

0xFF주차 보고서

202002489

Docker의 이해

도커는 컨테이너 기반의 오픈소스 가상화 플랫폼이다. 배에 싣는 화물 수송용 박스인 컨테이너를 생각해 보면 이해하기 쉽다. 서버에서 이야기하는 컨테이너 또한 그와 비슷하다는 점에서다. 다양한 프로그램, 실행환경을 컨테이너로 추상화하고 동일한 인터페이스를 제공하여 프로그램의 배포 및 관리를 단순하게 해준다. 백엔드, 프로그램, 데이터베이스, 서버 등 어떤 프로그램도 컨테이너로 추상화할 수 있고 어디에서든 실행할 수 있다는 장점이 있다. 격리된 공간에서 프로세스가 동작하는 기술으로, 가상화 기술의 하나지만 기존방식과는 차이가 있다. 기존의 가상화 방식은 주로 OS를 가상화하였는데, 이 방식은 여러 가지 OS를 가상화할 수 있고 비교적 사용법이 간단하지만 무겁고 느려서 운영환경에선 사용할 수 없었다. 이러한 상황을 개선하기 위해 CPU의 가상화 기술(HVM)을 이용한 KVM(Kernel-based Virtual Machine)과 반가상화 Paravirtualization방식의 Xen이 등장하게 된다. 이 방식은 전체 OS를 가상화하는 방식이 아니기 때문에 호스트형 가상화 방식보다 더 나은 성능을 보인다. 이러한 기술들은 OpenStack이나 AWS, Rackspace같은 클라우드 서비스에서 가상 컴퓨팅 기술의 기반이 되었다는 점에서 의의가 있다.

Docker 제출 도입의 목적

실행환경이 함께 포함된 코드를 제출하여, 데이터의 개입을 줄여 객관적이고 효율적으로 코드를 평가하기 위해서다. 기존에는 ipynb 파일과 이를 실행하기 위한 requirements.txt라는 최소한의 환경만 제출물로 받았다. 이때 버전 상의 충돌이나 운영체제상의 충돌 등 다양한 문제가 발생하여 제출자들의 일부 코드를 수정하게 되는 문제가 파생된다. 코드의 수정은 코드 평가의 객관성이 훼손되고 시간도 오래 걸린다는 문제점으로 이어진다. 이러한 문제점을 해결하기 위해서 Docker 제출을 도입하게 되었다.

Docker 설치

-submit 폴더 생성

-inference.py 작성

- (1) submit/data/ 폴더에서 데이터 불러오기
- (2) submit/input/ 폴더에서 모델 불러오기
- (3) submit/output/ 폴더에 추론 결과 저장하기

-Dockerfile 작성

- (1) 베이스 이미지 가져오기
- (2) 리눅스에 필요한 라이브러리 설치
- (3) 폴더 경로 설정하기
- (4) python 라이브러리 설치하기

느낀 점

도커에 대한 기본적인 개념을 숙지할 수 있어서 유익했다. 가장 기억에 남았던 부분은 도커 클라이언트의 커맨드 명령어가 굉장히 잘 구축되어 있다는 사실이다. 확장성이 좋으며 직관적이고 사용하기 쉬운 명령어로 시스템 구성을 이해하지 못하더라도 편하게 사용할 수 있다는 장점을 알 수 있었다.