUDA 11.5	CUDA 11.6	CUDA 11.7	CUDA 11.8	CUDA 12.x
Hopper	9.0	NO	NO	NO	NO	NO	NO	Yes	Yes
Ada Lovelace	8.9	NO	NO	NO	NO	NO	NO	Yes	Yes
Ampere	8.0-8.7	NO	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Turing	7.5	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Volta	7.x	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Pascal	6.x	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Maxwell	5.x	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Kepler	3.5	Yes	Yes	Yes	Yes	Yes	Yes	Yes	NO
Kepler	3.0	Yes	Yes	Yes	NO	NO	NO	NO	NO

代表的なGPUの商品名

Generation	CC	代表的な機種名 (GF: GeForce)
Hopper	9.0	H100
Ada Lovelace	8.9	GF 4090-4060, RTX 6000Ada, RTX4000SFF, L4, L40
Ampere	8.0-8.7	GF 3090-3050, RTX A6000-A2000, A100/40/30/16/10/2
Turing	7.5	Titan RTX, GF2080-2060, GTX1660, RTX8000, Tesla T4
Volta	7.x	Titan V, Quadro GV100, Tesla V100/V100S
Pascal	6.x	Titan X, GF 1080-1010, Quadro P6000-1000, Tesla P100/40
Maxwell	5.x	GF GTX Titan X/980, Quadro M6000, Tesla M60/40/10/6/4
Kepler	3.5	GF GTX Titan Z/780-710, Quadro K6000, Tesla K40/20x/20

(参考)

https://docs.nvidia.com/deploy/cuda-compatibility/index.html#support-hardware_table-hardware-support

https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html

Apple Metal Performance Shaders (MPS)の利用

NVIDIAのGPUは, CUDAとcuDNNを用いて深層学習の計算を高速化しました. Appleが提供するGPU環境であるMetal Performance Shaders (MPS)は, Apple Silicon (いわゆるM1やM2)に搭載されたGPU環境であり, PyTorchを利 用した深層学習においても活用できます.

Apple Siliconでは、メインメモリとGPUメモリが統合されるため、NVIDIAの GPUのようにGPU専用の物理メモリの概念がありません。CPUとGPUはメインメ モリを共有するため、メインメモリから自動的にGPU用のメモリを割り当てて実行 されます。その結果、メインメモリが十分に搭載されていれば、NVIDIAのGPUメモ リでは動作しなかったバッチサイズやモデルをMPSでは実行できる可能性がありま す。PyTorchにおけるMPSの利用方法は比較的簡単で、

- ・Apple Siliconを搭載のMacであること
- ・macOS12.3以上であること
- ・MPS対応のPyTorchをインストールすること(Phase 3, Step 3参照)

で環境構築ができます.実行時には、CUDAでは、以下のようにCUDAかCPUの判定を行いデバイスを指定していましたが、

device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')

MPSの場合は、以下のようにMPSの利用判定を行います.

device = torch.device('mps' if torch.backends.mps.is_available() else 'cpu')

deviceにmpsが指定されていれば, M1/M2のMPSがCUDAのように利用できるは ずです. deviceの指定の状況は,

print(device)

で確認できます. CUDA, MPS, CPUの判定を自動で行いたい場合には,

device = 'cuda' if torch.cuda.is_available() else 'mps' if torch.backends.mps.is_available() else 'cpu'

で指定できます.具体的なスクリプトは、CheckScript1を参考にしてください.

このようにデバイスにMPSを指定してCheckScript2のモデルの学習(30 epoch)で評価すると次のような結果になりました。ただし, macOS14.5を除き, PyTorch1.13系です.

OS version	種類	M1/M2	メインメモリ	計算時(秒)	Batch size
macOS 14.5	MB Air	M2, 10GPU	24GB	106	10000
macOS 13.1	MB Air	M2, 10GPU	24GB	203	変更なし
macOS 13.1	MB Air	M2, 10GPU	24GB	84	10000
macOS 13.0	Mac Studio	M1, 48GPU	128GB	186	変更なし
macOS 13.0	Mac Studio	M1, 48GPU	128GB	61	10000
macOS 13.0	MB Pro	M1, 32GPU	64GB	227	変更なし
macOS 12.5	MB Air	M1, 8GPU	16GB	394	変更なし
macOS 12.4	MB Pro	M1, 32GPU	64GB	366	変更なし
macOS 12.5	MB Air	M1, 8GPU	16GB	CPUのみ736	変更なし
macOS 12.4	MB pro	M1, 32GPU	64GB	CPUのみ737	変更なし

MPSを利用した場合の計算速度の比較

CUDAを利用した場合の計算速度の比較

OS version	種類	CPU	GPU	計算時(秒)	Batch size
Windows11	Desktop PC	Core i7	RTX 3060	110	変更なし
Windows11	Desktop PC	Core i7	RTX 3060	72	10000
Windows10	Laptop PC	Core i7	GTX 2070	266	変更なし
Windows10	Laptop PC	Core i7	GTX 1070	287	変更なし
Windows10	Laptop PC	Core i7	_	CPUのみ912	変更なし

Windows+CUDA+GeForceの組み合わせは,変更なしのバッチサイズでもかなり 早いことがわかります.以上のことから,

・M1/M2のGPUの数が少なくても高速化に効果はあり

・M1/M2のMPSの利用はGTX 2070のGPUと同程度の高速化が期待できる

・バッチサイズの大きさは計算時間に影響がある(新しい知識ではない)

とわかりました.

不明なトラブルメモ

OpenCV系が不安定:

インストール順に依存しそう torchvisionとの関連がありそう

<u>急にOpenCVの不具合がでる時の対応策:</u>

一度パッケージを削除して再インストールする
 conda uninstall opencv
 conda install conda-forge::opencv
 (もしくは conda install anaconda::opencv)

conda install -c conda-forge liblapack conda install -c conda-forge libopenblas