

NVIDIA RAPIDS LAB

RAPIDS 실습 환경

Chapter 01 GPU가 속한 이터 과학의 시작
준비하기

발표자

김형섭

모두의 연구소 NVIDIA RAPIDS Lab

오늘의 목차

1

RAPIDS란 무엇인가?

GPU 가속의 필요성과 RAPIDS 아키텍처

2

필수 요구사항

하드웨어 및 소프트웨어 요구사항

3

환경별 설치 가이드

Local, WSL2, Colab, Kaggle, AI Workbench

4

핵심 명령어 및

Quick Reference & FAQ
그리드를 가용

왜 RAPIDS가 필요한가?

현재의 한계

NumPy, Pandas, scikit-learn은 모두 CPU 기반
중소규모 데이터에서는 문제없지만, 빅데이터 처리 시 속도가
기하급수적으로 느려집니다.

RAPIDS의 해결책

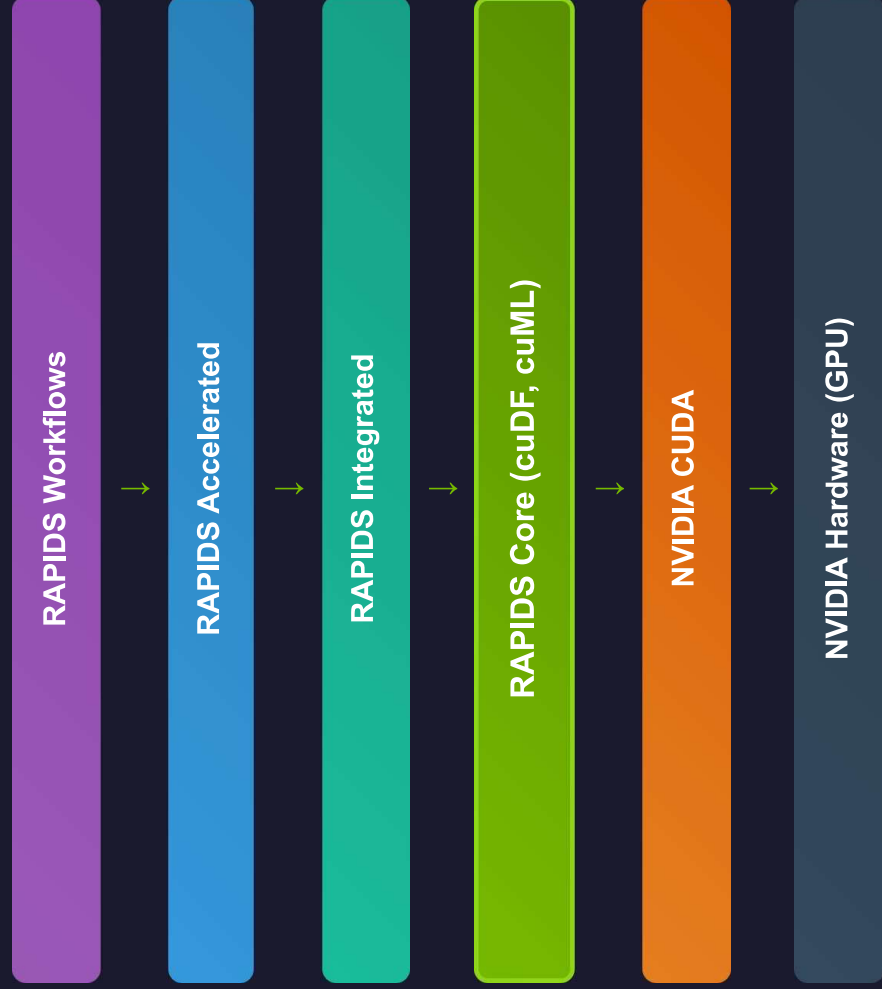
NVIDIA GPU를 활용한 데이터 처리 가속화
기존 코드를 약간만 수정해서 최소 2배 이상의 속도 향상을
경험할 수 있습니다.

기존 GPU 프레임워크 vs RAPIDS

항목	기존 방식	RAPIDS
진입 장벽	높음	낮음
새로운 방법 학습	필요	최소화
러닝커브	높음	낮음
기존 코드 재사용	어려움	약간 수월

핵심: Pandas/scikit-learn과 유사한 API로 GPU 가속!

RAPIDS 아키텍처 스택



초보자를 위한 설명

NVIDIA Hardware

실제 그래픽 카드 (RTX 시리즈 등)

CUDA

GPU를 프로그래밍할 수 있게 해주는 NVIDIA 플랫폼

RAPIDS Core (우리가 사용할 것!)

cuDF (Pandas 대체), cuML (scikit-learn 대체)

상위 레이어로 갈수록

더 추상화되고 사용하기 쉬워짐!

필수 요구사항 : 하드웨어

하드웨어 요구사항

항목	요구사항	설명
GPU	NVIDIA Volta+	GeForce 20 시리즈 이상
Compute Cap.	7.0 이상	GPU 연산 능력 지표
GPU RAM	16GB+ 권장	대용량 데이터 처리 시

Compute Capability 확인 방법

1. NVIDIA 공식 사이트에서 본인 GPU 모델 검색
2. 또는 `nvidia-smi` 명령어로 GPU 모델 확인 후 검색

GPU가 없다면?

걱정 마세요! 클라우드 옵션이 있습니다.

Google Colab (무료)
Kaggle (무료)

주의사항

Windows에서는 RAPIDS 직접 실행 불가!
WSL2를 통해 Linux 환경에서 실행하거나 클라우드 사용 권장

지원 GPU 예시

RTX 2060, 2070, 2080, 3060, 3070, 3080, 3090, 4060, 4070, 4080,
4090, A100, H100 등

필수 요구사항 : 소프트웨어

설치 순서 (의존성)

1. NVIDIA Driver



2. CUDA Toolkit



3. Python (Conda)



4. RAPIDS

중요: 버전 호환성 확인 필수!

RAPIDS 설치 전 반드시 지원 CUDA 버전 확인

공식 문서 확인

<https://docs.rapids.ai/install/>

핵심 확인 명령어

`nvidia-smi`
드라이버 + GPU 상태 확인

`nvcc --version`
CUDA Toolkit 버전 확인

SECTION 03

어떤 환경을 선택해야 할까?

NVIDIA GPU + Linux

Local

최대 성능

NVIDIA GPU + Windows

WSL2

Linux 환경

설정이 어려움

AI Workbench

원클릭 설정

GPU 없음 / 빠른 테스트

Google Colab

무료 GPU

데이터 분석 학습

Kaggle

GPU + 커뮤니티

환경별 비교				
환경	난이도	비용	성능	추천
Local	중간	무료	최고	Linux 사용자
WSL2	어려움	무료	높음	Windows + GPU
Colab	쉬움	제한된 무료	중간	입문자
Kaggle	쉬움	제한된 무료	중간	데이터 분석

Google Colab에서 RAPIDS 사용하기

Colab의 장점

- 설치 필요 없음 - 웹 브라우저지만 OK
- 무료 GPU 제공 (T4)
- cuDF, CuPy 이미 설치됨!

Step 1: GPU 런타임 설정

런타임 → 런타임 유형 변경 → T4 GPU

```
!nvidia-smi
```

Step 2: cuDF 바로 사용!

```
import cudf
import cupy
```

Pandas vs cuDF 비교

Pandas (CPU)

```
import pandas as pd
df = pd.DataFrame({"a": [1, 2, 3]})
```

cuDF (GPU)

```
import cudf
gdf = cudf.DataFrame({"a": [1, 2, 3]})
```

문법이 거의 동일!

cuML 사용 시 추가 설치

```
!git clone https://github.com/rapidsai/rapidsai-csp-utils.git
!python rapidsai-csp-utils/colab/pip-install.py
```

제한: 무료 최대 12시간, 유료 시 연결 해제

Kaggle에서 RAPIDS 사용하기

Kaggle이란?

- 데이터 분석 경연 플랫폼
- 현업 문제 데이터로 상금 경쟁
- 코드 공유 및 토론 가능
- 무료 GPU 제공 (T4 x2)

시작하기

1. <https://www.kaggle.com> 가입
Google 계정으로 간편 가입
2. Code → New Notebook
3. Competitions에서 대회 참가

주의사항

다른 사람 코드 Fork 후 수정 없이 Public 공유 시 표절로 계정 제재 가능!

RAPIDS 설정 방법

- 1 New Notebook 생성
- 2 Session options 설정
ACCELERATOR → GPU T4 x2
- 3 ENVIRONMENT 설정
Always use latest environment

바로 사용 가능!

```
import cudf
cudf.__version__
```

Colab과 마찬가지로 cudf가 기본 설치되어 있음

Colab vs Kaggle: GPU T4 x2 제공으로 Kaggle이 더 강력!

Linux Local에 RAPIDS 설치하기

Step 1: NVIDIA 드라이버

```
sudo apt update
sudo ubuntu-drivers autoinstall
nvidia-smi # 확인
```

Step 2: CUDA Toolkit

```
sudo sh cuda_<version>_linux.run
nvcc --version # 확인
```

Driver 체크 해제! (이미 설치됨)

Step 3: Miniforge 설치

```
wget "https://github.com/conda-forge/miniforge/..."
bash Miniforge3-*.sh
```

Step 4: RAPIDS 설치 (핵심!)

```
conda create -n rapids-24.08 \
-c rapidsai -c conda-forge -c nvidia \
rapids=24.08 python=3.11 \
cuda-version=12.5 # 본인 환경!
```

```
conda activate rapids-24.08
```

설치 옵션 설명

`-c rapidsai`
`-c conda-forge`
`cuda-version`

RAPIDS 공식 채널
커뮤니티 패키지
본인 CUDA 확인 필수!

설치 확인

```
python -c "import cudf; print(cudf.__version__)"
```

WSL2에서 RAPIDS 설치하기

WSL이란?

Windows Subsystem for Linux
Windows에서 Linux를 실행하는 기술

요구사항

- Windows 10 이상
- WSL2 (WSL 1 미지원)
- Compute Cap. 7.0+
- 단일 GPU만 지원

Step 1: WSL 설치

PowerShell (관리자):

```
ws1 --install Ubuntu-22.04  
ws1 --update
```

Step 2: Driver 설치

```
ws1 -d Ubuntu-22.04 --user root  
apt-get update && apt-get upgrade  
apt install nvidia-driver-535
```

Step 3: CUDA 설치

```
wget https://developer.download.  
nvidia.com/.../cuda-wsl-ubuntu.pin  
sudo apt-get -y install cuda  
sudo apt install nvidia-cuda-toolkit
```

Step 4: Miniconda

```
wget https://repo.anaconda.com/  
miniconda/Miniconda3-latest-Linux-x86_64.sh  
chmod +x ./Miniconda3-*.sh  
./Miniconda3-latest-Linux-x86_64.sh
```

Step 5: RAPIDS 설치

```
conda create -n rapids-24.04 \  
-c rapidsai -c conda-forge \  
-c nvidia rapids=24.04 \  
python=3.10 cuda-version=11.5
```

```
conda activate rapids-24.04
```

설치 확인

```
nvidia-smi --query-gpu=name,  
memory.total --format=csv
```

```
python -c "import cudf"  
python -c "import cuml"
```

난이도 높음!

어렵다면 Colab/Kaggle 또는 AI Workbench 추천

NVIDIA AI Workbench

AI Workbench란?

NVIDIA가 제공하는 통합 개발 도구
복잡한 환경 설정을 단일 플랫폼으로 간소화!

자동으로 설정되는 것들

WSL Docker NVIDIA Driver CUDA Conda

RAPIDS

장점

- 환경 설정 복잡도 대폭 감소
- 프로젝트별 독립 환경 관리
- JupyterLab 바로 사용 가능

시작하기

- 1 공식 문서 접속
docs.nvidia.com/workbench/
- 2 Workbench 다운로드 및 설치
- 3 GitHub에서 프로젝트 복제
- 4 빌드 완료 후 JupyterLab 실행!

필요한 준비물

WSL (Windows) NVIDIA 계정
Docker Git 계정

WSL2 수동 설치가 어렵다면 AI Workbench 추천!

SECTION 04

핵심 명령어 Quick Reference

설치 확인 명령어

```
nvidia-smi
```

드라이버 및 GPU 상태

```
nvcc --version
```

CUDA Toolkit 버전

```
nvidia-smi --query-gpu=name,memory.total --format=csv
```

GPU 상세 정보

RAPIDS 확인

```
python -c "import cudf; print(cudf.__version__)"
```

Conda 환경 관리

```
conda create -n [이름] [패키지]
```

가상환경 생성

```
conda activate [환경명]
```

환경 활성화

```
conda deactivate
```

환경 비활성화

```
conda env list
```

환경 목록

Tip: (rapids-24.08) 표시 = 환경 활성화 성공!

SECTION 04

자주 발생하는 문제 및 해결

Q1: **nvidia-smi** 실행 안 됨

원인: NVIDIA 드라이버 미설치

해결: **NVIDIA** 드라이버 재설치

Q3: **RAPIDS** 설치 시 버전 충돌

원인: CUDA와 RAPIDS 버전 불일치

해결: docs.rapids.ai/install/ 확인

Q2: **nvcc --version** 실행 안 됨

원인: CUDA 미설치 또는 PATH 미설정

해결: `~/bashrc`에 **CUDA PATH** 추가

Q4: **WSL**에서 **GPU** 인식 안 됨

원인: WSL1 사용 또는 드라이버 문제

`ws1 --set-version Ubuntu-22.04 2`

가장 중요한 것!

RAPIDS 설치 전 반드시 <https://docs.rapids.ai/install/> 에서 지원하는 CUDA 버전을 확인하세요.

추가 도움이 필요하다면?

[RAPIDS GitHub Issues](#) [NVIDIA Developer Forums](#) [Stack Overflow \(rapids 태그\)](#)

오늘의 핵심 정리

1

RAPIDS = GPU 가속

Pandas/scikit-learn 과 동일한 API로 2배+ 속도 향상

3

설치 전 필수 확인

RAPIDS 지원 CUDA 버전, GPU Compute Capability

2

다양한 환경 선택지

GPU: Local/WSL2, 없으면: Colab/Kaggle

4

핵심 명령어

nvidia-smi, nvcc --version

입문자 추천

순서:

Colab



Kaggle



Local/WSL2

THANK YOU

Q & A

궁금한 점이 있으시면 질문해 주세요!

발표자: Harrison

모두의 연구소 NVIDIA RAPIDS Lab

docs.rapids.ai

github.com/rapidsai