인공지능 모델 연구 환경 지원을 위한 연구소프트웨어 개발 통합 프레임워크

조민희[1], 김다솔[1], 송사광[1], 이상백[1], 이미경[1], 임형준[1]
[1]한국과학기술정보연구원 연구데이터공유센터
mini@kisti.re.kr, dskim@kisti.re.kr, esmallj@kisti.re.kr,
jerryis@kisti.re.kr, sb_lee@kisti.re.kr, hjlim@kisti.re.kr

Research SW Development Integrated Framework to Support AI Model Research Environments

Minhee Cho[1], Dasol Kim[1], Sa-kwang Song[1],
Sang-Baek Lee[1], Mikyoung Lee[1], Hyung-Jun Yim[1]
[1]Research Data Sharing Center,
Korea Institute of Science and Technology Information

요 약

소프트웨어를 개발하거나 실행하는 환경은 매우 다양하다. 최근에 혁신을 이끌고 있는 인공지능 모델은 오픈소스 프로젝트를 통해 공개되는 코드나 라이브러리를 활용하여 구현하는 경우가 많다. 하지만 실행을 위한 환경 설치 과정이 쉽지 않고, 데이터 혹은 기학습된 모델 사이즈가 대용량일 경우에는 로컬 컴퓨터에서 실행하는 것이 불가능한 경우도 발생하고, 동료와 작업을 공유하거나 수동 배포의 어려움 등 다양한 문제에 직면한다. 이러한 문제를 해결하기 위하여, 소프트웨어가 유연하게 동작할 수 있도록 효율적인 리소스를 관리할 수 있는 컨테이너 기술을 많이 활용한다. 이 기술을 활용하는 이유는 AI 모델이 시스템에 관계없이 정확히 동일하게 재현될 수 있도록 하기 위함이다. 본 연구에서는 인공지능 모델 개발과 관련하여 코드가 실행되는 환경을 편리하게 관리하기 위하여 소프트웨어를 컨테이너화하여 배포할 수 있는 기능을 제공하는 연구소프트웨어 개발 통합 프레임워크를 제안한다.

1. 서론

연구 재현성 및 투명성이 강조되면서, 연구소프트웨어 공유 및 재사용이 중요해지고 있다[1]. 소프트웨어를 공유하고 재사용함으로써 시간과 비용을 절감하고 개발의 효율성을 높일 수 있다. 또한 논문 혹은 커뮤니티를 통해 품질이 검증된 소프트웨어를 재사용함으로써 신뢰성과 안정성이 향상될 수 있다.

오픈소스코드 공유 사이트인 깃허브(Github) 리포지토리에는 AI와 관련된 모델 코드들이 상당히 많이 공개되어 있고, 최상위에 랭킹되어 있다. 이러한 각각의 오픈소스코드들은 모두 서로 다른 개발환경에서 개발되어 공유되고 있어 다운로드 받더라도 매번 실행이 쉽지 않다. 또한 인공지능모델 아키텍처가 점점 더 복잡해짐에 따라 처음부터 모델을 만들고 전문화하는 것은 점점 더 어려워지고 있다. 현재 인공지능 기반의 개발 방향은 더 나은 패키지 유지 관리 및 종속성 관리를 수행할 수 있도록 기존 소프트웨어 패키지의 재사용 사례가 많다.

최근 이렇게 빠르게 변화하는 인공지능 세계에서 오픈소스 AI 플랫폼인 허깅페이스(Hugging Face)[1)]가 모델 개발자들에게 각광받고 있다. 허깅페이스를 통해 다양한 트렌스포머 모델을 라이브러리 형태로 손쉽게 이용이 가능하다. 대표적으로 자연어처리 모델인 GPT, BERT, RoBERTa 등을 포함한 트랜스포머 아키텍처를 기반으로 사전 훈련된 NLP 모델의 풍부한 컬렉션을 제공함으로써 고품질 어플리케이션을 쉽게 구축할 수 있도록 지원하고 있다. 개발자는 공개된 머신러닝 레퍼런스를 통해 최신 모델을 스스로 구축할 수 있다. Hugging Face의 모델은 PyTorch 및 TensorFlow와 같은 인기 있는 딥 러닝 프레임워크를 기반으로 구축되어 쉽게 통합할 수 있다. 하지만 제공하는 모델이 제한적이고, 모든 모델이 동일한 고품질이거나 특정 작업에 대해 잘 최적화된 것은 아니다. 본 논문에서는 연구소프트웨어를 개발한 제공자가 코드, 라이브러리, API 등을 활용한 개발환경을 포함하여 컨테이너에 패키징하여 쿠

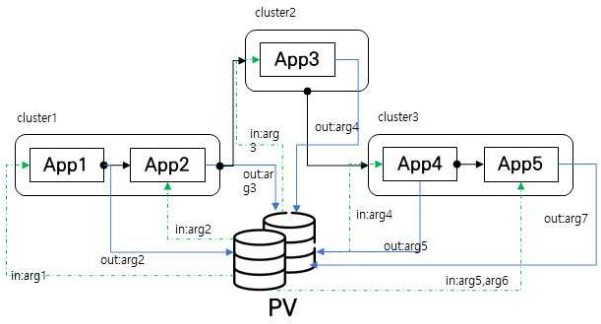
1) https://huggingface.co/

버네티스에 배포하는 과정을 자동으로 처리할 수 있도록 지원하는 연구소프트웨어 개발 통합 프레임워크 활용 방안을 제안한다.

2. 본론

KISTI에서는 2020년부터 국가 연구데이터 커먼즈(KRDC: Korea Research Data Commons)[2] 체계를 구축하고 있고, 그 일환으로 연구소프트웨어[3] 공유·활용 지원을 위해 연구소프트웨어 개발 통합 프레임워크를 개발하고 있다[2]. 기존의 개발된 코드들은 특정 컴퓨팅 환경에서 개발되었고, 코드가 이전될때마다 버그와 오류가 발생해 재현이 어려운 점이 발견된다. 이를 해결하기 위하여 코드 및 실행에 필요한 구성 파일, 라이브러리 및 종속성과 함께 패키징하는 컨테이너화 기술을 활용한다. 패키징된 각각을 컨테이너라고 하고, 각각이 독립적인 환경에서 실행되도록 관리하기 위하여 오케스트레이션 도구인 쿠버네티스를 활용한다. 이를 통해 개별 연구SW를 배포하는데 문제가 없고, 자동으로 다양한 실행환경을 지원하고, 다중의 사용자에게 유연하게 리소스를 제공한다. 따라서 KRDC프레임워크는 쿠버네티스를 기반으로 구축되었다.

인공지능 및 기계학습 연구과정에서 파이프라인은 데이터 수집, 전처리, 모델 학습, 학습 모델 배포, 예측 단계로 정형화되어 있다. 데이터를 준비하고, 정제해 피처를 가공하고 학습시켜 서빙하는 각 과정들을 모듈화하여 단계별로 순차적으로 진행할 수 있다. 이 과정중 데이터 전처리 과정은 데이터를 모델이 이해할 수 있는 형태로 변환하거나 품질을 올리는 작업을 수행하므로 데이터 분석 및 처리 과정에서 매우 중요한 단계이다. 예를 들면 이미지 데이터 전처리에는 이미지 리사이징, 정규화, 데이터증강(회전, 이동, 반전), 노이즈 제거, 객체분리, 컬러채널조작, 배경제거 등 여러 종류가 있다. 문제의 특성 및 데이터의 특성에 따라 적절한 방법을 선택하여 사용한다. 이러한 전처리 과정은 모델 생성 과정에서 누구나 진행하는 반복적인 과정이므로 재사용 빈도가 꽤 높다. KRDC 프레임워크에서는 쿠버네티스에 배포하는 패키징화된 연구 소프트웨어를 APP이라고 정의하고, APP이 외부 입출력을 통해 코드가 실행 가능한 형태로 동작할 수 있도록 템플릿을 제공하고 있다. 현재는 ML에서 많이 활용되는 파이썬 언어를 지원하는 것을 기반으로 한다. 개별 컨테이너에서 실행되는 코드들간에 데이터를 상호운용할 수 있도록 “args”라는 개념을 활용하여 사용자 지정 매개변수를 정의하고 공유스토리지를 활용하여 데이터를 연계할 수 있도록 지원한다. 서로 다른 클러스터에서 실행되더라도 공유스토리지를 통해 데이터 연계가 그림1과 같이 가능하다.



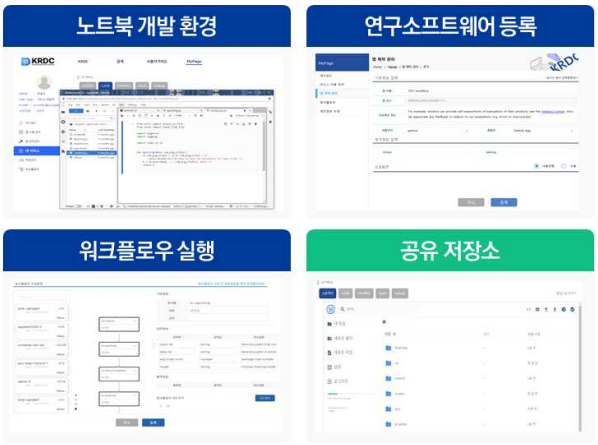
(그림 1) 저장소 공유

연구소프트웨어 개발 통합 프레임워크 주요 오픈소스를 기반으로 구현되었고, 제공하는 기능은 다음과 같다.

- 연구SW 개발: 빠른 AI 개발 환경 구축, 주피터 노트북을 활용한 협업을 통한 공동 개발
- 연구SW 등록: 온라인에서 실행 가능한 APP 형태로 변환하기 위하여 배포하고자 하는 연구소프트웨어의 코드를 업로드, dockerfile 작성, 메타정보 입력, 이미지 빌드
- 이미지 저장 및 공유: Harbor 기반 컨테이너 이미지 저장 관리
- 코드 저장 및 공유: Gitlab 기반 코드 저장 및 관리
- 저장소 공유: MINIO, Ceph 기반 사용자, 컨테이너별 전용 볼륨 할당, 모델의 입출력 데이터를 저장 관리하고, 아티팩트 공유 및 관리
- 워크플로우: 앱을 다양한 형태의 파이프라인으로 디자인 및 ArgoWorkflow 기반 쿠버네티스에 잡이 실행되는 것을 관리
- 검색카탈로그: 앱 및 워크플로우를 검색 및 공유

2) 연구데이터 활용 극대화를 위해 상호 운용 가능한 고품질의 컴퓨팅 리소스에 원활한 사용성(usability)을 제공하는 신뢰할 수 있는 공통 활용 체계로 정의한다

3) 연구 과정 중에 또는 연구 목적으로 생성된 소스코드 파일, 알고리즘, 스크립트, 전산 작업 흐름 및 실행 파일이 포함



(그림 2) 프레임워크 서비스

그림2와 같이 프레임워크가 제공하는 기능들을 통해 연구자는 하나의 인터페이스로 모든 인공지능 모델 배포에 필요한 과정을 실행할 수 있는 통합 환경을 제공받는다. 연구자들은 클릭 몇 번으로 개발한 코드의 자동화된 빌드 및 배포를 제공받으므로 연구소프트웨 개발 생산성 및 운영 안정성을 보장받을 수 있다.

3. 결론

연구소프트웨어 개발 통합 프레임워크는 인공지능 모델 연구 개발 및 공유 생산성을 높이기 위하여 코드를 컨테이너화하여 앱으로 제작 및 배포하는 과정을 자동화하여 지원한다. 또한 인공지능 모델 개발 전과정에서 필요한 다양한 종류의 앱들을 쉽고 간편하게 조합하여 재사용할 수 있는 워크플로우 서비스를 제공함으로써 코드 재사용 및 재현에 도움을 주고 있다. 향후 KRDC 프레임워크를 활용하는 사용자가 증가된다면, 국가 R&D 산출물로 개발된 다양한 인공지능 분야 연구소프트웨어들이 접근 가능하고 활용될 수 있을 것이다. 이는 인공지능 모델 연구 진입 장벽을 낮게 하고, 더 나아가 국가 R&D 연구 생산성을 높이는데 기여할 것이다.

ACKNOWLEDGMENT

이 논문은 2023년도 한국과학기술정보연구원(KISTI)의 기본사업으로 수행된 연구입니다.(과제번호: (KISTI)K-23-L01-C03-S01, (NTIS)1711198423)

참고문헌

[1] FAIR Principles for Research Software version 1.0. Research Data Alliance. (DOI: https://doi.org/10.15497/RDA00068)

[2] 임형준, 이미경, 송사광, 서동민, 조민희.(2022). 데이터 기반 연구개발을 위한 국가연구데이터커먼즈 설계 및 적용 방안.한국지능시스템학회 논문지, 32(5), 392-400.

[3] 조민희, 송사광, 임형준, 이미경, "연구소프트웨어 공유·활용을 위한 학술 인프라 설계", 한국경영과학회 학술대회논문집, 개최지, pp.1315-1317, 2023.