

Green-ML MANUAL

Green-ML 프로젝트 사용설명서

이 프로젝트는 AI 학습 및 추론 시 발생하는 탄소 배출량을 실시간으로 모니터링하고 시각화하는 솔루션입니다.

1. 프로젝트 주요 기능

- 실시간 탄소 대시보드: CPU/GPU 사용량에 기반한 탄소 배출량 실시간 트래킹
- 탄소 영수증 (**Carbon Receipt**): 학습 세션 종료 시 감성적인 UI의 영수증 발행
- AI 어드바이저: 탄소 배출량을 줄이기 위한 학습 중단 및 최적화 가이드 제공

2. 기술 스택

- **Backend**: FastAPI (Python 3.12)
- **Frontend**: React Native (Expo)
- **Data Agent**: psutil 기반 실시간 전력 소모량 측정

3. 시작하기 (Quick Start)

Backend 서버 실행

```
cd backend
python -m venv venv
source venv/bin/activate # Windows: venv\Scripts\activate
pip install -r requirements.txt
uvicorn main:app --reload
```

모바일 앱 실행

```
cd app
npm install
npx expo start
```

탄소 측정 에이전트 실행

```
cd measurement
python train_resnet_v3.py # 또는 inference_realesrgan.py
```

4. 디렉토리 구조

- app/ : React Native 모바일 앱 소스 코드

- backend/ : FastAPI 기반 백엔드 API 서버
- measurement/ : 탄소 배출량 측정 및 AI 모델 실행 스크립트
- docs/ : 프로젝트 관련 문서 및 아카이브

5. 외부 코드 연동 가이드 (Integration Guide)

기존의 Python 스크립트나 실험 코드에 Green-ML의 탄소 트래킹 기능을 추가하는 방법입니다.

1) 필수 패키지 설치

```
pip install codecarbon requests
```

2) 코드 삽입 (예시)

아래 코드를 참고하여 프로젝트에 삽입하세요.

```
import requests
from datetime import datetime
from codecarbon import OfflineEmissionsTracker

# 1. 트래커 초기화 및 시작
# country_iso_code="KOR" 설정으로 한국 전력 기반 탄소 배출량 계산
tracker = OfflineEmissionsTracker(
    country_iso_code="KOR",
    save_to_file=False,
    log_level="error"
)
tracker.start()

# --- 여기에 실험 코드를 작성하세요 ---
# 예: model.train() 또는 model.predict()
# -----

# 2. 트래킹 종료 및 데이터 수집
emissions_data = tracker.stop()
try:
    duration = tracker.final_emissions_data.duration
    energy = tracker.final_emissions_data.energy_consumed
except AttributeError:
    duration = 0.0
    energy = 0.0

# 3. Backend 서버로 전송 (영수증 발행용)
API_URL = "http://127.0.0.1:8000/record" # 백엔드 주소 확인
payload = {
    "project_name": "My-Awesome-Experiment", # 대시보드에 표시될 프로젝트 이름
    "emissions": float(emissions_data),
    "energy_consumed": float(energy),
```

```
"duration": float(duration),
"timestamp": datetime.now().isoformat()
}

try:
    requests.post(API_URL, json=payload, timeout=5)
    print("Green-ML: 세션 기록 성공!")
except Exception as e:
    print(f"Green-ML: 기록 실패 ({e})")
```

3) 결과 확인

1. `npx expo start` 로 모바일 앱을 실행합니다.
2. 앱 화면을 아래로 당기면 메인 화면 혹은 '기록' 탭에서 방금 전송한 프로젝트 이름(My-Awesome-Experiment)과 탄소 배출량을 확인할 수 있습니다.