

클라우드 기반 AI 플랫폼 기술 동향

김윤기 (고려사이버대학교), 전미향 (수원여자대학교),
전유부 (동국대학교), 한성수 (강원대학교)

목 차

1. 서 론
2. AI 플랫폼 기술 요소
3. AI 플랫폼 기술 동향 분석
4. 결 론

1. 서 론

최근 들어 AI를 이용한 데이터 분석에 관한 수요가 급증하고, AI를 활용한 서비스의 품질을 높이기 위한 여러 방법들이 고안되고 있다. 특히 신뢰성 있는 AI 서비스 모델을 개발하기 위해 AI 모델 개발의 전 과정을 플랫폼에서 지원받을 수 있도록 하는 기술들이 연구·개발되고 있다. AI 서비스 모델의 신뢰성을 높이기 위해서는 대규모의 학습 데이터 셋이 필요하고, 이를 학습하기 위한 컴퓨팅 자원의 규모도 커져야 한다. 학습 데이터도 시간이 지남에 따라 달라지며 이에 대응하는 컴퓨팅 자원의 규모도 달라질 수 있다. 또한 하이퍼파라미터 튜닝 및 모델 배포에 필요한 컴퓨팅 자원의 규모도 고려해야 한다. AI 모델을 개발하기 위한 절차가 복잡하며, 신뢰성 높은 모델을 구현하기 위해서는 더 많은 사항을 고려해야 한다. 하지만 분석가들이 이 모든 사항을 고려하여 AI 모델을 개발한다면, 많은 시간적 비용을 치러야

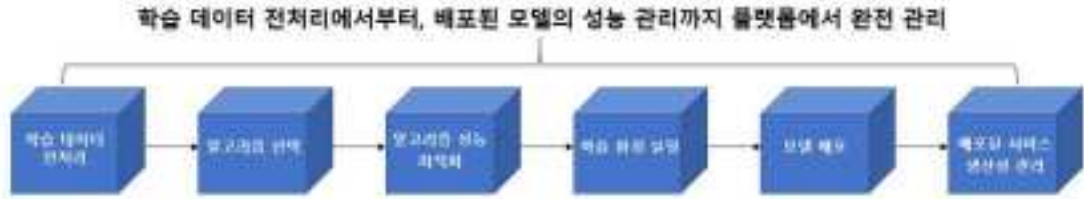
한다.

최신 클라우드 기반의 AI 플랫폼에서는 컴퓨팅 자원을 쉽게 제어할 수 있는 인프라 제어 기술을 활용하여 분석가들이 AI 모델 개발을 쉽게 할 수 있도록 하는 환경을 제공한다. 또한 학습 데이터의 전처리에서부터 모델을 개발하고 배포할 수 있는 기술을 플랫폼 안에서 모두 수행할 수 있도록 하는 End to End 기술로 진화하고 있다.

본 논문에서는 클라우드 기반의 AI 플랫폼의 최신 기술 동향을 소개함으로써, 현재 AI 플랫폼이 나아가고 있는 기술적 방향을 도출하고, 향후 연구 방향에 관해 정리해 본다.

2. AI 플랫폼의 기술 요소

AI 모델을 개발하기까지는 학습 데이터 전처리, 알고리즘 선택, 알고리즘 성능 최적화, 학습 환경 설정, 모델 배포 및 배포된 서비스 생산성 관리가 필요하다. 이러한 전 과정을 플랫폼에서 완전하게



(그림 1) AI 플랫폼에서의 AI 모델 개발 단계

관리하기 위하여 (그림 1)과 같은 과정을 모두 제공하는 환경을 제공한다.

각 분석 모델 개발 단계에서는 각 단계마다 다양한 기법이 수반된다. 첫 번째로 학습 데이터 전처리에서는 데이터 전처리, 피처엔지니어링, 탐색적 분석, 학습 데이터 라벨링과 같은 기술이 필요하다. 두 번째로 알고리즘을 선택 및 개발하는 단계에서는 분석가가 주피터 노트북[1]과 같은 가상 IDE을 활용할 수 있도록 하는 **notebook**, 미리 알고리즘을 사용할 수 있도록 이를 라이브러리로 제공하는 **Built-in Algorithm** 기술 또는 자동으로 알고리즘을 선택하는 **auto selection** 기술 등이 필요하다. 세 번째로, 알고리즘 성능을 최적화하는 단

계에서는 하이퍼파라미터 튜닝 및 모델의 라이프 사이클 관리하는 기술이 필요하다. 네 번째로 학습 환경을 설정하는 부분에서는 학습용 컴퓨팅 자원을 인프라로 지원하는 기술이 필요하며, 마찬가지로 다섯 번째, 모델을 배포하는 단계에서도 모델을 추론하는 데 필요한 컴퓨팅 자원을 관리하는 기술을 필요로 한다. 마지막으로 최적의 AI 모델이 개발되었다고 판단된다면 배포된 모델의 성능을 모니터링 해야 한다. 따라서 배포된 모델의 성능 모니터링 기술이 필요하다. 아래 <표 1>은 각 분석 단계마다 플랫폼에서 필요로 하는 기술들을 정리한 표이다.

분석가는 플랫폼을 활용하여 각 분석 단계로 필요한 모든 기술을 플랫폼에서 제공받으며, 최적화된 모델이 개발되기까지 모델 개발의 라이프 사이클[2]을 거친다. AI 모델 개발의 라이프 사이클이란, 모델 개발의 단계를 차례로 거친 후에도, 모델 성능 최적화를 위해 언제든지 분석의 앞 단계로 다시 돌아가 이를 수행하며 모델의 성능을 극대화하는 행위를 말한다. 분석 모델의 개발은 한 번에 끝나는 것이 아니라 각 단계를 순회하며 모델의 성능을 최적화할 수 있는 다양한 방법을 시도하게 되는데, 모델 개발이 완료되고 모델이 배포된 뒤에도 모델의 성능이 시간이 지남에 따라 떨어질 수 있다. 이와 같은 이유로 모델의 이상 현상을 탐지했다면, 모델 개발의 초기 단계로 다시 돌아갈 수 있다. 모델의 라이프 사이클을 관리하고 여

<표 1> 분석 단계별 플랫폼 기술 요소

분석 단계	플랫폼 기술 요소
학습 데이터 전처리	학습 데이터 전처리 피처 엔지니어링 탐색적 데이터 분석 학습 데이터 라벨링
알고리즘 선택 및 개발	가상 IDE Built-in Algorithm or Algorithm Auto selection
알고리즘 성능 최적화	Hyper Parameter 튜닝 모델 라이프 사이클 관리
학습 환경 설정	학습용 컴퓨팅 자원 관리
모델 배포	배포용 모델 컴퓨팅 자원 관리
배포된 서비스 생산성 관리	모델 성능 모니터링

리 단계를 순회하며 최적의 모델이 배포될 수 있도록 지원하는 기술을 플랫폼에서 지원하기도 한다. 더 나아가 플랫폼에서는 계속적으로 모델을 재생산해내고 이를 배포하는 Continuous Delivery for Machine Learning 기술[3, 4]도 등장하게 되었다.

3. AI 플랫폼 기술 동향 분석

앞에서 언급한 바와 같이 AI 플랫폼에서는 분석을 위한 데이터셋 관리 및 전처리에서부터 배포된 모델의 성능을 모니터링하는 분석의 전 과정을 지원하게 되었다. 이번 장에서는 이러한 환경을 제공하기 위해 각 플랫폼에서는 어떠한 기술을 어느 정도의 수준까지 제공하는지, 각 플랫폼의 기술이 어떤 방향으로 진화되고 있는지를 살펴보도록 한다.

3.1 학습 데이터 전처리

학습 데이터 전처리 부분에서는 분석가들이 주로 사용하는 알고리즘을 제공받도록 하는데, 크게 2가지 형태로 제공한다. 첫 번째는 코드베이스로 데이터를 전처리할 수 있도록 하는 환경을 제공하는 방법, 두 번째는 Drag & Drop 방식으로 데이터 파이프라인[5]을 구성할 수 있는 환경을 제공하는 방법이다. 코드베이스로 전처리 알고리즘을 사용하기 위해서는 기존의 알고리즘을 라이브러리 형태로 제공하는 여러 프레임워크를 손쉽게 사용할 수 있도록 각 플랫폼에서 built-in 시켜서 제공한다. 이것은 전통적으로 분석가들이 사용하는 친숙한 방법으로, 코드만 이식하면 어느 플랫폼에서도 사용할 수 있다는 장점이 존재하지만, 전처리의 결과가 어떻게 되는지 시각적으로 보이지 않는다는 단점이 있다. 반면에 Drag & Drop 방식의 데이

터 파이프라인 기법은 한번 파이프라인을 구성해 놓으면 다른 플랫폼으로의 이식이 어렵지만, 각 전처리 결과를 시각적으로 확인할 수 있다는 장점이 있다.

3.1.1 코드 베이스 학습 데이터 전처리

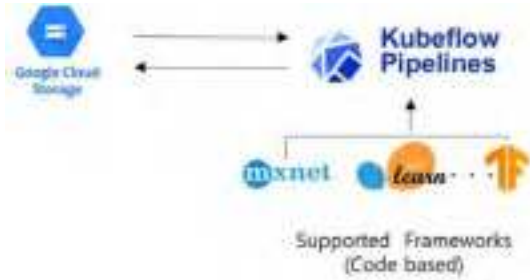
코드 베이스의 학습 데이터 전처리방법은 AWS의 Sagemaker[6], Google AI platform[7]에서 제공한다. (그림 2)는 AWS Sagemaker에서 제공하는 학습 데이터 전처리 환경을 나타낸다.

Sagemaker에서는 Mxnet, Tensorflow, Scikit Learn과 같은 AI 프레임워크들을 컨테이너 형태로 제공한다. 분석가는 각 프레임워크에서 제공하는 코드 방식을 그대로 사용하여 전처리할 수 있게 됨으로써, 개발된 코드는 어느 플랫폼으로도 이식이 가능할 수 있는 이식성이 생긴다. 학습 데이터 전처리를 위해 AWS의 S3에서 데이터 셋을 가져와 처리할 수 있도록 하는 라이브러리를 제공한다. 전처리를 수행한 결과도 S3에 저장하여 처리할 수 있도록 하는 메커니즘을 제공한다.

Google의 AI platform도 (그림 3)과 마찬가지로 코드 베이스로 학습 데이터를 전처리하는 방법을 제공한다. 이는 AWS Sagemaker와 마찬가지로 여러 프레임워크들을 가상화 컨테이너 형태로



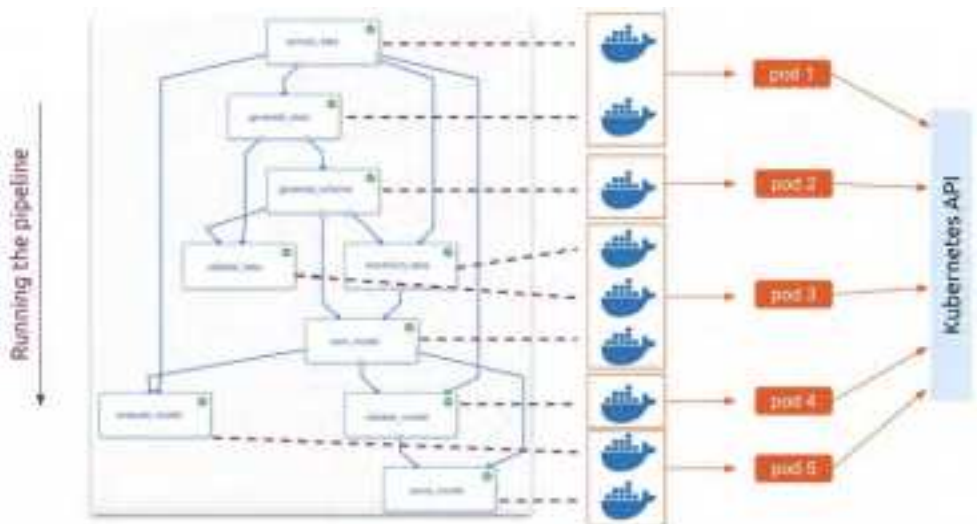
(그림 2) AWS Sagemaker에서의 학습 데이터 전처리 환경



(그림 3) Google AI Platform에서의 학습 전처리 환경

패키징하여 유저가 이를 사용할 수 있도록 하는 환경을 제공한다.

Google의 AI platform에서는 (그림 4)에서 처럼 kubeflow라고 하는 데이터 파이프라인을 제공하여 사용자가 작성한 데이터 조작의 흐름을 시각적으로 볼 수 있다는 장점이 있다. 이러한 파이프라인은 AI 및 빅데이터의 전처리에서부터 작업 처리 및 시각화까지의 작업 흐름을 정의하는데, 각각의 작업은 컨테이너[8]로 정의되어 kubernetes에서 작동한다. 컨테이너는 다른 작업에서 다시 재사용할 수 있다는 장점이 있다.



(그림 4) Google AI Platform에서 제공하는 Data pipeline 형식의 작업 흐름

3.1.2 Drag & Drop 학습 데이터 전처리

Azure ML에서도 학습 데이터 전처리뿐만 아니라, 대부분의 ML 작업을 Data Pipeline을 통해서 한다. 특히 Azure ML에서는 Drag & Drop 방식으로 Pipeline을 구성할 수 있도록 하는 환경을 제공한다. 이것은 코드로 파이프라인을 작성해 나가는 방식과 달리, 플랫폼상에서 미리 제공하는 알고리즘으로 사용자가 마우스를 사용하여 각 작업의 흐름을 구성해 나가는 방식이다. 다음 (그림 5)는 데이터셋에서부터 Drag & Drop 으로 학습 데이터를 전처리하는 Data pipeline의 예이다.

3.1.3 학습 데이터 전처리 자동화

학습 데이터 전처리는 유저가 직접 수행하는 경우 외에도, 주어진 학습 데이터 셋을 기반으로 컴퓨터가 자동으로 전처리를 수행하는 기술이 등장했다. DataRobot[9]은 (그림 6)에서와 같이 학습 데이터의 전처리를 자동화해주는 기술을 제공하여 사용자의 학습 데이터를 기반으로 결측치와 완



(그림 5) Azure ML에서 제공하는 Data pipeline

전하지 않은 데이터가 있는 것을 탐지하여 결측치에 어떤 값을 채울지를 추천한다. 이외에 적절한 값이 없는 데이터 형식, 일관성 없는 값 및 카테고리에 범주에 속하지 않은 데이터 등을 자동으로 탐지하여 적절한 값을 추천하거나 잘못된 형식의 값을 바로잡는다. 자동으로 이를 탐지하여 값을 수정하는 것은 분석에 있어서 상당한 작업 부담을 줄일 수 있다.

C	D
age	weight
[50-60)	?
[20-30)	[50-75)
[80-90)	?
[50-60)	?
[50-60)	?
[70-80)	?

(그림 6) data robot 상에서의 Missing Value 탐지

3.1.4 학습 데이터 자동 라벨링

인공지능 지도학습을 위해서는 학습 데이터에 라벨링이 필수적이다. 정확성이 높은 결과를 얻기

위해서는 많은 양의 정답 데이터셋을 확보 해야 한다. 하지만 데이터셋에 정답을 라벨링 하는 작업은 상당한 시간이 걸린다. 이에 많은 양의 라벨링 작업을 모두 수작업하지 않고, 일부 자동화할 수 있도록 하는 기술을 제공하고 있다. AWS의 Sagemaker에서는 (그림 7)과 같이 Groud Truth라는 서비스를 통하여 4개 영역에서의 학습 데이터 자동 라벨링 기술을 제공한다.

물체를 자동으로 인식해 물체에 Bounding Box를 설정해 주는 것, 이미지 분류 세그멘테이션, 텍스트 분류 및 사용자 정의 태스크에 관련한 라벨링 기술을 제공한다. Ground Truth의 오토라벨링 기술은 사용자가 일부 라벨링한 데이터셋을 토대로 자동으로 라벨링 되지 않은 데이터셋을 라벨링을 한다. Ground Truth에서는 기계가 자동 라벨링을 하면서도 정확하지 않은 데이터에 대해 사용자에게 보정을 요구하며 점차 정확성을 높여 가는 방향으로 작업한다. 또한 사람이 직접 라벨링 한 데이터셋도 추가해 주면 이를 토대로 더 정확성 높은 라벨링 작업을 수행하게 된다. 어느 정도 신뢰성 있는 결과 데이터를 얻기 위해서는 학습 데이터 셋의 최소 10%는 수동 라벨링하고 나머지 90%



(그림 7) Sage Maker의 Ground Truth에서의 오토 라벨링 기술

정도를 자동 라벨링 하도록 권장하고 있다.

3.2 알고리즘 선택 및 개발

학습 데이터 전처리 과정이 끝나면, 대개 분석을 위한 알고리즘 개발을 하게 된다. 알고리즘 개발 역시 크게 두 가지 방법으로 제공된다. 첫 번째는 사용자에게 모델을 개발에 필요한 여러 프레임워크를 삽입하여 IDE 개발환경으로 제공해주는 방법이다. 두 번째는 모델 개발에 여러 단계를 Data Pipeline 형태로 제공해주는 방법이 있고, Data Pipeline 역시 코드 기반 또는 Drag & Drop 기반 두 가지 방식으로 작성할 수 있다.

3.2.1 알고리즘 개발 IDE 제공

알고리즘 작성을 위한 개발 IDE를 제공하여 사용자가 편리하게 인공지능 모델을 개발할 수 있도록 한다. 사용자가 친숙한 여러 프레임워크를 선택적으로 제공하여, 언제든지 유저가 복잡한 설치 및 설정하지 않아도 프레임워크 사용에 최적화된 개발환경을 제공하게 된다. AWS Sagemaker, Google AI Platform에서는 최적화된 개발환경을

제공하기 위해 개발환경을 패키징하여 사용자에게 제공하게 되는데, 이때 사용하는 기술이 컨테이너 기술이다. Sagemaker와 Google AI Platform에서는 Docker를 활용하여 컨테이너 형태의 패키징 환경을 제공하는데, 각 알고리즘의 단계별로 마치 Data Pipeline을 형성하듯이 컨테이너 흐름을 정의할 수도 있다. (그림 8)은 Sagemaker에서의 도커 컨테이너를 활용한 알고리즘 개발환경을 보여준다. 첫 번째로 사용자는 자신이 알고리즘을 개발할 IDE를 요청하여 할당받는다. 두 번째로는 사용자가 의도하고자 하는 알고리즘을 프레임워크를 이용하여 작성하는데, 각각의 알고리즘 단계를 Script라고 부른다. 각 script는 사용자에게 친숙한 프레임워크의 프로그래밍 모델을 그대로 따를 수 있어, 만들어진 Script는 다른 플랫폼의 프레임워크로도 이식이 가능해진다. 세 번째로 Script 작성이 끝난다면, Sagemaker의 코드를 작성함으로써, 학습 인프라와 관련된 설정과 알고리즘 개발의 흐름을 정의할 수 있다. 유저는 같은 Script로도 Sagemaker의 코드 변경만 하면 서로 다른 학습 환경에서 학습하는 것이 가능해진다.



(그림 8) Docker 컨테이너를 활용한 알고리즘 개발환경

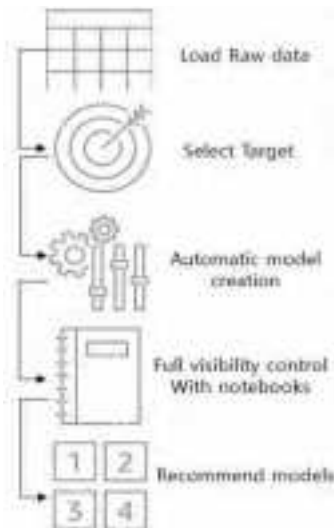
3.2.2 자동화된 알고리즘 개발 및 선택

사용자가 직접 알고리즘을 개발하는 방법과는 반대로, 학습 데이터 셋을 기반으로 알고리즘을 자동으로 선택해 주는 기술도 등장했다. 사용자가 분석하고자 하는 학습 데이터 셋을 올리지만 하면 자동으로 학습데이터셋에 관련된 최적화 된 알고

리즘을 추천해주는 기술을 제공한다. (그림 9)는 AWS의 Sagemaker에서 제공되는 Auto-Pilot 기능을 나타낸다.

3.3 학습 환경 설정 및 모델 배포

알고리즘 개발 후에도, 하이퍼파라미터를 최적화하며 모델 개발의 성능을 극대화하여야 한다. 하이퍼파라미터 최적화만으로도 알고리즘 성능을 극적으로 올리는 경우가 많지만, 어떤 방법으로 최적화 해야 한다는 기준은 존재하지 않는다. 이를테면, 학습률, 모델의 레이어 개수, 옵티마이저 등의 여러 조합 중 최고의 성능을 내는 경우의 수를 찾아야 한다. 어떤 경우에 모델이 최고의 성능을 낼지는 이유를 알 수 없는 블랙박스 모형인 경우가 대부분이다. 따라서 여러 경우의 수를 조합하여 가장 최적의 성능을 내는 조합을 찾아 모델 배포에 사용하게 된다. 그러나 사용자가 최적의 조합을 찾아내기에는 상당한 시간적 비용이 소모된다. 플랫폼에는 컴퓨팅 파워를 활용하여 여러 가지 조합을 수행하여 최적의 성능을 내는 하이퍼



(그림 9) Sagemaker의 Auto Pilot



(그림 10) Katib의 알고리즘 성능 최적화

파라미터 값을 빠르게 찾을 수 있게 한다. (그림 10)은 Google AI Platform에서 제공하는 Hyperparameter Optimization인 Katib을 나타낸다. Katib은 docker 기반으로 컨테이너를 생성하여 여러 하이퍼파라미터 조합을 병렬적으로 수행하여 단기간 내에 어떠한 하이퍼파라미터가 성능이 잘 나오는지를 계산할 수 있다.

(그림 11)에서는 여러 가지 하이퍼파라미터 조

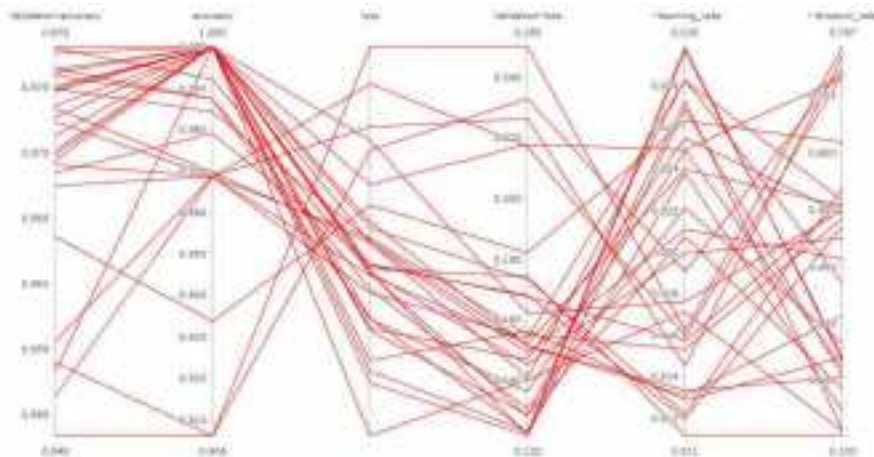
합 중 어떠한 조합이 성능이 우수함을 시각적으로 표시한 결과이다.

3.4 학습 환경 설정 및 모델 배포

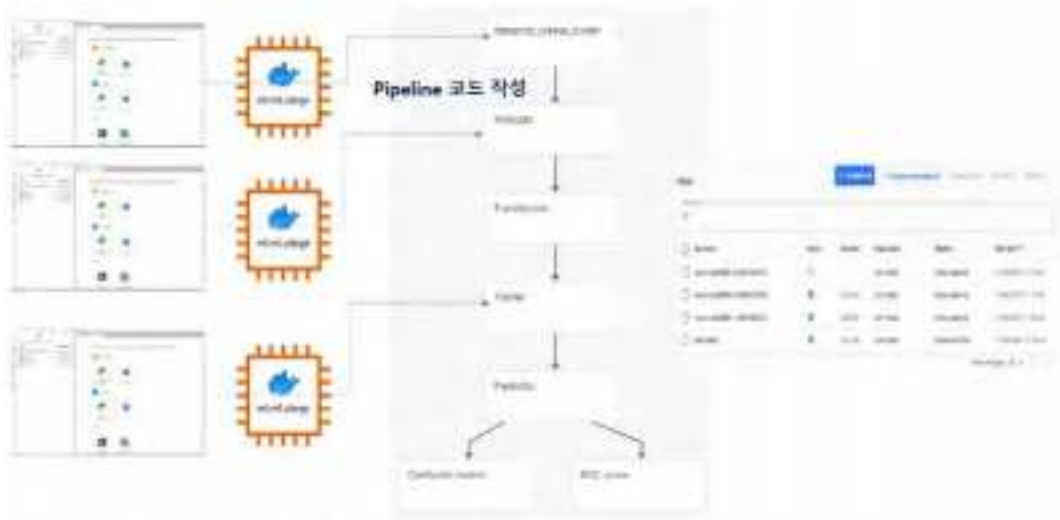
사용자가 학습 데이터를 학습하고, 모델 배포를 배포할 때는 상당한 컴퓨팅 자원을 필요로 한다. 일반적인 사용자는 이러한 컴퓨팅 자원을 능숙하게 다루는 데에는 한계가 있고, 시간이 오래 걸리는 일이다. 플랫폼에서는 사용자가 손쉽게 컴퓨팅 자원을 다룰 수 있도록 지원한다. 다음 (그림 12)는 Google AI Platform 에서의 컨테이너 기반 컴퓨팅 자원 할당 방법이다. AI 모델의 알고리즘은 파이프라인의 각 요소가 되고, 각 알고리즘은 컨테이너로 구동할 때 컴퓨팅 자원의 규모를 설정할 수 있다. 이런 방식으로 학습 환경 및 모델 배포 시 사용자는 언제든지 원하는 만큼의 컴퓨팅 자원을 할당하는 것이 가능하다.

3.5 배포된 모델의 생산성 관리

일반적으로 학습 데이터 셋을 학습 세트와 테스트



(그림 11) Katib의 하이퍼파라미터 최적화 결과



(그림 12) 학습 환경 및 모델 배포 시 할당되는 컨테이너 기반 컴퓨팅 자원 할당

트 세트로 나누어, 학습을 마치고 난 뒤 그 모델의 성능을 평가하게 된다. 그러나 이렇게 평가된 모델이 영구적으로 성능이 유지되는 것은 아니다. 시간이 지남에 따라 모델은 성능이 떨어지게 된다 [10]. 모델 모니터링은 시간이 지남에 따라 떨어지는 모델의 성능을 조기에 감지하고 모델을 운영하는 사람이 즉각적으로 새로운 모델로 업데이트시킬 수 있도록 하는 데에 필요한 기술이다. 정답

셋이 없는 데이터를 이용해 추론으로 들어가는 데이터를 기반으로서는 모델의 성능을 즉시 확인할 수 없다. 그러한 이유로 최신의 모델 모니터링 기술은 정답 셋이 없는 데이터에 한해서도 모델의 성능의 유효성을 확인할 수 있는 “Concept Drift” 방식을 최신 ML 플랫폼[4-6]에서 채택하고 있다. 이 “Concept Drift” 방식은 모델의 종류에 국한되지 않고 모델의 성능 저하를 탐지할 수 있다는 장



(그림 13) Sagemaker에서 제공하는 성능 모니터링 기술인 Data Drift 결과

점이 있다. 이 방식에서는 학습 데이터와 추론 데이터와의 분포도의 차이를 탐지한다. AWS의 경우 Sagemaker라는 서비스 상에서 Data Drift 기반의 Model Monitor를 제공한다. (그림 13)에서는 AWS 유저는 먼저 학습을 시킨 모델을 배포하고, 추론을 요청하여 Input으로 들어가는 데이터를 사전에 유저가 설정해 놓은 Schedule에 따라 모니터링 하게 된다. 모니터링중 이상이 감지되면, Amazon CloudWatch 서비스를 통해 유저에게 notification을 주며, 유저는 모델의 성능이 저하되었을 유추할 수 있다.

4. 결 론

최신 클라우드 기반의 AI 플랫폼에서는 컴퓨팅 자원을 쉽게 제어할 수 있는 인프라 제어 기술을 활용하여 분석가들이 AI 모델 개발을 쉽게 할 수 있도록 하는 환경을 제공한다. 특히 학습 데이터의 전처리에서부터 모델을 개발하고 배포할 수 있는 기술을 플랫폼 안에서 모두 수행할 수 있도록 하는 End to End 기술로 진화하고 있다. 이러한 플랫폼 기술은 분석가 컴퓨팅 자원 제어를 위한 이해가 없어도 인프라 기술과 인공지능 개발을 위해 들어가는 컴퓨팅 파워를 활용할 수 있게 하여, 빠르고 효율적인 AI 모델 및 운용을 가능하게 한다.

참 고 문 헌

- [1] Project Jupyter. <https://jupyter.org/>, Accessed June 21, 2021.
- [2] Hummer, Waldemar, Vinod Muthusamy, Thomas Rausch, Parijat Dube, Kaoutar El Maghraoui, Anupama Murthi, and Punleuk Oum. "Modelops: Cloud-based lifecycle management for reliable and trusted ai." In 2019 IEEE International Conference on Cloud Engineering (IC2E), pp. 113-120. IEEE, 2019.
- [3] Carter, Eric, and Matthew Hurst. "Continuous Delivery." In Agile Machine Learning, pp. 59-69. Apress, Berkeley, CA, 2019.
- [4] Lim, Junsung, Hoejoo Lee, Youngmin Won, and Hunje Yeon. "MLOP lifecycle scheme for vision-based inspection process in manufacturing." In 2019 {USENIX} Conference on Operational Machine Learning (OpML 19), pp. 9-11. 2019.
- [5] Klievink, Bram, Eveline Van Stijn, David Hesketh, Huib Aldewereld, Sietse Overbeek, Frank Heijmann, and Yao-Hua Tan. "Enhancing visibility in international supply chains: The data pipeline concept." International Journal of Electronic Government Research (IJEGR) 8, no. 4 (2012): 14-33.
- [6] Joshi, Ameet V. "Amazon's machine learning toolkit: Sagemaker." In Machine Learning and Artificial Intelligence, pp. 233-243. Springer, Cham, 2020.
- [7] Bisong, Ekaba. "Kubeflow and kubeflow pipelines." In Building Machine Learning and Deep Learning Models on Google Cloud Platform, pp. 671-685. Apress, Berkeley, CA, 2019.
- [8] Bernstein, David. "Containers and cloud: From lxc to docker to kubernetes." IEEE Cloud Computing 1, no. 3 (2014): 81-84.
- [9] "DataRobot: EnterPrise AI." DataRobot. <https://www.datarobot.com/>. Accessed June 21, 2021.
- [10] Tsymbal, Alexey. "The problem of concept drift: definitions and related work." Computer Science Department, Trinity College Dublin 106, no. 2, 2004.

저자약력



김 윤 기

이메일 : ykkim77@cuk.edu

- 2012년 서경대학교 컴퓨터공학과 (학사)
- 2019년 고려대학교 전산학과 (석박사)
- 2019년~2021년 SK주식회사 / Data Platform 그룹 수석
- 2021년~현재 고려사이버대학교 미래학부 조교수
- 관심 분야: 분산 병렬 시스템, AI 빅데이터 플랫폼, HPC



전 유 부

이메일 : jeonyb@dgu.edu

- 2013년 고려대학교 영상정보처리학과 (박사)
- 2013년~2014년 (주)파워그리드 CTO
- 2014년~2019년 순천향대학교 교수
- 2021년~현재 동국대학교 인공지능학과 교수
- 2021년~현재 한국정보처리학회 학회지 편집위원장
- 관심 분야: 유비쿼터스 컴퓨팅, 사물인터넷, 인공지능



전 미 향

이메일 : chic830@naver.com

- 1999년 University of Adelaide Performing art (학사)
- 2011년 한국방송통신대학교 영어영문학과 (학사)
- 2014년 동국대학교 예술경영 (석사)
- 2018년 경기대학교 미디어 커뮤니케이션 (박사)
- 2011년~2018년 안젤라 아트컴퍼니 대표
- 2018년~현재 수원여자대학교 겸임교수
- 관심 분야: 문화 콘텐츠, 글로벌 미디어, 콘텐츠 유통, 빅데이터



한 성 수

이메일 : sshan1@kangwon.ac.kr

- 2019년 고려대학교 영상정보처리학과 (박사)
- 2015년~2016년 오리온테크놀로지 이사
- 2018년~2019년 순천향대학교 교수
- 2019년~현재 강원대학교 자유전공학부 교수
- 2020년~현재 한국정보처리학회 상임이사
- 관심 분야: 빅데이터, 분산병렬알고리즘, 영상정보처리, 딥러닝