



본 영상 교재는
2025년도 과학기술정보통신부 및 정보통신기획평가원의
SW중심대학 사업의 지원을 받아 제작되었습니다.



한국항공대학교
Korea Aerospace University

—KIU—
시용합대학

SW중심대학

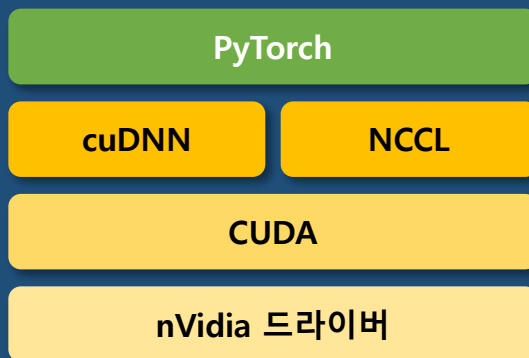


PyTorch 코딩

개발환경 구축

개발환경 구축

- PyTorch의 계층 구조
 - PyTorch를 효율적으로 활용하기 위한 하위 라이브러리



- CUDA** (Compute Unified Device Architecture)는 GPU 코어를 **일반 연산**에 활용하게 도와줌
 - ANN은 수많은 뉴런의 feed-forward와 back-propagation을 **병렬 처리**하여 빠르게 수행하는데 필요함
- cuDNN** (CUDA Deep Neural Network)은 CUDA를 **ANN**에 최적화시켜 ANN의 수행 속도를 높여주는 라이브러리
- NCCL** (nVidia Collective Communications Library)은 다수의 GPU가 **통신**할 수 있도록 도와주는 라이브러리
 - 단일 GPU로 처리하기 어려운 대형 ANN을 여러 GPU가 나누어 처리하는데 필요함

개발환경 구축

- OS에 PyTorch 직접 설치하기
 - CUDA 및 cuDNN 등 다양한 하위 라이브러리를 직접 설치해야 함
 - PyTorch 버전에 맞는 하위 라이브러리를 찾아 설치해야 함
 - PyTorch 버전을 바꿔야 할 때 하위 라이브러리의 버전도 바꿔야 하는 문제가 있음
- Docker를 통한 개발
 - Docker를 통해 특정 버전의 PyTorch에 필요한 하위 라이브러리가 설치되어 있는 가상 환경 제공
 - 설치, 재설치, 제거 등이 매우 간단함
- MS VScode
 - Microsoft의 통합 개발툴인 Visual Studio code
 - Docker 및 원격 개발 환경을 쉽게 활용할 수 있음
- 본 강의는 **MS VScode**와 **Docker**를 활용하여 진행

MS VScode

01

MS VScode

- Windows, Linux, iOS 모두 활용 가능
- MS VScode 설치 방법
 1. MS VScode 홈페이지에서 OS에 맞는 설치 파일 다운로드
 - <https://code.visualstudio.com/Download>
 2. 설치파일 실행하여 개발툴 설치

NVIDIA 드라이버

02

NVIDIA 드라이버

- NVIDIA 그래픽 카드를 활용하기 위한 드라이버
 - NVIDIA 그래픽 카드가 없는 경우 생략
- Windows
 1. NVIDIA 홈페이지에서 그래픽 카드에 맞는 설치 파일 다운로드
 - <https://www.nvidia.com/ko-kr/drivers/>
 2. 다운받은 설치파일 실행
- Ubuntu
 1. 터미널에서 아래 명령 실행
 - > `sudo apt update`
 - > `sudo apt upgrade`
 - > `ubuntu-drivers devices` # 권장 드라이버 이름 확인
 - > `sudo apt install "권장_드라이버_이름"`
 - > `sudo reboot` # 또는 로그아웃 후 다시 로그인

Docker

03

Docker

- Windows

1. Docker 홈페이지에서 **Docker desktop** 설치 파일 다운로드
 - <https://www.docker.com/>
2. 다운받은 설치파일 실행 후 **재부팅**
3. Docker desktop 실행
 - 직접 실행하기 귀찮은 경우, **설정**에서 Windows 로그인(sign in) 할 때마다 자동으로 Docker를 실행하게 할 수 있음

Docker

- Ubuntu

- Ubuntu용 Docker desktop도 있으나, Ubuntu는 원격 AI 서버에 많이 활용됨
- 따라서 본 강의에서는 원격 환경에서 GUI 없이 설치하는 방법으로 진행

1. Docker docs에서 Docker engine 설치 방법 따라하기

- <https://docs.docker.com/engine/install/ubuntu/#installation-methods>
- 터미널에서 **Installation methods**의 1, 2번 cell의 명령어 실행 후 재부팅, 또는 로그아웃 후 다시 로그인

2. NVIDIA docs에서 NVIDIA container toolkit 설치 방법 따라하기

- NVIDIA **그래픽 카드**가 없는 경우 **생략**
- <https://docs.nvidia.com/datacenter/cloud-native/container-toolkit/latest/install-guide.html#installing-with-apt>
- 터미널에서 **Installing with Apt**의 1, 2, 3번 cell의 명령어 실행 후 재부팅, 또는 로그아웃 후 다시 로그인

3. Docker 권한 변경을 위해 터미널에서 아래 명령 실행

- > `sudo usermod -aG docker $USER`
- > `sudo chmod 777 /var/run/docker.sock`

MS VScode와 Docker 연결

04

MS VScode와 Docker 연결

- Docker engine 실행
 - Windows에서는 docker desktop 실행(설정에 따라 자동 실행시킬 수 있음)
 - Ubuntu의 경우 자동 실행됨
- MS VScode에 Docker extension 설치
 1. MS VScode의 **사이드 바**에서 퍼즐모양 아이콘(**Extensions**) 클릭
 2. Docker 검색 후, **Microsoft 인증마크**가 있는 Docker extension 설치
 3. 설치 후, **사이드 바**에 새로 만들어진 고래모양 아이콘(**Docker**) 클릭
 - 느낌표 등 에러 메시지가 없는 경우 정상 설치 됨

MS VScode와 Docker 연결

- PyTorch Docker image 다운로드

1. Docker hub 홈페이지에서 "pytorch/pytorch" 검색

- Docker hub : <https://hub.docker.com/>
- Pytorch Docker image: <https://hub.docker.com/r/pytorch/pytorch>

2. Tag 탭 클릭 후 활용할 버전의 Docker image 다운받기

- 본 강의에서는 "**2.6.0-cuda11.8-cudnn9-runtime**" 활용
- 활용할 버전 옆에 다운로드 코드를 **VScode 터미널**에서 실행

MS VScode와 Docker 연결

- PyTorch Docker container 생성하기

- 개발에 활용할 폴더 생성 후, 터미널에서 해당 폴더로 이동

- 예) "docker"라는 폴더 생성 후, `cd docker`

- 아래 명령 실행

> `docker run --gpus=all --ipc=host --network=host --name=pytorch26 -it -v ${PWD}:/workspace pytorch/pytorch:2.6.0-cuda11.8-cudnn9-runtime`

- NVIDIA 그래픽 카드가 없는 경우 `--gpus=all` 옵션을 제거, 아래와 같이 실행

> `docker run --ipc=host --network=host --name=pytorch26 -it -v ${PWD}:/workspace pytorch/pytorch:2.6.0-cuda11.8-cudnn9-runtime`

- `--name` 옵션은 생성할 container의 이름
- `-v` 옵션은 Docker container 외부와 연결할 폴더를 지정하는 옵션
 - `${PWD}`는 터미널의 현재 위치, `/workspace`는 container 내부에서의 이름
- 실행 후 Docker container 리스트에 "pytorch26"가 생성됐는지 확인

Docker Container 진입

05

Docker Container 진입

- MS VScode에 Dev Containers 설치
 1. MS VScode의 **사이드 바**에서 퍼즐모양 아이콘(**Extensions**) 클릭
 2. Dev Containers 검색 후 설치
- Docker container 진입
 1. MS VScode의 **사이드 바**에서 고래모양 아이콘(**Docker**) 클릭
 2. 생성한 "**pytorch26**" container를 오른쪽 클릭하여 "**Attach visual studio code**" 실행
 - Container가 실행 중이어야 함
 3. 처음 진입했을 땐 초기 세팅을 위한 시간이 걸림
- Tip : 사이드 바에 고래모양 아이콘이 없다면 container 내부로 진입한 것

Docker Container 진입

- Docker container 내부 개발 환경 구축
 1. MS VScode **사이드 바**에서 파일모양 아이콘(**Explorer**) 클릭
 2. Explorer에서 "Open folder"클릭
 3. 접근할 폴더로 **"/workspace"**를 지정, 작업할 파일들을 Explorer에 표시
 4. 사이드 바에서 퍼즐모양 아이콘(**Extensions**) 클릭
 5. Python 검색 후 설치

Have a nice day!

