

Trusted AI를 위한 시각적이고 확장 가능한 구성 요소 라이브러리인 CLAIMED

로미오 카엔즐러¹ 2 와 이반 네사치

¹ IBM, 오픈 소스 데이터 및 AI 기술 센터(CODAIT)

² 바젤 대학병원 방사선 및 핵의학과

추상적인. 딥 러닝 모델은 점점 더 대중화되고 있지만 설명 가능성, 적대적 견고성 및 공정성에 대한 제약은 종종 프로덕션 배포의 주요 관심사입니다. 오픈 소스 에코시스템은 이러한 문제를 해결하는데 풍부하지만 완전히 통합된 엔드 투 엔드 시스템은 오픈 소스가 부족합니다. 따라서 우리는 IBM과 University Hospital Basel 간의 공동 노력으로 Kubernetes를 기반으로 프로덕션 등급 머신 러닝을 위한 완전한 오픈 소스, 재사용 가능한 구성 요소 프레임워크, 시각적 편집기 및 실행 엔진을 제공합니다. Kubeflow Pipelines, AI Explainability360 툴킷, AI Fairness360 툴킷 및 Adversarial Robustness Toolkit을 사용합니다.

ElyraAI, Kubeflow, Kubernetes 및 JupyterLab의 최상위. Elyra 파이프라인 편집기를 사용하면 일련의 jupyter 노트북으로 AI 파이프라인을 시각적으로 개발할 수 있습니다. CT 스캔을 기반으로 COVID-19 딥 러닝 분류 파이프라인을 만든 방법을 설명합니다. 툴킷을 사용하여 모델의 결정에 중요한 이미지 부분을 강조 표시합니다. 나이와 성별에 대한 편향을 감지하고 마지막으로 KFServing에 모델을 배포하여 Swiss Personalized Health Network의 여러 병원 데이터 센터에서 공유하는 방법을 보여줍니다.

키워드: 쿠버네티스 · Kubeflow · JupyterLab · ElyraAI · KFServing
· TrustedAI · AI 설명 가능성 · AI 공정성 · AI 적대적 견고성

1. 소개

개별 AI 파이프라인 작업을 수행하기 위한 오픈 소스 소프트웨어는 풍부하지만 커뮤니티에는 완전히 통합되고 신뢰할 수 있으며 확장 가능한 시각적 도구가 부족합니다.

따라서 우리는 모든 종류의 AI 파이프라인을 Kubernetes로 푸시할 수 있는 ElyraAI 위에서 실행되는 AI 기계 학습, ETL 및 데이터 과학을 위한 시각적 구성 요소 라이브러리인 CLAIMED를 구축했습니다. 모든 컨테이너화된 애플리케이션은 라이브러리의 구성 요소가 될 수 있습니다. CLAIMED는 Apache v2 오픈 소스 라이선스에 따라 릴리스되었습니다. 다음에서는 현재 릴리스에 통합하고 있는 오픈 소스 구성 요소를 소개하고 다음 설명과 함께 다양한 구성 요소 범주에 대한 개요를 설명합니다.

CODAIT(IBM Center for Open Source Data and AI Technologies)에서 지원 <https://github.com/elyra-ai/component-library>

2 Romeo Kienzler 외.

건강 관리에 사용되는 예시적인 구성 요소. 이 파이프라인은 개방형에서도 사용할 수 있습니다.

원천 ⁴.

1.1 컨테이너화와 쿠버네티스

가상화는 인프라 관리를 위한 많은 가능성, 주로 동일한 하드웨어에서 동시에 다른 운영 체제를 실행할 수 있는 기능을 열었습니다. 다음 격리 단계는 서버에서 실행되는 각 마이크로 서비스에 대해 수행할 수 있지만 호스트 운영 체제에서 액세스 권한 및 리소스를 관리하는 대신 자체 환경에서 별도의 패키지로 컨테이너화할 수 있습니다. 이것의 실질적인 효과는 우리가 마치 전용 가상 머신이 있는 것처럼 각 마이크로 서비스를 실행하지만 그러한 노력의 오버헤드 없이 실행한다는 것입니다. 이는 호스트 운영 체제 위에서 컨테이너를 실행하여 수행됩니다. 컨테이너화 플랫폼의 예는 Docker입니다.

컨테이너화를 통해 많은 수의 컨테이너를 실행할 수 있으므로 오케스트레이션이 필요했습니다. 이는 시스템이 원하는 상태에 있는지 지속적으로 관리해야 하고, 확장 또는 축소하고, 컨테이너 간 통신을 관리하고, 일정을 잡고, 인증을 관리하고, 로드 균형을 맞추는 등의 작업을 수행해야 한다는 의미입니다. Docker Swarm과 같은 다른 옵션에서 Kubernetes는 이 도메인의 시장 리더입니다.

구글이 CNCF에 기증한 것이기 때문에 구글의 노하우와 다년간의 경험이 많이 들어갔다. 시스템은 퍼블릭, 온프레미스 또는 하이브리드 클라우드에서 실행할 수 있습니다. 온프레미스 설치의 민감한 데이터를 다루는 기관에 매우 중요합니다. IBM의 경우 Kubernetes도 전략적입니다. CNCF에 합류하여 모든 Watson Services를 Kubernetes로 이전하고 RedHat을 인수했으며 IBM은 이제 Kubernetes에 대한 세 번째로 큰 커미터입니다.

1.2 TensorFlow를 사용한 딥러닝

Google Brain 프로젝트의 확장 가능한 분산 교육 및 추론 시스템인 DistBelief[10]의 두 번째 버전입니다. 훈련 및 추론을 위해 휴대폰에서 GPU/TPU 클러스터에 이르기까지 무수히 많은 하드웨어 플랫폼을 지원합니다. 데이터가 컴퓨터를 떠나지 않고도 클라이언트 측의 브라우저에서 실행할 수도 있습니다. 연구에서 귀중한 도구가 되는 것 외에도 까다로운 생산 환경에서도 사용되고 있습니다. 개발 측면에서 나무와 같은 구조의 기계 학습 알고리즘을 대표하는 것은 매우 좋은 표현 인터페이스가 됩니다. 마지막으로 성능 대 사용성 측면에서 열망 모드와 그래프 모드가 모두 지원됩니다. 의미 디버깅은 첫 번째 경우에 훨씬 간단하며 속도가 필요한 경우 후자를 사용할 수 있습니다.

⁴ <https://github.com/cloud-annotations/elyra-classification-training/tree/developer> 기사

1.3 큐브플로우

Kubeflow는 Kubernetes 위에서 일급 시민으로 실행되는 기계 학습 파이프라인 관리 및 실행 시스템입니다. Kubernetes 확장성을 활용하는 것 외에도 기계 학습 파이프라인으로 재현 가능한 작업을 허용하고 실행 결과 및 중간 아티팩트가 메타 데이터 저장소에 저장됩니다.

1.4 엘라라AI

ElyraAI는 JupyterLab 생태계를 위한 일련의 확장으로 시작되었습니다. 여기서 우리는 끌어서 놓기 편집기를 사용하여 기계 학습 워크플로를 표현하고 Kubeflow 또는 Airflow와 같은 다른 엔진 위에서 실행하도록 보낼 수 있는 ElyraAI의 파이프라인 편집기에 집중합니다. 이를 통해 프로그래머가 아닌 사람도 읽고 이해할 수 있을 뿐만 아니라 기계 학습 워크플로를 만들 수도 있습니다. ElyraAI는 또한 브라우저(예: github 리포지토리)에서 이러한 파이프라인 시각화를 지원합니다.

1.5 주피터랩

JupyterLab은 데이터 과학을 위한 가장 인기 있는 개발 환경 중 하나입니다. 그래서 먼저 JupyterLab을 지원하기 시작했습니다. 그러나 ElyraAI의 파이프라인 편집기는 다른 환경에서도 지원되며 VSCode가 그 다음 목록에 있습니다.

1.6 AI 설명 가능성

놀라운 성능 외에도 딥 러닝 모델은 블랙 박스로 간주되기 때문에 프로덕션 사용에 많은 저항에 직면합니다. 기술적으로(및 수학적으로) 딥 러닝 모델은 일련의 비선형 기능 공간 변환입니다. 무섭게 들리지만 다시 말해서 딥 러닝 네트워크가 수행하는 개별 처리 단계를 이해하는 것은 정의에 따라 매우 어렵습니다. 그러나 딥 러닝 모델의 어깨 너머로 볼 수 있는 기술이 존재합니다. 여기서 우리가 사용하고 있는 것은 LIME[8]입니다. LIME은 기존 분류 모델을 사용하고 잘못된 분류가 발생하는 한 유효성 검사 세트(따라서 실제 클래스 레이블이 알려짐)에서 가져온 이미지를 치환합니다. 그런 식으로 LIME을 사용하여 이미지 오버레이로 히트 맵을 생성하여 분류자가 가장 잘 수행할 수 있는 이미지 영역을 나타낼 수 있습니다. 즉, 분류자가 보고 있는 이미지의 영역을 식별합니다.

그림 1)과 같이. 예를 들어, COVID-19 분류를 위한 이미지에서 가장 관련성이 높은 영역은 특정 분류자에 문제가 있음을 나타내는 폐 조직 위의 뼈가 포함된 영역입니다.

1.7 AI 공정성과 편견

그렇다면 편견이란 무엇입니까? Wikipedia에서는 다음과 같이 말합니다.

4 Romeo Kienzler 외.

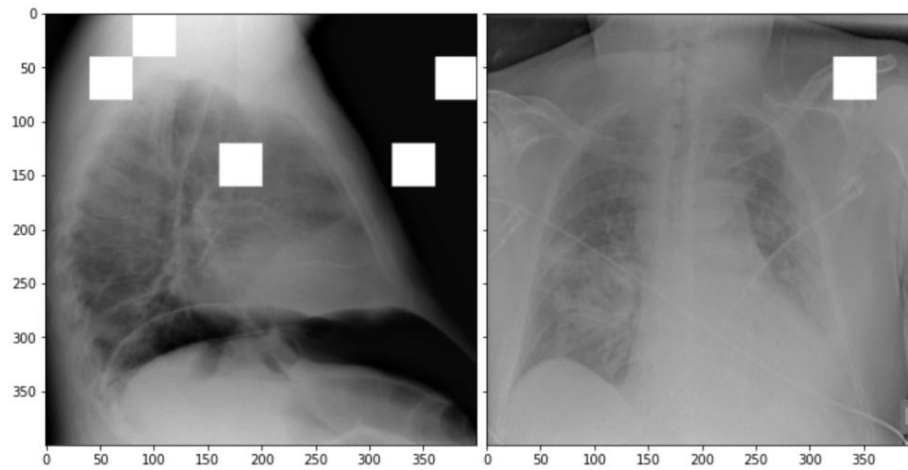


그림 1. LIME이 이미지의 분류 관련 영역을 식별하는 데 어떻게 도움이 되는지에 대한 예

또는 불공평합니다.”[1] 그래서 여기에 있습니다. 우리는 모델이 성별, 인종, 연령, 사회 경제적 지위, 종교 등과 같은 보호 속성에 대해 공정하고 편견이 없기를 바랍니다. 그렇다면 교육 중에 모델에 해당 데이터를 "제공"하지 않는 것이 쉽지 않을까요? 그렇게 간단하지 않다는 것이 밝혀졌습니다. 보호된 속성은 종종 다른 속성으로 인코딩됩니다. 예를 들어 인종, 종교 및 사회경제적 지위는 우편번호, 연락 방법 또는 구매한 제품 유형과 같은 속성에 잠재적으로 인코딩됩니다. 공정성 평가 및 편향 감지는 그 자체로 예술입니다. 다행스럽게도 데이터와 모델의 편향을 평가하기 위한 수많은 단일 숫자 메트릭이 존재합니다. 여기에서는 IBM이 Linux Foundation AI에 기부한 AIF360[2] 라이브러리를 사용하고 있으므로 개방형 거버넌스를 받고 있습니다.

1.8 AI 적대적 견고성

Trusted AI의 또 다른 기둥은 적대적 견고성입니다. 연구원들이 알아낸 바와 같이 적대적 잡음은 데이터(데이터 중독) 또는 모델(모델 중독)에 도입되어 적대자에게 유리한 모델 결정에 영향을 미칠 수 있습니다. Adversarial Robustness Toolbox ART[11]와 같은 라이브러리는 모든 최첨단 공격 및 방어를 지원합니다.

2 시스템 구현 및 데모 사용 사례

2.1 TrustedAI 이미지 분류 파이프라인

앞에서 이미 언급했듯이 파이프라인은 기계 학습에서 재현성, 확장성, 감사 가능성 및 협업을 도입하는 좋은 방법입니다. 파이프라인은 종종 ML-Ops 전략의 중심 부분입니다. 재현성과 감사 가능성이 훨씬 더 중요하기 때문에 이것은 특히 TrustedAI 파이프라인에 적용됩니다.

그림 2는 구성 요소 라이브러리를 사용하여 구축한 예시적인 TrustedAI 파이프라인을 보여주고 그림 3은 파이프라인 실행을 마친 후 표시하는 KubeFlow에서 가져온 스크린샷입니다.

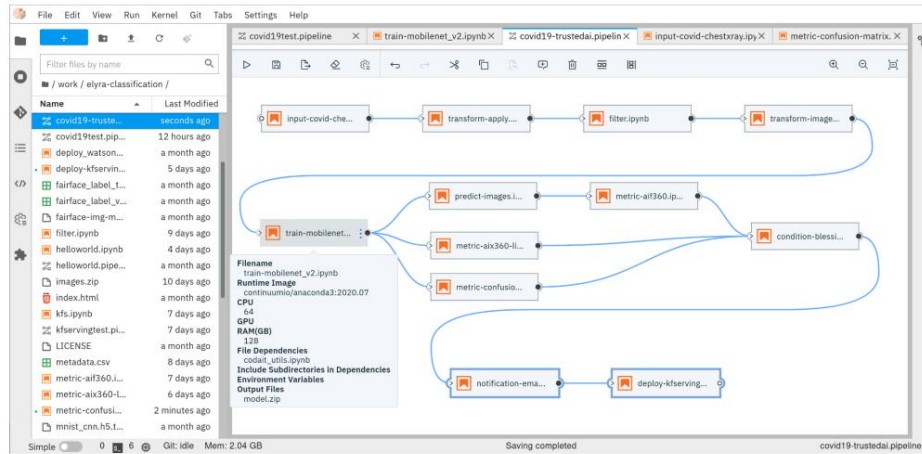


그림 2. 의료 사용 사례를 위한 예시적인 TrustedAI 파이프라인

2.2 파이프라인 구성 요소 다음과 같은 다

양한 범주의 파이프라인 구성 요소는 Trusted AI 이미지 분류 파이프라인에서 사용되는 구성 요소를 사용하여 예시됩니다.

입력 구성 요소 이 특별한 경우에는 공용 및 영구 링크를 통해 GitHub 리포지토리에서 직접 데이터를 가져옵니다. meta data.csv 및 images 폴더를 가져오기만 하면 됩니다. 구성 요소 라이브러리에는 파일 및 데이터 베이스와 같은 각기 다른 유형의 데이터 소스에 대한 구성 요소가 포함됩니다.

구성 요소 변환 때때로 메타데이터(또는 기타 구조화된 데이터 세트)에 대한 변환이 필요합니다. 따라서 일반적인 변환 구성 요소를 제공합니다. 이 경우에는 파일 시스템에서 사용하기 어렵게 만드는 슬래시가 포함된 원본 파일을 범주 형식으로 변경하는 데 사용했습니다. 해당 컬럼에 적용할 컬럼 이름과 함수만 지정하면 됩니다.

필터 구성 요소 데이터 세트에서 행의 내용을 변경하는 것과 유사하게 행을 제거하는 것도 데이터 엔지니어링에서 일반적인 작업이므로 필터 단계에서 정확히 이를 허용합니다. 술어를 제공하는 것으로 충분합니다. 이 경우 술어 `~metadata.filename.str.contains('.gz')`는 유효하지 않은 이미지를 제거합니다.

6 Romeo Kienzler 외.

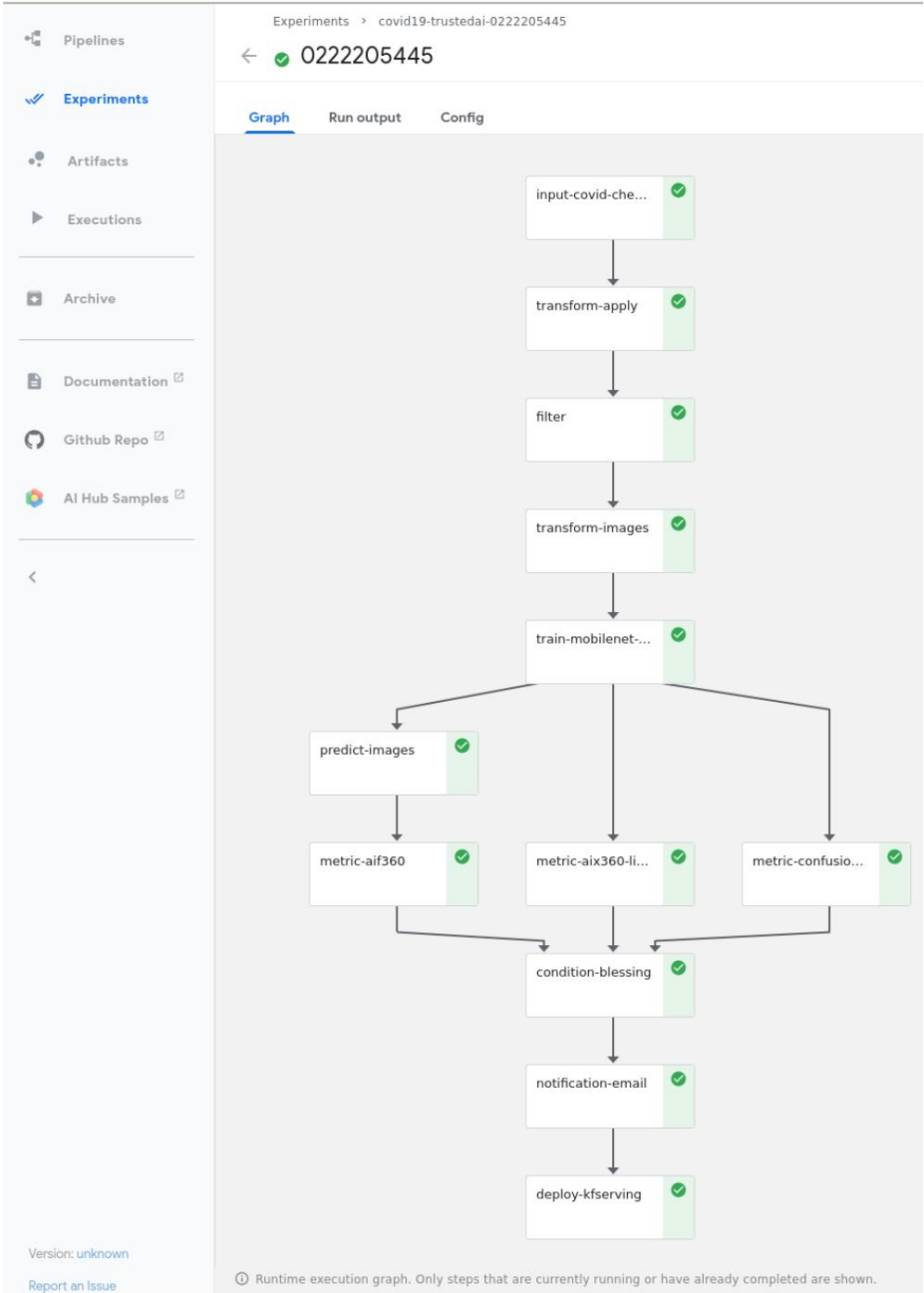


그림 3. Kubeflow에서 한 번 실행된 파이프라인

```
data
├── No Finding
│   ├── 000001-3.jpg
│   ├── 000001-6.jpg
│   └── Pneumonia_Viral_COVID-19
│       ├── 0007000c.jpg
│       └── 01E392EE-69F9-4E33-BFCE-E5C968654078.jpeg
```

그림 4. 이미지 분류 데이터 폴더 구조의 사실상 표준

이미지 변환기 구성 요소 레이블이 지정된 이미지 데이터에 대한 사실상의 표준은 이미지를 클래스/카테고리당 하나의 폴더에 넣는 것입니다. 그러나 이 특별한 경우에는 원시 데이터가 필요한 형식이 아닙니다. 이미지로 가득 찬 폴더일 뿐이며 해당 속성은 별도의 CSV 파일에 설명되어 있습니다. 클래스(또는 레이블)(이 경우 검색 결과) 외에도 이 CSV 파일에는 성별 및 연령에 대한 정보가 포함되어 있습니다. 따라서 먼저 CSV 파일에 제공된 결과 레이블에 대한 정보를 사용하고 그림 4와 같이 이미지를 적절한 폴더 구조로 정렬합니다.

교육 구성 요소 딥 러닝 모델을 이해, 정의 및 교육하는 것은 그 자체로 예술입니다. 딥 러닝 이미지 분류 모델을 교육하려면 적절하게 설계된 신경망 아키텍처가 필요합니다. 다행스럽게도 커뮤니티는 하이퍼 매개변수를 통해 매개변수화되는 사전 정의된 모델 아키텍처를 선호하는 경향이 있습니다. 이 단계에서 우리는 가장 일반적인 매개 변수 세트가 있는 작은 딥 러닝 신경망 아키텍처인 MobileNetV2를 사용하고 있습니다. TensorFlow 배포판과 함께 제공되며 뉴런이나 레이어를 추가로 정의하지 않고 바로 사용할 수 있습니다. 그림 5와 같이 몇 가지 매개변수만 지정하면 됩니다.

가능하지만 향후 Katib[9]를 통해 활용되는 KubeFlow의 하이퍼 매개변수 검색 기능을 활용하고자 하므로 이 처리 단계에서는 하이퍼 매개변수 검색을 고려하지 않습니다.

```
exec('input_shape = ("image_shape",3)')
model = tf.keras.applications.MobileNetV2(
    input_shape=input_shape, alpha=1.0, include_top=False,
    input_tensor=None, pooling=None, classes=num_classes,
    classifier_activation='softmax'
)
model = my_net(model, num_classes=num_classes)
```

그림 5. 래핑된 학습 구성 요소의 소스 코드

평가 구성 요소 정의, 컴파일 및 맞춤 외에도 모델은 생산에 들어가기 전에 평가해야 합니다. 대상 레이블에 대한 분류 성능을 평가하는 것은 기계 학습이 시작된 이래로 최신 기술이므로 혼동 행렬과 같은 구성 요소를 추가했습니다. 그러나 TrustedAI 조치를 고려하는 것은 새롭게 떠오르는 관행입니다. 따라서 AI Fairness, AI Explainability 및 AI Adversarial Robustness에 대한 구성 요소가 구성 요소 라이브러리에 추가되었습니다.

Blessing Components Trusted AI에서는 생성된 데이터, 모델 또는 다른 하위 시스템이나 사람이 게시하고 사용할 보고서와 같은 자산의 축복을 얻는 것이 중요합니다. 따라서 축복 구성 요소는 평가 구성 요소의 결과를 사용하여 자산을 게시할 준비가 되었는지 결정합니다.

구성 요소 게시 자산 유형에 따라 게시는 데이터 세트를 데이터 저장소에 유지하거나 기계 학습 모델을 배포하는 것을 의미합니다.

8 Romeo Kienzler 외.

다른 하위 시스템을 사용하거나 사람이 사용할 보고서를 게시하기 위해. 여기에서는 훈련된 TensorFlow 딥 러닝 모델을 Kubernetes에 게시하는 KFServing 구성 요소로 이 범주를 예시합니다. KNative 위에 있는 KFServing은 빌트인 Trusted AI 기능 외에도 카나리아 배포 및 확장성(0으로 확장 포함)과 같은 Kubernetes 기능에서 가져오기 때문에 특히 흥미롭습니다.

3 앞으로의 일

현재 각 범주에 대한 하나 이상의 대표 구성 요소가 릴리스되었습니다. 구성 요소는 매일 라이브러리에 추가됩니다. 게시할 다음 구성 요소는 TFJob을 사용한 병렬 Tensorflow 교육, Katib를 사용한 병렬 하이퍼 파라미터 튜닝 및 Apache Spark를 사용한 병렬 데이터 처리입니다. 또한 ElyraAI의 다음 릴리스(v.2.3.0)는 구성 요소의 구성 옵션 렌더링 기능을 개선할 것입니다. jupyter 노트북이나 python 스크립트에 있습니다.

4 결론

우리는 기계 학습 커뮤니티에서 사용하는 많은 사실상의 표준 구성 요소 위에 신뢰할 수 있고 코드가 적고 확장 가능한 오픈 소스 시각적 AI 파이프라인 시스템을 구축하고 제안했습니다. KubeFlow Pipelines를 사용하면 재현성과 감사 가능성이 제공됩니다. Kubernetes를 사용하면 확장성과 표준화가 제공됩니다.

시각적 개발에 ElyraAI를 사용하면 모든 내부 및 외부 이해 관계자가 모든 차원에서 시스템을 감사할 수 있는 권한을 부여받아 사용이 간편해집니다.

참조

1. 위키백과, <https://en.wikipedia.org/wiki/Bias>. 2021년 2월 18일에 마지막 액세스 2. AI 공정성 360 톨킷, <https://github.com/Trusted-AI/AIF360>. 마지막 액세스 2021년 2월 18일
3. AI Explainability 360 톨킷, <https://github.com/Trusted-AI/AIX360>. 마지막 접속일: 2021년 2월 18일
4. 엘라라 AI, <https://github.com/elyra-ai>. 마지막 액세스일: 2021년 2월 18일 5. Kubernetes, <https://kubernetes.io/>. 2021년 2월 18일에 마지막 액세스 6. JupyterLab, <https://jupyter.org/>. 2021년 2월 18일에 마지막 액세스 7. KFServing, <https://www.kubeflow.org/docs/components/serving/kfserving/>. 2021년 2월 18일에 마지막 액세스
8. Marco Tulio Ribeiro, Sameer Singh 및 Carlos Guestrin: "왜 내가 당신을 믿어야 합니까?": 모든 분류기의 예측 설명. 지식 발견 및 데이터 마이닝에 관한 22회 ACM SIGKDD 국제 회의 간행물, 샌프란시스코, 캘리포니아, 미국, pp. 1135–1144 (2016)
9. 카팁, <https://github.com/kubeflow/katib>. 2021년 2월 18일에 마지막 액세스

10. TensorFlow: 이기종 시스템에 대한 대규모 기계 학습, 백서
tensorflow.org 에서 Martin Abadi 외.
11. 적대적 견고성 도구 상자, <https://github.com/Trusted-AI/adversarial-robustness-toolbox> . 2021년 2월 18일에 마지막 액세스
12. CNCF에 합류한 IBM, <https://developer.ibm.com/technologies/containers/blogs/ibms-dication-to-open-source-and-its-involvement-with-the-cncf/>.
2021년 2월 18일에 마지막 액세스
13. 클라우드 네이티브 컴퓨팅 재단, <https://www.cncf.io>. 2월 18일에 마지막 액세스
2021년