사용 사례

NetApp Solutions

NetApp
April 20, 2024

목차

사용 사례

책임 **AI** 및 기밀 추론 **- NetApp AI** 및 **Protopia** 이미지 변환

TR-4928: AI 및 기밀 추론 지원 **- NetApp AI** 및 **Protopia** 이미지 및 데이터 변환

사티아가라잔, 마이클 오글즈비야, NetApp 병훈 안, 제니퍼 와겐버그, 프로토피아

시각적 해석은 이미지 캡처와 이미지 처리의 등장과 함께 커뮤니케이션의 핵심 요소가 되었습니다. 디지털 이미지 처리의 인공 지능(AI)은 암 및 기타 질병 식별을 위한 의료 분야, 환경 위험 연구, 패턴 인식, 범죄 퇴치를 위한 비디오 처리 등을 위한 지리공간 시각적 분석 등 새로운 비즈니스 기회를 제공합니다. 그러나 이 기회에는 특별한 책임이 있습니다.

AI에 대한 조직의 의사 결정이 내려질수록 데이터에 대한 개인 정보 보호, 보안, 법률, 윤리 및 규제 문제와 관련된 위험에 노출될 위험은 커집니다. 책임감 있는 AI를 통해 기업과 정부 조직이 대규모 기업에서 AI에 중요한 신뢰 및 거버넌스 구축을 지원할 수 있습니다. 이 문서에서는 Protopia 데이터 난독 처리 소프트웨어와 함께 NetApp 데이터 관리 기술을 사용하여 중요한 데이터를 민화하고 위험과 윤리적 문제를 줄임으로써 NetApp에서 검증한 AI 추론 솔루션에 대해 설명합니다.

소비자와 기업 모두 다양한 디지털 장치를 사용하여 매일 수백만 개의 이미지가 생성됩니다. 결과적으로 데이터가 폭발적으로 증가하고 컴퓨팅 작업 부하가 발생함에 따라 기업은 규모와 효율성을 위해 클라우드 컴퓨팅 플랫폼으로 전환하게 됩니다. 한편, 이미지 데이터에 포함된 민감한 정보에 대한 개인 정보 보호는 퍼블릭 클라우드로 이전될 때 발생합니다. 보안 및 개인 정보 보호 보장이 없기 때문에 이미지 처리 AI 시스템을 배포하는 데 주요 장애물이 되고 있습니다.

또한 이 있습니다 "삭제권" GDPR에 따라, 조직에서 모든 개인 데이터를 삭제하도록 요청할 수 있는 개인의 권리. 또한 도 있습니다 "개인정보보호법"이것은 공정 정보 관리 코드를 설정합니다. 사진과 같은 디지털 이미지는 데이터를 수집, 처리 및 지우는 방법을 제어하는 GDPR의 개인 데이터를 구성할 수 있습니다. 그렇지 않으면 GDPR을 준수하지 않아 조직에 심각한 피해를 줄 수 있는 규정 준수 위반에 대한 무거운 벌금이 부과될 수 있습니다. 개인 정보 보호 원칙은 머신 러닝(ML) 및 딥 러닝(DL) 모델 예측의 공정성을 보장하고 개인 정보 보호 또는 규정 준수 위반과 관련된 위험을 줄이는 책임 있는 AI 구현의 근간입니다.

이 문서에서는 개인 정보 보호 유지 및 책임 있는 AI 솔루션 배포와 관련된 이미지 난독 처리를 사용하는 경우와 사용하지 않는 세 가지 시나리오에서 검증된 설계 솔루션을 설명합니다.

- * 시나리오 1. * Jupyter 노트북 내 필요 시 추론.
- * 시나리오 2. * Kubernetes의 배치 추론
- * 시나리오 3. * NVIDIA Triton 추론 서버

이 솔루션에서는 FDDB(얼굴 인식 데이터 세트 및 벤치마크)를 사용합니다. FDDB는 페이스 보스의 구현을 위한 PyTorch 기계 학습 프레임워크와 함께, 구속되지 않은 얼굴 감지 문제를 연구하도록 설계된 얼굴 영역 데이터 세트입니다. 이 데이터 세트에는 다양한 해상도의 2845 이미지 세트에 있는 5171면에 대한 주석이 포함되어 있습니다. 또한 이 기술 보고서에서는 이 솔루션이 적용 가능한 상황에 대해 NetApp 고객 및 현장 엔지니어를 통해 수집된 일부 솔루션 영역 및 관련 사용 사례를 소개합니다.

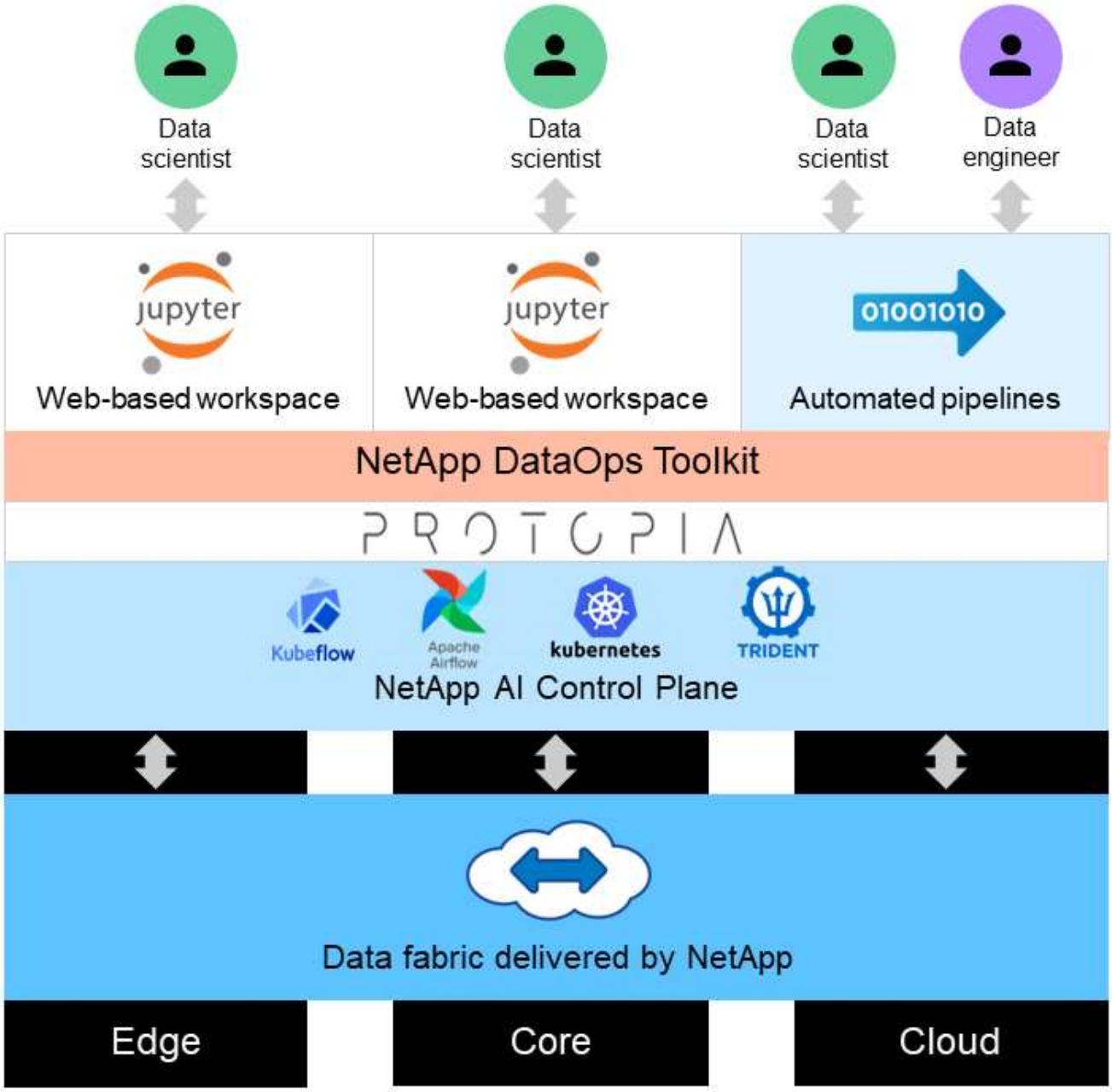
대상

이 기술 보고서는 다음 대상자를 대상으로 합니다.

- 책임 있는 AI를 설계 및 구축하고 공공 장소에서의 얼굴 이미지 처리와 관련된 데이터 보호 및 개인 정보 문제를 처리하고자 하는 비즈니스 리더 및 엔터프라이즈 설계자
- 개인 정보를 보호하고 유지하고자 하는 데이터 과학자, 데이터 엔지니어, AI/기계 학습(ML) 연구자 및 AI/ML 시스템 개발자.
- GDPR, CCPA 또는 국방부(DoD) 및 정부 조직의 개인정보 보호법과 같은 규정 표준을 준수하는 AI/ML 모델 및 애플리케이션에 대한 데이터 난독 처리 솔루션을 설계하는 엔터프라이즈 설계자
- 중요한 정보를 보호하는 딥 러닝(DL) 및 AI/ML/DL 추론 모델을 효율적으로 구축하는 방법을 찾고 있는 데이터 과학자 및 AI 엔지니어
- 에지 장치 관리자 및 에지 서버 관리자는 에지 추론 모델의 구축과 관리를 담당합니다.

솔루션 아키텍처

이 솔루션은 GPU의 처리 성능과 기존 CPU를 함께 사용하여 대규모 데이터 세트에서 AI 워크로드를 실시간으로 일괄 처리하고 추론하도록 설계되었습니다. 이 검증에서는 책임 있는 AI 배포를 원하는 조직에 필요한 ML에 대한 개인 정보 보호 추론과 최적의 데이터 관리를 보여줍니다. 이 솔루션은 Jupyter Lab 및 CLI 인터페이스를 사용하여 코어 사내 NetApp ONTAP AI와 상호 연결된 에지 및 클라우드 컴퓨팅을 위한 단일 또는 다중 노드 Kubernetes 플랫폼, NetApp DataOps 툴킷, Protopia 난독화 소프트웨어에 적합한 아키텍처를 제공합니다. 다음 그림에서는 DataOps Toolkit 및 Protopia를 지원하는 NetApp 기반의 Data Fabric에 대한 논리적 아키텍처 개요를 보여 줍니다.



Protopia 난독 처리 소프트웨어는 NetApp DataOps Toolkit에서 원활하게 실행되며 스토리지 서버를 떠나기 전에 데이터를 변환합니다.

솔루션 영역

디지털 이미지 프로세싱은 많은 이점을 제공하므로 많은 조직에서 시각적 표현과 관련된 데이터를 최대한 활용할 수 있습니다. 이 NetApp 및 Protopia 솔루션은 ML/DL 수명 주기 동안 AI/ML 데이터를 보호하고 민영화 할 수 있는 고유한 AI 추론 설계를 제공합니다. 고객은 이 모델을 통해 기밀 데이터에 대한 소유권을 유지하고, 개인 정보 보호와 관련된 문제를 완화하여 규모에 맞게 퍼블릭 또는 하이브리드 클라우드 구축 모델을 사용하거나, 에지에 AI 추론을 배포할 수 있습니다.

환경 인텔리전스

업계가 환경적 위험 영역에서 지리 공간 분석을 활용할 수 있는 방법은 여러 가지가 있습니다. 정부 및 공공사업 부서는 전염병 또는 산불과 같은 자연 재해 발생 시 공공 보건 및 기상 조건에 대한 실행 가능한 통찰력을 제공하여 일반에게 보다 나은 조언을 제공할 수 있습니다. 예를 들어, 영향을 받는 개인의 사생활을 침해하지 않고 공항 또는 병원과 같은 공공 장소에서 COVID-양성 환자를 식별하고 관련 당국과 인근 직원에게 필요한 안전 조치를 알릴 수 있습니다.

에지 장치 웨어러블

군대와 전쟁터에서 군인의 사생활을 보호하고 보호하면서 군인의 건강을 추적하고, 운전자의 행동을 모니터링하고, 군용 차량에 접근하는 안전 및 관련 위험에 대해 당국에게 경고하기 위해 엣지에서 AI 추론을 웨어러블 장치로 사용할 수 있습니다. 군대의 미래는 전투지 사물 인터넷(IoT)과 웨어러블 전투기용 사물 인터넷(IoMT)을 통해 첨단 기술로 나아가고 있습니다. 이 장비는 신속한 에지 컴퓨팅을 통해 군인들이 적들을 식별하고 전투에서 더 나은 성과를 달성할 수 있도록 지원합니다. 드론과 웨어러블 기어와 같은 에지 장치에서 수집된 시각적 데이터를 보호하고 보존하는 것은 해커와 적에게 항상 보안을 유지하는 데 매우 중요합니다.

비전투원 대피 작전

비전투원 대피 작전(Neos)은 미 국방부가 미국 시민과 국적을 대피시키기 위해, 국방부 민간인 직원, 그리고 적절한 안전한 피난처로 목숨을 걸고 있는 지정된 사람(HN)과 제3국 국민(TCN)을 대피시키기 위해 실시합니다. 현재 행정적 통제는 대부분 수작업을 통해 선별검사 프로세스를 비우는 데 사용됩니다. 그러나 AI/ML 비디오 난독 처리 기술과 결합된 고도로 자동화된 AI/ML 도구를 사용하면 비우는 ID, 비우는 방법 추적 및 위협 선별의 정확성, 보안 및 속도를 개선할 수 있습니다.

의료 및 생의학 연구

영상 처리는 컴퓨터 단층촬영(CT) 또는 자기공명영상(MRI)에서 얻은 3D 영상에서 외과적 계획을 위한 병리를 진단하는 데 사용됩니다. HIPAA 개인 정보 보호 규칙은 모든 개인 정보 및 사진과 같은 디지털 이미지에 대해 조직에서 데이터를 수집, 처리 및 지우는 방법을 규정합니다. HIPAA 세이프 하버 규정에 따라 데이터를 공유할 수 있는 자격을 갖추려면 전체 사진 이미지와 비교 이미지를 제거해야 합니다. 구조적 CT/MR 영상에서 개인의 얼굴 특징을 모호하게 하는 데 사용되는 식별 해제 또는 두개골 스트리핑 알고리즘과 같은 자동화 기술은 생물 의학 연구 기관의 데이터 공유 프로세스에서 필수적인 부분이 되었습니다.

AI/ML 분석의 클라우드 마이그레이션

엔터프라이즈 고객은 전통적으로 AI/ML 모델을 온프레미스에 교육 및 배포했습니다. 규모의 경제 및 효율성 측면에서 이러한 고객은 AI/ML 기능을 퍼블릭, 하이브리드 또는 멀티 클라우드 클라우드 구현 환경으로 이전하기 위해 확장하고 있습니다. 그러나 다른 인프라에 노출될 수 있는 데이터에 바인딩됩니다. NetApp 솔루션은 에 필요한 모든 범위의 사이버 보안 위협을 해결합니다 "데이터 보호" 또한, 보안 평가 및 프로토피아 데이터 변환과 결합하면 이미지 처리 AI/ML 워크로드를 클라우드로 마이그레이션하는 데 따른 위험을 최소화할 수 있습니다.

다른 업계에서의 에지 컴퓨팅 및 AI 추론의 추가 사용 사례는 를 참조하십시오 "TR-4886 Ai Inferencing at the Edge(가장자리 기준 TR-4886 AI 추론" 그리고 NetApp AI 블로그, "인텔리전스와 개인 정보 보호 비교".

기술 개요

이 섹션에서는 이 솔루션을 완료하는 데 필요한 다양한 기술 구성 요소에 대해 간략하게 설명합니다.

프로토피아

Protopia AI는 오늘날 시장에서 기밀 추론을 위한 방해되지 않는 소프트웨어 전용 솔루션을 제공합니다. Protopia

솔루션은 중요한 정보의 노출을 최소화하여 추론 서비스를 위한 탁월한 보호 기능을 제공합니다. AI는 현재 작업을 수행하는 데 꼭 필요한 데이터 레코드의 정보만 제공합니다. 대부분의 추론 작업에서는 모든 데이터 레코드에 있는 모든 정보를 사용하지 않습니다. AI가 이미지, 음성, 비디오 또는 구조화된 표 형식 데이터를 소비하고 있는지 여부와 관계없이 Protopia는 추론 서비스에 필요한 것만 제공합니다. 특허를 획득한 코어 기술은 수학적으로 선별된 소음을 사용하여 데이터를 변환하고 지정된 ML 서비스에서 필요하지 않은 정보를 제공합니다. 이 솔루션은 데이터를 마스킹하지 않고 선별된 랜덤 노이즈를 사용하여 데이터 표현을 변경합니다.

프로토피아 솔루션은 모델의 기능과 관련하여 입력 기능 공간에 관련 정보를 계속 유지하는 경사 기반 퍼터버레이션 최대화 방법으로 표현을 변경하는 문제를 공식화합니다. 이 검색 프로세스는 ML 모델 교육을 마칠 때 미세 조정 통과로 실행됩니다. 통과가 자동으로 확률 분포 세트를 생성한 후 오버헤드가 낮은 데이터 변환은 이러한 분포의 노이즈 샘플을 데이터에 적용하고 추론을 위해 모델에 전달하기 전에 난독 처리 합니다.

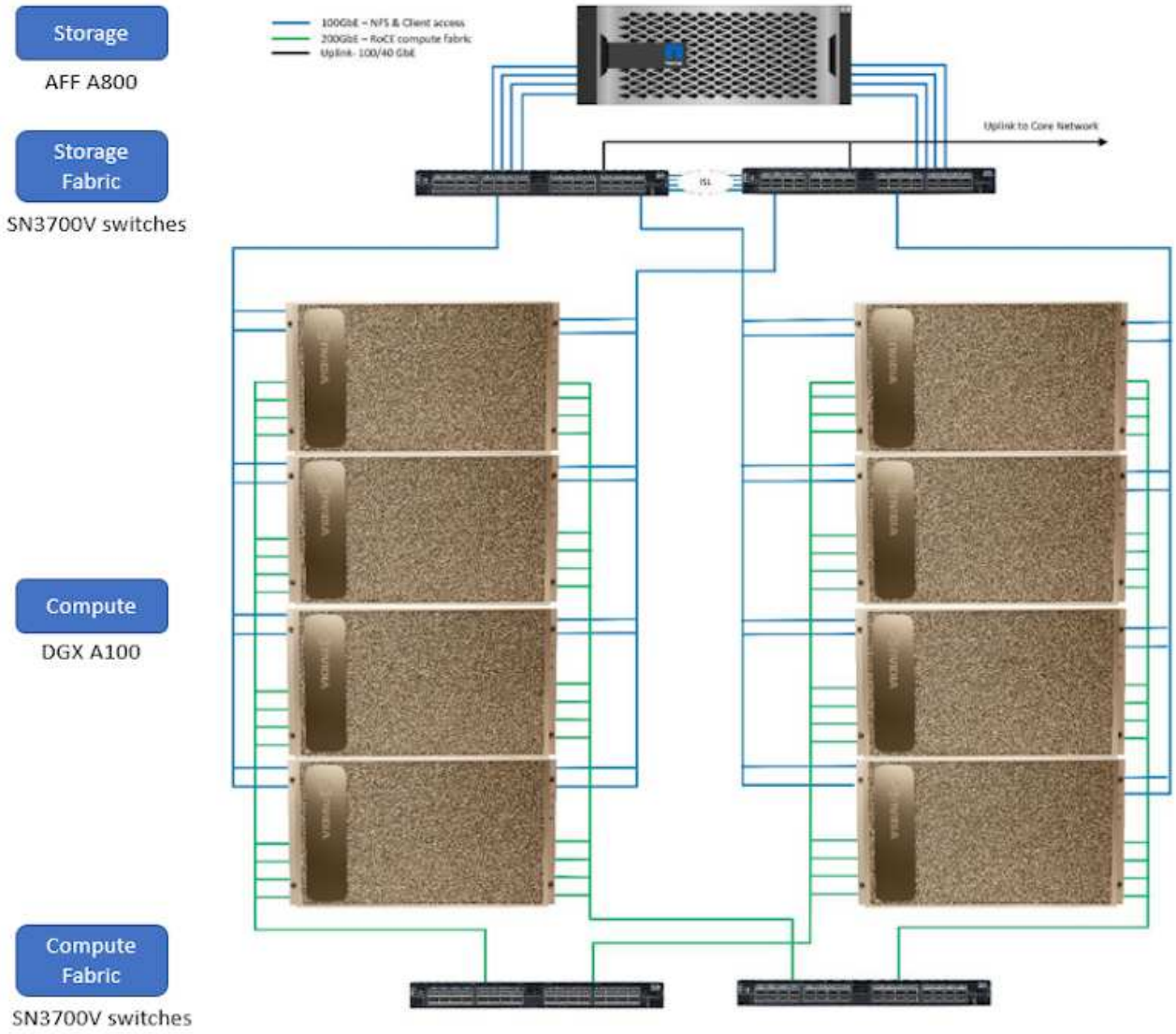
NetApp ONTAP AI를 참조하십시오

DGX A100 시스템과 NetApp 클라우드 연결형 스토리지 시스템에 기반한 NetApp ONTAP AI 참조 아키텍처는 NetApp과 NVIDIA가 개발 및 검증했습니다. 이 아키텍처는 IT 조직에 다음과 같은 이점을 제공하는 아키텍처를 제공합니다.

- 설계 복잡성 제거
- 컴퓨팅과 스토리지를 독립적으로 확장할 수 있습니다
- 고객이 작은 규모로 시작한 후 원활하게 확장할 수 있도록 지원
- 다양한 성능 및 비용 요소에 부합하는 폭넓은 스토리지 옵션을 제공합니다

ONTAP AI는 DGX A100 시스템과 NetApp AFF A800 스토리지 시스템을 최첨단 네트워킹과 긴밀하게 통합합니다. ONTAP AI는 설계 복잡성과 추측을 제거함으로써 AI 배포를 단순화합니다. 고객은 작은 규모로 시작한 후 에지에서 코어 및 클라우드까지 포괄하여 데이터를 지능적으로 관리하면서 중단 없이 확장할 수 있습니다.

다음 그림은 DGX A100 시스템과 ONTAP AI 솔루션 제품군의 다양한 변형을 보여줍니다. 최대 8개의 DGX A100 시스템에서 AFF A800 시스템 성능 검증 ONTAP 클러스터에 스토리지 컨트롤러 쌍을 추가하면 아키텍처를 여러 개의 랙으로 확장하여 선형 성능으로 많은 DGX A100 시스템과 페타바이트급 스토리지 용량을 지원할 수 있습니다. 이 접근 방식은 사용되는 DL 모델의 크기와 필요한 성능 메트릭을 기준으로 컴퓨팅 대 스토리지 비율을 독립적으로 변경할 수 있는 유연성을 제공합니다.



ONTAP AI에 대한 자세한 내용은 를 참조하십시오 "NVA-1153: NVIDIA DGX A100 시스템 및 Mellanox Spectrum 이더넷 스위치를 포함하는 NetApp ONTAP AI"

NetApp ONTAP를 참조하십시오

NetApp의 최신 세대 스토리지 관리 소프트웨어인 ONTAP 9.11을 통해 기업은 인프라를 현대화하고 클라우드 지원 데이터 센터로 전환할 수 있습니다. ONTAP는 업계 최고 수준의 데이터 관리 기능을 활용하여 데이터가 상주하는 위치와 상관없이 단일 툴셋으로 데이터를 관리하고 보호할 수 있습니다. 필요에 따라 에지, 코어, 클라우드 등 어느 위치로도 데이터를 자유롭게 이동할 수 있습니다. ONTAP 9.11에는 데이터 관리를 단순화하고, 중요 데이터를 가속화하고, 보호하며, 하이브리드 클라우드 아키텍처 전반에 걸쳐 차세대 인프라 기능을 지원하는 다양한 기능이 포함되어 있습니다.

NetApp DataOps 툴킷

NetApp DataOps Toolkit은 개발자, 데이터 과학자, DevOps 엔지니어 및 데이터 엔지니어가 새로운 데이터 볼륨의 거의 즉각적인 프로비저닝 또는 JupyterLab 작업 공간, 데이터 볼륨의 거의 즉각적인 클론 복제 또는 JupyterLab 작업 공간과 같은 다양한 데이터 관리 작업을 간단하게 수행할 수 있는 Python 라이브러리입니다. 추적 기능 또는 베이스라인 기능을 위해 데이터 볼륨의 스냅샷 또는 JupyterLab 작업 공간을 거의 즉각적으로 생성합니다. 이 Python 라이브러리는 명령행 유틸리티 또는 Python 프로그램이나 Jupyter 노트북으로 가져올 수 있는 기능의 라이브러리로 작동할 수

있습니다.

NVIDIA Triton Inference Server를 참조하십시오

NVIDIA Triton Inference Server는 모델 구축 및 실행을 표준화하여 운영 환경에서 신속하고 확장 가능한 AI를 제공하는 오픈 소스 추론 제공 소프트웨어입니다. Triton Inference Server는 팀이 GPU 또는 CPU 기반 인프라의 모든 프레임워크에서 훈련된 AI 모델을 구축, 실행 및 확장할 수 있도록 지원하여 AI 추론을 간소화합니다. Triton Inference Server는 TensorFlow, NVIDIA TensorRT, PyTorch, MXNet, OpenVINO 등 Triton은 모든 주요 퍼블릭 클라우드 AI 및 Kubernetes 플랫폼에서 사용할 수 있는 오케스트레이션 및 확장을 위해 Kubernetes와 통합됩니다. 또한 많은 MLOps 소프트웨어 솔루션과 통합됩니다.

PyTorch

"PyTorch" 는 오픈 소스 ML 프레임워크입니다. GPU 및 CPU를 사용하는 딥 러닝용으로 최적화된 텐서 라이브러리입니다. PyTorch 패키지에는 여러 유용한 유틸리티 간에 효율적인 서너 직렬화를 위한 여러 유틸리티를 제공하는 다차원 Tensor용 데이터 구조가 포함되어 있습니다. 또한 컴퓨팅 기능이 있는 NVIDIA GPU에서 텐서 컴퓨팅을 실행할 수 있는 CUDA 상대가 있습니다. 이 검증에서는 OpenCV-Python(CV2) 라이브러리를 사용하여 모델을 검증하고 Python의 가장 직관적인 컴퓨터 비전 개념을 활용합니다.

데이터 관리를 단순화하십시오

데이터 관리는 AI 애플리케이션에 적합한 리소스를 사용하고 AI/ML 데이터 세트를 교육할 수 있도록 엔터프라이즈 IT 운영 및 데이터 과학자에게 매우 중요합니다. NetApp 기술에 대한 다음 추가 정보는 이 검증의 범위에 포함되지 않지만, 배포에 따라 달라질 수 있습니다.

ONTAP 데이터 관리 소프트웨어에는 운영을 간소화 및 단순화하고 총 운영 비용을 절감하는 다음과 같은 기능이 있습니다.

- 인라인 데이터 컴팩션 및 확대된 중복제거: 데이터 컴팩션은 스토리지 블록 내부의 낭비되는 공간을 줄이고, 중복제거는 실제 용량을 상당히 늘려줍니다. 이는 로컬에 저장된 데이터와 클라우드로 계층화된 데이터에 적용됩니다.
- 최소, 최대 및 적응형 서비스 품질(AQoS): 세부적인 서비스 품질(QoS) 제어로 고도의 공유 환경에서 중요 애플리케이션의 성능 수준을 유지할 수 있습니다.
- NetApp FabricPool를 참조하십시오. AWS(Amazon Web Services), Azure, NetApp StorageGRID 스토리지 솔루션을 포함한 퍼블릭 클라우드 및 프라이빗 클라우드 스토리지에 콜드 데이터를 자동으로 계층화합니다. FabricPool에 대한 자세한 내용은 를 참조하십시오 "TR-4598: FabricPool 모범 사례".

데이터 가속화 및 보호

ONTAP는 탁월한 수준의 성능과 데이터 보호를 제공하며 다음과 같은 방법으로 이러한 기능을 확장합니다.

- 성능 및 짧은 지연 시간: ONTAP는 가장 짧은 지연 시간으로 가장 높은 처리량을 제공합니다.
- 데이터 보호: ONTAP는 모든 플랫폼에서 공통 관리를 지원하는 내장 데이터 보호 기능을 제공합니다.
- NVE(NetApp 볼륨 암호화). ONTAP는 온보드 및 외부 키 관리를 모두 지원하는 기본 볼륨 레벨 암호화를 제공합니다.
- 멀티테넌시 및 다단계 인증. ONTAP를 사용하면 인프라 리소스를 최고 수준의 보안으로 공유할 수 있습니다.

미래 지향형 인프라

ONTAP은 다음과 같은 기능을 통해 끊임없이 변화하는 까다로운 비즈니스 요구사항을 충족할 수 있도록 지원합니다.

- 원활한 확장 및 무중단 운영: ONTAP은 운영 중단 없이 기존 컨트롤러 및 스케일아웃 클러스터에 용량을 추가할 수 있도록 지원합니다. 고객은 고비용이 따르는 데이터 마이그레이션이나 운영 중단 없이 NVMe 및 32Gb FC와 같은 최신 기술로 업그레이드할 수 있습니다.
- 클라우드 연결: ONTAP은 클라우드에 가장 많이 연결된 스토리지 관리 소프트웨어로, 모든 퍼블릭 클라우드에서 ONTAP Select(소프트웨어 정의 스토리지) 및 NetApp Cloud Volumes Service(클라우드 네이티브 인스턴스)에 대한 옵션을 제공합니다.
- 새로운 애플리케이션과 통합: ONTAP은 기존 엔터프라이즈 앱을 지원하는 인프라와 동일한 인프라를 사용하여 자율주행 차량, 스마트 시티, Industry 4.0과 같은 차세대 플랫폼 및 애플리케이션을 위한 엔터프라이즈급 데이터 서비스를 제공합니다.

NetApp Astra Control

NetApp Astra 제품군은 온프레미스 및 퍼블릭 클라우드에서 Kubernetes 애플리케이션을 위한 스토리지 및 애플리케이션 인식 데이터 관리 서비스를 제공하며, NetApp 스토리지 및 데이터 관리 기술을 기반으로 합니다. Kubernetes 애플리케이션을 쉽게 백업하고, 데이터를 다른 클러스터로 마이그레이션하고, 작업 중인 애플리케이션 클론을 즉시 생성할 수 있습니다. 퍼블릭 클라우드에서 실행 중인 Kubernetes 애플리케이션을 관리해야 하는 경우에는 의 문서를 참조하십시오 "Astra 제어 서비스". Astra Control Service는 GKE(Google Kubernetes Engine) 및 AKS(Azure Kubernetes Service)에서 Kubernetes 클러스터의 애플리케이션 인식 데이터 관리를 제공하는 NetApp 관리 서비스입니다.

NetApp Astra Trident

아스트라 "트라이던트" NetApp은 Docker 및 Kubernetes용 오픈 소스 동적 스토리지 오케스트레이터로서 영구 스토리지의 생성, 관리 및 사용을 단순화합니다. Kubernetes 네이티브 애플리케이션인 Trident는 Kubernetes 클러스터 내에서 직접 실행됩니다. Trident를 사용하면 고객이 DL 컨테이너 이미지를 NetApp 스토리지에 원활하게 배포하고 AI 컨테이너 배포를 위한 엔터프라이즈급 경험을 제공할 수 있습니다. Kubernetes 사용자(ML 개발자, 데이터 과학자 등)는 오케스트레이션 및 클론 복제를 생성, 관리 및 자동화하여 NetApp 기술이 제공하는 고급 데이터 관리 기능을 활용할 수 있습니다.

NetApp BlueXP 복사 및 동기화

"BlueXP 복사 및 동기화" 는 빠르고 안전한 데이터 동기화를 제공하는 NetApp 서비스입니다. 온프레미스 NFS 또는 SMB 파일 공유 간에 파일을 전송해야 하는 경우, NetApp StorageGRID, NetApp ONTAP S3, NetApp Cloud Volumes Service, Azure NetApp Files, Amazon Simple Storage Service(Amazon S3), Amazon Elastic File System(Amazon EFS), Azure Blob, Google Cloud Storage, 또는 IBM Cloud Object Storage인 BlueXP Copy and Sync는 필요한 파일을 빠르고 안전하게 이동합니다. 데이터가 전송되면 소스와 타겟 모두에서 사용할 수 있습니다. BlueXP Copy 및 Syncc 는 미리 정의된 일정에 따라 데이터를 지속적으로 동기화하므로 변경된 부분만 이동하므로 데이터 복제에 소비되는 시간과 비용이 최소화됩니다. BlueXP Copy and Sync는 설정 및 사용이 매우 간편한 서비스형 소프트웨어(SaaS) 툴입니다. BlueXP Copy 및 Sync에 의해 트리거되는 데이터 전송은 데이터 브로커에 의해 수행됩니다. AWS, Azure, Google Cloud Platform 또는 사내에 BlueXP Copy 및 Sync 데이터 브로커를 배포할 수 있습니다.

NetApp BlueXP 분류

강력한 AI 알고리즘을 기반으로 "NetApp BlueXP 분류" 전체 데이터 자산에 걸쳐 자동화된 제어 및 데이터 거버넌스를 제공합니다. 비용 절감 효과를 쉽게 파악하고 규정 준수 및 개인 정보 보호에 대한 우려 사항을 파악하며 최적화 기회를 찾을 수 있습니다. BlueXP Classification 대시보드를 통해 중복 데이터를 식별하여 중복을 제거하고 개인, 비개인 및 중요 데이터를 매핑하고 기밀 데이터 및 이상 상황에 대한 알림을 설정할 수 있습니다.

테스트 및 검증 계획

이 솔루션 설계의 경우 다음 세 가지 시나리오를 검증했습니다.

- Kubernetes용 NetApp DataOps Toolkit을 사용하여 조율된 JupyterLab 작업 공간 내의 프로토피아 난독 처리 기능을 사용하거나 사용하지 않는 추론 작업.
- Kubernetes용 NetApp DataOps Toolkit을 사용하여 오케스트레이션된 데이터 볼륨을 사용하는 Kubernetes에서 프로토피아 난독 처리를 사용하거나 사용하지 않는 배치 추론 작업
- Kubernetes용 NetApp DataOps 툴킷을 사용하여 조정된 NVIDIA Triton Inference Server 인스턴스를 사용한 추론 작업 네트워크를 통해 전송되는 모든 데이터가 난독 처리되어야 한다는 일반적인 요구 사항을 시뮬레이션하기 위해 Triton 추론 API를 호출하기 전에 이미지에 Protopia 난독 처리를 적용했습니다. 이 워크플로는 신뢰할 수 있는 영역 내에서 데이터를 수집하지만 추론을 위해 신뢰할 수 있는 영역 외부로 전달해야 하는 사용 사례에 적용됩니다. Protopia 난독 처리를 사용하지 않으면 중요한 데이터가 신뢰할 수 있는 영역을 벗어나지 않으면 이러한 유형의 워크플로를 구현할 수 없습니다.

구성을 테스트합니다

다음 표에서는 솔루션 설계 검증 환경을 간략하게 보여 줍니다.

구성 요소	버전
쿠버네티스	1.21.6
NetApp Astra Trident CSI 드라이버	22.01.0
Kubernetes용 NetApp DataOps 툴킷	2.3.0
NVIDIA Triton Inference Server를 참조하십시오	21.11-3장

테스트 절차

이 섹션에서는 검증을 완료하는 데 필요한 작업에 대해 설명합니다.

필수 구성 요소

이 섹션에 설명된 작업을 실행하려면 다음 도구가 설치 및 구성되어 있는 Linux 또는 macOS 호스트에 대한 액세스 권한이 있어야 합니다.

- Kubectl(기존 Kubernetes 클러스터에 액세스하도록 구성)
  - 설치 및 구성 지침을 찾을 수 있습니다 "여기".
- Kubernetes용 NetApp DataOps 툴킷
  - 설치 지침을 찾을 수 있습니다 "여기".

시나리오 **1 – JupyterLab**의 온디맨드 추론

1. AI/ML 추론 워크로드를 위한 Kubernetes 네임스페이스를 생성합니다.

```
$ kubectl create namespace inference
namespace/inference created
```

2. NetApp DataOps 툴킷을 사용하여 추론을 수행할 데이터를 저장할 영구 볼륨을 프로비저닝합니다.

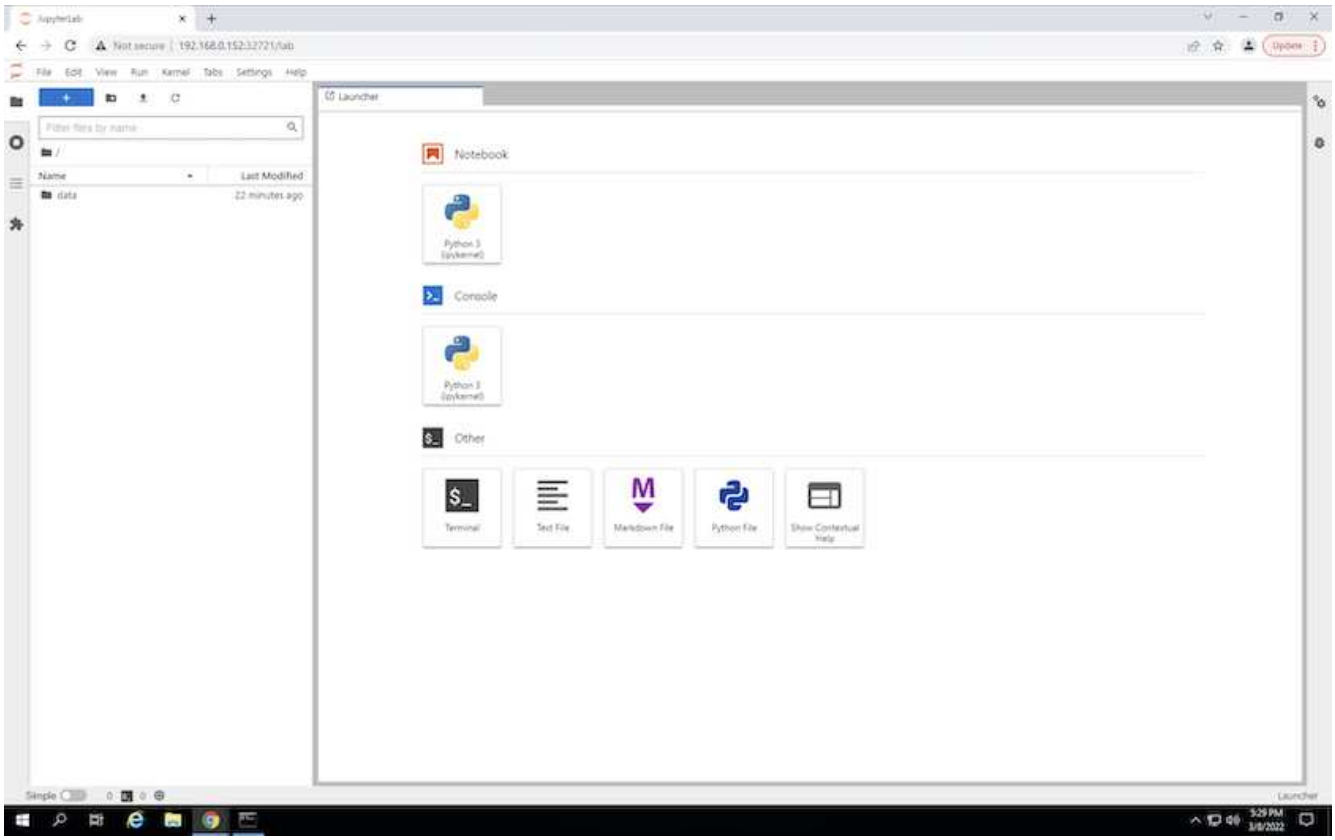
```
$ netapp_dataops_k8s_cli.py create volume --namespace=inference --pvc
-name=inference-data --size=50Gi
Creating PersistentVolumeClaim (PVC) 'inference-data' in namespace
'inference'.
PersistentVolumeClaim (PVC) 'inference-data' created. Waiting for
Kubernetes to bind volume to PVC.
Volume successfully created and bound to PersistentVolumeClaim (PVC)
'inference-data' in namespace 'inference'.
```

3. NetApp DataOps Toolkit을 사용하여 새로운 JupyterLab 작업 공간을 생성합니다. '--mount-PVC' 옵션을 사용하여 이전 단계에서 생성한 영구 볼륨을 마운트합니다. 필요한 경우 '--nVidia-GPU' 옵션을 사용하여 NVIDIA GPU를 작업 공간에 할당합니다.

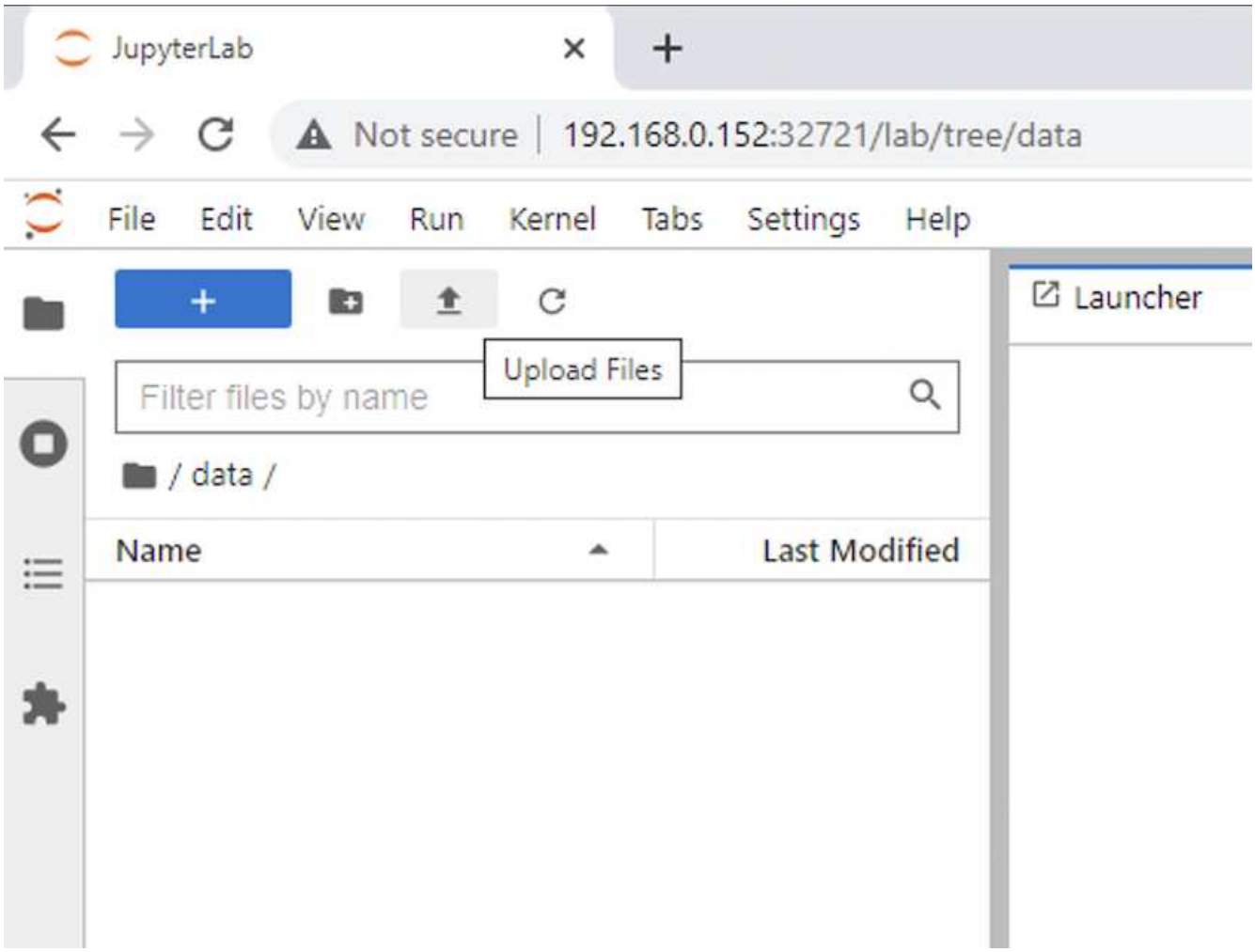
   다음 예에서는 영구 볼륨 '추론 데이터'가 '/home/jovyan/data'의 JupyterLab 작업 영역 컨테이너에 마운트됩니다. Jupytter의 공식 컨테이너 이미지를 사용할 때 JupyterLab 웹 인터페이스 내의 최상위 디렉토리로 /home/jovyan이 표시됩니다.

```
$ netapp_dataops_k8s_cli.py create jupyterlab --namespace=inference
--workspace-name=live-inference --size=50Gi --nvidia-gpu=2 --mount
-pvc=inference-data:/home/jovyan/data
Set workspace password (this password will be required in order to
access the workspace):
Re-enter password:
Creating persistent volume for workspace...
Creating PersistentVolumeClaim (PVC) 'ntap-dsutil-jupyterlab-live-
inference' in namespace 'inference'.
PersistentVolumeClaim (PVC) 'ntap-dsutil-jupyterlab-live-inference'
created. Waiting for Kubernetes to bind volume to PVC.
Volume successfully created and bound to PersistentVolumeClaim (PVC)
'ntap-dsutil-jupyterlab-live-inference' in namespace 'inference'.
Creating Service 'ntap-dsutil-jupyterlab-live-inference' in namespace
'inference'.
Service successfully created.
Attaching Additional PVC: 'inference-data' at mount_path:
'/home/jovyan/data'.
Creating Deployment 'ntap-dsutil-jupyterlab-live-inference' in namespace
'inference'.
Deployment 'ntap-dsutil-jupyterlab-live-inference' created.
Waiting for Deployment 'ntap-dsutil-jupyterlab-live-inference' to reach
Ready state.
Deployment successfully created.
Workspace successfully created.
To access workspace, navigate to http://192.168.0.152:32721
```

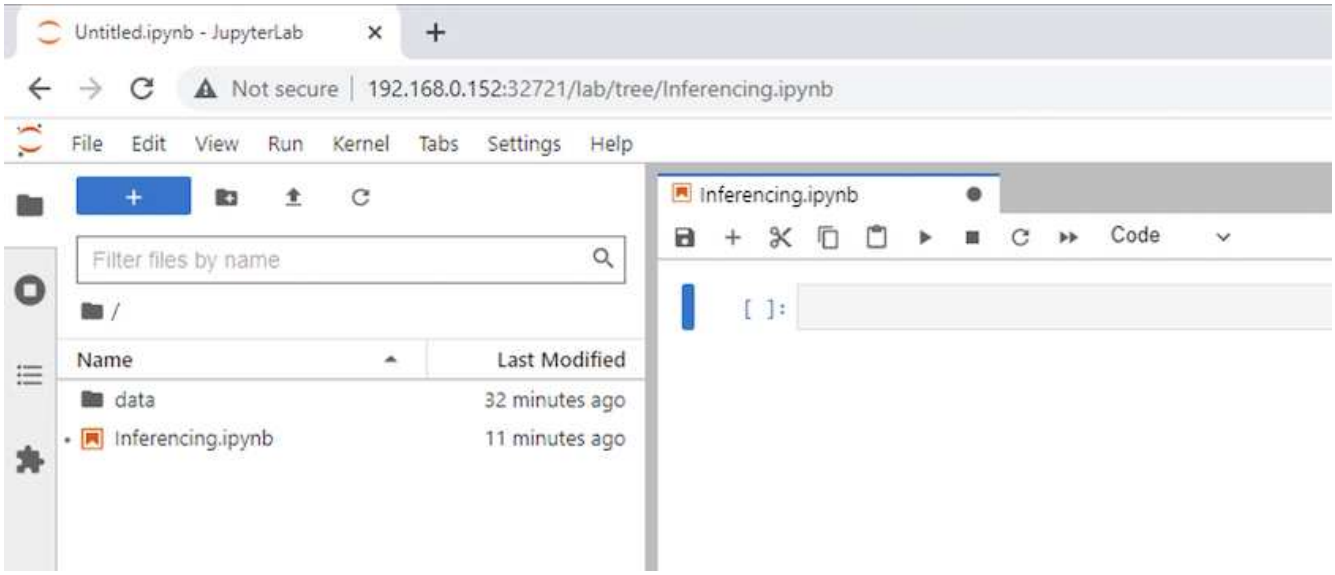
4. 'jupyterlab 생성' 명령의 출력에 지정된 URL을 사용하여 JupyterLab 작업 영역에 액세스합니다. 데이터 디렉토리는 작업 공간에 마운트된 영구 볼륨을 나타냅니다.



5. "ata" 디렉토리를 열고 추론을 수행할 파일을 업로드합니다. 파일이 데이터 디렉토리에 업로드되면 작업 공간에 마운트된 영구 볼륨에 자동으로 저장됩니다. 파일을 업로드하려면 다음 이미지와 같이 파일 업로드 아이콘을 클릭합니다.



6. 최상위 디렉토리로 돌아가서 새 전자 필기장을 만듭니다.



7. 노트북에 추론 코드를 추가합니다. 다음 예에서는 이미지 감지 사용 사례에 대한 추론 코드를 보여 줍니다.

Launcher | image-demo-pytorch.ipynb

Code | Python 3 (ipykernel)

STEP 3-1: Clean (Without obfuscation) detection

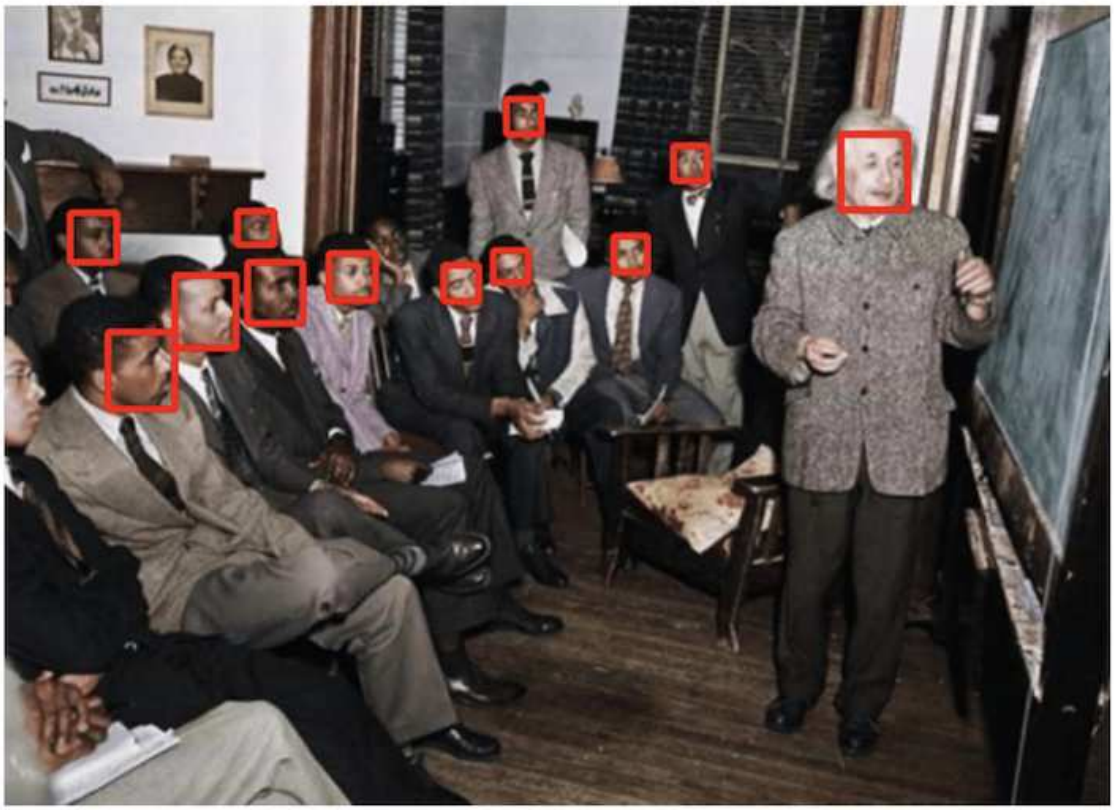
```
[9]: # get current frame
frame = input_image

# preprocess input
preprocessed_input = preprocess_input(frame)
preprocessed_input = torch.Tensor(preprocessed_input).to(device)

# run forward pass
clean_activation = clean_model.forward_head(preprocessed_input)  # runs the first few layers
loc, pred = clean_model.forward_tail(clean_activation)  # runs rest of the layers

# postprocess output
clean_pred = (loc.detach().cpu().numpy(), pred.detach().cpu().numpy())
clean_outputs = postprocess_outputs(
    clean_pred, [[input_image_width, input_image_height]], priors, THRESHOLD
)

# draw rectangles
clean_frame = copy.deepcopy(frame)  # needs to be deep copy
for (x1, y1, x2, y2, s) in clean_outputs[0]:
    x1, y1 = int(x1), int(y1)
    x2, y2 = int(x2), int(y2)
    cv2.rectangle(clean_frame, (x1, y1), (x2, y2), (0, 0, 255), 4)
```

Launcher | image-demo-pytorch.ipynb

Code | Python 3 (ipykernel)

Visualize Clean (Without obfuscation) detection

```
[10]: show_cv2_image(clean_frame, scale=2)
```



8. 추론 코드에 Protopia 난독 처리를 추가합니다. Protopia는 고객과 직접 협력하여 사용 사례별 문서를 제공하며 이 기술 보고서의 범위를 벗어납니다. 다음 예제에서는 Protopia 난독 처리를 추가한 이미지 검색 사용 사례에 대한 추론 코드를 보여 줍니다.

Launcher | image-demo-pytorch.ipynb

Code | Python 3 (ipykernel)

STEP 3-2: Protopia AI (With obfuscation) detection

[11]:
```
# get current frame
frame = input_image

# preprocess input
preprocessed_input = preprocess_input(frame)
preprocessed_input = torch.Tensor(preprocessed_input).to(device)

# run forward pass
not_noisy_activation = noisy_model.forward_head(preprocessed_input)  # runs the first few layers
##################################################################
#          SINGLE ADITIONAL LINE FOR PRIVATE INFERENCE           #
##################################################################
noisy_activation = noisy_model.forward_noise(not_noisy_activation)
##################################################################
loc, pred = noisy_model.forward_tail(noisy_activation)  # runs rest of the layers

# postprocess output
noisy_pred = (loc.detach().cpu().numpy(), pred.detach().cpu().numpy())
noisy_outputs = postprocess_outputs(
    noisy_pred, [[input_image_width, input_image_height]], priors, THRESHOLD * 0.5
)

# get reconstruction of the noisy activation
noisy_reconstruction = decoder_function(noisy_activation)
noisy_reconstruction = noisy_reconstruction.detach().cpu().numpy()[0]
noisy_reconstruction = unpreprocess_output(
    noisy_reconstruction, (input_image_width, input_image_height), True
).astype(np.uint8)

# draw rectangles
for (x1, y1, x2, y2, s) in noisy_outputs[0]:
    x1, y1 = int(x1), int(y1)
    x2, y2 = int(x2), int(y2)
    cv2.rectangle(noisy_reconstruction, (x1, y1), (x2, y2), (0, 0, 255), 4)
```

Launcher | image-demo-pytorch.ipynb

Code | Python 3 (ipykernel)

Visualize Protopia AI (With obfuscation) detection

[12]:
```
show_cv2_image(noisy_reconstruction, scale=2)
```

시나리오 **2 – Kubernetes**의 배치 추론

1. AI/ML 추론 워크로드를 위한 Kubernetes 네임스페이스를 생성합니다.

```
$ kubectl create namespace inference
namespace/inference created
```

2. NetApp DataOps 툴킷을 사용하여 추론을 수행할 데이터를 저장할 영구 볼륨을 프로비저닝합니다.

```
$ netapp_dataops_k8s_cli.py create volume --namespace=inference --pvc
-name=inference-data --size=50Gi
Creating PersistentVolumeClaim (PVC) 'inference-data' in namespace
'inference'.
PersistentVolumeClaim (PVC) 'inference-data' created. Waiting for
Kubernetes to bind volume to PVC.
Volume successfully created and bound to PersistentVolumeClaim (PVC)
'inference-data' in namespace 'inference'.
```

3. 추론을 수행할 데이터로 새 영구 볼륨을 채웁니다.

   PVC로 데이터를 로드하는 방법은 여러 가지가 있습니다. 현재 데이터가 NetApp StorageGRID 또는 Amazon S3와 같은 S3 호환 오브젝트 스토리지 플랫폼에 저장되어 있는 경우 를 사용할 수 있습니다 "NetApp DataOps 툴킷 S3 Data Mover 기능". 또 하나의 간단한 방법은 JupyterLab 작업 공간을 만든 다음, 섹션 “의 3-5단계에 설명된 대로 JupyterLab 웹 인터페이스를 통해 파일을 업로드하는 것입니다시나리오 1 – JupyterLab의 온디맨드 추론.”

4. 배치 추론 작업을 위해 Kubernetes 작업을 생성합니다. 다음 예는 이미지 감지 사용 사례에 대한 배치 추론 작업을 보여줍니다. 이 작업은 이미지 세트의 각 이미지에서 추론을 수행하고 추론 정확도 메트릭을 stdout에 씁니다.

```
$ vi inference-job-raw.yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: netapp-inference-raw
  namespace: inference
spec:
  backoffLimit: 5
  template:
    spec:
      volumes:
      - name: data
        persistentVolumeClaim:
          claimName: inference-data
      - name: dshm
        emptyDir:
          medium: Memory
      containers:
      - name: inference
        image: netapp-protopia-inference:latest
        imagePullPolicy: IfNotPresent
        command: ["python3", "run-accuracy-measurement.py", "--dataset",
"/data/netapp-face-detection/FDDB"]
        resources:
          limits:
            nvidia.com/gpu: 2
        volumeMounts:
        - mountPath: /data
          name: data
        - mountPath: /dev/shm
          name: dshm
      restartPolicy: Never
$ kubectl create -f inference-job-raw.yaml
job.batch/netapp-inference-raw created
```

5. 추론 작업이 성공적으로 완료되었는지 확인합니다.

```
$ kubectl -n inference logs netapp-inference-raw-255sp
100%|██████████| 89/89 [00:52<00:00,  1.68it/s]
Reading Predictions : 100%|██████████| 10/10 [00:01<00:00,  6.23it/s]
Predicting ... : 100%|██████████| 10/10 [00:16<00:00,  1.64s/it]
==================== Results ====================
FDDB-fold-1 Val AP: 0.9491256561145955
FDDB-fold-2 Val AP: 0.9205024466101926
FDDB-fold-3 Val AP: 0.9253013871078468
FDDB-fold-4 Val AP: 0.9399781485863011
FDDB-fold-5 Val AP: 0.9504280149478732
FDDB-fold-6 Val AP: 0.9416473519339292
FDDB-fold-7 Val AP: 0.9241631566241117
FDDB-fold-8 Val AP: 0.9072663297546659
FDDB-fold-9 Val AP: 0.9339648715035469
FDDB-fold-10 Val AP: 0.9447707905560152
FDDB Dataset Average AP: 0.9337148153739079
==================================================
mAP: 0.9337148153739079
```

6. 추론 작업에 Protopia 난독 처리를 추가합니다. 이 기술 보고서의 범위를 벗어나는 Protopia에서 직접 Protopia 난독 처리를 추가하기 위한 사용 사례별 지침을 찾을 수 있습니다. 다음 예제는 알파 값 0.8을 사용하여 Protopia 난독 처리가 추가된 얼굴 인식 사용 사례에 대한 일괄 추론 작업을 보여 줍니다. 이 작업은 이미지 세트의 각 이미지에 대한 추론을 수행하기 전에 Protopia 난독 처리를 적용한 다음 추론 정확도 메트릭을 stdout에 기록합니다.

   알파 값 0.05, 0.1, 0.2, 0.4, 0.6, 0.8, 0.9 및 0.95. 에서 결과를 볼 수 있습니다 ""추론 정확도 비교.""

```
$ vi inference-job-protopia-0.8.yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: netapp-inference-protopia-0.8
  namespace: inference
spec:
  backoffLimit: 5
  template:
    spec:
      volumes:
      - name: data
        persistentVolumeClaim:
          claimName: inference-data
      - name: dshm
        emptyDir:
          medium: Memory
      containers:
      - name: inference
        image: netapp-protopia-inference:latest
        imagePullPolicy: IfNotPresent
        env:
        - name: ALPHA
          value: "0.8"
        command: ["python3", "run-accuracy-measurement.py", "--dataset",
"/data/netapp-face-detection/FDDB", "--alpha", "$(ALPHA)", "--noisy"]
        resources:
          limits:
            nvidia.com/gpu: 2
        volumeMounts:
        - mountPath: /data
          name: data
        - mountPath: /dev/shm
          name: dshm
      restartPolicy: Never
$ kubectl create -f inference-job-protopia-0.8.yaml
job.batch/netapp-inference-protopia-0.8 created
```

7. 추론 작업이 성공적으로 완료되었는지 확인합니다.

```
$ kubectl -n inference logs netapp-inference-protopia-0.8-b4dkz
100%|██████████| 89/89 [01:05<00:00,  1.37it/s]
Reading Predictions : 100%|██████████| 10/10 [00:02<00:00,  3.67it/s]
Predicting ... : 100%|██████████| 10/10 [00:22<00:00,  2.24s/it]
==================== Results ====================
FDDB-fold-1 Val AP: 0.8953066115834589
FDDB-fold-2 Val AP: 0.8819580264029936
FDDB-fold-3 Val AP: 0.8781107458462862
FDDB-fold-4 Val AP: 0.9085731346308461
FDDB-fold-5 Val AP: 0.9166445508275378
FDDB-fold-6 Val AP: 0.9101178994188819
FDDB-fold-7 Val AP: 0.8383443678423771
FDDB-fold-8 Val AP: 0.8476311547659464
FDDB-fold-9 Val AP: 0.8739624502111121
FDDB-fold-10 Val AP: 0.8905468076424851
FDDB Dataset Average AP: 0.8841195749171925
==================================================
mAP: 0.8841195749171925
```

시나리오 **3 – NVIDIA Triton Inference Server**

1. AI/ML 추론 워크로드를 위한 Kubernetes 네임스페이스를 생성합니다.

```
$ kubectl create namespace inference
namespace/inference created
```

2. NetApp DataOps 툴킷을 사용하여 NVIDIA Triton Inference Server의 모델 저장소로 사용할 영구 볼륨을 프로비저닝합니다.

```
$ netapp_dataops_k8s_cli.py create volume --namespace=inference --pvc
-name=triton-model-repo --size=100Gi
Creating PersistentVolumeClaim (PVC) 'triton-model-repo' in namespace
'inference'.
PersistentVolumeClaim (PVC) 'triton-model-repo' created. Waiting for
Kubernetes to bind volume to PVC.
Volume successfully created and bound to PersistentVolumeClaim (PVC)
'triton-model-repo' in namespace 'inference'.
```

3. 의 새 영구 볼륨에 모델을 저장합니다 "형식" 이 기능은 NVIDIA Triton Inference Server에서 인식됩니다.

   PVC로 데이터를 로드하는 방법은 여러 가지가 있습니다. 간단한 방법은 “의 3-5단계에 설명된 대로 JupyterLab 작업 공간을 만든 다음 JupyterLab 웹 인터페이스를 통해 파일을 업로드하는 것입니다시나리오 1 – JupyterLab의 온디맨드 추론. ”

4. NetApp DataOps 툴킷을 사용하여 새 NVIDIA Triton Inference Server 인스턴스를 구축합니다.

```
$ netapp_dataops_k8s_cli.py create triton-server --namespace=inference
--server-name=netapp-inference --model-repo-pvc-name=triton-model-repo
Creating Service 'ntap-dsutil-triton-netapp-inference' in namespace
'inference'.
Service successfully created.
Creating Deployment 'ntap-dsutil-triton-netapp-inference' in namespace
'inference'.
Deployment 'ntap-dsutil-triton-netapp-inference' created.
Waiting for Deployment 'ntap-dsutil-triton-netapp-inference' to reach
Ready state.
Deployment successfully created.
Server successfully created.
Server endpoints:
http: 192.168.0.152: 31208
grpc: 192.168.0.152: 32736
metrics: 192.168.0.152: 30009/metrics
```

5. Triton 클라이언트 SDK를 사용하여 추론 작업을 수행합니다. 인용된 다음 Python 코드는 Triton Python 클라이언트 SDK를 사용하여 얼굴 감지 사용 사례에 대한 추론 작업을 수행합니다. 이 예에서는 Triton API를 호출하고 추론을 위해 이미지를 전달합니다. 그런 다음 Triton Inference Server가 요청을 수신하고 모델을 호출하고 추론 출력을 API 결과의 일부로 반환합니다.

```
# get current frame
frame = input_image
# preprocess input
preprocessed_input = preprocess_input(frame)
preprocessed_input = torch.Tensor(preprocessed_input).to(device)
# run forward pass
clean_activation = clean_model_head(preprocessed_input)  # runs the
first few layers
#######################################################################
##############
#          pass clean image to Triton Inference Server API for
inferencing           #
#######################################################################
##############
triton_client =
httpclient.InferenceServerClient(url="192.168.0.152:31208",
verbose=False)
model_name = "face_detection_base"
inputs = []
outputs = []
inputs.append(httpclient.InferInput("INPUT__0", [1, 128, 32, 32],
```

```
"FP32"))
inputs[0].set_data_from_numpy(clean_activation.detach().cpu().numpy(),
binary_data=False)
outputs.append(httpclient.InferRequestedOutput("OUTPUT__0",
binary_data=False))
outputs.append(httpclient.InferRequestedOutput("OUTPUT__1",
binary_data=False))
results = triton_client.infer(
    model_name,
    inputs,
    outputs=outputs,
    #query_params=query_params,
    headers=None,
    request_compression_algorithm=None,
    response_compression_algorithm=None)
#print(results.get_response())
statistics =
triton_client.get_inference_statistics(model_name=model_name,
headers=None)
print(statistics)
if len(statistics["model_stats"]) != 1:
    print("FAILED: Inference Statistics")
    sys.exit(1)

loc_numpy = results.as_numpy("OUTPUT__0")
pred_numpy = results.as_numpy("OUTPUT__1")
#######################################################################
##############
# postprocess output
clean_pred = (loc_numpy, pred_numpy)
clean_outputs = postprocess_outputs(
    clean_pred, [[input_image_width, input_image_height]], priors,
THRESHOLD
)
# draw rectangles
clean_frame = copy.deepcopy(frame)  # needs to be deep copy
for (x1, y1, x2, y2, s) in clean_outputs[0]:
    x1, y1 = int(x1), int(y1)
    x2, y2 = int(x2), int(y2)
    cv2.rectangle(clean_frame, (x1, y1), (x2, y2), (0, 0, 255), 4)
```

6. 추론 코드에 Protopia 난독 처리를 추가합니다. Protopia에서 직접 Protopia 난독 처리를 추가하기 위한 사용 사례별 지침을 찾을 수 있지만 이 프로세스는 이 기술 보고서의 범위를 벗어납니다. 다음 예제에서는 앞의 5단계에서 표시되지만 Protopia 난독 처리를 추가한 것과 동일한 Python 코드를 보여 줍니다.

   이 경우, Triton API로 전달되기 전에 Protopia 난독 처리 기능이 이미지에 적용됩니다. 따라서, 난독 처리된

이미지가 로컬 시스템에서 절대 빠져나가지는 않습니다. 난독 처리된 이미지만 네트워크를 통해 전달됩니다. 이 워크플로는 신뢰할 수 있는 영역 내에서 데이터를 수집한 다음 추론을 위해 신뢰할 수 있는 영역 외부로 전달해야 하는 사용 사례에 적용됩니다. Protopia 난독 처리를 사용하지 않으면 중요한 데이터가 신뢰할 수 있는 영역을 벗어나지 않으면 이러한 유형의 워크플로를 구현할 수 없습니다.

```
# get current frame
frame = input_image
# preprocess input
preprocessed_input = preprocess_input(frame)
preprocessed_input = torch.Tensor(preprocessed_input).to(device)
# run forward pass
not_noisy_activation = noisy_model_head(preprocessed_input)  # runs the
first few layers
#################################################################
#          obfuscate image locally prior to inferencing          #
#          SINGLE ADITIONAL LINE FOR PRIVATE INFERENCE           #
#################################################################
noisy_activation = noisy_model_noise(not_noisy_activation)
#################################################################
#######################################################################
####################
#          pass obfuscated image to Triton Inference Server API for
inferencing           #
#######################################################################
####################
triton_client =
httpclient.InferenceServerClient(url="192.168.0.152:31208",
verbose=False)
model_name = "face_detection_noisy"
inputs = []
outputs = []
inputs.append(httpclient.InferInput("INPUT__0", [1, 128, 32, 32],
"FP32"))
inputs[0].set_data_from_numpy(noisy_activation.detach().cpu().numpy(),
binary_data=False)
outputs.append(httpclient.InferRequestedOutput("OUTPUT__0",
binary_data=False))
outputs.append(httpclient.InferRequestedOutput("OUTPUT__1",
binary_data=False))
results = triton_client.infer(
    model_name,
    inputs,
    outputs=outputs,
    #query_params=query_params,
    headers=None,
    request_compression_algorithm=None,
```

```
    response_compression_algorithm=None)
#print(results.get_response())
statistics =
triton_client.get_inference_statistics(model_name=model_name,
headers=None)
print(statistics)
if len(statistics["model_stats"]) != 1:
    print("FAILED: Inference Statistics")
    sys.exit(1)

loc_numpy = results.as_numpy("OUTPUT__0")
pred_numpy = results.as_numpy("OUTPUT__1")
######################################################################
###################

# postprocess output
noisy_pred = (loc_numpy, pred_numpy)
noisy_outputs = postprocess_outputs(
    noisy_pred, [[input_image_width, input_image_height]], priors,
THRESHOLD * 0.5
)
# get reconstruction of the noisy activation
noisy_reconstruction = decoder_function(noisy_activation)
noisy_reconstruction = noisy_reconstruction.detach().cpu().numpy()[0]
noisy_reconstruction = unpreprocess_output(
    noisy_reconstruction, (input_image_width, input_image_height), True
).astype(np.uint8)
# draw rectangles
for (x1, y1, x2, y2, s) in noisy_outputs[0]:
    x1, y1 = int(x1), int(y1)
    x2, y2 = int(x2), int(y2)
    cv2.rectangle(noisy_reconstruction, (x1, y1), (x2, y2), (0, 0, 255),
4)
```

추론 정확도 비교

이 검증에서는 일련의 원시 이미지를 사용하여 이미지 검색 사용 사례에 대한 추론을 수행했습니다. 그런 다음 추론 전에 Protopia 난독 처리를 추가하여 동일한 이미지 집합에서 동일한 추론 작업을 수행했습니다. 우리는 프로토피아 난독 처리 구성 요소에 대해 서로 다른 알파 값을 사용하여 작업을 반복했습니다. Protopia 난독 처리 컨텍스트에서 알파 값은 더 높은 난독 처리 수준을 나타내는 더 높은 알파 값으로 적용되는 난독 처리 양을 나타냅니다. 그런 다음 이 서로 다른 실행 간에 추론 정확도를 비교합니다.

다음 두 표에서는 사용 사례에 대한 자세한 내용과 결과에 대해 설명합니다.

Protopia는 고객과 직접 협력하여 특정 사용 사례에 적합한 알파 값을 결정합니다.

구성 요소	세부 정보
모델	FaceBoXes(PyTorch) -
데이터 세트	FDDB 데이터 세트

프로토피아 난독화	알파	정확성
아니요	해당 없음	0.9337148153739079
예	0.05	0.902876627325002
예	0.1	0.9024301009661478
예	0.2	0.9081836283186224
예	0.4	0.9073066107482036
예	0.6	0.8847816568680239
예	0.8	0.8841195749171925
예	0.9	0.8455427675252052
예	0.95	0.8455427675252052

난독화 속도

이 검증을 위해 Protopia 난독 처리를 1920 x 1080 픽셀 이미지에 5회 적용하고 난독 처리 단계가 매번 완료되는 데 걸리는 시간을 측정했습니다.

난독 처리를 적용하기 위해 단일 NVIDIA V100 GPU에서 실행되는 PyTorch를 사용했고 실행 간에 GPU 캐시를 지웠습니다. 난독화 단계는 5회 실행에서 각각 5.47ms, 5.27ms, 4.54ms, 5.24ms, 4.84ms를 완료하는 데 각각 걸렸습니다. 평균 속도는 5.072ms였습니다.

결론

데이터는 세 가지 상태, 즉 유휴 상태, 전송 중 상태 및 계산 중에 있습니다. AI 추론 서비스의 중요한 부분은 전체 프로세스 동안 위협으로부터 데이터를 보호하는 것이 되어야 합니다. 추론 도중에 데이터를 보호하는 것은 매우 중요합니다. 이 프로세스에서는 외부 고객과 추론 서비스를 제공하는 회사 모두에 대한 비공개 정보를 표시할 수 있기 때문입니다. Protopia AI는 오늘날의 시장에서 기밀 AI 추론을 위한 비간섭 소프트웨어 전용 솔루션입니다. Protopia를 통해 AI는 현재 AI/ML 작업을 수행하는 데 필수적인 데이터 레코드에 변환된 정보만 제공합니다. 이 확률적 변환은 마스킹의 형태가 아니며 큐레이션 노이즈를 사용하여 데이터의 표현을 수학적으로 변경하는 것을 기반으로 합니다.

ONTAP 기능을 갖춘 NetApp 스토리지 시스템은 로컬 SSD 스토리지와 동일하거나 더 우수한 성능을 제공하며 NetApp DataOps Toolkit과 함께 데이터 과학자, 데이터 엔지니어, AI/ML 개발자 및 비즈니스 또는 엔터프라이즈 IT 의사 결정자에게 다음과 같은 이점을 제공합니다.

- AI 시스템, 분석 및 기타 중요한 비즈니스 시스템 간에 데이터를 손쉽게 공유 이러한 데이터 공유는 인프라 오버헤드를 줄이고 성능을 향상하며 기업 전체에서 데이터 관리를 간소화합니다.

- 컴퓨팅과 스토리지를 독립적으로 확장하므로 비용을 최소화하고 리소스 사용량을 높일 수 있습니다.
- 즉각적이고 공간 효율적인 사용자 작업 공간, 통합 버전 제어 및 자동화된 구축을 위해 통합 Snapshot 복사본과 클론을 사용하여 개발 및 구축 워크플로우를 간소화했습니다.
- 재해 복구, 비즈니스 연속성 및 규정 요구사항을 충족하는 엔터프라이즈급 데이터 보호 및 데이터 거버넌스
- Jupyter 노트북에 있는 NetApp DataOps Toolkit에서 데이터 과학자 작업 공간의 Snapshot 복사본을 신속하게 만들어 백업 및 추적 기능을 제공합니다.

NetApp 및 Protopia 솔루션은 엔터프라이즈급 AI 추론 구축에 이상적인 유연한 스케일아웃 아키텍처를 제공합니다. 이 솔루션은 사내 및 하이브리드 클라우드 구축 모두에서 책임 있는 AI 사례를 충족할 수 있는 기밀 AI 추론 요구사항을 충족하는 중요한 정보에 대해 데이터 보호를 지원합니다.

추가 정보 및 승인 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 문서 및/또는 웹 사이트를 참조하십시오.

- NetApp ONTAP 데이터 관리 소프트웨어 - ONTAP 정보 라이브러리

  http://mysupport.netapp.com/documentation/productlibrary/index.html?productID=62286

- 컨테이너용 NetApp 영구 스토리지 - NetApp Trident

  "https://netapp.io/persistent-storage-provisioner-for-kubernetes/"

- NetApp DataOps 툴킷

  "https://github.com/NetApp/netapp-dataops-toolkit"

- 컨테이너용 NetApp 영구 스토리지 - NetApp Astra Trident

  "https://netapp.io/persistent-storage-provisioner-for-kubernetes/"

- 토폴로지 AI - 기밀 추론

  "https://protopia.ai/blog/protopia-ai-takes-on-the-missing-link-in-ai-privacy-confidential-inference/"

- NetApp BlueXP 복사 및 동기화

  "https://docs.netapp.com/us-en/occm/concept_cloud_sync.html#how-cloud-sync-works"

- NVIDIA Triton Inference Server를 참조하십시오

  "https://developer.nvidia.com/nvidia-triton-inference-server"

- NVIDIA Triton Inference Server 설명서

  "https://docs.nvidia.com/deeplearning/triton-inference-server/index.html"

- PyTorch의 FaceBoxes

  "https://github.com/zisianw/FaceBoxes.PyTorch"

감사의 말

- Mark Cates, 수석 제품 관리자, NetApp
- NetApp 기술 마케팅 엔지니어 Sufian Ahmad
- Hadi Esmaeilzadeh, 최고 기술 책임자 및 Protopia AI 교수

NetApp AI를 통한 감정 분석

TR-4910: NetApp AI를 통한 고객 커뮤니케이션의 감정 분석

Rick Huang, Sathish Thyagarajan, David Arnette, NetApp Diego Sosa-Coba, SFL Scientific

이 기술 보고서에서는 이전 학습 및 대화형 AI를 사용하는 NVIDIA 소프트웨어 프레임워크와 NetApp 데이터 관리 기술을 함께 사용하여 엔터프라이즈 수준의 글로벌 지원 센터에서 NetApp 데이터 관리 기술을 수행하는 고객에 대한 감정 분석을 수행할 수 있는 설계 지침을 제공합니다. 이 솔루션은 채팅 로그, 이메일 및 기타 텍스트 또는 오디오 통신을 나타내는 녹음된 음성 또는 텍스트 파일을 통해 고객 통찰력을 얻고자 하는 모든 산업에 적용됩니다. NetApp은 NetApp 클라우드 연결 All-Flash 스토리지를 통해 GPU 가속 컴퓨팅 클러스터에서 자동 음성 인식, 실시간 감정 분석, 딥 러닝 자연어 처리 모델 재교육 기능을 시연하기 위해 엔드 투 엔드 파이프라인을 구축했습니다. 방대한 최신 언어 모델을 훈련 및 최적화하여 글로벌 지원 센터와 신속하게 추론을 수행하여 탁월한 고객 경험과 객관적이고 장기적인 직원 성과 평가를 생성할 수 있습니다.

정서 분석은 자연어 처리(NLP) 내 연구 분야로서 텍스트에서 긍정적, 부정적 또는 중립적 감정을 도출합니다. 점점 더 많은 사람들이 대화하는 AI 시스템은 거의 세계적인 수준의 통합으로 부상했습니다. 감정 분석은 지원 센터 직원의 통화 성과를 확인하고 적절한 자동 챗봇 응답을 제공하는 등 다양한 활용 사례를 통해 분기별 수익 통화 시 기업 담당자와 대상 간의 상호 작용을 기반으로 회사의 주식 가격을 예측해 볼 수 있습니다. 또한, 감정 분석을 사용하여 브랜드가 제공하는 제품, 서비스 또는 지원에 대한 고객의 관점을 결정할 수 있습니다.

이 엔드 투 엔드 솔루션은 NLP 모델을 사용하여 지원 센터 분석 프레임워크를 지원하는 고수준 정서 분석을 수행합니다. 오디오 녹음은 서면 텍스트로 처리되며 대화의 각 문장에서 감정은 추출됩니다. 대시보드로 집계된 결과는 역사적, 실시간으로 대화 감정을 분석하기 위해 만들 수 있습니다. 이 솔루션은 유사한 데이터 양식 및 출력 요구가 있는 다른 솔루션으로 일반화할 수 있습니다. 적절한 데이터를 사용하여 다른 사용 사례를 수행할 수 있습니다. 예를 들어 동일한 종단간 파이프라인을 사용하여 기업 수익 통화를 분석하여 감정을 분석할 수 있습니다. 또한 파이프라인의 유연한 특성 때문에 주제 모델링 및 NER(명명된 엔티티 인식)과 같은 다른 형태의 NLP 분석이 가능합니다.

이러한 AI 구현은 NVIDIA Riva, NVIDIA TAO 툴킷 및 NetApp DataOps 툴킷을 함께 사용하여 가능했습니다. NVIDIA의 툴은 사전 구축된 모델 및 파이프라인을 사용하여 고성능 AI 솔루션을 신속하게 배포하는 데 사용됩니다. NetApp DataOps 툴킷은 다양한 데이터 관리 작업을 단순화하여 개발 속도를 높여줍니다.

고객 가치

기업은 감성 분석을 위해 텍스트, 오디오 및 비디오 대화를 위한 직원 평가 및 고객 반응 도구를 통해 가치를 확인합니다. 관리자는 대시보드에 표시되는 정보를 활용하여 대화 양쪽을 기준으로 직원 및 고객 만족도를 평가할 수 있습니다.

또한 NetApp DataOps 툴킷은 고객 인프라 내에서 데이터의 버전 관리 및 할당을 관리합니다. 따라서 복잡하지 않은 데이터 스토리지 비용을 발생시키지 않고 대시보드 내에 제공되는 분석 내용이 자주 업데이트됩니다.

사용 사례

이러한 지원 센터에서 처리하는 통화 수 때문에 수동으로 수행할 경우 통화 성능 평가에 상당한 시간이 걸릴 수 있습니다. 단어 개수 계산 및 기타 방법과 같은 기존 방법은 일부 자동화를 달성할 수 있지만 이러한 방법은 동적 언어의 보다 미묘한 측면과 의미 컨텍스트를 캡처하지 않습니다. AI 모델링 기법을 사용하면 이러한 고급 분석 중 일부를 자동화된 방식으로 수행할 수 있습니다. 또한, NVIDIA, AWS, Google 등에서 제공하는 최신 기술 및 사전 교육 모델링 툴을 사용하여 복잡한 모델을 가진 엔드 투 엔드 파이프라인을 상대적으로 쉽게 구축 및 사용자 지정할 수 있습니다.

지원 센터 정서 분석을 위한 엔드 투 엔드 파이프라인은 직원들이 통화자와 대화하면서 실시간으로 오디오 파일을 수집합니다. 그런 다음 이러한 오디오 파일을 텍스트 형식으로 변환하는 텍스트 음성 변환 구성 요소에서 사용할 수 있도록 처리됩니다. 대화의 각 문구에는 정서(긍정적, 부정적 또는 중립적)를 나타내는 레이블이 표시됩니다.

감정 분석은 통화 성과를 평가하기 위한 대화의 필수 요소를 제공할 수 있습니다. 이러한 감정은 직원과 통화자 간의 상호 작용에 대한 심도 있는 수준을 더하고 있습니다. AI 지원 정서 대시보드는 관리자가 대화 내에서 감정을 실시간으로 추적할 수 있도록 하며 직원의 과거 통화 내역을 후향적 분석합니다.

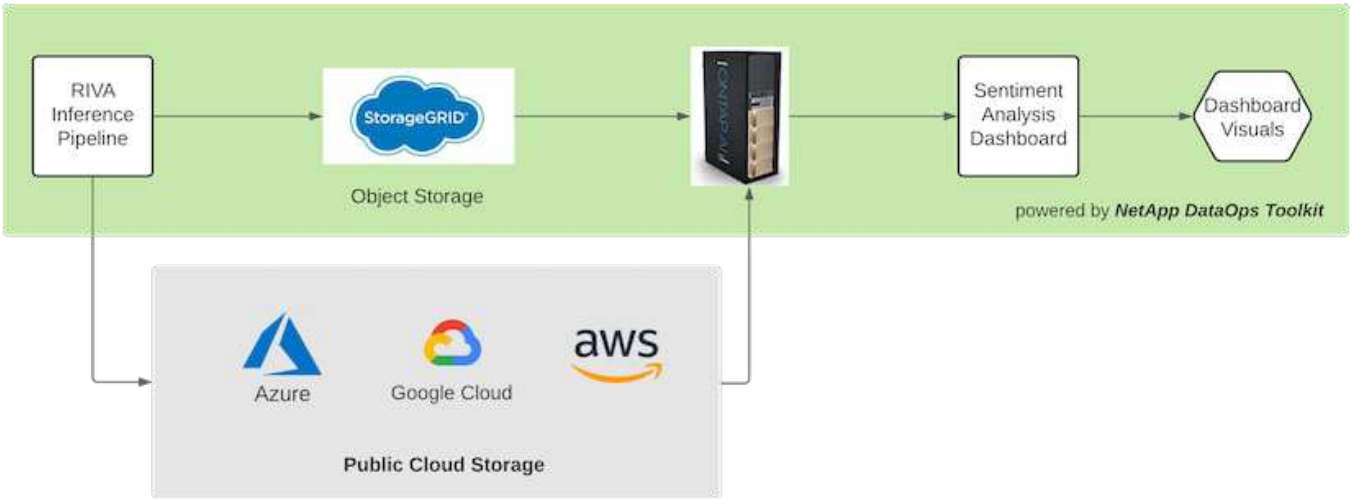
사전 구축된 툴을 강력한 방법으로 결합하여 이 문제를 해결하기 위한 엔드 투 엔드 AI 파이프라인을 빠르게 구축할 수 있습니다. 이 경우 NVIDIA Riva 라이브러리를 사용하여 두 개의 직렬 내 작업(오디오 전사 및 정서 분석)을 수행할 수 있습니다. 첫 번째는 감시 방식 학습 신호 처리 알고리즘이고 두 번째는 감시 방식 학습 NLP 분류 알고리즘입니다. NVIDIA TAO 툴킷을 사용하여 비즈니스 관련 데이터와 관련된 모든 사용 사례에 맞게 즉시 사용 가능한 알고리즘을 미세 조정할 수 있습니다. 따라서 훨씬 적은 비용과 리소스로 더 정확하고 강력한 솔루션을 구축할 수 있습니다. 고객은 을 통합할 수 있습니다 "NVIDIA Maxine" 지원 센터 설계의 GPU 가속 비디오 회의 응용 프로그램용 프레임워크

다음 사용 사례가 이 솔루션의 핵심입니다. 두 사용 사례 모두 모델 세부 조정에는 TAO Toolkit을 사용하고 모델 배포에는 Riva를 사용합니다.

- 텍스트 음성 변환
- 정서 분석

직원과 고객 간의 지원 센터 상호 작용을 분석하기 위해 오디오 통화 형식으로 각 고객 대화를 파이프라인을 통해 실행하여 문장별 감정을 추출할 수 있습니다. 그런 다음, 그 감정은 인간이 검증하여 정서를 정당화하거나 필요에 따라 조정할 수 있다. 그런 다음 레이블이 지정된 데이터가 미세 조정 단계로 전달되어 감정의 예측을 개선합니다. 레이블이 지정된 감정 데이터가 이미 있으면 모델 세부 조정을 신속하게 처리할 수 있습니다. 어느 경우든 파이프라인은 오디오를 수집하여 문장을 분류해야 하는 다른 솔루션에 일반화할 수 있습니다.

AI 정서 출력은 외부 클라우드 데이터베이스 또는 회사에서 관리하는 스토리지 시스템에 업로드됩니다. 감성 출력은 관리자의 정서 분석을 표시하는 대시보드 내에서 사용할 수 있도록 이 큰 데이터베이스에서 로컬 스토리지로 전송됩니다. 대시보드의 주요 기능은 고객 서비스 직원과 실시간으로 상호 작용하는 것입니다. 관리자는 통화 중에 각 문장의 감정을 실시간으로 업데이트하고 직원의 과거 성과 또는 고객 반응에 대한 과거의 평가를 통해 직원에 대한 피드백을 평가 및 제공할 수 있습니다.



를 클릭합니다 "NetApp DataOps 툴킷" 는 Riva 추론 파이프라인에서 정서 레이블을 생성한 후에도 데이터 스토리지 시스템을 계속 관리할 수 있습니다. 이러한 AI 결과는 NetApp DataOps 툴킷에서 관리하는 데이터 스토리지 시스템에 업로드할 수 있습니다. 데이터 스토리지 시스템은 수백 개의 인서트를 관리할 수 있어야 하며 매 분마다 선택해야 합니다. 로컬 디바이스 스토리지 시스템은 더 큰 데이터 스토리지를 실시간으로 쿼리하여 압축을 풉니다. 또한 대규모 데이터 스토리지 인스턴스를 쿼리하여 기간별 데이터를 쿼리하면 대시보드 환경을 더욱 향상시킬 수 있습니다. NetApp DataOps 툴킷은 데이터를 빠르게 복제하고 이를 사용하는 모든 대시보드에 배포하여 이러한 두 용도 모두를 촉진합니다.

대상

이 솔루션의 대상 고객은 다음과 같은 그룹을 포함합니다.

- 직원 관리자
- 데이터 엔지니어/데이터 과학자
- IT 관리자(사내, 클라우드 또는 하이브리드)

대화 전반에 걸쳐 감정을 추적하는 것은 직원의 성과를 평가할 수 있는 귀중한 도구입니다. 관리자는 AI 대시보드를 사용하여 직원과 발신자가 어떻게 자신의 감정을 실시간으로 변화시킵니다. 이를 통해 실시간 평가와 안내 세션을 진행할 수 있습니다. 또한, 기업은 음성 대화, 텍스트 챗봇 및 화상 회의에 참여하는 고객으로부터 중요한 고객 통찰력을 얻을 수 있습니다. 이러한 고객 분석은 최신 최첨단 AI 모델 및 워크플로우와 함께 규모에 따른 다중 모드 처리 기능을 사용합니다.

데이터 측면에서 많은 수의 오디오 파일이 지원 센터에 의해 매일 처리됩니다. NetApp DataOps 툴킷은 모델 및 정서 분석 대시보드의 주기적인 미세 조정을 위해 이 데이터 처리 작업을 용이하게 합니다.

IT 관리자는 NetApp DataOps 툴킷을 사용하여 구축 환경과 운영 환경 간에 데이터를 빠르게 이동할 수 있습니다. 또한, 실시간 추론을 위해 NVIDIA 환경과 서버를 관리하고 분산해야 합니다.

있습니다

이 지원 센터 솔루션의 아키텍처는 NVIDIA의 사전 구축된 툴과 NetApp DataOps 툴킷을 중심으로 돌아가고 있습니다. NVIDIA의 도구는 사전 구축된 모델 및 파이프라인을 사용하여 고성능 AI 솔루션을 신속하게 배포하는 데 사용됩니다. NetApp DataOps 툴킷은 다양한 데이터 관리 작업을 단순화하여 개발 속도를 높여줍니다.

솔루션 기술

"NVIDIA Riva" GPU에 실시간 성능을 제공하는 멀티모달 대화형 AI 애플리케이션을 구축하기 위한 GPU 가속 SDK NVIDIA Train, 조정 및 최적화(TAO) 툴킷은 교육을 가속화하고 매우 정확하고 성능 높은 도메인 특정 AI 모델을 빠르게 생성할 수 있는 더 빠르고 쉬운 방법을 제공합니다.

NetApp DataOps Toolkit은 개발자, 데이터 과학자, DevOps 엔지니어 및 데이터 엔지니어가 다양한 데이터 관리 작업을 수행할 수 있도록 지원하는 Python 라이브러리입니다. 여기에는 새로운 데이터 볼륨 또는 JupyterLab 작업 공간의 거의 즉각적인 프로비저닝, 데이터 볼륨 또는 JupyterLab 작업 공간의 거의 즉각적인 클론 복제, 추적 및 베이스라인 기능을 위한 데이터 볼륨 또는 JupyterLab 작업 공간의 거의 즉각적인 스냅샷 생성이 포함됩니다.

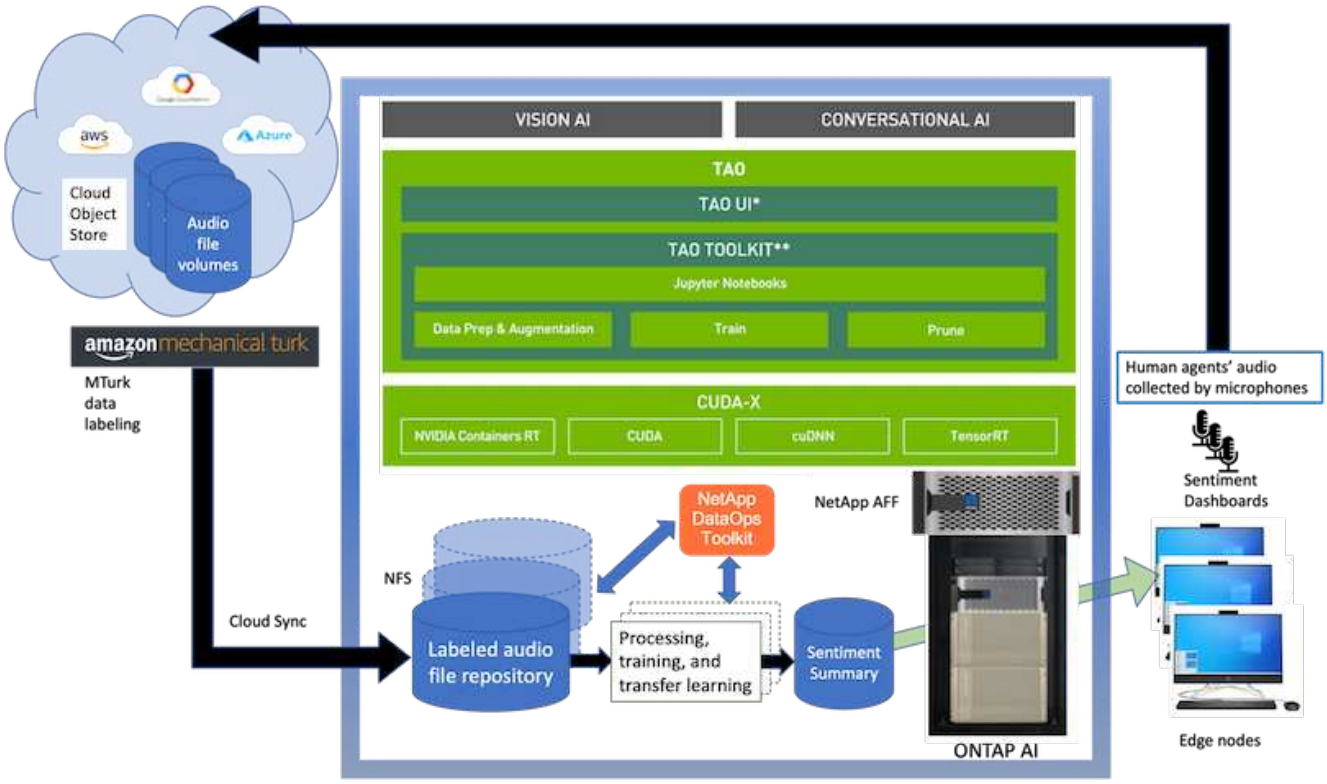
아키텍처 다이어그램

다음 다이어그램에서는 솔루션 아키텍처를 보여 줍니다. 클라우드, 코어, 에지의 세 가지 주요 환경 범주가 있습니다. 각 범주는 지리적으로 분산될 수 있습니다. 예를 들어, 클라우드에는 여러 지역의 버킷에 오디오 파일이 있는 오브젝트 저장소가 포함되어 있는 반면, 코어에는 고속 네트워크 또는 NetApp BlueXP 복사본 및 동기화 를 통해 연결된 데이터 센터가 포함될 수 있습니다. 에지 노드는 개별 상담원의 일상 업무 플랫폼을 나타내며, 대화형 대시보드 도구 및 마이크를 사용하여 감정을 시각화하고 고객과의 대화에서 오디오 데이터를 수집할 수 있습니다.

GPU 가속 데이터 센터에서 기업은 NVIDIA를 사용할 수 있습니다 "리바" 대화형 AI 애플리케이션을 구축하기 위한 프레임워크로, 이 애플리케이션은 에서 사용할 수 있습니다 "Tao 툴킷" TRANSFER L-Learning 기술을 사용하여 모델 마무리 및 재교육을 위한 연결 이러한 컴퓨팅 애플리케이션과 워크플로우가 에서 제공됩니다 "NetApp DataOps 툴킷 "ONTAP이 제공하는 최고의 데이터 관리 기능을 활용할 수 있습니다. 이 툴킷은 기업 데이터 팀이 추적 가능성, 버전 관리, A/B 테스트를 위해 스냅샷 및 클론을 통해 관련 정형 및 비정형 데이터와 함께 모델을 신속하게 프로토타입화할 수

있도록 해 줍니다. 따라서 보안, 거버넌스, 및 규정 준수: 섹션을 참조하십시오 "스토리지 설계" 를 참조하십시오.

이 솔루션은 오디오 파일 처리, NLP 모델 교육, 전송 학습 및 데이터 관리 세부 정보 단계를 보여 줍니다. 결과적으로 전체 파이프라인은 인적 지원 상담원의 대시보드에 실시간으로 표시되는 정서 요약을 생성합니다.



하드웨어 요구 사항

다음 표에는 솔루션을 구현하는 데 필요한 하드웨어 구성요소가 나와 있습니다. 이 솔루션을 구체적으로 구축하는 데 사용되는 하드웨어 구성요소는 고객 요구사항에 따라 다를 수 있습니다.

응답 지연 시간 테스트	시간(밀리초)
데이터 처리	10
추론	10

이러한 응답 시간 테스트는 560건의 대화에서 50,000개 이상의 오디오 파일로 실행되었습니다. 각 오디오 파일의 크기는 MP3로 최대 100KB, WAV로 변환될 경우 최대 1MB였습니다. 데이터 처리 단계에서는 MP3를 WAV 파일로 변환합니다. 추론 단계에서는 오디오 파일을 텍스트로 변환하고 텍스트에서 감정을 추출합니다. 이러한 단계는 모두 서로 독립적이며 병렬화를 통해 프로세스 속도를 높일 수 있습니다.

매장 간의 데이터 전송 지연 시간을 고려하여 관리자는 문장의 끝 후 1초 이내에 실시간 감정 분석에 대한 업데이트를 볼 수 있어야 합니다.

NVIDIA Riva 하드웨어

하드웨어	요구 사항
OS	Linux x86_64

하드웨어	요구 사항
GPU 메모리(ASR)	스트리밍 모델: ~5600 MB 비스트리밍 모델: ~3100 MB
GPU 메모리(NLP)	BERT 모델당 최대 500MB

NVIDIA TAO 툴킷 하드웨어

하드웨어	요구 사항
시스템 RAM	32GB
GPU RAM	32GB
CPU	8코어
GPU	NVIDIA(A100, V100 및 RTX 30x0)
SSD를 지원합니다	100GB

플래시 스토리지 시스템

NetApp ONTAP 9

NetApp의 최신 세대 스토리지 관리 소프트웨어인 ONTAP 9.9를 통해 기업은 인프라를 현대화하고 클라우드 지원 데이터 센터로 전환할 수 있습니다. ONTAP는 업계 최고 수준의 데이터 관리 기능을 활용하여 데이터가 상주하는 위치와 상관없이 단일 툴셋으로 데이터를 관리하고 보호할 수 있습니다. 필요에 따라 에지, 코어, 클라우드 등 어느 위치로도 데이터를 자유롭게 이동할 수 있습니다. ONTAP 9.9에는 데이터 관리를 단순화하고, 중요 데이터를 더 빨리 처리하고, 보호하며, 하이브리드 클라우드 아키텍처 전체에서 차세대 인프라 기능을 지원하는 다양한 기능이 포함되어 있습니다.

NetApp BlueXP 복사 및 동기화

"BlueXP 복사 및 동기화" 은(는) 빠르고 안전한 데이터 동기화를 제공하는 NetApp 서비스로, 사용자는 온프레미스 NFS 또는 SMB 파일 공유 간에 파일을 다음 타겟으로 전송할 수 있습니다.

- NetApp StorageGRID를 참조하십시오
- NetApp ONTAP S3
- NetApp Cloud Volumes Service를 참조하십시오
- Azure NetApp Files
- Amazon Simple Storage Service(Amazon S3)
- Amazon Elastic File System(Amazon EFS)
- Azure Blob
- Google 클라우드 스토리지
- IBM 클라우드 오브젝트 스토리지

BlueXP 복사 및 동기화는 필요한 파일을 빠르고 안전하게 이동합니다. 데이터가 전송되면 소스와 타겟 모두에서 사용할 수 있습니다. BlueXP 복사 및 동기화는 미리 정의된 일정에 따라 데이터를 지속적으로 동기화하므로 변경된 부분만 이동하므로 데이터 복제에 소비되는 시간과 비용이 최소화됩니다. BlueXP Copy and Sync는 간편하게 설정하고 사용할 수 있는 서비스형 소프트웨어(SaaS) 툴입니다. BlueXP Copy 및 Sync에 의해 트리거되는 데이터 전송은 데이터 브로커에 의해 수행됩니다. AWS, Azure, Google Cloud Platform 또는 사내에 BlueXP Copy 및 Sync 데이터

브로커를 배포할 수 있습니다.

NetApp StorageGRID를 참조하십시오

StorageGRID 소프트웨어 정의 오브젝트 스토리지 제품군은 퍼블릭, 프라이빗, 하이브리드 멀티 클라우드 환경에서 다양한 사용 사례를 원활하게 지원합니다. 업계 최고 수준의 혁신적인 NetApp StorageGRID은 오랫동안 자동 라이프사이클 관리를 포함하여 다목적 사용을 위해 비정형 데이터를 저장, 보안, 보호 및 보존합니다. 자세한 내용은 를 참조하십시오 "NetApp StorageGRID를 참조하십시오" 사이트.

소프트웨어 요구 사항

다음 표에는 이 솔루션을 구축하는 데 필요한 소프트웨어 구성요소가 나와 있습니다. 이 솔루션을 구체적으로 구축하는 데 사용되는 소프트웨어 구성요소는 고객 요구사항에 따라 다를 수 있습니다.

호스트 시스템	요구 사항
Riva(이전 명칭 JARVIS)	1.4.0
Tao 툴킷(이전 명칭: 학습 툴킷)	3.0
ONTAP	9.9.1
DGX OS	5.1
생년월일	2.0.0

NVIDIA Riva 소프트웨어

소프트웨어	요구 사항
Docker 를 참조하십시오	>19.02(NVIDIA-Docker 설치 시) >=19.03(DGX를 사용하지 않는 경우
NVIDIA 드라이버	465.19.01 + 418.40+, 440.33+, 450.51+, 460.27+(데이터 센터 GPU용
컨테이너 OS	Ubuntu 20.04
CUDA	11.3.0
큐블라스	11.5.1.101
큐드NN	8.2.0.41
NCCL	2.9.6
TensorRT	7.2.3.4
Triton Inference Server를 참조하십시오	2.9.0

NVIDIA TAO 툴킷 소프트웨어

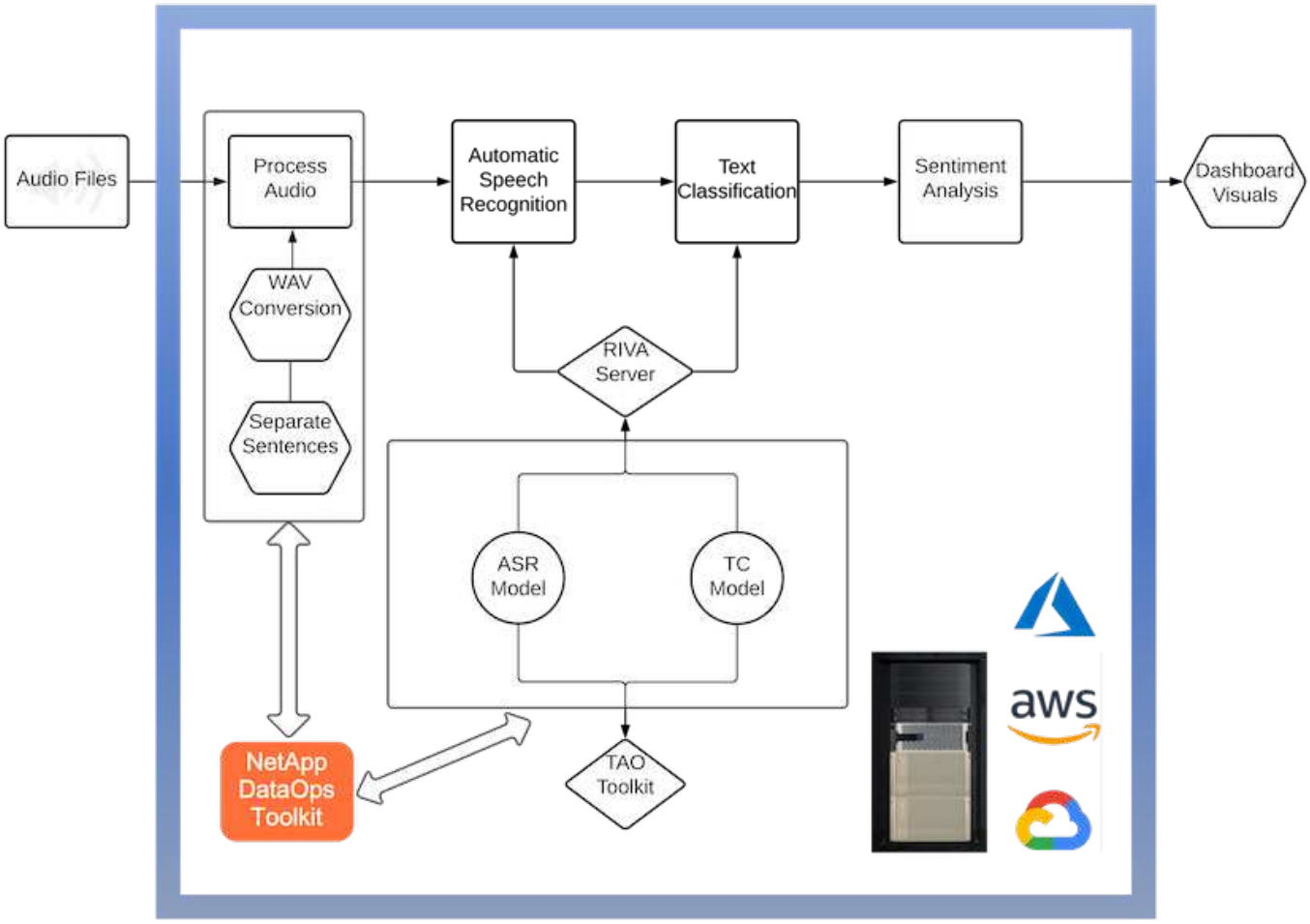
소프트웨어	요구 사항
Ubuntu 18.04 LTS	18.04
파이썬	>= 3.6.9
Docker-CE 를 참조하십시오	>19.03.5

소프트웨어	요구 사항
Docker-API를 지원합니다	1.40
NVIDIA - 컨테이너 - 툴킷	>1.3.0-1
nvidia-container-runtime	3.4.0-1
nVidia-docker2	2.5.0-1
nVidia - 드라이버	>455
Python-PIP	>21.06
nVidia-pyindex	최신 버전

사용 사례 세부 정보

이 솔루션은 다음과 같은 사용 사례에 적용됩니다.

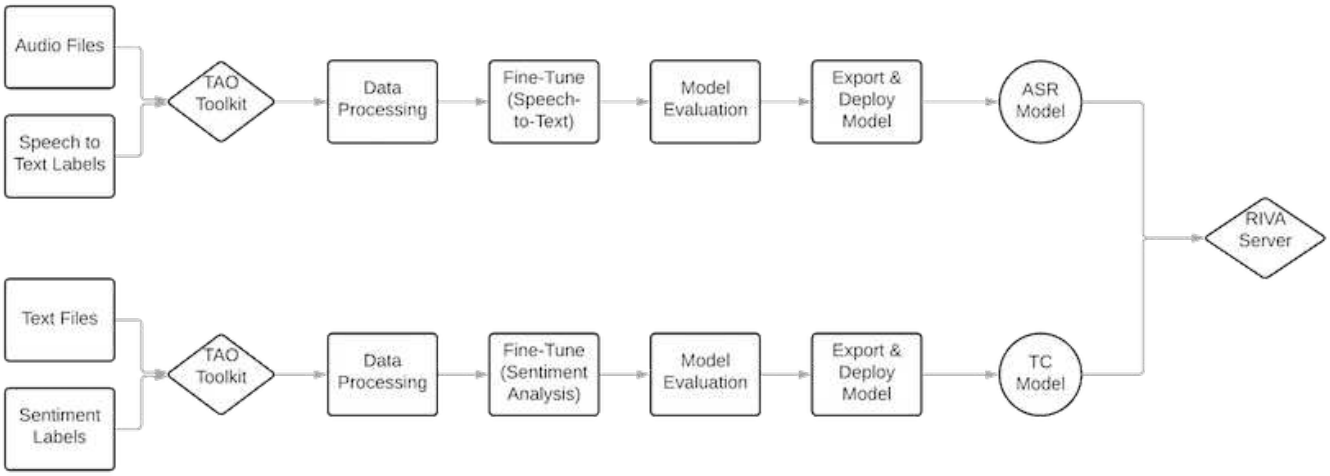
- 텍스트 음성 변환
- 정서 분석



텍스트 음성 변환 사용 사례는 지원 센터의 오디오 파일을 수집하여 시작합니다. 그런 다음 이 오디오는 Riva가 요구하는 구조에 맞게 처리됩니다. 오디오 파일이 아직 분석 단위로 분할되지 않은 경우 Riva에 오디오를 전달하기 전에 이 작업을 수행해야 합니다. 오디오 파일이 처리되면 Riva 서버에 API 호출로 전달됩니다. 서버는 호스팅 중인 여러 모델 중 하나를

사용하고 응답을 반환합니다. 이 텍스트 음성 변환(자동 음성 인식의 일부)은 오디오의 텍스트 표현을 반환합니다. 여기서 파이프라인은 감정 분석 부분으로 전환됩니다.

감정 분석의 경우 자동 음성 인식의 텍스트 출력은 텍스트 분류에 대한 입력 역할을 합니다. 텍스트 분류는 텍스트를 다양한 범주로 분류하는 NVIDIA 구성 요소입니다. 지원 센터 대화의 경우 긍정적 범주에서 부정적 범주에 이르기까지 다양합니다. 미세 조정 단계의 성공을 결정하기 위해 홀드아웃 세트를 사용하여 모델의 성능을 평가할 수 있습니다.



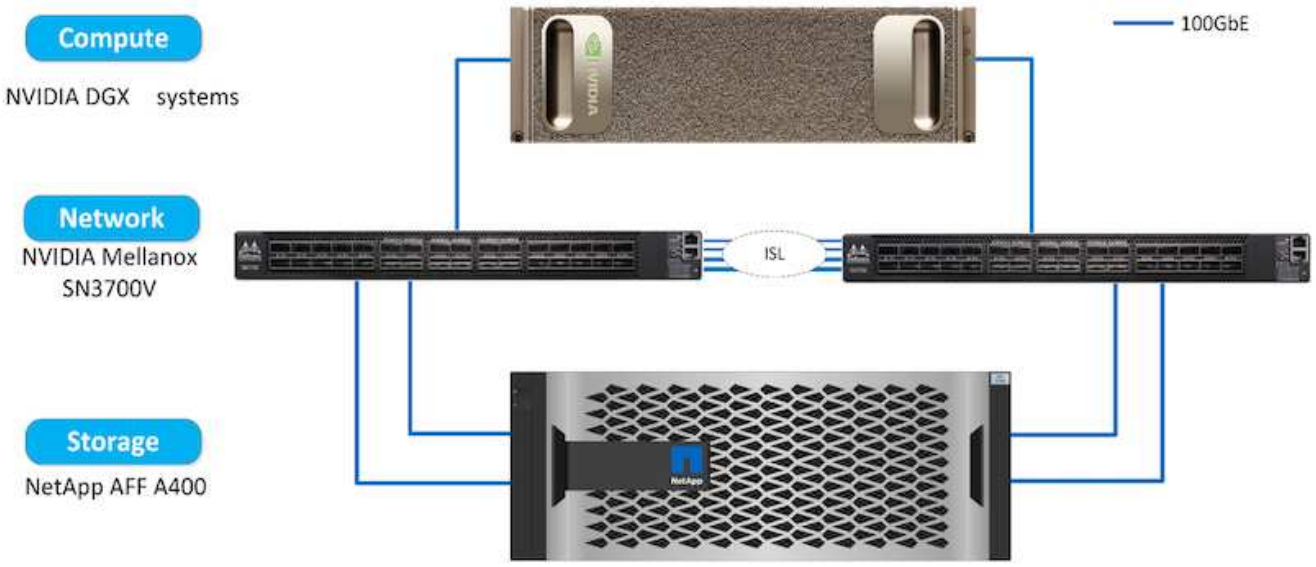
TAO 툴키트의 텍스트 음성 및 정서 분석에 비슷한 파이프라인이 사용됩니다. 주요 차이점은 모델의 미세 조정에 필요한 라벨 사용입니다. TAO 툴킷 파이프라인은 데이터 파일 처리부터 시작합니다. 그런 다음 미리 훈련된 모델(에서 제공)을 사용합니다 "NVIDIA NGC 카탈로그")는 지원 센터 데이터를 사용하여 미세 조정됩니다. 미세 조정된 모델은 해당 성능 메트릭을 기준으로 평가되며, 사전 훈련된 모델보다 성능 기준에 더 적합한 경우 Riva 서버에 배포됩니다.

설계 고려 사항

이 섹션에서는 이 솔루션의 다양한 구성 요소에 대한 설계 고려 사항에 대해 설명합니다.

네트워크 및 컴퓨팅 설계

데이터 보안 제한에 따라 모든 데이터는 고객의 인프라 또는 보안 환경 내에 있어야 합니다.



스토리지 설계

NetApp DataOps 툴킷은 스토리지 시스템 관리를 위한 1차 서비스 역할을 합니다. DataOps Toolkit은 개발자, 데이터 과학자, DevOps 엔지니어 및 데이터 엔지니어가 새로운 데이터 볼륨의 거의 즉각적인 프로비저닝 또는 JupyterLab 작업 공간, 데이터 볼륨의 거의 즉각적인 클론 복제 또는 JupyterLab 작업 공간과 같은 다양한 데이터 관리 작업을 간단하게 수행할 수 있는 Python 라이브러리입니다. 추적 기능 또는 베이스라인 기능을 위한 데이터 볼륨 또는 JupyterLab 작업 공간의 거의 즉각적인 스냅샷 기능을 제공합니다. 이 Python 라이브러리는 명령줄 유틸리티 또는 모든 Python 프로그램 또는 Jupyter Notebook로 가져올 수 있는 기능 라이브러리 중 하나로 작동할 수 있습니다.

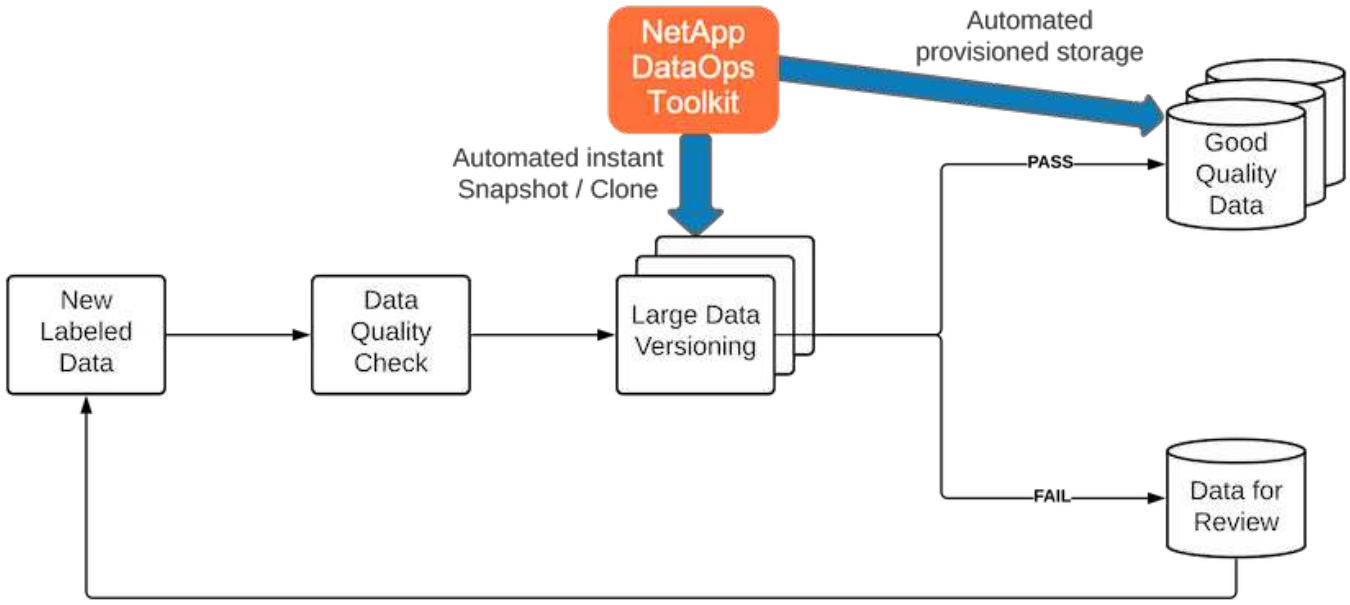
Riva 모범 사례

NVIDIA는 몇 가지 일반적인 기능을 제공합니다 "모범 데이터 사례" Riva 사용:

- * 가능한 경우 무손실 오디오 형식을 사용합니다. * MP3와 같은 손실 코덱을 사용하면 품질이 저하될 수 있습니다.
- * 교육 데이터를 보강합니다. * 오디오 교육 데이터에 배경 잡음을 추가하면 처음에는 정확도가 떨어되지만 견고성이 향상됩니다.
- * 스크레핑된 텍스트를 사용할 경우 어휘 크기를 제한합니다. * 많은 온라인 출처에는 오타 또는 부수적인 대명사 및 일반적이지 않은 단어가 포함되어 있습니다. 이러한 언어를 제거하면 언어 모델이 개선될 수 있습니다.
- * 가능한 경우 최소 16kHz의 샘플링 속도를 사용하십시오. * 그러나 리샘플링을 시도하지 마십시오. 리샘플링을 하면 오디오 품질이 저하됩니다.

이러한 모범 사례 외에도 고객은 파이프라인의 각 단계에 대해 정확한 레이블이 있는 대표적인 샘플 데이터 세트를 우선적으로 수집해야 합니다. 즉, 샘플 데이터 세트는 타겟 데이터 세트에 예시된 지정된 특성을 비율에 맞게 반영해야 합니다. 마찬가지로 데이터 세트의 주석 역시 데이터의 품질과 양을 모두 최대화하도록 정확도와 레이블 지정 속도를 조율할 책임이 있습니다. 예를 들어, 이 지원 센터 솔루션에는 오디오 파일, 텍스트 레이블 및 정서 레이블이 필요합니다. 이 솔루션의 순차적 특성은 파이프라인 시작 부분의 오류가 끝까지 전파된다는 것을 의미합니다 오디오 파일의 품질이 좋지 않으면 텍스트 사본과 정서 레이블도 함께 표시됩니다.

이 오류 전파는 이 데이터에 대한 교육을 받은 모델에도 비슷하게 적용됩니다. 감정의 예측이 100% 정확하지만 텍스트 음성 변환 모델이 제대로 작동하지 않는 경우, 최종 파이프라인은 초기 오디오-텍스트 사본으로 제한됩니다. 개발자는 각 모델의 성능을 개별적으로, 대규모 파이프라인의 구성 요소로 고려하는 것이 중요합니다. 이 경우 최종 목표는 감정을 정확하게 예측할 수 있는 파이프라인을 개발하는 것입니다. 따라서 파이프라인을 평가하는 전반적인 지표는 음성-텍스트 전사가 직접적으로 영향을 미치는 정서 정확도입니다.



NetApp DataOps 툴킷은 즉각적인 데이터 클론 복제 기술을 사용하여 데이터 품질 점검 파이프라인을 보완합니다. 레이블이 지정된 각 파일을 평가하고 기존의 레이블 파일과 비교해야 합니다. 이러한 품질 검사를 다양한 데이터 스토리지 시스템에 분산하면 이러한 검사가 빠르고 효율적으로 실행됩니다.

지원 센터 정서 분석 배포

솔루션 배포에는 다음 구성 요소가 포함됩니다.

1. NetApp DataOps 툴킷
2. NGC 구성
3. NVIDIA Riva 서버
4. NVIDIA TAO 툴킷
5. TAO 모델을 Riva로 내보냅니다

배포를 수행하려면 다음 단계를 수행하십시오.

NetApp DataOps 툴킷: 지원 센터 정서 분석

를 사용합니다 "NetApp DataOps 툴킷"에서 다음 단계를 완료합니다.

1. PIP 도구 키트를 설치합니다.

```
python3 -m pip install netapp-dataops-traditional
```

2. 데이터 관리를 구성합니다

```
netapp_dataops_cli.py config
```

NGC 구성: 지원 센터 정서 분석

를 눌러 설정합니다 "NGC"에서 다음 단계를 완료합니다.

1. NGC를 다운로드합니다.

```
wget -O ngccli_linux.zip
https://ngc.nvidia.com/downloads/ngccli_linux.zip && unzip -o
ngccli_linux.zip && chmod u+x ngc
```

2. 현재 디렉터리를 경로에 추가합니다.

```
echo "export PATH=\"\$PATH:$(pwd)\"" >> ~/.bash_profile && source
~/.bash_profile
```

3. 명령을 실행할 수 있도록 NGC CLI를 구성해야 합니다. 메시지가 나타나면 API 키를 포함하여 다음 명령을 입력합니다.

```
ngc config set
```

Linux 기반이 아닌 운영 체제는 을 참조하십시오 "여기".

NVIDIA Riva 서버: 지원 센터 정서 분석

를 눌러 설정합니다 "NVIDIA Riva"에서 다음 단계를 완료합니다.

1. NGC에서 Riva 파일을 다운로드합니다.

```
ngc registry resource download-version
nvidia/riva/riva_quickstart:1.4.0-beta
```

2. Riva 설정 초기화('Riva_init.sh')
3. Riva 서버('Riva_start.sh')를 시작합니다.
4. Riva client('Riva_start_client.sh')를 시작합니다.
5. Riva 클라이언트 내에서 오디오 처리 라이브러리("FFmpeg")

```
apt-get install ffmpeg
```

6. 를 시작합니다 "Jupyter를 선택합니다" 서버.
7. Riva Inference Pipeline 노트북을 실행합니다.

NVIDIA TAO Toolkit: 지원 센터 정서 분석

NVIDIA TAO 툴킷을 설정하려면 다음 단계를 수행하십시오.

1. 를 준비하고 활성화합니다 "가상 환경" TAO 툴킷을 참조하십시오.
2. 를 설치합니다 "필수 패키지".
3. 교육 및 미세 조정 중에 사용된 이미지를 수동으로 당깁니다.

```
docker pull nvcr.io/nvidia/tao/tao-toolkit-pyt:v3.21.08-py3
```

4. 를 시작합니다 "Jupyter를 선택합니다" 서버.
5. TAO 미세 조정 노트북을 실행합니다.

TAO 모델을 **Riva**로 내보내기**:** 지원 센터 정서 분석

사용합니다 "Riva의 Tao 툴킷 모델"에서 다음 단계를 완료합니다.

1. TAO 미세 조정 노트북에 모델을 저장합니다.
2. TAO 교육을 받은 모델을 Riva 모델 디렉토리에 복사합니다.
3. Riva 서버('Riva_start.sh')를 시작합니다.

구축 방해

다음은 자체 솔루션을 개발할 때 고려해야 할 몇 가지 사항입니다.

- NetApp DataOps 툴킷은 데이터 스토리지 시스템이 최적으로 실행되도록 하기 위해 먼저 설치됩니다.
- NVIDIA NGC는 이미지와 모델의 다운로드를 인증하기 때문에 다른 무엇보다도 먼저 설치해야 합니다.
- TAO 툴킷을 설치하기 전에 Riva를 설치해야 합니다. Riva 설치는 필요에 따라 Docker 데몬을 구성하여 이미지를 가져옵니다.
- 모델을 다운로드하려면 DGX 및 Docker에 인터넷 액세스 권한이 있어야 합니다.

검증 결과

이전 섹션에서 언급한 바와 같이, 두 개 이상의 기계 학습 모델이 순서대로 실행될 때마다 오류가 파이프라인 전체에 전파됩니다. 이 솔루션을 위해, 이 회사의 주식 리스크 수준을 측정하는 데 있어 가장 중요한 요소는 문장의 감정입니다. 파이프라인에 필수적인 스피치-텍스트 모델은 정서를 예측할 수 있는 전처리부 역할을 합니다. 진짜 중요한 것은 근거 있는 진실과 예측된 문장 사이의 감정의 차이이다. 이는 WER(Error Rate)의 프록시 역할을 합니다. 음성-텍스트 정확도는 중요하지만 WER은 최종 파이프라인 메트릭에 직접 사용되지 않습니다.

```
PIPELINE_SENTIMENT_METRIC = MEAN(DIFF(GT_sentiment, ASR_sentiment))
```

이러한 정서 메트릭은 각 문장의 F1 점수, 리콜 및 정밀도에 대해 계산할 수 있습니다. 그런 다음 결과를 집계하여 각 메트릭의 신뢰 간격과 함께 혼란 매트릭스 내에 표시할 수 있습니다.

전송 학습 기능을 사용하면 적은 데이터 요구사항, 교육 시간 및 비용으로 모델 성능을 향상할 수 있습니다. 또한 세부 조정된 모델을 기준 버전과 비교하여 전송 학습이 페어링되지 않고 성능을 향상시키도록 해야 합니다. 다시 말해, 세부 조정된 모델은 사전 교육 모델보다 지원 센터 데이터의 성능이 더 우수해야 합니다.

파이프라인 평가

테스트 케이스	세부 정보
테스트 번호	파이프라인 정서 지표
테스트 필수 구성 요소	음성-텍스트 및 정서 분석 모델을 위해 미세 조정된 모델
예상 결과	미세 조정된 모델의 정서 측정 기준은 원래 사전 교육 모델보다 성능이 뛰어납니다.

파이프라인 정서 지표

1. 기준 모델의 정서 메트릭을 계산합니다.
2. 미세 조정된 모델의 정서 메트릭을 계산합니다.
3. 이러한 메트릭 간의 차이를 계산합니다.
4. 모든 문장에 걸친 평균 차이입니다.

비디오 및 데모

감정 분석 파이프라인이 포함된 두 개의 노트북이 있습니다. ""“Support-Center-Model-Transfer-Learning-and-fine-Tuning.ipynb”" 및 ""“지원 센터-정서-분석-파이프라인.iynb”". 이 노트북은 함께 지원 센터 데이터를 수집하고 사용자 데이터에 맞게 조정된 최첨단 딥 러닝 모델을 사용하여 각 문장에서 감정을 추출하는 파이프라인을 개발하는 방법을 보여줍니다.

지원 센터 **-** 정서 분석 파이프라인**. ipynb**

이 노트북에는 오디오 인제스트, 텍스트로 변환, 외부 대시보드에서 사용할 정서 추출용 추론 Riva 파이프라인이 포함되어 있습니다. 아직 완료되지 않은 경우 데이터 세트가 자동으로 다운로드되고 처리됩니다. 전자 필기장의 첫 번째 섹션은 오디오 파일을 텍스트로 변환하는 작업을 처리하는 텍스트 음성 변환 섹션입니다. 그 다음에는 각 텍스트 문장에 대한 감정을 추출하고 제안된 대시보드와 유사한 형식으로 결과를 표시하는 감정 분석 섹션이 이어집니다.

> 이 노트북은 모델 훈련 및 미세 조정 전에 실행해야 합니다. MP3 데이터 세트를 다운로드하여 올바른 형식으로 변환해야 하기 때문입니다.

Call Center - Sentiment Analysis Pipeline

This notebook demonstrates how to build a pipeline for sentiment analysis of call center conversations. The goal of this pipeline is to develop sentiment analysis for use within an external dashboard.

This tutorial will guide you through the use of NVIDIA's RIVA for automatic speech recognition and text classification. This tutorial uses NetApp cloud storage for data storage and a pre-trained RIVA model.

Channels

These are the channels on which RIVA is hosting models.

- speech: `51051`
- voice: `61051`

These channels **must** be aligned with `riva_speech_api_port` and `riva_vision_api_port` within `config.sh`

In [4]:
```
speech_channel = "localhost:51051"
voice_channel = "localhost:61051"
```

Speech-To-Text

Automatic Speech Recognition (ASR) takes as input an audio stream or audio buffer and returns one or more text transcripts, along with additional optional metadata. ASR represents a full speech recognition pipeline that is GPU accelerated with optimized performance and accuracy. ASR supports synchronous and streaming recognition modes.

For more information on NVIDIA RIVA's Automatic Speech Recognition, visit here.

Constants

Use these constants to affect different aspects of this pipeline:

- `DATA_DIR` : base folder where data is stored
- `DATASET_NAME` : name of the call center dataset
- `COMPANY_DATE` : folder name identifying the particular call center conversation

Support Center - 모델 교육 및 미세 조정**. ipynb**

TAO 툴킷 가상 환경은 노트북을 실행하기 전에 설정해야 합니다(설치 지침은 명령 개요 의 TAO 툴킷 섹션 참조).

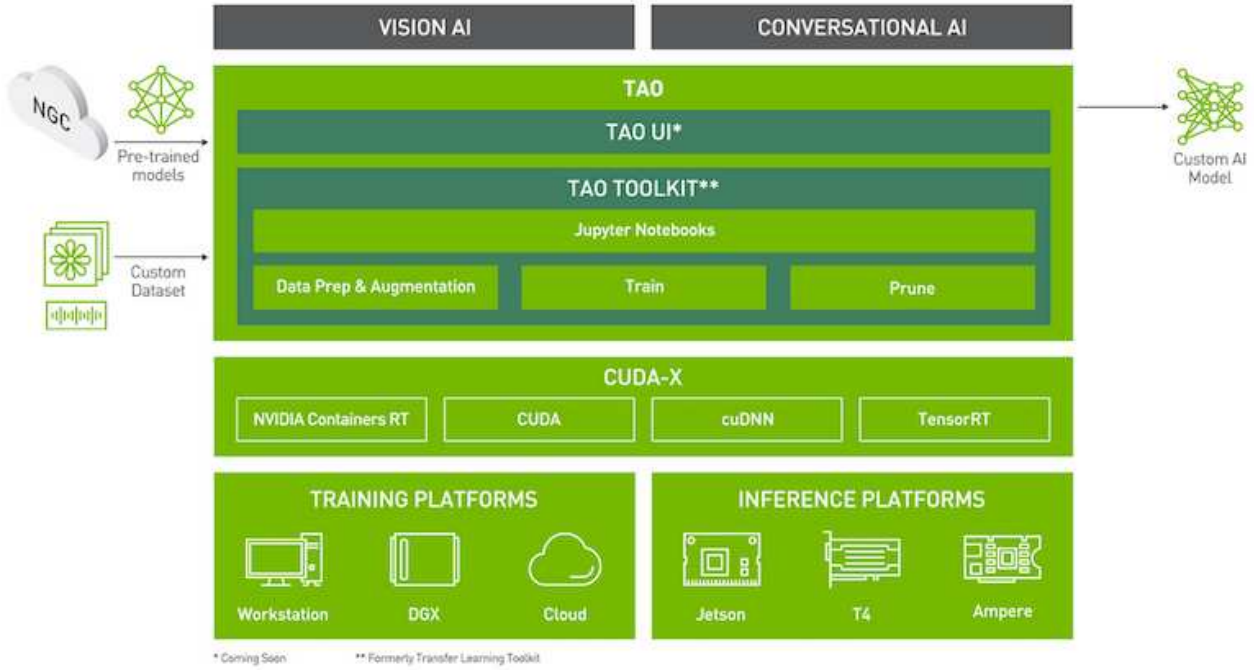
이 노트북은 TAO 툴킷을 사용하여 고객 데이터에 대한 딥 러닝 모델을 미세 조정합니다. 이전 전자 필기장과 마찬가지로 이 섹션은 텍스트 음성 대 텍스트 및 감정 분석 구성 요소에 대한 두 섹션으로 구분됩니다. 각 섹션은 데이터 처리, 모델 교육 및 세부 조정, 결과 평가 및 모델 내보내기를 거치게 됩니다. 마지막으로, Riva에서 사용할 미세 조정된 모델을 모두 배포하기 위한 마지막 섹션이 있습니다.

Call Center - Model Transfer Learning and Fine-Tuning

TAO Toolkit is a python based AI toolkit for taking purpose-built pre-trained AI models and customizing them with your own data. Transfer learning extracts learned features from an existing neural network to a new one. Transfer learning is often used when creating a large training dataset is not feasible in order to enhance the base performance of state-of-the-art models.

For this call center solution, the speech-to-text and sentiment analysis models are fine-tuned on call center data to augment the model performance on business specific terminology.

For more information on the TAO Toolkit, please visit here.



Installing necessary dependencies

For ease of use, please install TAO Toolkit inside a python virtual environment. We recommend performing this step first and then launching the notebook from the virtual environment. Please refer to the README for these instructions.

결론

고객 경험이 점점 더 경쟁이 치열해지는 주요 전장으로 간주됨에 따라, 거의 모든 산업의 기업이 간과할 수 없는 중요한 구성 요소가 AI 강화 글로벌 지원 센터가 되었습니다. 이 기술 보고서에서 제안된 솔루션은 이러한 탁월한 고객 경험의 제공을 지원하는 것으로 입증되었으며, 이제 기업이 AI 인프라 및 워크플로의 현대화를 위한 조치를 취하도록 하는 것이 과제입니다.

고객 서비스에서 AI를 가장 잘 구현하는 것은 상담원을 대체하지 않는 것입니다. 오히려 AI를 사용하면 실시간 감정 분석, 분쟁 에스컬레이션, 다중 모달 Affective 컴퓨팅을 통해 탁월한 고객 경험을 창출하여 포괄적인 AI 모델이 규모에 따라 권장사항을 제시하고 개별 상담원의 부족한 사항을 보완할 수 있는 언어, 비언어적, 안면 신호를 감지할 수 있습니다. 또한 AI는 특정 고객과 현재 사용 가능한 에이전트 간에 더 나은 일치를 제공할 수 있습니다. AI를 사용하는 기업은 공급자의 제품, 서비스 및 브랜드 이미지에 대한 생각과 인상에 대해 귀중한 고객 감정을 끌어낼 수 있습니다.

이 솔루션을 사용하여 지원 상담원이 객관적인 성능 평가 메트릭으로 사용할 시계열 데이터를 구성할 수도 있습니다. 일반적인 고객 만족도 설문 조사에는 대개 충분한 응답이 없습니다. 고용주는 장기적인 직원 및 고객 감정을 수집하여 지원 상담원의 성과에 대해 충분한 정보를 바탕으로 의사 결정을 내릴 수 있습니다.

NetApp, SFL Scientific, 오픈 소스 오케스트레이션 프레임워크 및 NVIDIA의 결합을 통해 최신 기술을 관리형 서비스로 통합하고 기술 도입을 가속하고 새로운 AI/ML 애플리케이션의 출시 시기를 앞당길 수 있습니다. 이러한 고급 서비스는 클라우드 네이티브 환경과 하이브리드 구축 아키텍처에 대해 쉽게 이식할 수 있는 온프레미스 서비스입니다.

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대해 자세히 알아보려면 다음 문서 및/또는 웹 사이트를 검토하십시오.

- 3D 대화형 데모

  "www.netapp.com/ai"

- NetApp AI 전문가와의 직접 연결

  "https://www.netapp.com/artificial-intelligence/"

- NVDIA Base Command Platform with NetApp 솔루션 개요

  https://www.netapp.com/pdf.html?item=/media/32792-DS-4145-NVIDIA-Base-Command-Platform-with-NetApp.pdf

- AI를 위한 NetApp 10가지 이유 인포그래픽

  "https://www.netapp.com/us/media/netapp-ai-10-good-reasons.pdf"

- 의료 부문의 AI: 폐 CT 스캔에서 COVID-19 병변을 식별하는 딥 러닝 백서

  https://www.netapp.com/pdf.html?item=/media/31240-WP-7342.pdf

- 의료 부문의 AI: 의료 설정에서 안면 마스크 사용 모니터링 백서

  https://www.netapp.com/pdf.html?item=/media/37490-NA-611-Monitoring-face-mask-usage-in-healthcare-settings.pdf

- 의료 부문의 AI: 진단 이미징 기술 보고서

  https://www.netapp.com/pdf.html?item=/media/7395-tr4811.pdf

- 소매 분야 AI: NVIDIA Riva를 통한 NetApp의 대화형 AI

  "https://docs.netapp.com/us-en/netapp-solutions/ai/cainvidia_executive_summary.html"

- NetApp ONTAP AI 솔루션 개요

  https://www.netapp.com/pdf.html?item=/media/6736-sb-3939.pdf

- NetApp DataOps 툴킷 솔루션 요약 정보

  https://www.netapp.com/pdf.html?item=/media/21480-SB-4111-1220-NA-Data-Science-Toolkit.pdf

- NetApp AI Control Plane 솔루션 요약

  https://www.netapp.com/pdf.html?item=/media/6737-sb-4055.pdf

- 데이터 드라이브를 통한 산업 혁신 AI eBook

  "https://www.netapp.com/us/media/na-337.pdf"

- NetApp EF-Series AI 솔루션 개요

  https://www.netapp.com/pdf.html?item=/media/26708-SB-4136-NetApp-AI-E-Series.pdf

- NetApp AI 및 Lenovo ThinkSystem for AI Inferencing 솔루션 개요

  https://www.netapp.com/pdf.html?item=/media/25316-SB-4129.pdf

- 엔터프라이즈 AI 및 ML용 NetApp AI 및 Lenovo ThinkSystem 솔루션 개요

  https://www.netapp.com/pdf.html?item=/media/25317-SB-4128.pdf

- NetApp과 NVIDIA – AI 비디오를 통해 가능한 것을 새롭게 정의합니다

  https://www.youtube.com/watch?v=38xw65SteUc

Azure-Click-Through Rate Prediction의 분산 교육

TR-4904: Azure에서 제공되는 분산 교육 - 클릭 비율 예측

Rick Huang, Verron Martina, Muneer Ahmad, NetApp

데이터 과학자의 작업은 머신 러닝(ML) 및 인공 지능(AI) 모델의 훈련 및 튜닝에 중점을 두어야 합니다. 그러나 구글의 조사에 따르면, 데이터 과학자들은 약 80%의 시간을 들여 모델을 엔터프라이즈 애플리케이션과 연동하고 대규모로 실행하는 방법을 찾아내고 있습니다.

엔드 투 엔드 AI/ML 프로젝트를 관리하려면 엔터프라이즈 구성 요소를 더 잘 이해해야 합니다. DevOps가 정의, 통합 및 구축을 인수했지만, 이러한 유형의 구성요소는 AI/ML 프로젝트를 포함하는 유사한 흐름을 타겟으로 합니다. 엔터프라이즈에서 엔드 투 엔드 AI/ML 파이프라인이 어떤 영향을 받는지 알아보려면 다음 필수 구성요소 목록을 참조하십시오.

- 스토리지
- 네트워킹
- 데이터베이스를 지원합니다
- 파일 시스템
- 컨테이너
- CI/CD(Continuous Integration and Continuous Deployment) 파이프라인
- IDE(통합 개발 환경)
- 보안

- 데이터 액세스 정책
- 하드웨어
- 클라우드
- 포함되었습니다
- 데이터 과학 도구 세트 및 라이브러리

대상

데이터 과학의 세계는 IT와 비즈니스의 여러 분야를 아우릅니다.

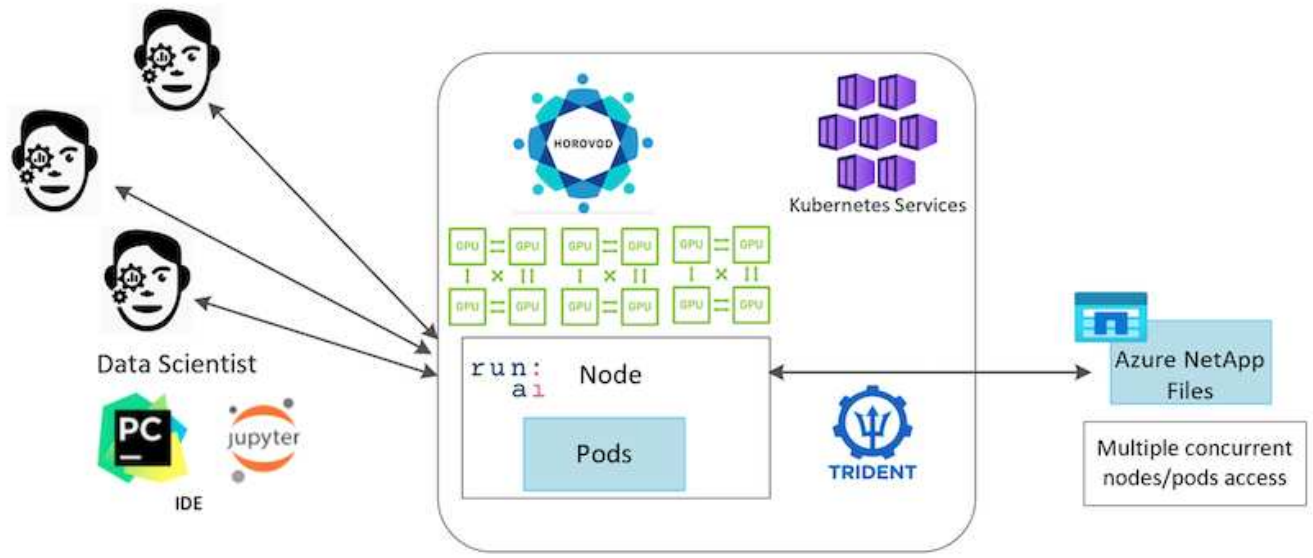
- 데이터 과학자는 자신이 선택한 도구와 라이브러리를 사용할 수 있는 유연성이 필요합니다.
- 데이터 엔지니어는 데이터 흐름과 데이터 위치를 알아야 합니다.
- DevOps 엔지니어는 새로운 AI/ML 애플리케이션을 CI/CD 파이프라인에 통합하는 툴을 필요로 합니다.
- 클라우드 관리자와 설계자는 Azure 리소스를 설정하고 관리할 수 있어야 합니다.
- 비즈니스 사용자는 AI/ML 애플리케이션에 액세스할 수 있기를 원합니다.

이 기술 보고서에서는 Azure NetApp Files, RAPIDS AI, DASK, Azure가 이러한 각 역할이 비즈니스에 어떤 가치를 제공하는지 설명합니다.

솔루션 개요

이 솔루션은 AI/ML 애플리케이션의 라이프사이클 뒤에 있습니다. 먼저 데이터 과학자의 작업을 통해 데이터를 준비하고 모델을 교육하는 데 필요한 다양한 단계를 정의합니다. DASK에서 RAPIDS를 활용하여 Azure Kubernetes Service(AKS) 클러스터 전반에 걸쳐 분산 교육을 수행하여 기존 Python 좌식 키트 학습 접근법과 비교하여 교육 시간을 크게 줄였습니다. 전체 주기를 완료하기 위해 Azure NetApp Files과 파이프라인을 통합합니다.

Azure NetApp Files는 다양한 성능 계층을 제공합니다. 고객은 표준 계층으로 시작하여 데이터를 이동하지 않고도 고성능 계층으로 스케일아웃 및 스케일업할 수 있습니다. 이 기능을 통해 데이터 과학자는 성능 문제 없이 규모에 맞게 모델을 교육할 수 있으므로 아래 그림과 같이 클러스터 전체에서 데이터 사일로를 피할 수 있습니다.



기술 개요

이 페이지에서는 이 솔루션에 사용된 기술에 대해 간략하게 설명합니다.

Microsoft 및 NetApp

2019년 5월부터 Microsoft는 NetApp ONTAP 기술을 기반으로 엔터프라이즈 NFS 및 SMB 파일 서비스를 위한 Azure 네이티브 자사 포털 서비스를 제공해 왔습니다. 이러한 개발을 위해 Microsoft와 NetApp의 전략적 파트너십을 활용하고 세계적인 수준의 ONTAP 데이터 서비스를 Azure로 확장합니다.

Azure NetApp Files

Azure NetApp Files 서비스는 엔터프라이즈급 고성능 용량제 파일 스토리지 서비스입니다. Azure NetApp Files은 모든 워크로드 유형을 지원하며 기본적으로 고가용성을 제공합니다. 서비스를 통해 서비스 및 성능 수준을 선택하고 스냅샷 복사본을 설정할 수 있습니다. Azure NetApp Files은 코드 변경 없이 데이터베이스, SAP, 고성능 컴퓨팅 애플리케이션 등 클라우드에서 가장 까다로운 엔터프라이즈 파일 워크로드를 마이그레이션 및 실행하기 위한 Azure 퍼스트 파티 서비스입니다.

이 참조 아키텍처는 IT 조직이 다음과 같은 이점을 얻을 수 있도록 해 줍니다.

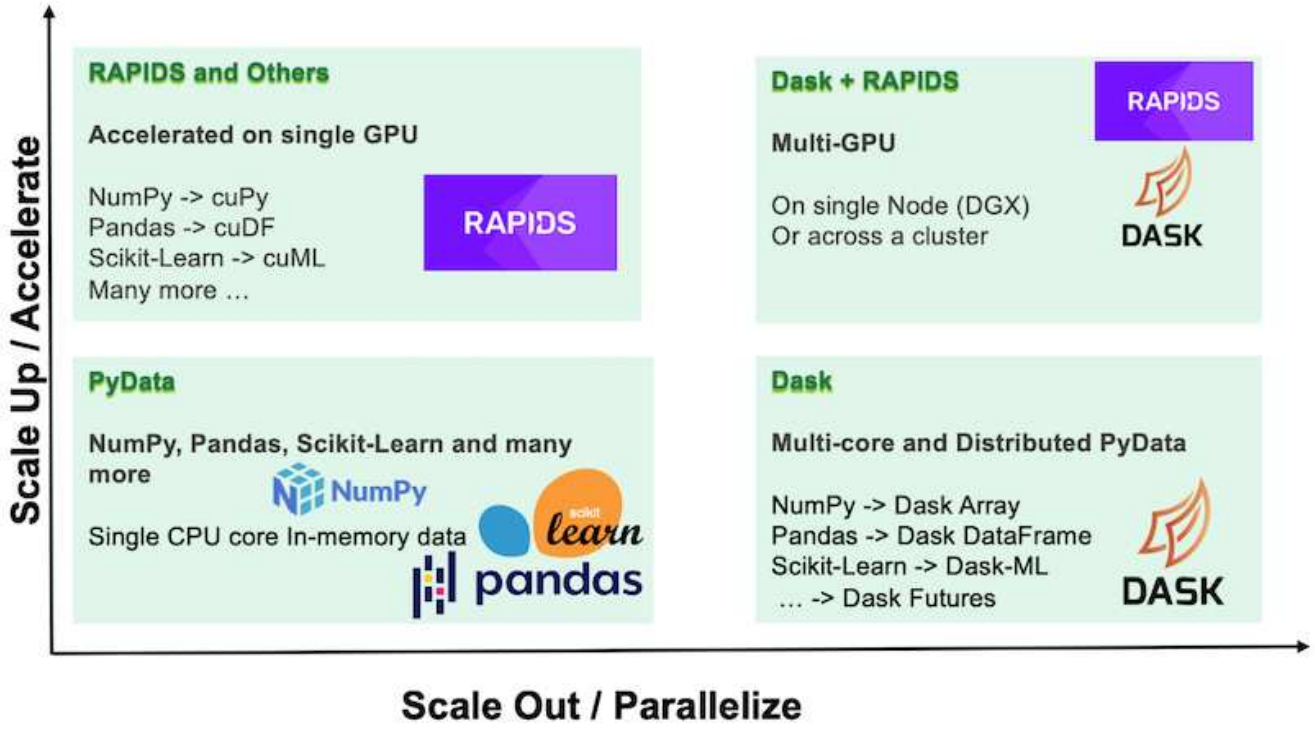
- 설계 복잡성 제거
- 컴퓨팅과 스토리지의 독립적인 확장 지원
- 고객이 작은 규모로 시작한 후 원활하게 확장할 수 있도록 지원
- 다양한 성능 및 비용 요소에 부합하는 폭넓은 스토리지 계층을 제공합니다

Dask 및 NVIDIA RAPIDS 개요

Dask는 여러 시스템에서 Python 라이브러리를 확장하고 대용량 데이터의 처리 속도를 높이는 오픈 소스 병렬 컴퓨팅 도구입니다. Pandas, Numpy 및 scikit-learn과 같은 단일 스레드 기존 Python 라이브러리와 유사한 API를 제공합니다. 따라서 기본 Python 사용자는 클러스터 전체에서 리소스를 사용하기 위해 기존 코드의 많은 부분을 변경하지 않아도 됩니다.

NVIDIA RAPIDS는 전체 GPU에서 엔드 투 엔드 ML 및 데이터 분석 워크플로우를 실행하도록 지원하는 오픈 소스 라이브러리 제품군입니다. DASK와 함께 사용하면 GPU 워크스테이션(스케일업)에서 다중 노드, 다중 GPU 클러스터(스케일아웃)까지 쉽게 확장할 수 있습니다.

클러스터에서 Dask를 구축할 경우 Kubernetes를 리소스 오케스트레이션에 사용할 수 있습니다. 다음 그림과 같이 프로세스 요구 사항에 따라 작업자 노드를 확장하거나 축소할 수도 있습니다. 그러면 클러스터 리소스 소비를 최적화할 수 있습니다.



소프트웨어 요구 사항

다음 표에는 이 솔루션에 필요한 소프트웨어 요구 사항이 나열되어 있습니다.

소프트웨어	버전
Azure Kubernetes 서비스	1.18.14
RAPIDS 및 Dask 컨테이너 이미지입니다	리포지토리: "rapidsai/rapidsai" 태그: 0.17-ca11.0-runtime-uubuntu18.04
NetApp 트라이던트	20.01.1
헬름	3.0.0

클라우드 리소스 요구사항

이 페이지에서는 Azure NetApp Files에 대한 클라우드 리소스 구성에 대해 설명합니다.

Azure NetApp Files를 구성합니다

에 설명된 대로 Azure NetApp Files를 구성합니다 "QuickStart: Azure NetApp Files를 설정하고 NFS 볼륨을 생성합니다".

Trident를 통해 볼륨을 생성하므로 "Azure NetApp Files용 NFS 볼륨 생성" 섹션을 계속 진행할 수 있습니다. 계속하기 전에 다음 단계를 완료하십시오.

1. Azure Shell을 통해 Azure NetApp Files 및 NetApp 리소스 공급자 에 등록("링크")를 클릭합니다.
2. Azure NetApp Files("링크")를 클릭합니다.
3. 용량 풀 설정(필요에 따라 최소 4TB Standard 또는 Premium)("링크"). 다음 표에는 클라우드에서 설정을 위한

네트워크 구성 요구 사항이 나와 있습니다. Dask 클러스터와 Azure NetApp Files는 동일한 Azure VNet(Virtual Network) 또는 피어링된 VNET에 있어야 합니다.

리소스	유형/버전
Azure Kubernetes 서비스	1.18.14
에이전트 노드	3x Standard_DS2_v2
GPU 노드	표준 _NC6s_v3 3개
Azure NetApp Files	표준 용량 풀
용량(TB	4

클릭률 예측 사용 사례 요약

이 사용 사례는 공개적으로 제공되는 를 기반으로 합니다 "테라바이트 로그를 클릭합니다" 데이터 세트 시작 "Criteo AI Lab을 참조하십시오". 최근 ML 플랫폼 및 애플리케이션의 발전으로 이제 대규모 학습에 많은 관심이 집중되고 있습니다. 클릭 비율(CTR)은 온라인 광고 노출 100회 당 평균 클릭 수(백분율로 표시)로 정의됩니다. 디지털 마케팅, 소매, 전자 상거래 및 서비스 공급자를 포함한 다양한 산업 및 사용 사례에서 핵심 메트릭으로 널리 채택되고 있습니다. CTR을 잠재적인 고객 트래픽에 대한 중요한 메트릭으로 사용하는 예는 다음과 같습니다.

- * 디지털 마케팅: * in "Google 웹로그 분석", CTR은 광고주 또는 상인의 키워드, 광고 및 무료 리스팅이 얼마나 잘 수행되고 있는지 측정하는 데 사용할 수 있습니다. 높은 CTR은 사용자가 귀하의 광고 및 리스팅을 유용하고 관련성 있는 것으로 찾도록 하는 좋은 지표입니다. 또한 CTR은 의 구성 요소인 키워드의 예상 CTR에 기여합니다 "광고 순위".
- * 전자 상거래: * 활용은 물론 "Google 웹로그 분석"전자 상거래 백엔드에는 최소 일부 방문자 통계가 있습니다. 이러한 통계는 한 눈에 유용하지 않을 수 있지만 일반적으로 읽기 쉽고 다른 정보보다 정확할 수 있습니다. 이러한 통계로 구성된 타사 데이터 세트는 독점 데이터이므로 전자 상거래 셀러, 구매자 및 플랫폼과 가장 관련이 있습니다. 이러한 데이터 세트는 추가 분석을 위해 시계시리즈를 구성하여 결과를 작년도와 어제와 비교하여 벤치마크 설정에 사용할 수 있습니다.
- * 소매: * 오프라인 유통업체는 방문자 수와 고객 수를 CTR과 연관시킬 수 있습니다. 고객 수는 POS(Point of Sale) 기록에서 확인할 수 있습니다. 소매업체의 웹 사이트 또는 광고 트래픽에서 CTR을 사용하면 앞서 언급한 판매량이 발생할 수 있습니다. 로열티 프로그램은 온라인 광고나 다른 웹 사이트에서 리디렉션된 고객이 보상을 받기 위해 참여할 수 있기 때문에 또 다른 활용 사례입니다. 소매업체는 로열티 프로그램을 통해 고객을 확보하고 판매 기록에서 고객 행동을 기록하여 다양한 범주의 소비자 구매 행동을 예측할 뿐만 아니라 쿠폰을 개인화하고 이탈을 줄이는 추천 시스템을 구축할 수 있습니다.
- * 서비스 공급자: * 통신 회사 및 인터넷 서비스 공급자는 통찰력 있는 AI, ML 및 분석 사용 사례를 위한 수많은 제3자 사용자 원격 측정 데이터를 보유하고 있습니다. 예를 들어, 통신 회사는 모바일 가입자의 웹 브라우징 최상위 도메인 기록 로그를 매일 활용하여 기존 모델을 세부 조정하여 최신 사용자 세분화, 고객 행동 예측, 온라인 경험 개선을 위한 실시간 광고 제작을 위해 광고업체와 협업할 수 있습니다. 이러한 데이터 중심 마케팅 워크플로에서 CTR은 변환을 반영하는 중요한 지표입니다.

디지털 마케팅의 맥락에서 "Criteo Terabyte 클릭 로그" 이제 ML 플랫폼 및 알고리즘의 확장성을 평가하는 데 필요한 참조 데이터세트가 되었습니다. 광고주는 클릭 비율을 예측함으로써 광고에 반응할 가능성이 가장 높은 방문자를 선택하고, 검색 기록을 분석하고, 사용자의 관심사에 따라 가장 관련성이 높은 광고를 표시할 수 있습니다.

이 기술 보고서에 제공된 솔루션은 다음과 같은 이점을 제공합니다.

- Azure NetApp Files는 분산 또는 대규모 교육에서 이점을 제공합니다

- RAPIDS CUDA 지원 데이터 처리(cuDF, cuPy 등) 및 ML 알고리즘(cuML)
- 분산 교육을 위한 Dask 병렬 컴퓨팅 프레임워크입니다

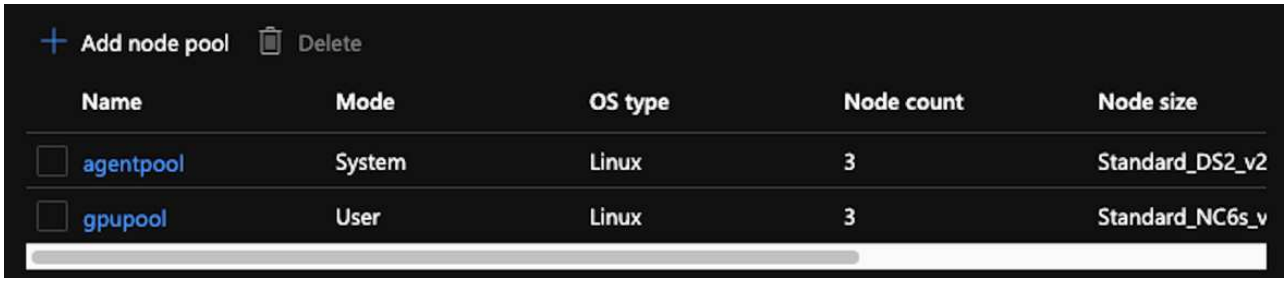
RAPIDS AI 및 Azure NetApp Files를 기반으로 하는 엔드 투 엔드 워크플로우에서 랜덤 포리스트 모델 훈련 시간을 크게 두 배나 단축한 것으로 입증되었습니다. 이러한 개선은 매일 45GB의 구조화된 표 형식 데이터(평균)를 사용하여 실제 클릭 로그를 처리할 때 기존의 Pandas 접근 방식과 비교했을 때 매우 중요합니다. 이는 약 20억 개의 행이 포함된 DataFrame과 같습니다. 클러스터 환경 설정, 프레임워크 및 라이브러리 설치, 데이터 로드 및 처리, 기존 교육과 분산 교육 비교, 시각화 및 모니터링, 이 기술 보고서의 중요한 엔드 투 엔드 런타임 결과를 비교합니다.

설정

AKS 클러스터를 설치하고 설정합니다

AKS 클러스터를 설치 및 설정하려면 웹 페이지를 참조하십시오 "AKS 클러스터를 생성합니다" 그런 다음 다음 다음 단계를 완료합니다.

1. 노드 유형(시스템 [CPU] 또는 작업자 [GPU] 노드)을 선택할 때 다음을 선택합니다.
   a. 기본 시스템 노드는 표준 DS2v2('agentpool' 기본 3개 노드)여야 합니다.
   b. 그런 다음 이름이 "gpupool"인 사용자 그룹(GPU 노드의 경우)에 대해 작업자 노드 Standard_NC6s_v3 풀(최소 3개 노드)을 추가합니다.



2. 배포에는 5~10분이 소요됩니다. 완료되면 Connect to Cluster를 클릭합니다.
3. 새로 생성된 AKS 클러스터에 연결하려면 로컬 환경(랩톱/PC)에서 다음을 설치합니다.
   a. 를 사용하는 Kubernetes 명령줄 툴입니다 "특정 OS에 대한 지침이 제공됩니다"
   b. 문서에 설명된 대로 Azure CLI를 사용할 수 있습니다. "Azure CLI를 설치합니다"
4. 터미널에서 AKS 클러스터에 액세스하려면 'az login'을 입력하고 자격 증명을 입력합니다.
5. 다음 두 명령을 실행합니다.

```
az account set --subscription xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx
aks get-credentials --resource-group resourcegroup --name aksclustername
```

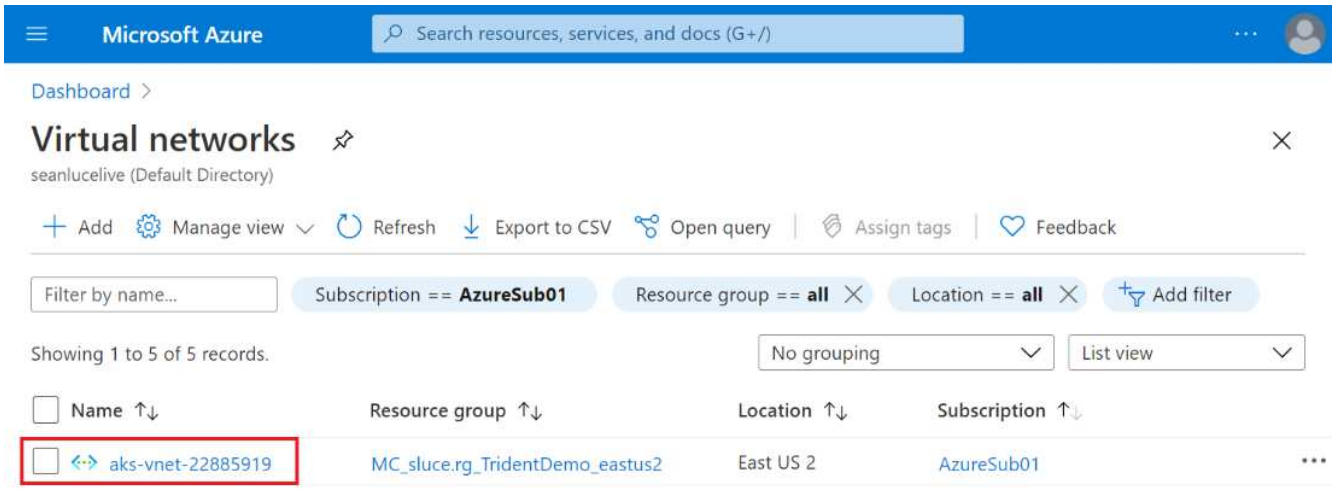
6. Azure CLI:kubeck get nodes를 입력합니다.
7. 다음 예와 같이 6개 노드가 모두 가동되어 실행 중인 경우 AKS 클러스터가 로컬 환경에 준비 및 연결됩니다

```
verronmartina@verron-mac-0 ~ % kubectl get nodes
NAME                                STATUS   ROLES    AGE   VERSION
aks-agentpool-34613062-vmss000000   Ready    agent    22m   v1.18.14
aks-agentpool-34613062-vmss000001   Ready    agent    22m   v1.18.14
aks-agentpool-34613062-vmss000002   Ready    agent    22m   v1.18.14
aks-gpupool-34613062-vmss000000     Ready    agent    20m   v1.18.14
aks-gpupool-34613062-vmss000001     Ready    agent    20m   v1.18.14
aks-gpupool-34613062-vmss000002     Ready    agent    20m   v1.18.14
verronmartina@verron-mac-0 ~ %
```

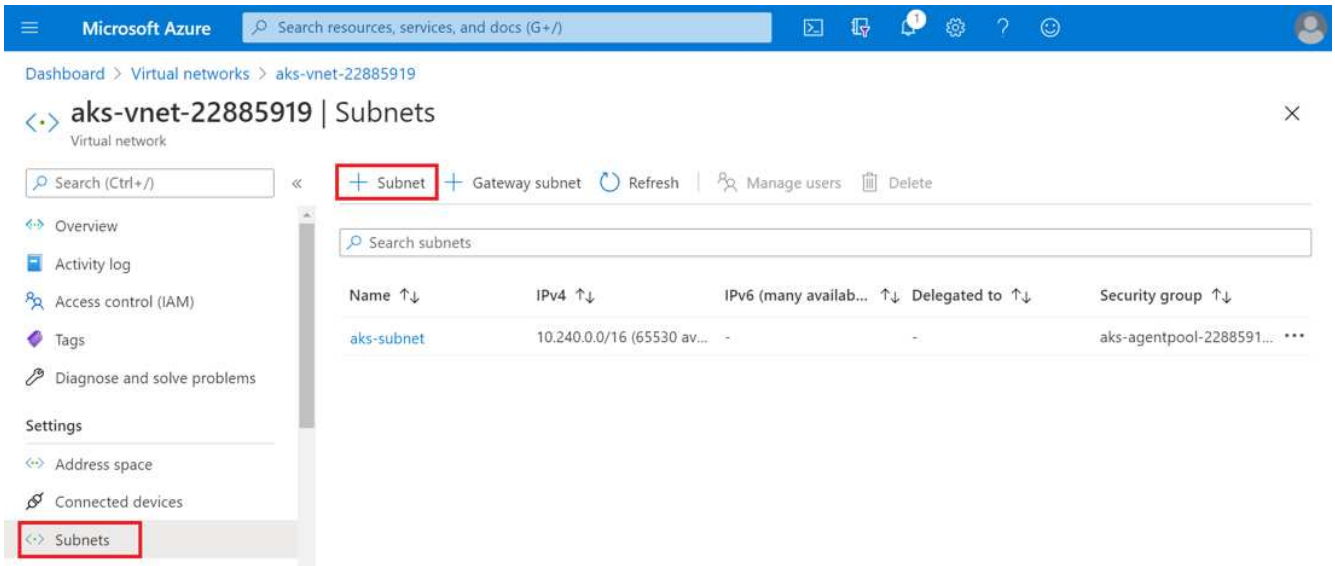
Azure NetApp Files에 대해 위임된 서브넷을 생성합니다

Azure NetApp Files에 대해 위임된 서브넷을 만들려면 다음 단계를 수행하십시오.

1. Azure 포털에서 가상 네트워크로 이동합니다. 새로 생성한 가상 네트워크를 찾습니다. AKS-VNET와 같은 접두사가 있어야 합니다.
2. VNET의 이름을 클릭합니다.

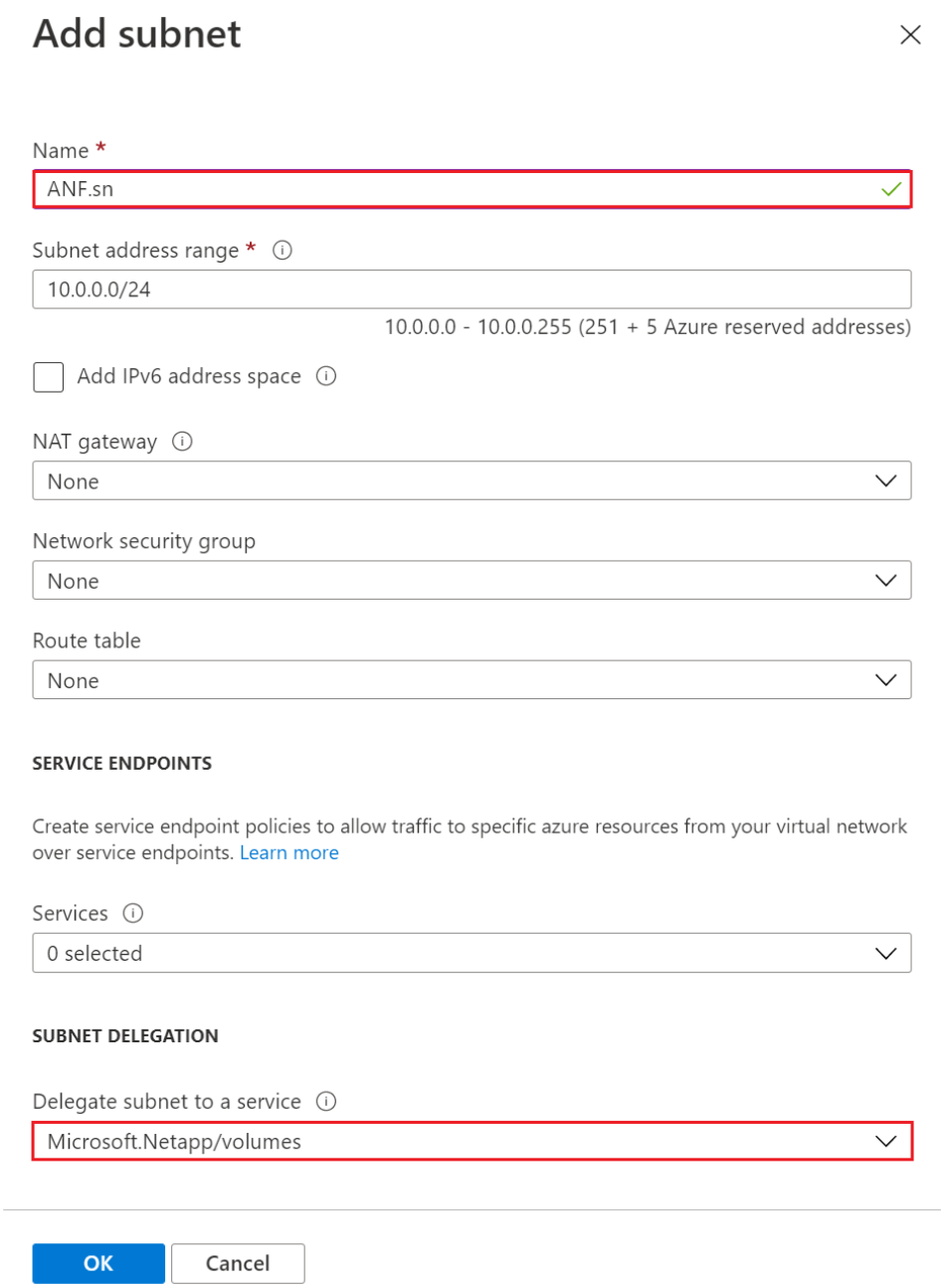


3. 서브넷 을 클릭하고 상단 도구 모음에서 + 서브넷 을 클릭합니다.



4. 서브넷에는 ANF.SN과 같은 이름을 입력하고 Subnet Delegation 제목 아래에서 microsoft.Netapp/volumes` 을

선택합니다. 다른 어떤 것도 변경하지 마십시오. 확인 을 클릭합니다.



Azure NetApp Files 볼륨은 애플리케이션 클러스터에 할당되며 Kubernetes에서 영구 볼륨 청구(PVC)로 사용됩니다. 또한 이 프로세스를 통해 Jupyter 노트북, 서버리스 기능 등과 같은 다양한 서비스에 유연하게 매핑할 수 있습니다.

서비스 사용자는 다양한 방법으로 플랫폼의 스토리지를 사용할 수 있습니다. 이 기술 보고서에서 NFS에 대해 설명함에 따라 Azure NetApp Files의 주요 이점은 다음과 같습니다.

- 사용자에게 스냅샷 복사본 사용 기능 제공
- 사용자가 Azure NetApp Files 볼륨에 대량의 데이터를 저장할 수 있도록 지원
- 대규모 파일 세트에서 모델을 실행할 때 Azure NetApp Files 볼륨의 성능 이점을 사용합니다.

피어 **AKS VNET** 및 **Azure NetApp Files VNET**

AKS VNET를 Azure NetApp Files VNET와 상호 운용하려면 다음 단계를 수행하십시오.

1. 검색 필드에 가상 네트워크를 입력합니다.
2. VNET AKS-VNET-NAME을 선택합니다 이 버튼을 클릭하고 검색 필드에 '북경'을 입력합니다.
3. 추가 를 클릭합니다.
4. 다음 설명을 입력합니다.
   a. 피어링 링크명은 AKS-VNET-NAME_to_anf입니다.
   b. VNET 피어링 파트너로 구독하는 Azure NetApp Files VNET와 가입자 ID입니다.
   c. 별표가 아닌 모든 섹션은 기본값을 사용하여 남겨 둡니다.
5. 추가 를 클릭합니다.

자세한 내용은 을 참조하십시오 "가상 네트워크 피어링을 생성, 변경 또는 삭제합니다".

Trident를 설치합니다

Hrom을 사용하여 Trident를 설치하려면 다음 단계를 완료하십시오.

1. Helm을 설치합니다(설치 지침은 를 참조하십시오 "출처")를 클릭합니다.
2. Trident 20.01.1 설치 프로그램을 다운로드하고 압축을 풉니다.

```
$wget
$tar -xf trident-installer-21.01.1.tar.gz
```

3. 디렉터리를 '트리덴트 - 설치자'로 변경합니다.

```
$cd trident-installer
```

4. 시스템 '$path'의 디렉토리에 tridentctl을 복사합니다.

```
$sudo cp ./tridentctl /usr/local/bin
```

5. Kubernetes(K8s) 클러스터에 Trident를 설치하고 H제어("출처"):
   a. 디렉터리를 'helm' 디렉토리로 변경합니다.

   ```
   $cd helm
   ```

   b. Trident를 설치합니다.

```
$helm install trident trident-operator-21.01.1.tgz --namespace
trident --create-namespace
```

c. Trident Pod의 상태를 확인합니다.

```
$kubectl -n trident get pods
```

모든 Pod가 가동되어 실행 중이면 Trident가 설치되어 앞으로 이동할 수 있습니다.

6. AKS에 대한 Azure NetApp Files 백엔드 및 스토리지 클래스를 설정합니다.

a. Azure 서비스 원칙을 만듭니다.

서비스 보안 주체는 Trident가 Azure와 통신하여 Azure NetApp Files 리소스를 조작하는 방법입니다.

```
$az ad sp create-for-rbac --name ""
```

출력은 다음 예와 같이 표시되어야 합니다.

```
{
"appId": "xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
"displayName": "netapptrident",
"name": "",
"password": "xxxxxxxxxxxxxxx.xxxxxxxxxxxxxx",
"tenant": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxx"
}
```

7. Trident 백엔드 json 파일(예: "anf-backend.json")을 생성합니다.
8. 원하는 텍스트 편집기를 사용하여 'anf-backend.json' 파일 안에 다음 필드를 입력합니다.

```
{
    "version": 1,
    "storageDriverName": "azure-netapp-files",
    "subscriptionID": "fakec765-4774-fake-ae98-a721add4fake",
    "tenantID": "fakef836-edc1-fake-bff9-b2d865eefake",
    "clientID": "fake0f63-bf8e-fake-8076-8de91e57fake",
    "clientSecret": "SECRET",
    "location": "westeurope",
    "serviceLevel": "Standard",
    "virtualNetwork": "anf-vnet",
    "subnet": "default",
    "nfsMountOptions": "vers=3,proto=tcp",
    "limitVolumeSize": "500Gi",
    "defaults": {
    "exportRule": "0.0.0.0/0",
    "size": "200Gi"
}
```

9. 다음 필드로 대체합니다.
   - '스크립트 ID'입니다. Azure 구독 ID입니다.
   - 텐antID. 이전 단계에서 'az ad sp'의 출력에서 Azure 테넌트 ID입니다.
   - '클라이언트 ID'. 이전 단계에서 'az ad sp'의 출력에서 귀하의 appID.
   - 'clientSecret' 이전 단계에서 사용한 'az ad sp' 출력의 암호입니다.
10. 구성 파일로 anf-backend.json을 사용하여 trident 네임스페이스에 Azure NetApp Files 백엔드를 생성하도록 Trident에 지시합니다.

```
$tridentctl create backend -f anf-backend.json -n trident
```

```
+------------------------+--------------------+--------------------------------------+--------+---------+
|          NAME          |   STORAGE DRIVER   |                 UUID                 | STATE  | VOLUMES |
+------------------------+--------------------+--------------------------------------+--------+---------+
| azurenetappfiles_86181 | azure-netapp-files | 2ca85462-59ac-4946-be05-c03f5575a2ad | online |       0 |
+------------------------+--------------------+--------------------------------------+--------+---------+
```

11. 스토리지 클래스를 생성합니다. Kubernetes 사용자는 이름으로 스토리지 클래스를 지정하는 PVC를 사용하여 볼륨을 프로비저닝합니다. K8s에게 이전 단계에서 만든 Trident 백엔드를 참조하는 스토리지 클래스 "azurenetappfiles"를 생성하도록 지시합니다.
12. 스토리지 클래스 및 복사본을 위한 YAML('anf-storage-class.yAML') 파일을 생성합니다.

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
name: azurenetappfiles
provisioner: netapp.io/trident
parameters:
backendType: "azure-netapp-files"
$kubectl create -f anf-storage-class.yaml
```

13. 스토리지 클래스가 생성되었는지 확인합니다.

```
kubectl get sc azurenetappfiles
```

```
NAME              PROVISIONER            RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
azurenetappfiles  csi.trident.netapp.io  Delete          Immediate           false                  98s
```

Helm을 사용하여 **AKS**에서 **RAPIDS**를 사용하여 **Dask**를 설정합니다

Hrom을 사용하여 AKS에서 RAPIDS를 사용하여 Dask를 설정하려면 다음 단계를 수행하십시오.

1. RAPIDS를 사용하여 Dask를 설치하기 위한 네임스페이스를 생성합니다.

```
kubectl create namespace rapids-dask
```

2. PVC를 생성하여 클릭률 데이터 세트를 저장합니다.
   a. 다음 YAML 콘텐츠를 파일에 저장하여 PVC를 생성합니다.

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: pvc-criteo-data
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 1000Gi
  storageClassName: azurenetappfiles
```

   b. Kubernetes 클러스터에 YAML 파일을 적용하십시오.

```
kubectl -n rapids-dask apply -f <your yaml file>
```

3. "rapidsai git" 리포지토리 복제("https://github.com/rapidsai/helm-chart")를 클릭합니다.

```
git clone https://github.com/rapidsai/helm-chart helm-chart
```

4. "Values.YAML"을 수정하고, 작업자와 Jupyter 작업공간을 위해 앞서 만든 PVC를 포함합니다.
   a. 리포지토리의 "rapidsai" 디렉토리로 이동합니다.

```
cd helm-chart/rapidsai
```

   b. Values.YAML 파일을 업데이트하고 PVC를 사용해 볼륨을 마운트합니다.

```
dask:
  …
  worker:
    name: worker
    …
    mounts:
      volumes:
        - name: data
          persistentVolumeClaim:
            claimName: pvc-criteo-data
      volumeMounts:
        - name: data
          mountPath: /data
    …
  jupyter:
    name: jupyter
    …
    mounts:
      volumes:
        - name: data
          persistentVolumeClaim:
            claimName: pvc-criteo-data
      volumeMounts:
        - name: data
          mountPath: /data
    …
```

5. 리포지토리의 홈 디렉토리로 이동하여 H제어 를 사용하여 AKS에 작업자 노드 3개가 있는 Dask를 배포합니다.

```
cd ..
helm dep update rapidsai
helm install rapids-dask --namespace rapids-dask rapidsai
```

Azure NetApp Files 성능 계층

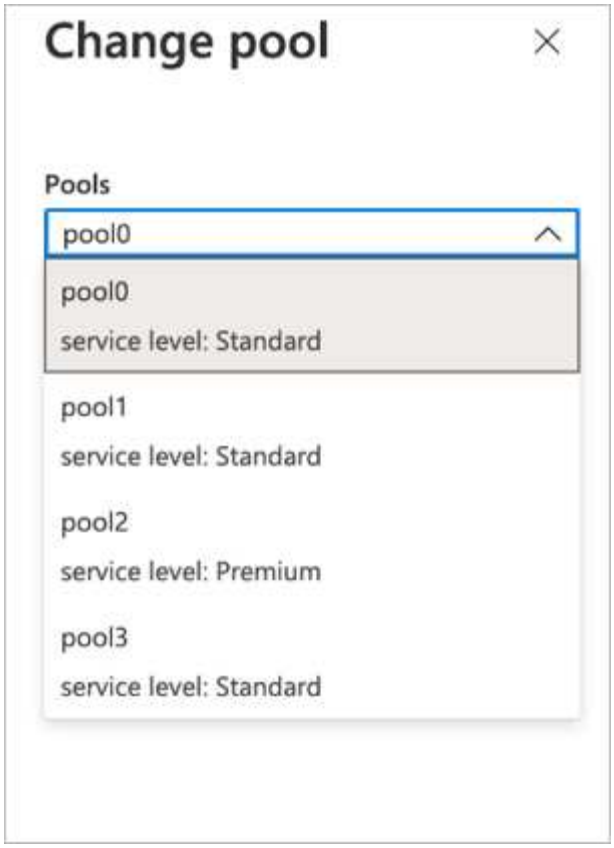
볼륨에 대해 원하는 서비스 수준을 사용하는 다른 용량 풀로 볼륨을 이동하여 기존 볼륨의 서비스 수준을 변경할 수 있습니다. 이 솔루션을 통해 고객은 작은 데이터 세트와 Standard Tier의 적은 수의 GPU로 시작한 후 데이터 및 GPU가 증가함에 따라 프리미엄 계층으로 스케일아웃 또는 스케일업할 수 있습니다. Premium Tier는 Standard Tier보다 테라바이트당 처리량이 4배 더 향상되었으며, 볼륨의 서비스 수준을 변경하기 위해 데이터를 이동할 필요 없이 스케일업이 가능합니다.

볼륨의 서비스 수준을 동적으로 변경합니다

볼륨의 서비스 수준을 동적으로 변경하려면 다음 단계를 수행하십시오.

1. 볼륨 페이지에서 서비스 수준을 변경할 볼륨을 마우스 오른쪽 단추로 클릭합니다. 풀 변경 을 선택합니다.

2. Change Pool 창에서 볼륨을 이동할 용량 풀을 선택합니다.



3. 확인 을 클릭합니다.

성능 계층 변경을 자동화합니다

다음 옵션을 사용하여 성능 계층 변경을 자동화할 수 있습니다.

- 동적 서비스 수준 변경은 현재 Public Preview에 있으며 기본적으로 활성화되어 있지 않습니다. Azure 구독에서 이 기능을 활성화하려면 이 설명서에서 방법을 참조하십시오 "볼륨의 서비스 수준을 동적으로 변경합니다".
- Azure CLI 볼륨 풀 변경 명령은 에 나와 있습니다 "볼륨 풀 변경 설명서" 및 의 예는 다음과 같습니다.

```
az netappfiles volume pool-change -g mygroup --account-name myaccname
--pool-name mypoolname --name myvolname --new-pool-resource-id
mynewresourceid
```

- PowerShell "Set-AzNetAppFilesVolumePool cmdlet" Azure NetApp Files 볼륨의 풀을 변경하며 다음 예에 표시됩니다.

```
Set-AzNetAppFilesVolumePool
-ResourceGroupName "MyRG"
-AccountName "MyAnfAccount"
-PoolName "MyAnfPool"
-Name "MyAnfVolume"
-NewPoolResourceId 7d6e4069-6c78-6c61-7bf6-c60968e45fbf
```

클릭률 예측 데이터 처리 및 모델 교육을 통해

데이터 처리 및 모델 교육을 위한 라이브러리

다음 표에서는 이 작업을 만드는 데 사용된 라이브러리와 프레임워크를 보여 줍니다. 이러한 모든 구성 요소는 Azure의 역할 기반 액세스 및 보안 제어와 완벽하게 통합됩니다.

라이브러리/프레임워크	설명
Dask cuML	GPU에서 작업할 ML의 경우 "cuML 라이브러리" Dask를 사용하여 RAPIDS cuML 패키지에 대한 액세스를 제공합니다. RAPIDS cuML은 고성능 GPU 기반 구축을 통해 클러스터링, 차원 축소, 회귀 접근 방식을 비롯한 인기 있는 ML 알고리즘을 구현하여 CPU 기반 접근 방식에 비해 최대 100배 빠른 속도를 제공합니다.
Dask cuDF	CuDF에는 데이터 하위 설정, 변환, 핫 인코딩 등 GPU 가속 추출, 변환, 로드(ETL)를 지원하는 다양한 함수가 포함되어 있습니다. RAPIDS 팀은 을 유지합니다 "dask-cudf 라이브러리" 여기에는 Dask 및 cuDF를 사용하는 도우미 메서드가 포함됩니다.
Scikit 학습	Scikit-Learn은 견적기라고 하는 수십 가지의 기계 학습 알고리즘과 모델을 제공합니다. 각각 "평가자" 를 사용하여 일부 데이터에 장착할 수 있습니다 "맞춤" 방법.

비교를 위해 두 대의 노트북을 사용해 ML 파이프라인을 구축했으며, 하나는 기존의 Pandas scikit-learn 접근방식이고, 다른 하나는 RAPIDS 및 Dask를 사용한 분산 훈련입니다. 각 노트북을 개별적으로 테스트하여 시간과 규모의 측면에서 성능을 확인할 수 있습니다. RAPIDS 및 DASK를 사용한 분산 훈련의 이점을 설명하기 위해 각 노트북을 개별적으로 다룹니다.

Pandas에서 **Logs day 15**를 로드하여 좌골키트을 훈련합니다**.** 무작위 포리스트 모델을 학습하십시오

이 섹션에서는 Pandas 및 Dask DataFrames를 사용하여 Criteo Terabyte 데이터 세트에서 Click Logs 데이터를 로드하는 방법을 설명합니다. 이 사용 사례는 광고 교환을 위한 디지털 광고에서 광고 클릭 여부를 예측하여 사용자의 프로필을 작성하는 데 사용됩니다. 또한 교환에서 자동화된 파이프라인에서 정확한 모델을 사용하지 않는 경우도 해당됩니다.

Click Logs 데이터 세트에서 15일차 데이터를 로드하여 총 45GB를 기록했습니다. Jupyter Notebook CTR-PandasRF-Collated에서 다음 셀을 실행하면 처음 5000만 개의 행이 포함된 Pandas DataFrame을 생성하고 좌골키트학습 무작위 포리스트 모델을 생성합니다.

```
%%time
import pandas as pd
import numpy as np
header = ['col'+str(i) for i in range (1,41)] #note that according to
criteo, the first column in the dataset is Click Through (CT). Consist of
40 columns
first_row_taken = 50_000_000 # use this in pd.read_csv() if your compute
resource is limited.
# total number of rows in day15 is 20B
# take 50M rows
"""
Read data & display the following metrics:
1. Total number of rows per day
2. df loading time in the cluster
3. Train a random forest model
"""
df = pd.read_csv(file, nrows=first_row_taken, delimiter='\t',
names=header)
# take numerical columns
df_sliced = df.iloc[:, 0:14]
# split data into training and Y
Y = df_sliced.pop('col1') # first column is binary (click or not)
# change df_sliced data types & fillna
df_sliced = df_sliced.astype(np.float32).fillna(0)
from sklearn.ensemble import RandomForestClassifier
# Random Forest building parameters
# n_streams = 8 # optimization
max_depth = 10
n_bins = 16
n_trees = 10
rf_model = RandomForestClassifier(max_depth=max_depth,
n_estimators=n_trees)
rf_model.fit(df_sliced, Y)
```

훈련된 무작위 포리스트 모델을 사용하여 예측을 수행하려면 이 전자 필기장에서 다음 단락을 실행하십시오. 중복을 피하기 위해 15일째부터 마지막 100만 행을 테스트 세트로 테스트했습니다. 또한 이 셀은 예측의 정확도를 계산하고, 사용자가 광고를 클릭하는지 여부를 모델이 정확하게 예측한 발생 비율로 정의됩니다. 이 노트북에서 잘 모르는 구성 요소를 검토하려면 을 참조하십시오 "공식 좌골 키트 - 학습 문서".

```
# testing data, last 1M rows in day15
test_file = '/data/day_15_test'
with open(test_file) as g:
    print(g.readline())

# dataFrame processing for test data
test_df = pd.read_csv(test_file, delimiter='\t', names=header)
test_df_sliced = test_df.iloc[:, 0:14]
test_Y = test_df_sliced.pop('col1')
test_df_sliced = test_df_sliced.astype(np.float32).fillna(0)
# prediction & calculating error
pred_df = rf_model.predict(test_df_sliced)
from sklearn import metrics
# Model Accuracy
print("Accuracy:",metrics.accuracy_score(test_Y, pred_df))
```

Dask에서 **Day 15**를 로드하고 **Dask cuML** 무작위 포리스트 모델을 교육합니다

이전 섹션과 비슷한 방식으로, Pandas에서 Criteo Click Logs day 15 를 로드하고 좌골키트학습 무작위 포리스트 모델을 훈련합니다. 이 예에서는 Dask cuDF를 사용하여 DataFrame 로드를 수행하고 Dask cuML에서 임의의 포리스트 모델을 교육했습니다. 이 섹션에서는 교육 시간과 규모의 차이를 비교했습니다 ""교육 시간 비교.""

criteo_dask_rf.ipynb입니다

이 노트북은 다음 예와 같이 'numpy', 'cuml', 필요한 'dask' 라이브러리를 가져옵니다.

```
import cuml
from dask.distributed import Client, progress, wait
import dask_cudf
import numpy as np
import cudf
from cuml.dask.ensemble import RandomForestClassifier as cumlDaskRF
from cuml.dask.common import utils as dask_utils
```

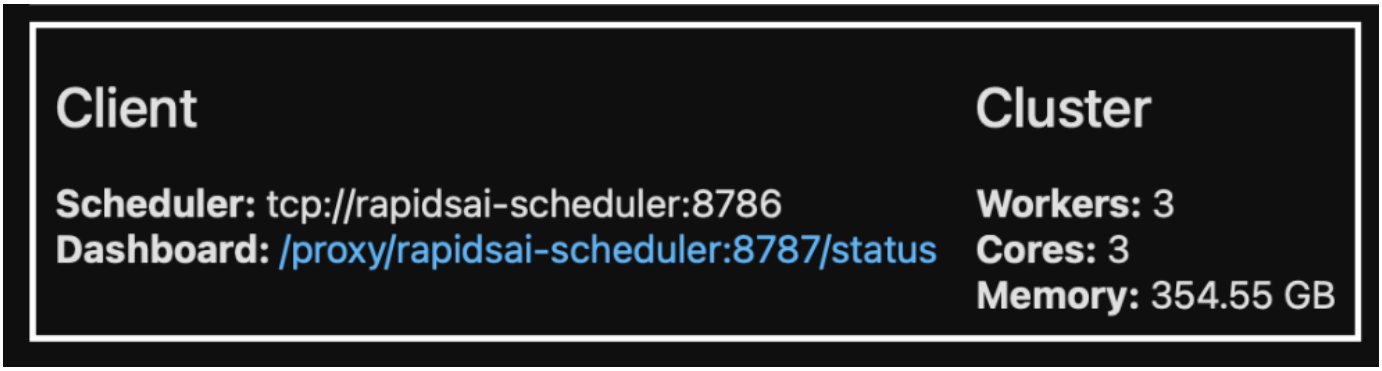
Dask 클라이언트()를 시작합니다.

```
client = Client()
```

클러스터가 올바르게 구성된 경우 작업자 노드의 상태를 확인할 수 있습니다.

```
client
workers = client.has_what().keys()
n_workers = len(workers)
n_streams = 8 # Performance optimization
```

AKS 클러스터에서 다음 상태가 표시됩니다.



DASK는 처리 코드를 즉시 실행하는 대신 실행 대신 실행 대신 대상 지정 DAG(Acyclic Graph)를 생성합니다. DAG에는 각 작업자가 실행해야 하는 일련의 작업과 상호 작용이 포함되어 있습니다. 이 레이아웃은 사용자가 Dask에서 한 가지 방식 또는 다른 방식으로 작업을 실행하도록 지시할 때까지 작업이 실행되지 않음을 의미합니다. Dask를 사용하면 다음과 같은 세 가지 주요 옵션을 사용할 수 있습니다.

- * DataFrame의 컴퓨팅()을 호출합니다. * 이 호출은 모든 파티션을 처리한 다음 결과를 스케줄러에 반환하여 최종 집계 및 cuDF DataFrame으로 변환합니다. 이 옵션은 스케줄러 노드의 메모리가 부족하지 않는 한 적은 결과에만 사용해야 합니다.
- * Call persist() on a DataFrame. * 이 호출은 그래프를 실행하지만 결과를 스케줄러 노드로 반환하는 대신 클러스터의 전체 노드를 메모리에 유지하여 사용자가 이러한 중간 결과를 다시 사용하지 않고도 파이프라인에서 재사용할 수 있도록 합니다.
- * DataFrame의 Call head(). * cuDF와 마찬가지로 이 호출은 10개의 레코드를 스케줄러 노드로 다시 반환합니다. 이 옵션을 사용하면 DataFrame 에 원하는 출력 형식이 포함되어 있는지 또는 레코드 자체가 타당한지 여부를 처리 및 계산에 따라 빠르게 확인할 수 있습니다.

따라서 사용자가 이러한 작업을 호출하지 않는 한 작업자는 스케줄러가 처리를 시작할 때까지 유휴 상태로 있습니다. 이러한 게으른 실행 패러다임은 Apache Spark와 같은 오늘날의 병렬적이고 분산된 컴퓨팅 프레임워크에서 흔히 볼 수 있습니다.

다음 단락에서는 분산 GPU 가속 컴퓨팅에 Dask cuML을 사용하여 임의 포리스트 모델을 교육하고 모델 예측 정확도를 계산합니다.

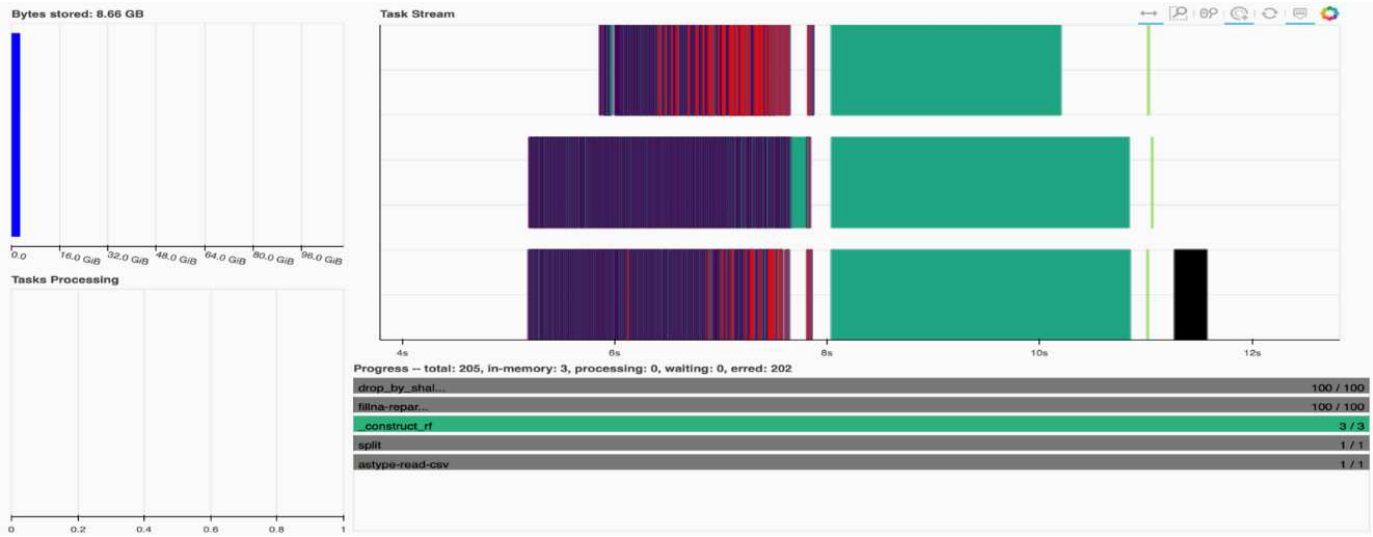
```
Adsf
# Random Forest building parameters
n_streams = 8 # optimization
max_depth = 10
n_bins = 16
n_trees = 10
cuml_model = cumlDaskRF(max_depth=max_depth, n_estimators=n_trees,
n_bins=n_bins, n_streams=n_streams, verbose=True, client=client)
cuml_model.fit(gdf_sliced_small, Y)
# Model prediction
pred_df = cuml_model.predict(gdf_test)
# calculate accuracy
cu_score = cuml.metrics.accuracy_score( test_y, pred_df )
```

기본 작업 스트림 대시보드를 사용하여 **Dask**를 모니터링합니다

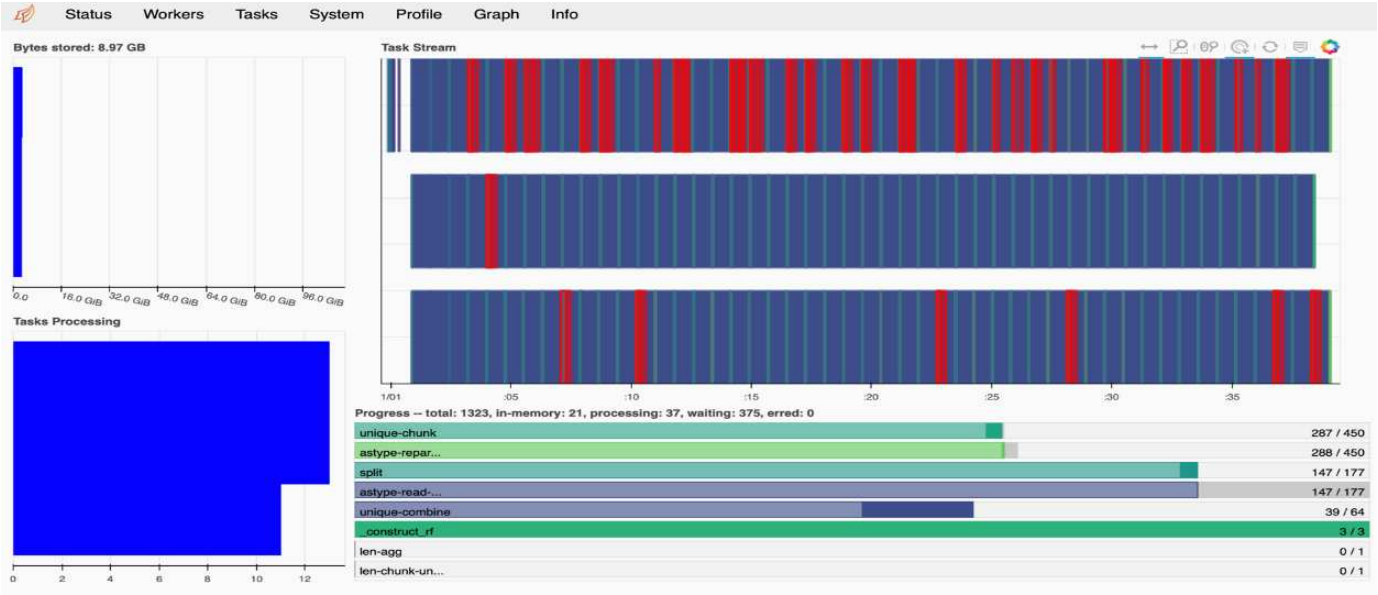
를 클릭합니다 "Dask 분산 스케줄러입니다" 다음과 같은 두 가지 형태로 실시간 피드백을 제공합니다.

- 실시간 정보가 포함된 여러 플롯과 테이블이 포함된 대화형 대시보드
- 콘솔 또는 노트북에서 대화형 사용에 적합한 진행률 표시줄

이 경우 다음 그림은 저장된 바이트, 스트림 수에 대한 자세한 분석 결과가 있는 작업 스트림, 실행된 관련 함수와 함께 작업 이름별 진행 상황 등을 포함하여 작업 진행 상황을 모니터링하는 방법을 보여 줍니다. 이 경우 작업자 노드가 3개이므로 스트림의 기본 청크가 3개 있고 색상 코드는 각 스트림 내의 서로 다른 작업을 나타냅니다.



개별 작업을 분석하고 실행 시간을 밀리초 단위로 점검하거나 장애물이나 장애물을 식별할 수 있는 옵션이 있습니다. 예를 들어 다음 그림에서는 랜덤 포리스트 모델 피팅 단계에 대한 작업 스트림을 보여 줍니다. DataFrame 프로세싱을 위한 고유한 청크, 랜덤 포리스트에 맞는 _Construct_RF 등 훨씬 더 많은 함수가 실행되고 있습니다. Criteo Click Logs에서 하루 동안 수집한 데이터의 크기가 45GB로 인해 대부분의 시간이 DataFrame 작업에 소비되었습니다.



교육 시간 비교

이 섹션에서는 기존 Pandas를 사용한 모델 교육 시간을 DASK와 비교합니다. Pandas의 경우, 메모리 오버플로를 방지하기 위해 처리 시간이 느려지기 때문에 더 적은 양의 데이터를 로드했습니다. 따라서 공정한 비교를 제공하기 위해 결과를 보간했습니다.

다음 표에서는 Pandas 무작위 포리스트 모델에 사용되는 데이터(데이터 세트의 15일 당 20억 개 중 5천만 개 행)가 상당히 적은 경우의 원시 교육 시간 비교를 보여 줍니다. 이 샘플은 사용 가능한 모든 데이터의 0.25% 이하만을 사용합니다. Dask-cuML의 경우 사용 가능한 모든 20억 행에 대해 무작위 포리스트 모델을 교육했습니다. 두 가지 접근 방식은 비슷한 훈련 시간을 낳았습니다.

접근 방식	교육 시간
Scikit-learn: 15일째에 50M 행만 교육 데이터로 사용합니다	47분 21초
RAPIDS-DASK: 15일 안에 20B 행을 모두 훈련 데이터로 사용	1시간, 12분, 11초

다음 표와 같이 교육 시간 결과를 선형적으로 보간할 경우 Dask와 함께 분산 훈련을 사용하면 큰 이점이 있습니다. 기존의 Pandas scikit-learn 접근 방식을 통해 클릭 로그를 하루에 45GB의 데이터를 처리하고 교육하는 데 13일이 걸리는 반면, RAPIDS-Dask 방식을 사용하면 같은 양의 데이터를 262.39배 더 빠르게 처리할 수 있습니다.

접근 방식	교육 시간
Scikit-learn: 15일째에 20B 행을 모두 훈련 데이터로 사용합니다	13일, 3시간, 40분, 11초
RAPIDS-DASK: 15일 안에 20B 행을 모두 훈련 데이터로 사용	1시간, 12분, 11초

이전 표에서 RAPIDS와 DASK를 사용하여 여러 GPU 인스턴스에 데이터 처리 및 모델 훈련을 분산하면, Scikit-learn 모델 훈련을 통해 기존 PandDataFrame 처리에 비해 실행 시간이 상당히 짧아진다는 것을 알 수 있습니다. 이 프레임워크를 통해 다중 노드, 다중 GPU 클러스터의 온프레미스뿐만 아니라 클라우드에서 스케일업 및 스케일아웃이 가능합니다.

Prometheus 및 Grafana로 Dask 및 RAPIDS를 모니터링합니다

모든 것이 배포된 후 새 데이터에 대한 추론을 실행합니다. 이 모델은 사용자가 검색 활동을 기반으로 광고를 클릭하는지 여부를 예측합니다. 예측 결과는 Dask cuDF에 저장됩니다. Prometheus를 사용하여 결과를 모니터링하고 Grafana 대시보드에서 시각화할 수 있습니다.

자세한 내용은 다음을 참조하십시오 "RAPIDS AI 중간 포스트".

NetApp DataOps 툴킷을 사용하여 데이터 세트 및 모델 버전 관리

Kubernetes용 NetApp DataOps 툴킷은 스토리지 리소스와 Kubernetes 워크로드를 데이터 과학 작업 공간 수준까지 추상화합니다. 이러한 기능은 데이터 과학자와 데이터 엔지니어를 위해 설계된 간단하고 사용하기 쉬운 인터페이스로 패키징되어 있습니다. 데이터 과학자와 엔지니어는 익숙한 형식의 Python 프로그램을 사용하여 JupyterLab 작업 공간을 단 몇 초 만에 프로비저닝 및 폐기할 수 있습니다. 이러한 작업 공간에는 테라바이트나 페타바이트급의 스토리지 용량이 포함될 수 있으므로 데이터 과학자는 모든 훈련 데이터 세트를 프로젝트 작업 공간에 직접 저장할 수 있습니다. 작업 영역과 데이터 볼륨을 별도로 관리하는 시대는 지났습니다.

자세한 내용은 툴킷 을 참조하십시오 "GitHub 리포지토리".

Jupyter 노트북을 참조하십시오

이 기술 보고서와 관련하여 2개의 Jupyter 노트북이 있습니다.

- "* CTR-PandasRF-Collated.ipynb. *" 이 노트북은 Criteo Terabyte Click Logs 데이터 세트에서 15일을 로드하고 Pandas DataFrame으로 데이터를 처리 및 포맷하고 Scikit-Learn 무작위 포리스트 모델을 교육하며 예측을 수행하고 정확도를 계산합니다.
- "criteo_dask_rf.ipynb. * 를 사용합니다" 이 전자 필기장은 Criteo Terabyte Click Logs 데이터 집합에서 15일을 로드하고, 데이터를 처리하여 Dask cuDF로 서식을 지정하고, Dask cuML 임의 포리스트 모델을 교육하고, 예측을 수행하고, 정확도를 계산합니다. GPU에 여러 작업자 노드를 활용함으로써 이러한 분산 데이터 및 모델 처리 및 교육 접근 방식이 매우 효율적입니다. 처리하는 데이터가 많을수록 기존 ML 방식에 비해 시간 절감 효과가 더 커집니다. 네트워킹 설정을 통해 데이터 및 모델 배포를 자유롭게 이동할 수 있는 한, 이 메모장을 클라우드, 온프레미스 또는 Kubernetes 클러스터에 다른 위치의 컴퓨팅 및 스토리지가 포함된 하이브리드 환경에 배포할 수 있습니다.

결론

Azure NetApp Files, RAPIDS 및 DASK는 Docker, Kubernetes 등의 오케스트레이션 툴과 통합하여 대규모 ML 처리 및 훈련 구축을 간소화하고 있습니다. 이 솔루션은 엔드 투 엔드 데이터 파이프라인을 통합함으로써 수많은 고급 컴퓨팅 워크로드에서 발생하는 지연 시간과 복잡성을 줄여 개발과 운영 간의 격차를 효과적으로 해소합니다. 데이터 과학자는 대규모 데이터 세트에서 쿼리를 실행하고 교육 단계 동안 다른 사용자와 데이터 및 알고리즘 모델을 안전하게 공유할 수 있습니다.

자체 AI/ML 파이프라인을 구축할 때는 아키텍처 구성 요소의 통합, 관리, 보안 및 접근성을 구성하는 것이 매우 어렵습니다. 개발자가 자신의 환경에 액세스하고 제어하도록 하는 것은 또 다른 도전 과제입니다.

클라우드에 엔드 투 엔드 분산 교육 모델 및 데이터 파이프라인을 구축하여 총 워크플로우 완료 시간을 GPU 가속 데이터 처리 및 컴퓨팅 프레임워크를 활용하지 않는 기존의 오픈 소스 접근 방식에 비해 크게 두 배나 단축한 것으로

입증되었습니다.

NetApp, Microsoft, 오픈 소스 오케스트레이션 프레임워크 및 NVIDIA가 결합되어 최신 기술을 유연한 관리 서비스로 통합하여 기술 채택을 가속화하고 새로운 AI/ML 애플리케이션의 출시 시기를 앞당길 수 있습니다. 이러한 고급 서비스는 사내 및 하이브리드 구축 아키텍처용으로 쉽게 포팅할 수 있는 클라우드 네이티브 환경에서 제공됩니다.

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 리소스를 참조하십시오.

- Azure NetApp Files:
  - Azure NetApp Files용 솔루션 아키텍처 페이지

    "https://docs.microsoft.com/azure/azure-netapp-files/azure-netapp-files-solution-architectures"
- 컨테이너용 Trident 영구 스토리지:
  - Azure NetApp Files 및 Trident

    "https://netapptrident.readthedocs.io/en/stablev20.07/kubernetes/operations/tasks/backends/anf.html"
- Dask 및 RAPIDS:
  - Dask(질문)

    "https://docs.dask.org/en/latest/"
  - Dask를 설치합니다

    "https://docs.dask.org/en/latest/install.html"
  - Dask API

    "https://docs.dask.org/en/latest/api.html"
  - Dask 기계 학습

    "https://examples.dask.org/machine-learning.html"
  - Dask 분산 진단

    "https://docs.dask.org/en/latest/diagnostics-distributed.html"
- ML 프레임워크 및 도구:
  - TensorFlow: 모두를 위한 오픈 소스 머신 러닝 프레임워크

    "https://www.tensorflow.org/"
  - Docker 를 참조하십시오

    "https://docs.docker.com"
  - 쿠버네티스

"https://kubernetes.io/docs/home/"

- Kubeflow

  "http://www.kubeflow.org/"

- Jupyter 노트북 서버

  "http://www.jupyter.org/"

TR-4896: Azure에서 분산된 교육: 차선 감지 - 솔루션 설계

Muneer Ahmad and Verron Martina, NetApp Ronden Dar, run:AI

2019년 5월부터 Microsoft는 NetApp ONTAP 기술을 기반으로 엔터프라이즈 NFS 및 SMB 파일 서비스를 위한 Azure 네이티브 자사 포털 서비스를 제공합니다. 이러한 개발을 위해 Microsoft와 NetApp의 전략적 파트너십을 활용하고 세계적인 수준의 ONTAP 데이터 서비스를 Azure로 확장합니다.

업계 최고의 클라우드 데이터 서비스 공급자인 NetApp이 Run:AI와 팀을 이루어 AI 인프라를 가상화하여 AI의 전체 GPU 활용률을 더욱 빠르게 실험할 수 있도록 했습니다. 이 파트너십을 통해 팀에서는 데이터를 빠르게 활용하고 컴퓨팅 리소스를 무제한으로 활용하여 여러 실험을 병렬로 실행하여 AI 속도를 높일 수 있습니다. 실행: AI는 리소스 할당을 자동화하여 전체 GPU 활용률을 지원하며, 검증된 Azure NetApp Files 아키텍처를 통해 데이터 파이프라인의 장애물을 제거하여 모든 실험을 최대 속도로 실행할 수 있습니다.

NetApp과 RUN TO NETAPP: AI는 고객에게 Azure에서의 AI 전환을 위한 미래 지향형 플랫폼을 제공하기 위해 힘을 합했습니다. 분석 및 고성능 컴퓨팅(HPC)에서 자율적 결정(고객이 필요한 시점에 필요한 비용만 지불하여 IT 투자를 최적화할 수 있음)에 이르기까지, NetApp과 실행 시 AI는 Azure Cloud에서 통합된 단일 경험을 제공합니다.

솔루션 개요

이 아키텍처에서 초점은 AI 또는 머신 러닝(ML) 분산 훈련 프로세스 중 가장 컴퓨팅 집약적인 레인 감지 프로세스에 있습니다. 차선 감지는 자동 주행에서 가장 중요한 작업 중 하나로서, 차선 표시를 현지화함으로써 차량을 인도하는 데 도움이 됩니다. 차선 표시와 같은 정적 구성 요소는 차량이 고속도로를 대화식으로 안전하게 주행하도록 안내합니다.

합성신경망(CNN) 기반 접근 방식은 장면에 대한 이해와 세그멘테이션을 새로운 차원으로 끌어올려 왔습니다. 폐쇄될 수 있는 긴 구조 및 영역(예: 폴, 차선의 음영 등)이 있는 물체에는 이 기능이 제대로 작동하지 않습니다. 공간 컨벌루ional Neural Network(SCNN)는 CNN을 풍부한 공간 수준으로 일반화합니다. 동일한 레이어에서 뉴런 간에 정보를 전달할 수 있으므로 폐색이 있는 레인, 폴 또는 트럭과 같은 구조적 개체에 가장 적합합니다. 이러한 호환성은 공간 정보를 보강할 수 있고 매끄러움과 지속성을 유지하기 때문입니다.

모델이 데이터세트의 다양한 구성 요소를 학습하고 구분할 수 있도록 시스템에 수천 개의 화면 이미지를 삽입해야 합니다. 이러한 이미지에는 날씨, 주간 또는 야간, 다차선 고속도로 도로 및 기타 교통 상황이 포함됩니다.

교육에는 양질의 데이터와 많은 양의 데이터가 필요합니다. 단일 GPU 또는 여러 GPU를 사용하여 교육을 완료하는 데 며칠~몇 주가 걸릴 수 있습니다. 데이터 분산 교육을 통해 여러 노드 및 GPU를 사용하여 프로세스를 가속화할 수 있습니다. Horovod는 분산 교육을 제공하지만 GPU 클러스터 간에 데이터를 읽는 것이 방해가 될 수 있는 프레임워크 중 하나입니다. Azure NetApp Files은 컴퓨팅 용량의 최고에 GPU를 활용할 수 있도록 초고속, 높은 처리량, 지속적으로 짧은 지연 시간을 제공합니다. 이번 실험에서 클러스터 전체의 모든 GPU가 SCNN을 사용하여 차선 감지를 교육하기 위해 평균 96% 이상 사용되고 있다는 것을 확인했습니다.

대상

데이터 과학은 IT 및 비즈니스 분야의 여러 분야를 통합하므로 여러 페르소나가 대상 고객을 대상으로 합니다.

- 데이터 과학자는 자신이 선택한 도구와 라이브러리를 사용할 수 있는 유연성이 필요합니다.
- 데이터 엔지니어는 데이터 흐름과 데이터 위치를 알아야 합니다.
- 자율 주행 사용 사례 전문가
- 클라우드 관리자 및 설계자는 Azure(클라우드) 리소스를 설정하고 관리합니다.
- DevOps 엔지니어는 새로운 AI/ML 애플리케이션을 CI/CD(Continuous Integration and Continuous Deployment) 파이프라인에 통합하는 툴을 필요로 합니다.
- 비즈니스 사용자는 AI/ML 애플리케이션에 액세스할 수 있기를 원합니다.

이 문서에서는 Azure NetApp Files, RUN:AI 및 Microsoft Azure가 각 역할이 비즈니스에 제공하는 데 어떤 도움이 되는지 설명합니다.

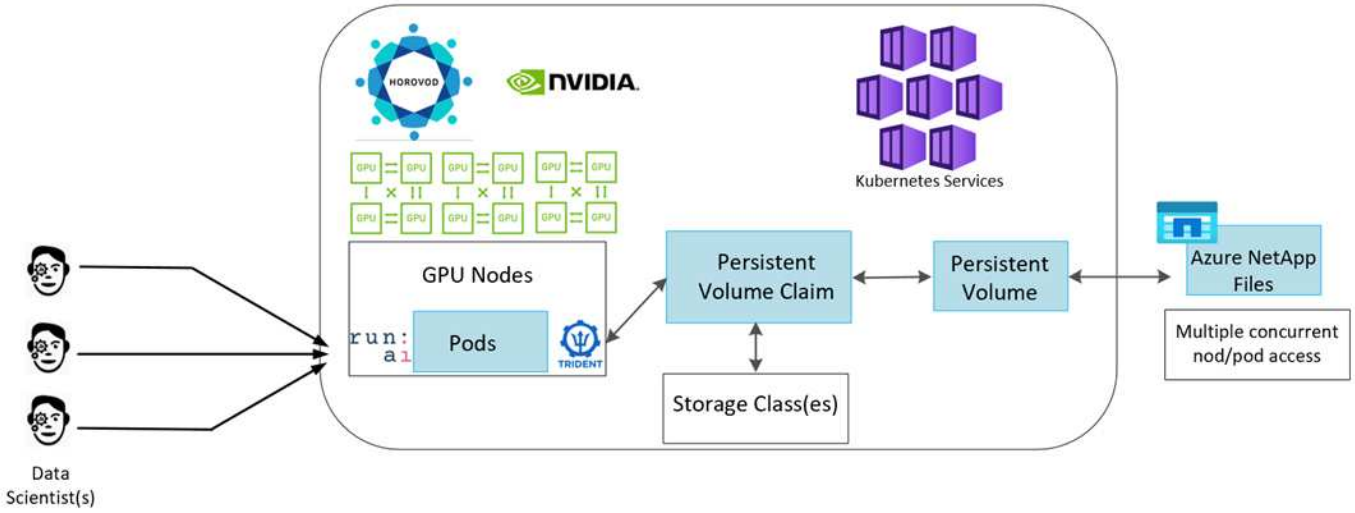
솔루션 기술

이 섹션에서는 Azure 클라우드에서 완벽하게 실행되는 분산 교육 솔루션을 구현하여 레인 감지 사용 사례에 대한 기술 요구 사항을 다룹니다. 아래 그림은 솔루션 아키텍처를 간략하게 보여 줍니다.

이 솔루션에 사용되는 요소는 다음과 같습니다.

- Azure Kubernetes 서비스(AKS)
- NVIDIA GPU를 사용하는 Azure Compute SKU
- Azure NetApp Files
- 실행: AI
- NetApp 트라이던트

여기에 언급된 모든 요소에 대한 링크는 에 나와 있습니다 "추가 정보" 섹션을 참조하십시오.



클라우드 리소스 및 서비스 요구사항

다음 표에는 솔루션을 구현하는 데 필요한 하드웨어 구성요소가 나와 있습니다. 솔루션 구현에 사용되는 클라우드 구성요소는 고객 요구사항에 따라 다를 수 있습니다.

클라우드	수량
AKS	최소 3개의 시스템 노드 및 3개의 GPU 작업자 노드
가상 머신(VM) SKU 시스템 노드입니다	Standard_DS2_v2 3개
VM SKU GPU 작업자 노드입니다	표준 _NC6s_v3 3개
Azure NetApp Files	4TB 표준 계층

소프트웨어 요구 사항

다음 표에는 솔루션을 구현하는 데 필요한 소프트웨어 구성요소가 나와 있습니다. 솔루션 구현에 사용되는 소프트웨어 구성요소는 고객 요구사항에 따라 다를 수 있습니다.

소프트웨어	버전 또는 기타 정보
AKS - Kubernetes 버전	1.18.14
실행: AI CLI	v2.2.25
실행: AI Orchestration Kubernetes Operator version	1.0.109
호로브	0.21.2
NetApp 트라이던트	20.01.1
헬름	3.0.0

차선 감지 – AI를 통한 분산 훈련

이 섹션에서는 RUN:AI Orchestrator를 사용하여 규모에 따라 차선 감지 분산 교육을 수행할 수 있는 플랫폼을 설정하는 방법에 대해 자세히 설명합니다. 모든 솔루션 요소의 설치와 해당 플랫폼에서 분산된 교육 작업을 실행하는 방법에 대해 설명합니다. ML 버전 관리는 데이터 및 모델 재현성을 달성하기 위한 Run:AI 실험과 연결된 NetApp SnapshotTM을 사용하여 완료됩니다. ML 버전 관리는 모델 추적, 팀 구성원 간 작업 공유, 결과 재현성, 새로운 모델 버전을 운영 환경에 롤링하며 데이터 관리에 중요한 역할을 합니다. NetApp ML 버전 제어(Snapshot)는 각 실험과 관련된 데이터, 훈련된 모델 및 로그의 시점 버전을 캡처할 수 있습니다. 풍부한 API 지원을 통해 run:AI 플랫폼과 쉽게 통합할 수 있습니다. 교육 상태에 따라 이벤트를 트리거하기만 하면 됩니다. 또한, Kubernetes(K8s) 상에서 실행 중인 코드나 컨테이너의 아무 것도 변경하지 않고 전체 실험의 상태를 포착해야 합니다.

마지막으로, 이 기술 보고서에서는 AKS의 여러 GPU 지원 노드에 대한 성능 평가를 마무리합니다.

TuSimple 데이터 세트를 사용하여 차선 감지 사용 사례에 대한 분산 교육

이 기술 보고서에서 분산된 교육은 차선 감지를 위한 TuSimple 데이터 세트에 대해 수행됩니다. Horovod는 AKS를 통해 Kubernetes 클러스터의 여러 GPU 노드에 대해 동시에 데이터 분산 교육을 수행하기 위한 교육 코드에 사용됩니다. 코드는 TuSimple 데이터 다운로드 및 처리를 위한 컨테이너 이미지로 패키징됩니다. 처리된 데이터는

NetApp Trident 플러그인에서 할당한 영구 볼륨에 저장됩니다. 교육에서는 하나 이상의 컨테이너 이미지가 생성되고 데이터를 다운로드하는 동안 생성된 영구 볼륨에 저장된 데이터를 사용합니다.

데이터 및 교육 작업을 제출하려면 RUN:AI를 사용하여 리소스 할당 및 관리를 오케스트레이션하십시오. Run: AI를 사용하면 Horovod에 필요한 MPI(Message Passing Interface) 작업을 수행할 수 있습니다. 이 레이아웃을 통해 여러 GPU 노드가 서로 통신하여 각 교육 미니 일괄 처리 후 교육 가중치를 업데이트할 수 있습니다. 또한 UI 및 CLI를 통해 교육을 모니터링할 수 있으므로 실험 진행 상황을 쉽게 모니터링할 수 있습니다.

NetApp Snapshot은 교육 코드 내에 통합되어 모든 실험에 대한 데이터 상태 및 훈련 모델을 캡처합니다. 이 기능을 사용하면 사용된 데이터 및 코드의 버전과 생성된 관련 교육 모델을 추적할 수 있습니다.

AKS 설정 및 설치

AKS 클러스터의 설정 및 설치는 로 이동하십시오 "AKS 클러스터를 생성합니다". 그런 다음 다음 다음 일련의 단계를 수행합니다.

1. 노드 유형(시스템(CPU) 또는 작업자(GPU) 노드인지 여부)을 선택할 때 다음을 선택합니다.

   a. S tandard_DS2_v2 크기의 1차 시스템 노드 agentpool을 추가합니다. 기본 3개 노드를 사용합니다.

   b. Standard_NC6s_v3 풀 크기로 작업자 노드 'gpupool'을 추가합니다. GPU 노드에 대해 최소 3개의 노드를 사용합니다.

   배포에는 5~10분이 소요됩니다.

2. 구축이 완료되면 Connect to Cluster를 클릭합니다. 새로 생성한 AKS 클러스터에 연결하려면 로컬 환경(랩톱 /PC)에서 Kubernetes 명령줄 도구를 설치하십시오. 를 방문하십시오 "도구를 설치합니다" OS에 따라 설치합니다.
3. "로컬 환경에 Azure CLI를 설치합니다".
4. 터미널에서 AKS 클러스터에 액세스하려면 먼저 'az login'을 입력하고 자격 증명을 입력하십시오.
5. 다음 두 명령을 실행합니다.

```
az account set --subscription xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx
aks get-credentials --resource-group resourcegroup --name aksclustername
```

6. Azure CLI에서 다음 명령을 입력합니다.

```
kubectl get nodes
```

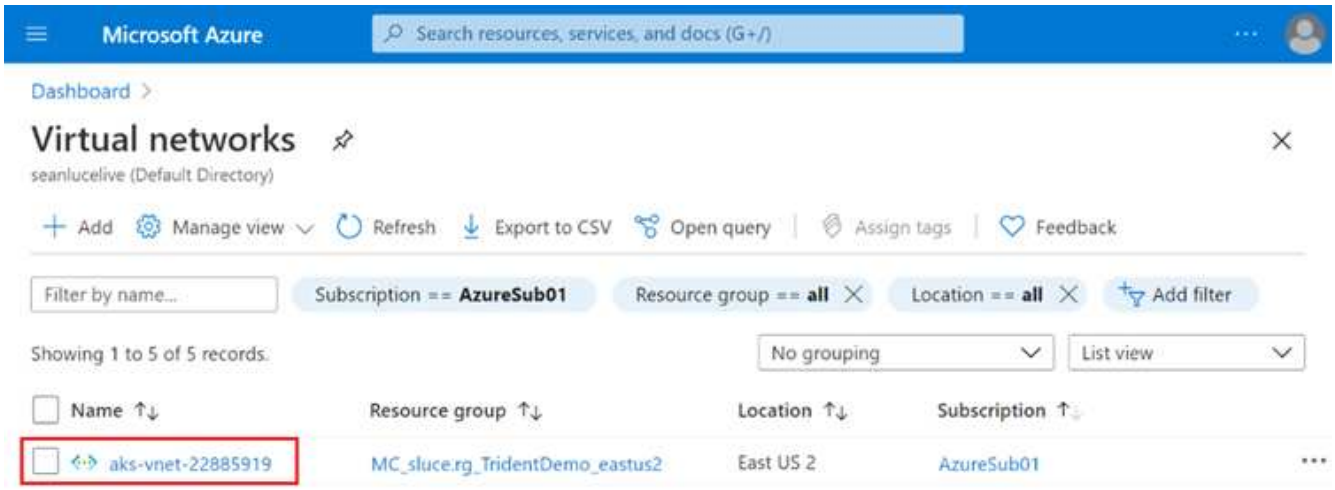
여기에 표시된 대로 6개 노드가 모두 가동되어 실행 중이면 AKS 클러스터가 로컬 환경에 연결되고 준비됩니다.

```
verronmartina@verron-mac-0 ~ % kubectl get nodes
NAME                                STATUS   ROLES    AGE   VERSION
aks-agentpool-34613062-vmss000000   Ready    agent    22m   v1.18.14
aks-agentpool-34613062-vmss000001   Ready    agent    22m   v1.18.14
aks-agentpool-34613062-vmss000002   Ready    agent    22m   v1.18.14
aks-gpupool-34613062-vmss000000     Ready    agent    20m   v1.18.14
aks-gpupool-34613062-vmss000001     Ready    agent    20m   v1.18.14
aks-gpupool-34613062-vmss000002     Ready    agent    20m   v1.18.14
verronmartina@verron-mac-0 ~ %
```

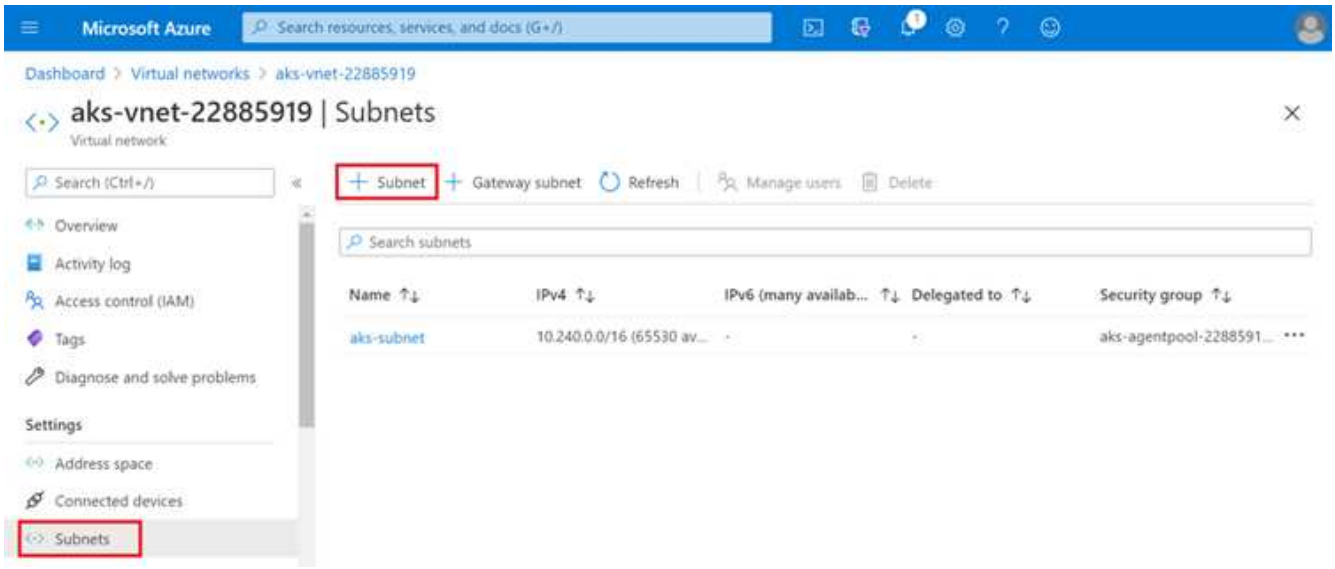
Azure NetApp Files에 대해 위임된 서브넷을 생성합니다

Azure NetApp Files에 대해 위임된 서브넷을 만들려면 다음 단계를 수행하십시오.

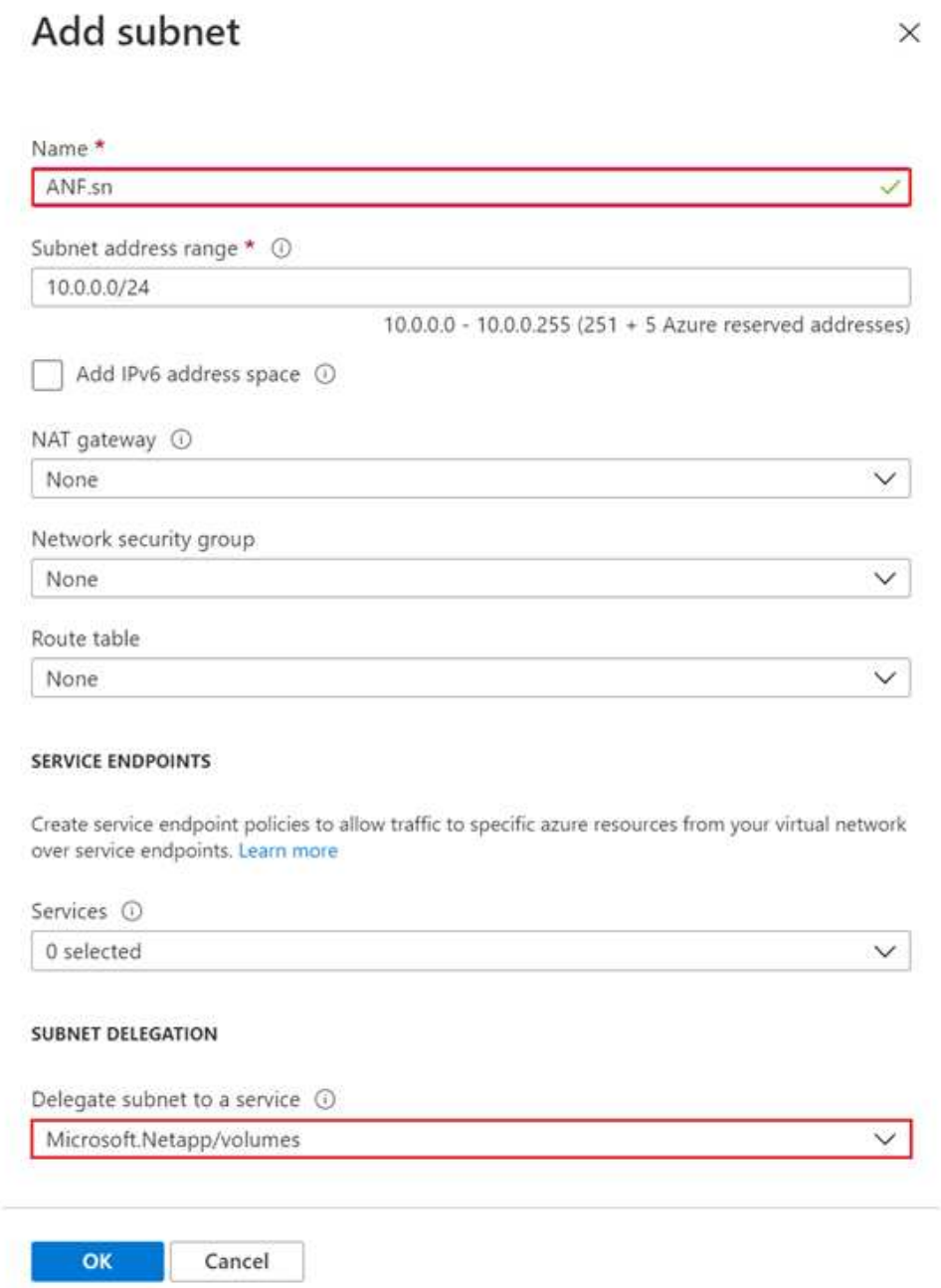
1. Azure 포털 내의 가상 네트워크로 이동합니다. 새로 생성한 가상 네트워크를 찾습니다. 여기에 표시된 대로 AKS-VNET와 같은 접두사가 있어야 합니다. 가상 네트워크의 이름을 클릭합니다.



2. 서브넷 을 클릭하고 상단 도구 모음에서 + 서브넷 을 선택합니다.



3. 서브넷에 ANF.SN과 같은 이름을 입력하고 Subnet Delegation 제목에서 Microsoft.NetApp/volumes 을 선택합니다. 다른 어떤 것도 변경하지 마십시오. 확인 을 클릭합니다.



Azure NetApp Files 볼륨은 애플리케이션 클러스터에 할당되며 Kubernetes에서 영구 볼륨 청구(PVC)로 사용됩니다. 또한, 이러한 할당을 통해 볼륨을 Jupyter 노트북, 서버리스 기능 등과 같은 다양한 서비스에 유연하게 매핑할 수 있습니다

서비스 사용자는 다양한 방법으로 플랫폼의 스토리지를 사용할 수 있습니다. Azure NetApp Files의 주요 이점은 다음과 같습니다.

- 사용자에게 스냅샷을 사용할 수 있는 기능을 제공합니다.
- 사용자가 Azure NetApp Files 볼륨에 대량의 데이터를 저장할 수 있도록 지원
- 대규모 파일 세트에서 모델을 실행할 때 Azure NetApp Files 볼륨의 성능 이점을 확보

Azure NetApp Files 설정

Azure NetApp Files 설정을 완료하려면 먼저 에 설명된 대로 구성해야 합니다 "QuickStart: Azure NetApp Files를 설정하고 NFS 볼륨을 생성합니다".

하지만 Trident를 통해 볼륨을 생성하므로 Azure NetApp Files용 NFS 볼륨을 생성하는 단계를 생략할 수 있습니다. 계속하기 전에 다음 사항을 확인하십시오.

1. "Azure NetApp Files 및 NetApp 리소스 공급자에 등록(Azure 클라우드 셸 이용)".
2. "Azure NetApp Files에서 계정을 생성했습니다".
3. "용량 풀을 설정합니다" (요구사항에 따라 최소 4TiB 표준 또는 프리미엄).

AKS 가상 네트워크 및 **Azure NetApp Files** 가상 네트워크 피어링

다음으로 다음 단계를 수행하여 Azure NetApp Files VNET를 사용하여 AKS 가상 네트워크(VNET)를 수행하십시오.

1. Azure 포털 맨 위의 검색 상자에 가상 네트워크를 입력합니다.
2. VNET AKS-VNET-NAME을 클릭한 다음 검색 필드에 Pebsearch를 입력합니다.
3. 추가 를 클릭하고 아래 표에 제공된 정보를 입력합니다.

필드에 입력합니다	값 또는 설명입니다
피어링 링크 이름	AKS-VNET-NAME_to_anf
SubscriptionID(하위 스크립트 ID)	피어링을 사용하는 Azure NetApp Files VNET의 구독
VNET 피어링 파트너	Azure NetApp Files VNET

> ⓘ 모든 별표 이외의 섹션은 기본적으로 그대로 둡니다

4. 추가 또는 확인 을 클릭하여 가상 네트워크에 피어링을 추가합니다.

자세한 내용은 를 참조하십시오 "가상 네트워크 피어링을 생성, 변경 또는 삭제합니다".

트라이던트

Trident는 NetApp에서 애플리케이션 컨테이너 영구 스토리지를 위해 유지하는 오픈 소스 프로젝트입니다. Trident는 Pod 자체로 실행되는 외부 공급자 컨트롤러로 구축되어 볼륨을 모니터링하고 프로비저닝 프로세스를 완전히 자동화했습니다.

NetApp Trident를 사용하면 교육 데이터 세트 및 교육 받은 모델을 저장하기 위한 영구 볼륨을 생성하여 K8s와 원활하게 통합할 수 있습니다. 이 기능을 사용하면 데이터 과학자와 데이터 엔지니어가 데이터 세트를 수동으로 저장하고 관리해야 하는 번거로움 없이 K8s를 더 쉽게 사용할 수 있습니다. 또한 Trident는 논리적 API 통합을 통해 데이터 관리 관련 작업을 통합하므로 데이터 과학자가 새로운 데이터 플랫폼 관리에 대해 배울 필요가 없습니다.

Trident를 설치합니다

Trident 소프트웨어를 설치하려면 다음 단계를 완료하십시오.

1. "첫 번째 설치 Helm".

2. Trident 21.01.1 설치 프로그램을 다운로드하고 압축을 풉니다.

```
wget
https://github.com/NetApp/trident/releases/download/v21.01.1/trident-
installer-21.01.1.tar.gz
tar -xf trident-installer-21.01.1.tar.gz
```

3. 디렉터리를 '트리덴트 - 설치자'로 변경합니다.

```
cd trident-installer
```

4. tridentctl을 시스템 '$path'의 디렉토리에 복사합니다

```
cp ./tridentctl /usr/local/bin
```

5. Helm을 사용하여 K8s 클러스터에 Trident 설치:
   a. 디렉터리를 Helm 디렉토리로 변경합니다.

   ```
   cd helm
   ```

   b. Trident를 설치합니다.

   ```
   helm install trident trident-operator-21.01.1.tgz --namespace trident
   --create-namespace
   ```

   c. Trident Pod의 상태를 확인합니다. 일반적인 K8s 방식:

   ```
   kubectl -n trident get pods
   ```

   d. 모든 Pod가 가동되어 실행 중이면 Trident가 설치되어 앞으로 이동하기에 좋습니다.

Azure NetApp Files 백엔드 및 스토리지 클래스 설정

Azure NetApp Files 백엔드 및 스토리지 클래스를 설정하려면 다음 단계를 수행하십시오.

1. 홈 디렉토리로 다시 전환합니다.

```
cd ~
```

2. 의 클론을 생성합니다 "프로젝트 리포지토리" 차선 감지 SCNN-horovod.
3. 트리덴트-구성 디렉토리로 이동합니다.

```
cd ./lane-detection-SCNN-horovod/trident-config
```

4. Azure 서비스 원칙 생성(서비스 원칙은 Trident가 Azure와 통신하여 Azure NetApp Files 리소스에 액세스하는 방법입니다.)

```
az ad sp create-for-rbac --name
```

출력은 다음 예와 같이 표시되어야 합니다.

```
{
  "appId": "xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
   "displayName": "netapptrident",
    "name": "http://netapptrident",
    "password": "xxxxxxxxxxxxxxx.xxxxxxxxxxxxxx",
    "tenant": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxx"
 }
```

5. Trident의 백엔드 json 파일을 생성합니다.
6. 원하는 텍스트 편집기를 사용하여 아래 표의 "anf-backend.json" 파일 안에 있는 다음 필드를 작성합니다.

필드에 입력합니다	값
구독 ID	Azure 구독 ID입니다
텐antID	Azure 테넌트 ID(이전 단계의 az ad SP 출력에서)
클라이언트 ID입니다	appID(이전 단계의 az ad SP 출력에서)
clientSecret	암호(이전 단계의 az ad SP 출력에서)

파일은 다음 예제와 같습니다.

```
{
    "version": 1,
    "storageDriverName": "azure-netapp-files",
    "subscriptionID": "fakec765-4774-fake-ae98-a721add4fake",
    "tenantID": "fakef836-edc1-fake-bff9-b2d865eefake",
    "clientID": "fake0f63-bf8e-fake-8076-8de91e57fake",
    "clientSecret": "SECRET",
    "location": "westeurope",
    "serviceLevel": "Standard",
    "virtualNetwork": "anf-vnet",
    "subnet": "default",
    "nfsMountOptions": "vers=3,proto=tcp",
    "limitVolumeSize": "500Gi",
    "defaults": {
    "exportRule": "0.0.0.0/0",
    "size": "200Gi"
}
```

7. 다음과 같이 구성 파일로 anf-backend.json을 사용하여 trident 네임스페이스에 Azure NetApp Files 백엔드를 생성하도록 Trident에 지시합니다.

```
tridentctl create backend -f anf-backend.json -n trident
```

8. 스토리지 클래스를 생성합니다.

   a. K8 사용자는 이름별로 저장소 클래스를 지정하는 PVC를 사용하여 체적을 프로비저닝합니다. K8s에게 다음을 사용하여 이전 단계에서 생성한 Azure NetApp Files 백엔드를 참조하는 스토리지 클래스 "azurenetappfiles"를 생성하도록 지시합니다.

   ```
   kubectl create -f anf-storage-class.yaml
   ```

   b. 다음 명령을 사용하여 스토리지 클래스가 생성되었는지 확인합니다.

   ```
   kubectl get sc azurenetappfiles
   ```

   출력은 다음 예와 같이 표시되어야 합니다.

```
NAME               PROVISIONER             RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
azurenetappfiles   csi.trident.netapp.io   Delete          Immediate           false                  98s
```

AKS에 볼륨 스냅샷 구성 요소를 구축하고 설정합니다

클러스터에서 올바른 볼륨 스냅샷 구성 요소가 사전 설치되지 않은 경우 다음 단계를 실행하여 이러한 구성 요소를 수동으로 설치할 수 있습니다.

AKS 1.18.14에는 Snapshot Controller가 사전 설치되어 있지 않습니다.

1. 다음 명령을 사용하여 스냅샷 베타 CRD를 설치합니다.

```
kubectl create -f https://raw.githubusercontent.com/kubernetes-
csi/external-snapshotter/release-
3.0/client/config/crd/snapshot.storage.k8s.io_volumesnapshotclasses.yaml
kubectl create -f https://raw.githubusercontent.com/kubernetes-
csi/external-snapshotter/release-
3.0/client/config/crd/snapshot.storage.k8s.io_volumesnapshotcontents.yam
l
kubectl create -f https://raw.githubusercontent.com/kubernetes-
csi/external-snapshotter/release-
3.0/client/config/crd/snapshot.storage.k8s.io_volumesnapshots.yaml
```

2. GitHub에서 다음 문서를 사용하여 Snapshot Controller를 설치합니다.

```
kubectl apply -f https://raw.githubusercontent.com/kubernetes-
csi/external-snapshotter/release-3.0/deploy/kubernetes/snapshot-
controller/rbac-snapshot-controller.yaml
kubectl apply -f https://raw.githubusercontent.com/kubernetes-
csi/external-snapshotter/release-3.0/deploy/kubernetes/snapshot-
controller/setup-snapshot-controller.yaml
```

3. K8s 'volumesnapshotclass'를 설정합니다. 볼륨 스냅샷을 생성하기 전에 "볼륨 스냅샷 클래스입니다" 설정해야 합니다. Azure NetApp Files용 볼륨 스냅샷 클래스를 생성하고 NetApp Snapshot 기술을 사용하여 ML 버전 관리를 달성하는 데 사용합니다. volumesapshotclass NetApp-CSI-snapclass를 생성하고 다음과 같이 기본 'volumesnapshotclass'로 설정합니다.

```
kubectl create -f netapp-volume-snapshot-class.yaml
```

출력은 다음 예와 같이 표시되어야 합니다.

```
volumesnapshotclass.snapshot.storage.k8s.io/netapp-csi-snapclass created
```

4. 다음 명령을 사용하여 볼륨 스냅샷 복사본 클래스가 생성되었는지 확인합니다.

```
kubectl get volumesnapshotclass
```

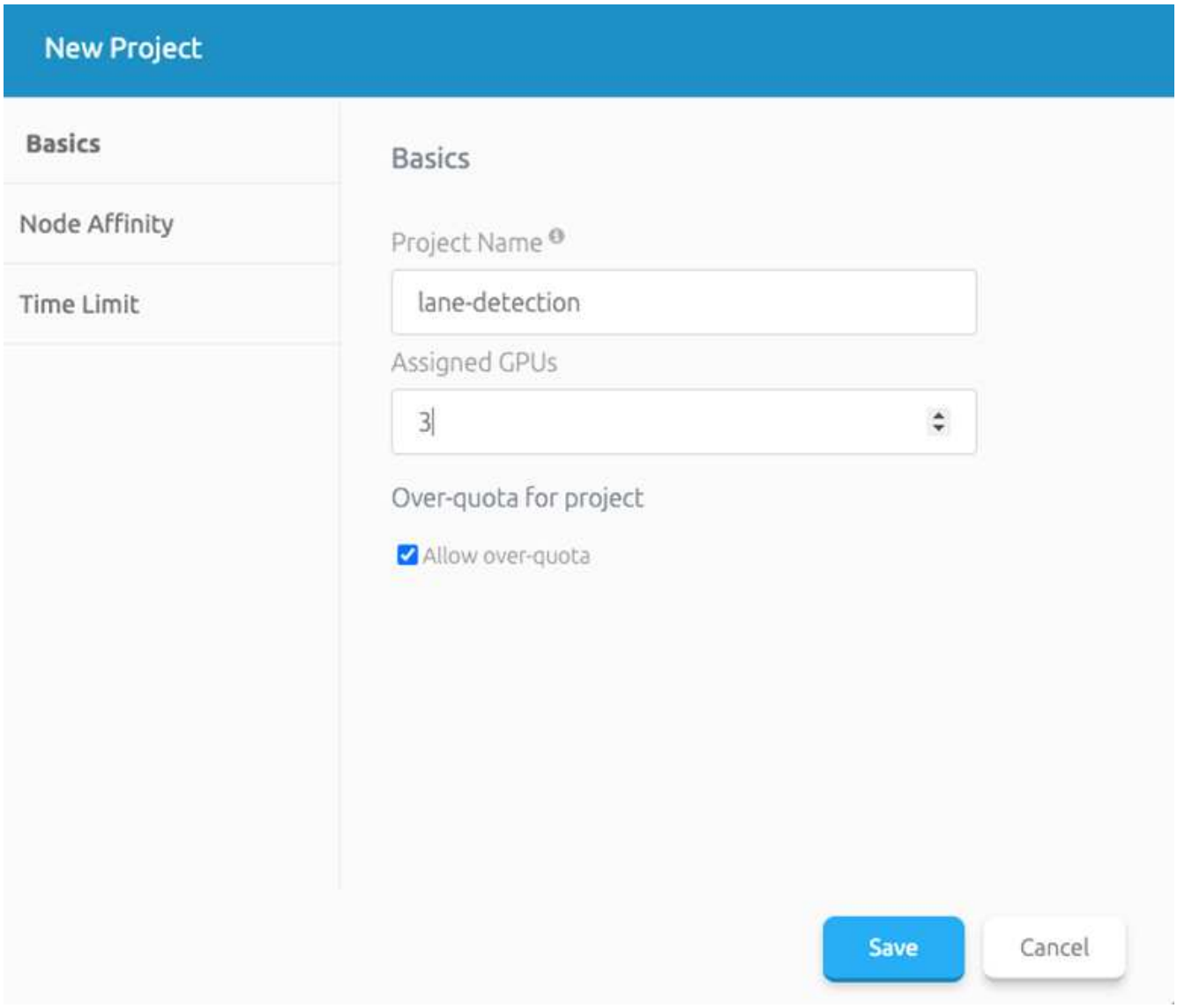
출력은 다음 예와 같이 표시되어야 합니다.

```
NAME                    DRIVER                  DELETIONPOLICY   AGE
netapp-csi-snapclass    csi.trident.netapp.io   Delete           63s
```

AI 설치 를 실행하십시오

run:AI를 설치하려면 다음 단계를 완료하십시오.

1. "설치 실행: AKS에 AI 클러스터".
2. app.runai.ai 으로 이동하여 새 프로젝트 만들기 를 클릭하고 이름을 차선 감지 로 지정합니다. 이렇게 하면 runai로 시작하는 K8s 클러스터의 이름 뒤에 프로젝트 이름이 붙습니다. 이 경우 생성된 네임스페이스는 runai-lane-detection입니다.



3. "설치 실행: AI CLI".
4. 터미널에서 다음 명령을 사용하여 레인 감지를 기본 run:AI 프로젝트로 설정합니다.

```
`runai config project lane-detection`
```

출력은 다음 예와 같이 표시되어야 합니다.

```
Project lane-detection has been set as default project
```

5. 프로젝트 네임스페이스(예: lane-detection)에 대해 ClusterRole 및 ClusterRoleBinding을 만들어 runai-lane-detection 네임스페이스에 속한 기본 서비스 계정은 작업 실행 중에 'volumesnapshot' 작업을 수행할 수 있는 권한을 갖습니다.

   a. 다음 명령을 사용하여 'runai-lane-detection'이 존재하는지 확인하기 위한 네임스페이스를 나열합니다.

   ```
   kubectl get namespaces
   ```

   출력은 다음 예와 같이 나타나야 합니다.

   ```
   NAME                    STATUS   AGE
   default                 Active   130m
   kube-node-lease         Active   130m
   kube-public             Active   130m
   kube-system             Active   130m
   runai                   Active   4m44s
   runai-lane-detection    Active   13s
   trident                 Active   102m
   ```

6. 다음 명령을 사용하여 ClusterRole의 "netaprosnapshot" 및 ClusterRoleBinding" netappsnapshot을 생성합니다.

   ```
   `kubectl create -f runai-project-snap-role.yaml`
   `kubectl create -f runai-project-snap-role-binding.yaml`
   ```

실행**:AI** 작업으로 **TuSimple** 데이터 세트를 다운로드하고 처리합니다

실행 시 TuSimple 데이터 세트를 다운로드하고 처리하는 프로세스는 선택 사항입니다. AI 작업은 선택 사항입니다. 여기에는 다음 단계가 포함됩니다.

1. 기존 Docker 이미지(예: muneer7589/download-tusimple:1.0)를 사용하려면 Docker 이미지를 빌드하고 푸시하거나 이 단계를 생략합니다

   a. 홈 디렉토리로 이동합니다.

   ```
   cd ~
   ```

   b. 'lane-detection-SCNN-horovod' 프로젝트의 데이터 디렉토리로 이동합니다.

```
cd ./lane-detection-SCNN-horovod/data
```

c. build_image.sh 쉘 스크립트를 수정하고 Docker 리포지토리를 사용자 위치로 변경합니다. 예를 들어, 'muneer7589'를 Docker 리포지토리 이름으로 바꿉니다. Docker 이미지 이름과 태그(예: dowload-tusimple, 1.0)를 변경할 수도 있습니다.

```
#!/bin/bash
#
# A simple script to build the Docker image.
#
# $ build_image.sh
set -ex

IMAGE=muneer7589/download-tusimple
TAG=1.0

# Build image
echo "Building image: "$IMAGE
docker build . -f Dockerfile \
 --tag "${IMAGE}:${TAG}"
echo "Finished building image: "$IMAGE

# Push image
echo "Pushing image: "$IMAGE
docker push "${IMAGE}:${TAG}"
echo "Finished pushing image: "$IMAGE
```

d. 스크립트를 실행하여 Docker 이미지를 구축하고 다음 명령을 사용하여 Docker 저장소로 푸시합니다.

```
chmod +x build_image.sh
./build_image.sh
```

2. Run:AI 작업을 제출하여 NetApp Trident가 동적으로 생성한 'PVC'에 TuSimple 레인 감지 데이터 세트를 다운로드, 추출, 전처리 및 저장합니다.

a. 다음 명령을 사용하여 run:AI 작업을 제출하십시오.

```
runai submit
--name download-tusimple-data
--pvc azurenetappfiles:100Gi:/mnt
--image muneer7589/download-tusimple:1.0
```

b. 실행:AI 작업을 제출하려면 아래 표의 정보를 입력하십시오.

필드에 입력합니다	값 또는 설명입니다
-이름	작업의 이름입니다
-PVC	[StorageClassName]:Size:ContainerMountPath 형식의 PVC 위의 작업 제출에서 스토리지 클래스 azurenetappfiles가 있는 Trident를 사용하여 필요 시 PVC를 만듭니다. 여기서 영구 볼륨 용량은 100Gi 이며 경로 /mnt에 마운트됩니다.
?곡깊	이 작업에 대한 컨테이너를 생성할 때 사용할 Docker 이미지입니다

출력은 다음 예와 같이 표시되어야 합니다.

```
The job 'download-tusimple-data' has been submitted successfully
You can run `runai describe job download-tusimple-data -p lane-detection` to check the job status
```

c. 제출된 RUN:AI 작업을 나열합니다.

```
runai list jobs
```

```
Showing jobs for project lane-detection
NAME                  STATUS            AGE  NODE                          IMAGE                          TYPE  PROJECT        USER          GPUs Allocated (Requested)
PODs Running (Pending)  SERVICE URL(S)
download-tusimple-data  ContainerCreating  1m   aks-agentpool-34613062-vmss00000a  muneer7589/download-tusimple:1.0  Train  lane-detection  verronmartina  0 (0)
1 (0)
```

d. 제출된 작업 로그를 확인하십시오.

```
runai logs download-tusimple-data -t 10
```

```
751150K .......... .......... .......... .......... ..........  6% 16.2M 20m37s
751200K .......... .......... .......... .......... ..........  6% 11.1M 20m37s
751250K .......... .......... .......... .......... ..........  6% 12.5M 20m36s
751300K .......... .......... .......... .......... ..........  6% 11.3M 20m36s
751350K .......... .......... .......... .......... ..........  6% 15.2M 20m36s
751400K .......... .......... .......... .......... ..........  6% 10.5M 20m36s
751450K .......... .......... .......... .......... ..........  6% 15.2M 20m36s
751500K .......... .......... .......... .......... ..........  6% 14.1M 20m36s
751550K .......... .......... .......... .......... ..........  6% 24.3M 20m36s
751600K .......... .......... .......... .......... ..........  6% 26.3M 20m36s
```

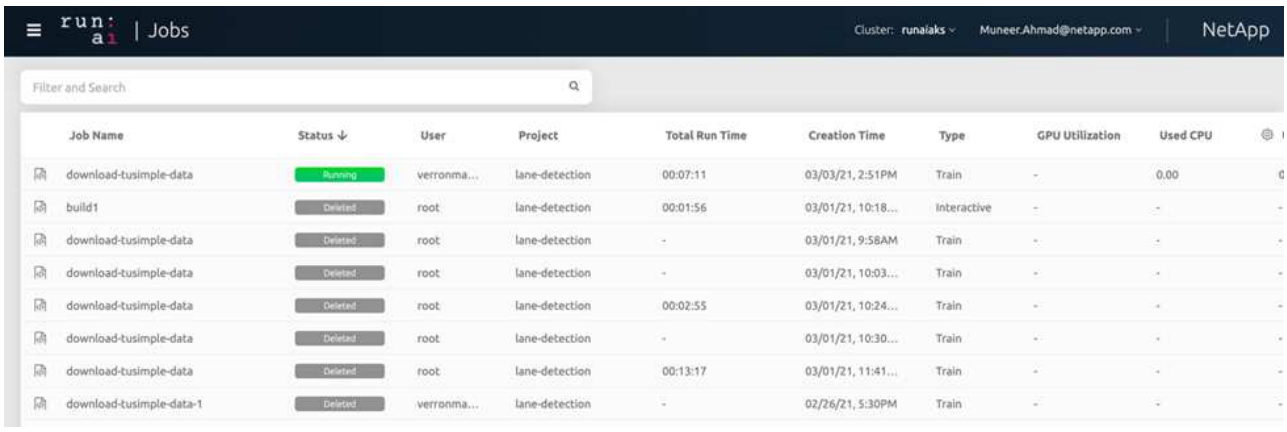
e. 만든 PVC를 나열합니다. 다음 단계에서 이 'PVC' 명령을 사용하여 훈련하십시오.

```
kubectl get pvc | grep download-tusimple-data
```

출력은 다음 예와 같이 표시되어야 합니다.

```
pvc-download-tusimple-data-0   Bound    pvc-bb03b74d-2c17-40c4-a445-79f3de8d16d5   100Gi      RWO            azurenetappfiles   4m47s
```

a. 실행 중인 작업 확인: AI UI (또는 'app.run.ai`).



Horovod를 사용하여 분산 차선 감지 교육을 수행합니다

Horovod를 사용하여 분산 차선 감지 교육을 수행하는 것은 선택적 프로세스입니다. 그러나 다음과 같은 단계가 있습니다.

1. 기존 Docker 이미지(예: 'muneer7589/dist-lane-detection: 3.1):'를 사용하려면 Docker 이미지를 빌드하고 푸시하거나 이 단계를 건너뜁니다

   a. 홈 디렉토리로 이동합니다.

   ```
   cd ~
   ```

   b. 프로젝트 디렉터리 레인 감지 SCNN-horovod로 이동합니다

   ```
   cd ./lane-detection-SCNN-horovod
   ```

   c. 'build_image.sh' 쉘 스크립트를 수정하고 Docker 리포지토리를 사용자 이름으로 변경합니다(예: 'muneer7589'를 Docker 리포지토리 이름으로 대체). Docker 이미지 이름과 태그(dist-lane-detection, 3.1 등)도 변경할 수 있습니다.

```
#!/bin/bash
#
# A simple script to build the distributed Docker image.
#
# $ build_image.sh
set -ex

IMAGE=muneer7589/dist-lane-detection
TAG=3.0

# Build image
echo "Building image: "$IMAGE
docker build . -f Dockerfile \
 --tag "${IMAGE}:${TAG}"
echo "Finished building image: "$IMAGE

# Push image
echo "Pushing image: "$IMAGE
docker push "${IMAGE}:${TAG}"
echo "Finished pushing image: "$IMAGE
```

d. 스크립트를 실행하여 Docker 이미지를 구축하고 Docker 저장소로 이동합니다.

```
chmod +x build_image.sh
./build_image.sh
```

2. 배포 교육(MPI)을 수행하기 위한 AI 작업 제출:

a. 러닝 제출 사용: 이전 단계에서 PVC를 자동으로 생성하기 위한 AI(데이터 다운로드용)만 RWO 액세스를 허용할 수 있습니다. 이 경우 여러 Pod 또는 노드가 동일한 PVC에 대한 분산 교육 액세스를 허용하지 않습니다. 액세스 모드를 ReadWriteMany로 업데이트하고 Kubernetes 패치를 사용하여 업데이트합니다.

b. 먼저 다음 명령을 실행하여 PVC의 볼륨 이름을 가져옵니다.

```
kubectl get pvc | grep download-tusimple-data
```

```
root@ai-w-gpu-2:/mnt/ai_data/anf_runai/lane-detection-SCNN-horovod# kubectl get pvc | grep download-tusimple-data
pvc-download-tusimple-data-0   Bound    pvc-bb03b74d-2c17-40c4-a445-79f3de8d16d5   100Gi      RWX            azurenetappfiles   2d4h
```

c. 볼륨을 패치하고 ReadWriteMany에 대한 액세스 모드를 업데이트합니다(다음 명령에서 볼륨 이름을 사용자 이름으로 바꾸기).

```
kubectl patch pv pvc-bb03b74d-2c17-40c4-a445-79f3de8d16d5 -p
'{"spec":{"accessModes":["ReadWriteMany"]}}'
```

d. 아래 표의 정보를 사용하여 배포된 교육 작업을 실행하기 위한 AI MPI 작업 제출:

```
runai submit-mpi
--name dist-lane-detection-training
--large-shm
--processes=3
--gpu 1
--pvc pvc-download-tusimple-data-0:/mnt
--image muneer7589/dist-lane-detection:3.1
-e USE_WORKERS="true"
-e NUM_WORKERS=4
-e BATCH_SIZE=33
-e USE_VAL="false"
-e VAL_BATCH_SIZE=99
-e ENABLE_SNAPSHOT="true"
-e PVC_NAME="pvc-download-tusimple-data-0"
```

필드에 입력합니다	값 또는 설명입니다
이름	분산된 교육 작업의 이름입니다
대형 shm	대용량 /dev/shm 디바이스 마운트 RAM에 마운트된 공유 파일 시스템이며 여러 CPU 작업자가 CPU RAM에 배치를 처리 및 로드할 수 있을 만큼 충분한 크기의 공유 메모리를 제공합니다.
프로세스	분산된 교육 프로세스 수
GPU	이 작업에서 작업에 할당할 GPU/프로세스 수, GPU 작업자 프로세스 3개(--프로세스=3)가 있으며, 각각 단일 GPU(--GPU 1)로 할당됩니다.
PVC	이전 작업(download-tusimple-data-0)에서 생성한 기존 영구 볼륨(PVC-download-tusimple-data-0)을 사용하고 path /mnt에 마운트됩니다
이미지	이 작업에 대한 컨테이너를 생성할 때 사용할 Docker 이미지입니다
컨테이너에 설정할 환경 변수를 정의합니다	
작업자 사용	인수를 true로 설정하면 다중 프로세스 데이터 로드가 설정됩니다
작업자 수	데이터 로더 작업자 프로세스의 수입니다
batch_size를 선택합니다	교육 배치 크기
VAL을 사용합니다	인수를 TRUE로 설정하면 유효성 검사가 허용됩니다
Val_batch_size를 선택합니다	검증 배치 크기
snapshot을 설정합니다	인수를 TRUE로 설정하면 ML 버전 관리를 위해 데이터 및 훈련된 모델 스냅샷을 생성할 수 있습니다

필드에 입력합니다	값 또는 설명입니다
PVC_이름	스냅샷을 생성할 PVC의 이름입니다. 위의 작업 제출에서 데이터 세트 및 교육 모델로 구성된 PVC-download-tusimple-data-0의 스냅샷을 촬영하고 있습니다

출력은 다음 예와 같이 표시되어야 합니다.

```
The job 'dist-lane-detection-training' has been submitted successfully
You can run `runai describe job dist-lane-detection-training -p lane-detection` to check the job status
```

e. 제출된 작업을 나열합니다.

```
runai list jobs
```

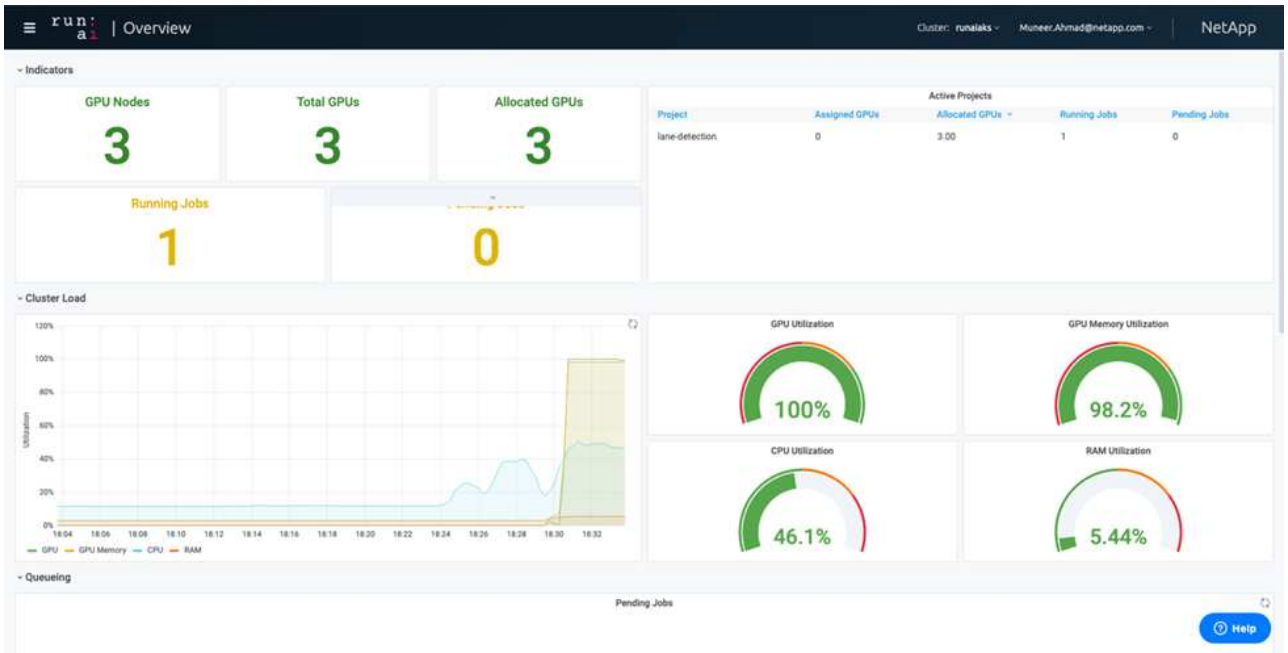
```
NAME                          STATUS    AGE  NODE        IMAGE                                TYPE  PROJECT        USER          GPUs Allocated (Requested)  PODs
 SERVICE URL(S)
download-tusimple-data        Succeeded 1d               muneer7589/download-tusimple:1.0     Train lane-detection verronmartina  - (0)                       0 (0)

dist-lane-detection-training  Init:0/1  2m   <multiple>  muneer7589/dist-lane-detection:3.1   Train lane-detection root          3 (3)                       4 (0)
```

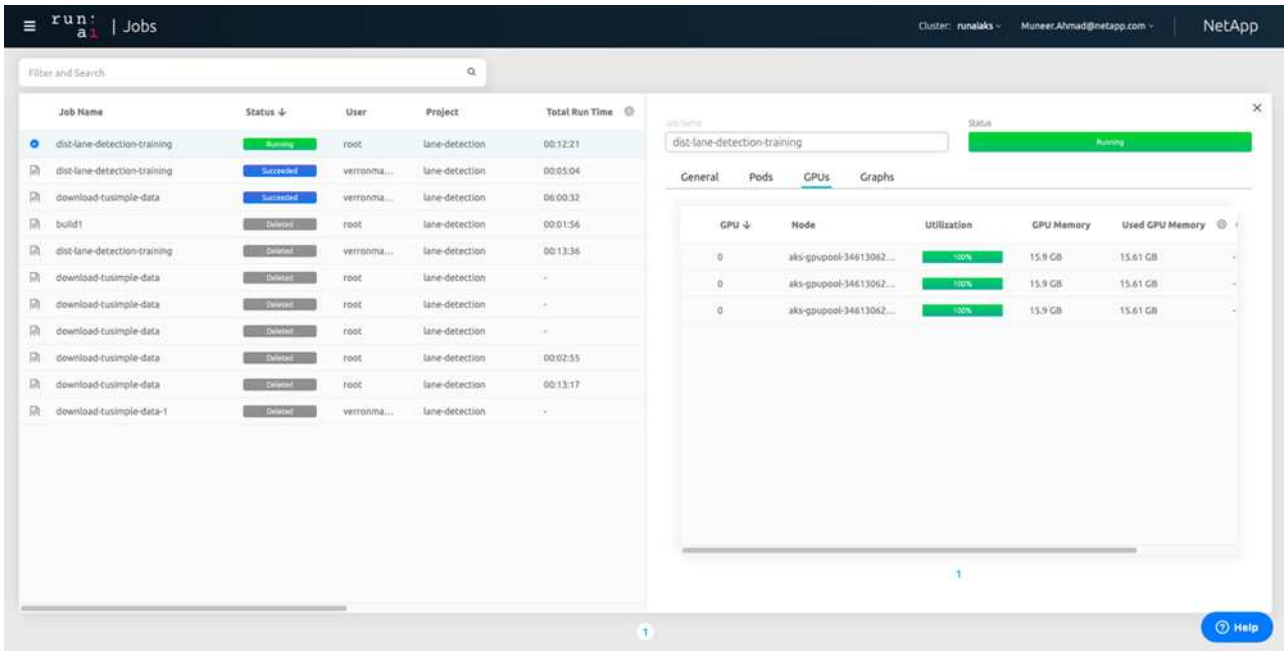
f. 제출된 작업 로그:

```
runai logs dist-lane-detection-training
```

```
root@ai-w-gpu-2:~/runai# runai logs dist-lane-detection-training
Running with 3 workers
2021-03-04 17:29:23.158449: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dynamic library libcudart.so.10.1
+ POD_NAME=dist-lane-detection-training-worker-0
+ [ d = - ]
+ shift
+ /opt/kube/kubectl cp /opt/kube/hosts dist-lane-detection-training-worker-0:/etc/hosts_of_nodes
+ POD_NAME=dist-lane-detection-training-worker-2
+ [ d = - ]
+ shift
+ /opt/kube/kubectl cp /opt/kube/hosts dist-lane-detection-training-worker-2:/etc/hosts_of_nodes
+ POD_NAME=dist-lane-detection-training-worker-1
```

g. 아래 그림과 같이 RUN TO/RUN TO/AI GUI(또는 app.runai.ai): RUN:AI 대시보드 에서 교육 작업을 확인하십시오. 첫 번째 그림에서는 분산 훈련 작업에 할당된 3개의 GPU를 AKS의 3개 노드에 분산시키고, 두 번째 실행인 AI 작업에 대해 자세히 설명합니다.





h. 교육이 완료되면 RUN:AI 작업과 연결되고 생성된 NetApp Snapshot 복사본이 있는지 확인하십시오.

```
runai logs dist-lane-detection-training --tail 1
```

```
[1,0]<stdout>:Snapshot snap-pvc-download-tusimple-data-0-dist-lane-detection-training-launcher-2021-03-05-16-23-42 created in namespace runai-lane-detection
```

```
kubectl get volumesnapshots | grep download-tusimple-data-0
```

NetApp 스냅샷 복사본에서 데이터를 복원합니다

NetApp Snapshot 복사본에서 데이터를 복원하려면 다음 단계를 수행하십시오.

1. 홈 디렉토리로 이동합니다.

```
cd ~
```

2. 프로젝트 디렉터리 'lane-detection-SCNN-horovod'로 이동합니다.

```
cd ./lane-detection-SCNN-horovod
```

3. restore-snaphot-vc.yaML을 수정하고 데이터 복원을 원하는 스냅샷 사본으로 dataSource의 이름 필드를 업데이트합니다. 이 예제에서는 데이터 복원 위치를 PVC 이름으로 변경할 수도 있습니다.

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: restored-tusimple
spec:
  storageClassName: azurenetappfiles
  dataSource:
    name: snap-pvc-download-tusimple-data-0-dist-lane-detection-training-launcher-2021-03-05-16-23-42
    kind: VolumeSnapshot
    apiGroup: snapshot.storage.k8s.io
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 100Gi
```

4. restore-snapshot-pvc.yAML을 사용하여 새로운 PVC를 생성한다.

```
kubectl create -f restore-snapshot-pvc.yaml
```

출력은 다음 예와 같이 표시되어야 합니다.

```
persistentvolumeclaim/restored-tusimple created
```

5. 방금 복원한 데이터를 교육에 사용하려는 경우, 작업 제출은 이전과 동일하게 유지되며, 교육 작업을 제출할 때 다음 명령에 표시된 것처럼 'PVC_NAME'만 복원된 'PVC_NAME'으로 교체합니다.

```
runai submit-mpi
--name dist-lane-detection-training
--large-shm
--processes=3
--gpu 1
--pvc restored-tusimple:/mnt
--image muneer7589/dist-lane-detection:3.1
-e USE_WORKERS="true"
-e NUM_WORKERS=4
-e BATCH_SIZE=33
-e USE_VAL="false"
-e VAL_BATCH_SIZE=99
-e ENABLE_SNAPSHOT="true"
-e PVC_NAME="restored-tusimple"
```

성능 평가

솔루션의 선형 확장성을 보여주기 위해 GPU 1개와 GPU 3개 등 두 가지 시나리오에서 성능 테스트를 수행했습니다. TuSimple 레인 감지 데이터 세트에 대한 교육 중에 GPU 할당, GPU 및 메모리 사용률, 다양한 단일 및 3노드 메트릭이 캡처되었습니다. 교육 프로세스 중 리소스 활용도를 분석하기 위해 데이터가 5배 증가합니다.

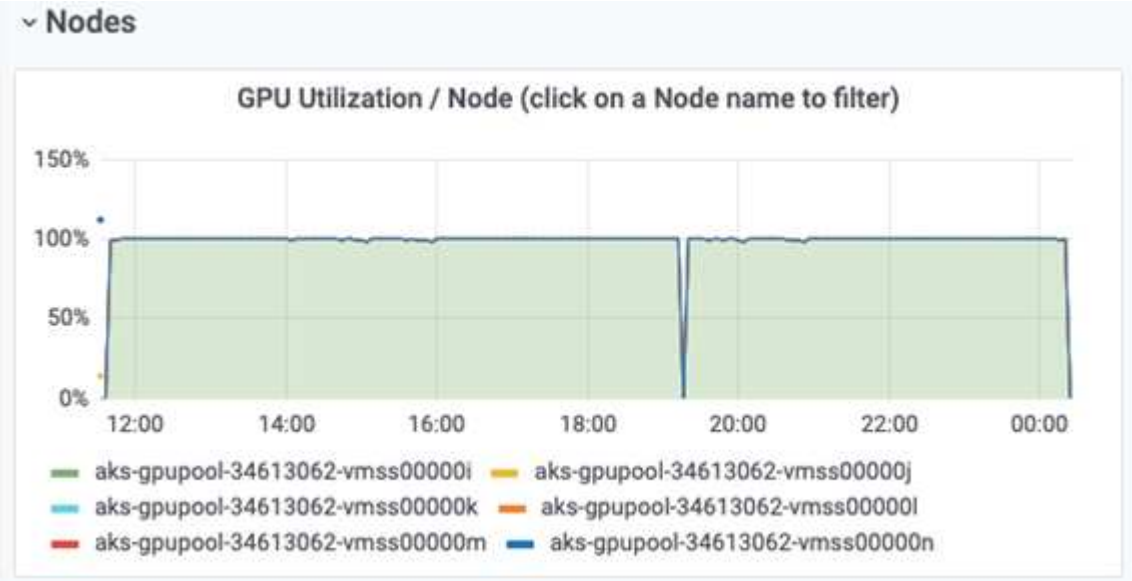
이 솔루션을 통해 고객은 작은 데이터 세트와 몇 개의 GPU로 시작할 수 있습니다. 데이터의 양과 GPU 수요가 증가하면 고객은 표준 계층의 테라바이트를 동적으로 확장하고 프리미엄 계층까지 신속하게 확장하여 데이터 이동 없이 테라바이트당 처리량의 4배를 얻을 수 있습니다. 이 프로세스는 섹션, "Azure NetApp Files 서비스 레벨".

GPU 1개의 처리 시간은 12시간 45분이었습니다. 3개 노드에서 3개의 GPU를 처리하는 데 약 4시간 30분이 소요되었습니다.

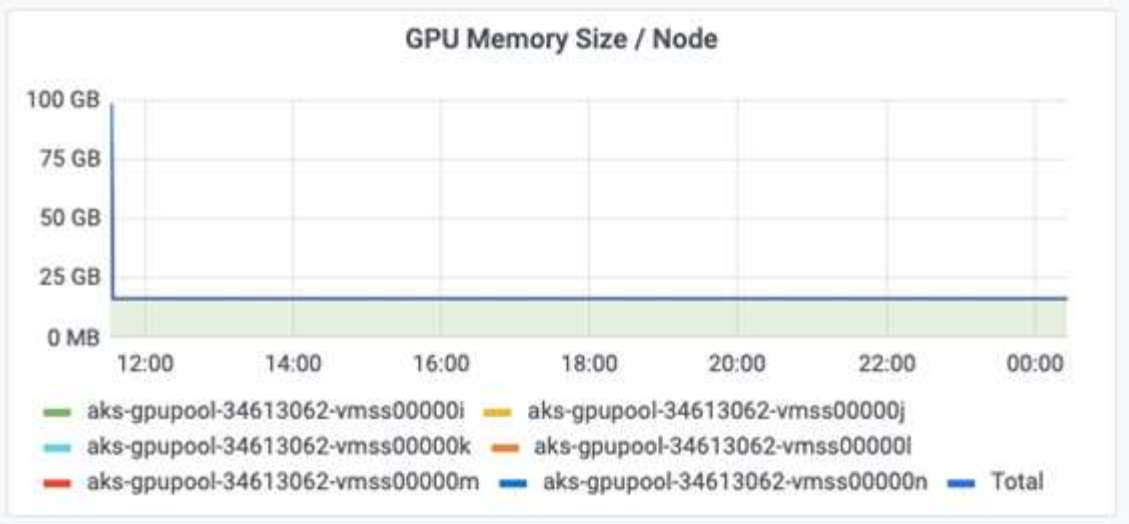
이 문서의 나머지 부분에서는 개별 비즈니스 요구 사항에 따른 성능 및 확장성의 예를 보여 줍니다.

아래 그림은 1 GPU 할당 및 메모리 활용률을 보여 줍니다.

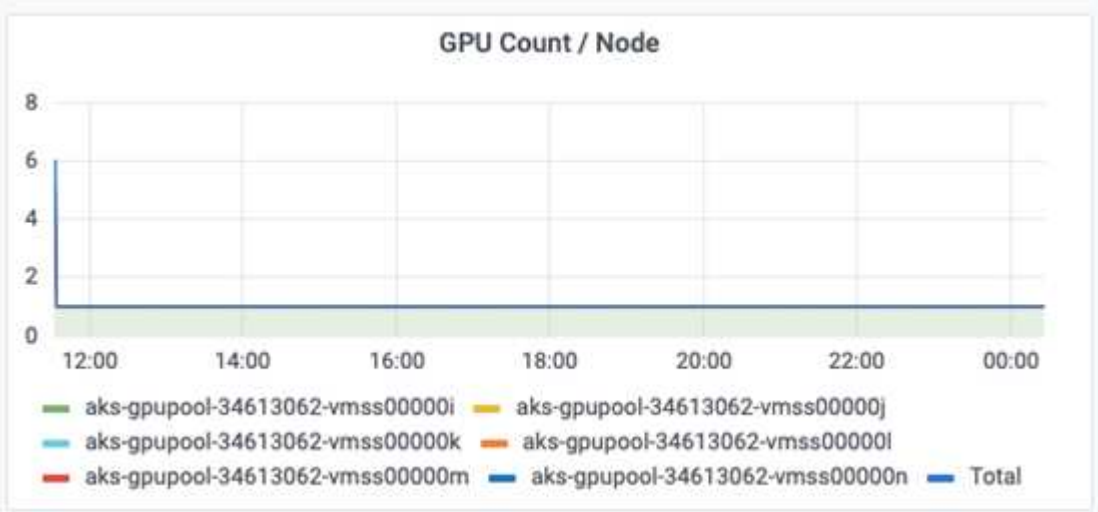
아래 그림은 단일 노드 GPU 활용률을 보여 줍니다.



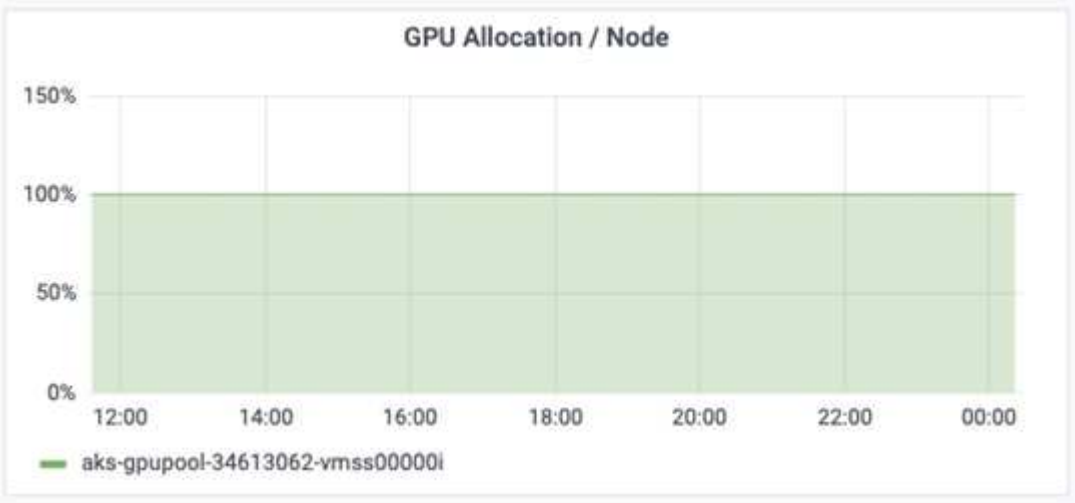
아래 그림은 단일 노드 메모리 크기(16GB)를 보여줍니다.



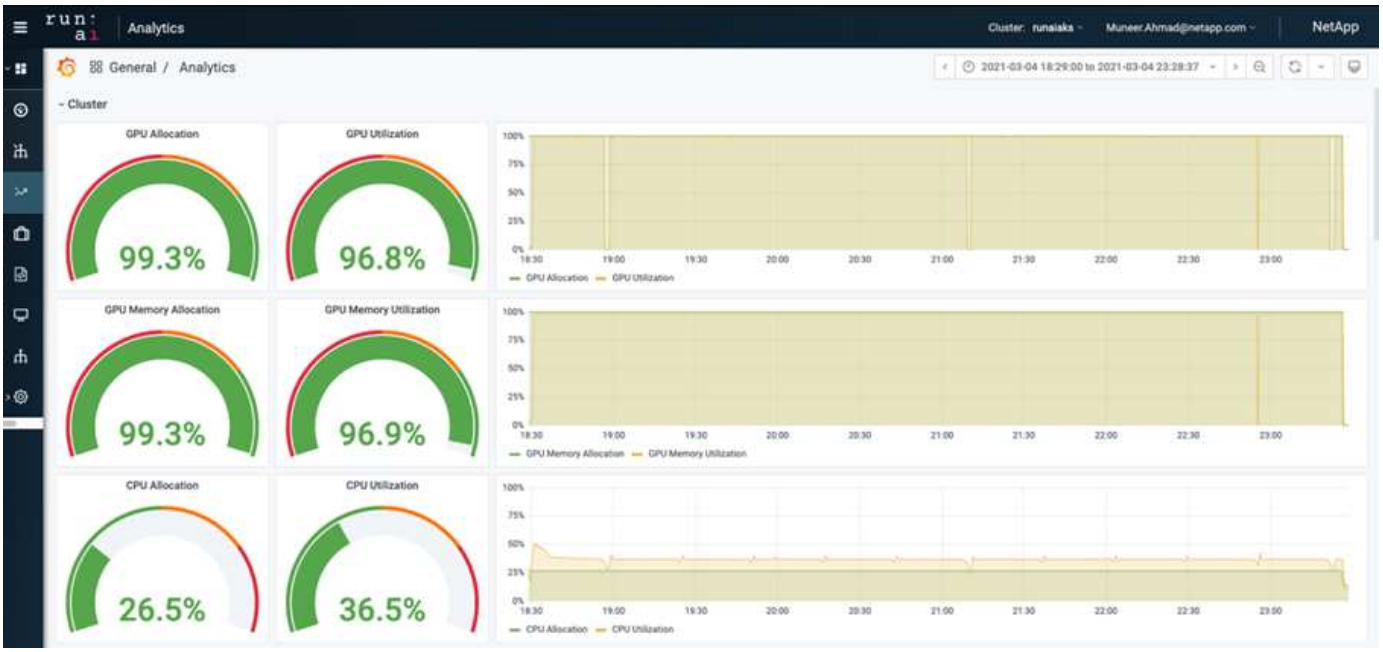
아래 그림은 단일 노드 GPU 수(1)를 보여줍니다.



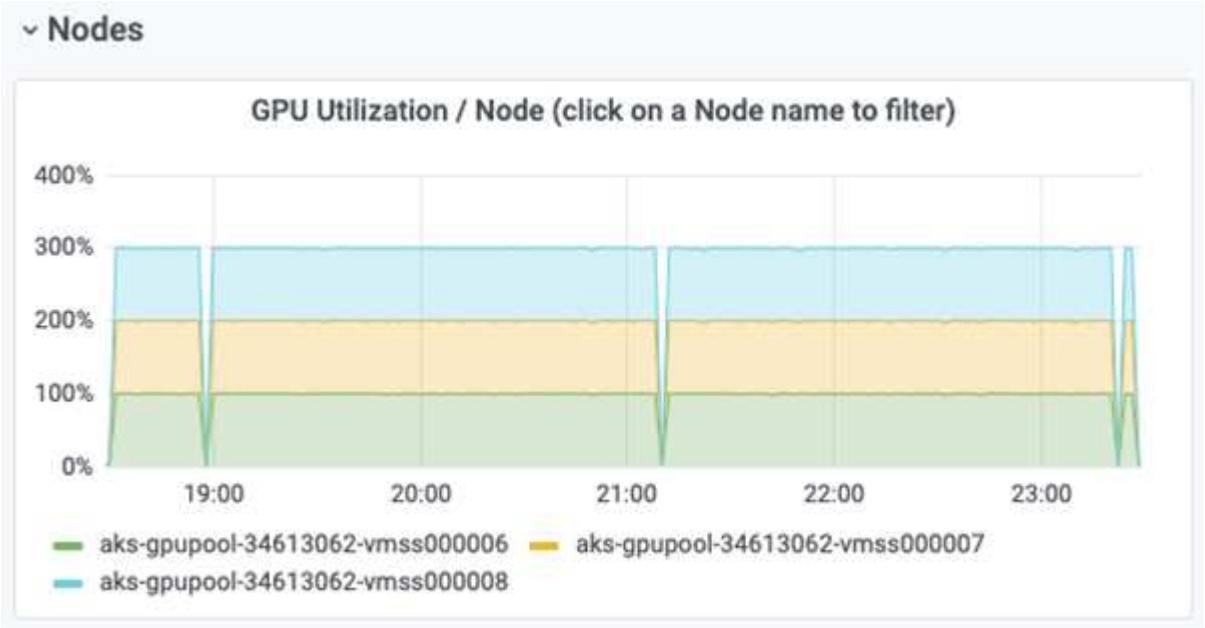
아래 그림은 단일 노드 GPU 할당(%)을 보여줍니다.



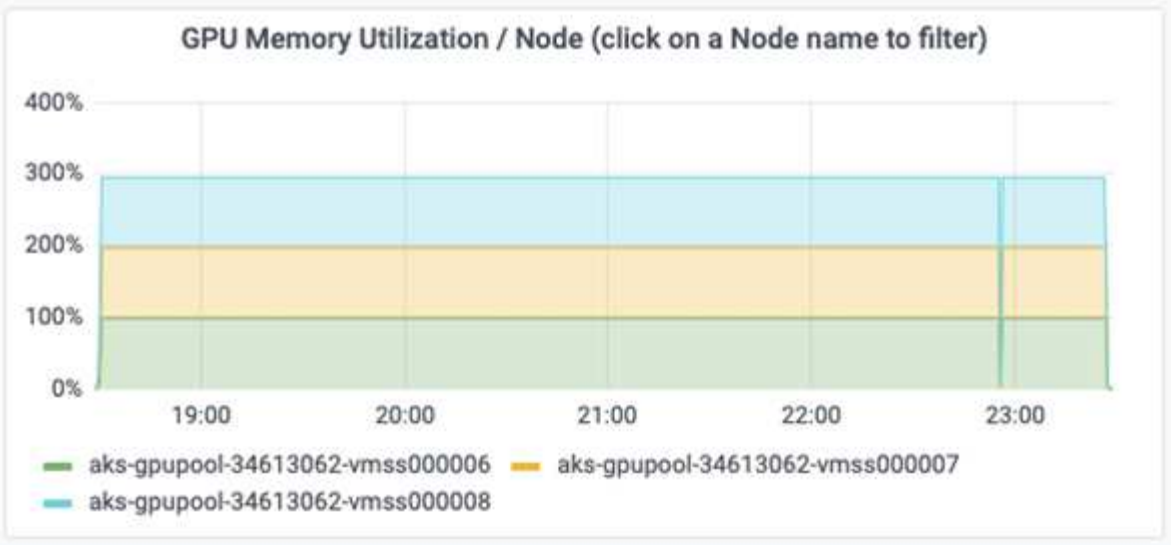
아래 그림은 3개 노드에서 GPU 할당 및 메모리인 3개의 GPU를 보여줍니다.



아래 그림은 3개 노드의 사용률(%)에서 3개의 GPU를 보여줍니다.



아래 그림은 3개 노드의 메모리 사용률(%)에서 3개의 GPU를 보여줍니다.



Azure NetApp Files 서비스 레벨

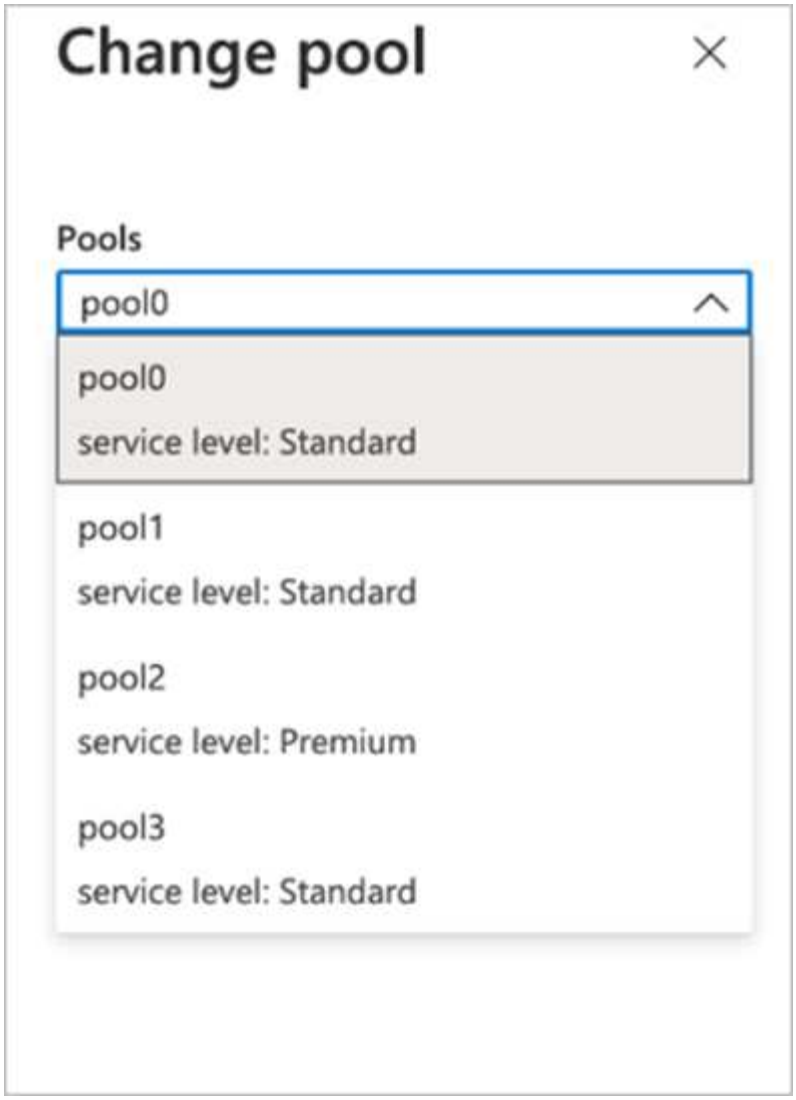
볼륨을 를 사용하는 다른 용량 풀로 이동하여 기존 볼륨의 서비스 수준을 변경할 수 있습니다 "서비스 레벨" 볼륨에 대한 을 선택합니다. 볼륨에 대한 이 기존 서비스 수준 변경 사항은 데이터를 마이그레이션할 필요가 없습니다. 볼륨에 대한 액세스에도 영향을 주지 않습니다.

볼륨의 서비스 수준을 동적으로 변경합니다

볼륨의 서비스 수준을 변경하려면 다음 단계를 수행하십시오.

1. 볼륨 페이지에서 서비스 수준을 변경할 볼륨을 마우스 오른쪽 단추로 클릭합니다. 풀 변경 을 선택합니다.

2. Change Pool 창에서 볼륨을 이동할 용량 풀을 선택합니다. 그런 다음 확인을 클릭합니다.



서비스 수준 변경 자동화

동적 서비스 수준 변경은 현재 공개 미리 보기에 있지만 기본적으로 활성화되어 있지 않습니다. Azure 구독에서 이 기능을 활성화하려면 " 문서에 제공된 다음 단계를 수행하십시오 "볼륨의 서비스 수준을 동적으로 변경합니다"."

- Azure:CLI에 대해 다음 명령을 사용할 수도 있습니다. Azure NetApp Files의 풀 크기 변경에 대한 자세한 내용은 를 참조하십시오 "AZ NetApp 파일 볼륨: ANF(Azure NetApp Files) 볼륨 리소스 관리".

```
az netappfiles volume pool-change -g mygroup
--account-name myaccname
-pool-name mypoolname
--name myvolname
--new-pool-resource-id mynewresourceid
```

- 여기에 표시된 'et-aznetapfilesvolumepool' cmdlet은 Azure NetApp Files 볼륨의 풀을 변경할 수 있습니다. 볼륨 풀 크기 및 Azure PowerShell 변경에 대한 자세한 내용은 을 참조하십시오 "Azure NetApp Files 볼륨의 풀을 변경합니다".

```
Set-AzNetAppFilesVolumePool
-ResourceGroupName "MyRG"
-AccountName "MyAnfAccount"
-PoolName "MyAnfPool"
-Name "MyAnfVolume"
-NewPoolResourceId 7d6e4069-6c78-6c61-7bf6-c60968e45fbf
```

결론

NetApp과 RUN: AI는 AI 워크로드 오케스트레이션을 단순화하기 위한 RUN:AI 플랫폼과 함께 Azure NetApp Files의 고유한 기능을 시연하기 위해 이 기술 보고서를 작성하는 데 협력했습니다. 이 기술 보고서는 분산 차선 감지 교육을 위한 데이터 파이프라인 및 워크로드 오케스트레이션 프로세스의 간소화를 위한 참조 아키텍처를 제공합니다.

결론적으로, 규모에 따른 분산 교육(특히 퍼블릭 클라우드 환경)과 관련하여 리소스 오케스트레이션 및 스토리지 구성요소를 솔루션의 중요한 요소로 간주하게 됩니다. 데이터를 관리하여 여러 GPU 처리를 방해해서는 안 되므로 GPU 사이클을 최적으로 활용할 수 있습니다. 따라서 대규모 분산 교육 용도로 시스템을 최대한 비용 효율적으로 만들 수 있습니다.

NetApp에서 제공하는 Data Fabric을 사용하면 데이터 과학자와 데이터 엔지니어가 사내 및 클라우드에 연결하여 수동 개입 없이 동기식 데이터를 사용할 수 있으므로 문제를 극복할 수 있습니다. 다시 말해, Data Fabric은 여러 위치에 분산되어 있는 AI 워크플로우를 관리하는 프로세스를 원활하게 처리합니다. 또한, 데이터를 컴퓨팅 가까이에 두고 분석, 교육, 검증을 필요할 때 언제 어디서나 수행하여 수요 기반 데이터 가용성을 지원합니다. 이 기능을 사용하면 데이터 통합뿐만 아니라 전체 데이터 파이프라인의 보호 및 보안을 실현할 수 있습니다.

추가 정보

이 문서에 설명된 정보에 대해 자세히 알아보려면 다음 문서 및/또는 웹 사이트를 검토하십시오.

- 데이터 세트: TuSimple

  "https://github.com/TuSimple/tusimple-benchmark/tree/master/doc/lane_detection"

- 딥 러닝 네트워크 아키텍처: 공간 컨볼루ional 신경망

  "https://arxiv.org/abs/1712.06080"

- 분산형 딥 러닝 교육 프레임워크: Horovod

  "https://horovod.ai/"

- 실행: AI 컨테이너 오케스트레이션 솔루션: 실행: AI 제품 소개

  "https://docs.run.ai/home/components/"

- AI 설치 설명서를 실행하십시오

  "https://docs.run.ai/Administrator/Cluster-Setup/cluster-install/#step-3-install-runai"
  "https://docs.run.ai/Administrator/Researcher-Setup/cli-install/#runai-cli-installation"

- 실행 중인 작업 제출: AI CLI

  "https://docs.run.ai/Researcher/cli-reference/runai-submit/"

  "https://docs.run.ai/Researcher/cli-reference/runai-submit-mpi/"

- Azure 클라우드 리소스: Azure NetApp Files

  "https://docs.microsoft.com/azure/azure-netapp-files/"

- Azure Kubernetes 서비스

  "https://azure.microsoft.com/services/kubernetes-service/-features"

- Azure VM SKU

  "https://azure.microsoft.com/services/virtual-machines/"

- GPU SKU가 포함된 Azure VM

  "https://docs.microsoft.com/azure/virtual-machines/sizes-gpu"

- NetApp 트라이던트

  "https://github.com/NetApp/trident/releases"

- NetApp 구현 Data Fabric

  "https://www.netapp.com/data-fabric/what-is-data-fabric/"

- NetApp 제품 설명서

  "https://www.netapp.com/support-and-training/documentation/"

TR-4841: 데이터 캐싱을 지원하는 하이브리드 클라우드 AI 운영 체제

Rick Huang, David Arnette, NetApp Yochay Ettun, cnvrg.io

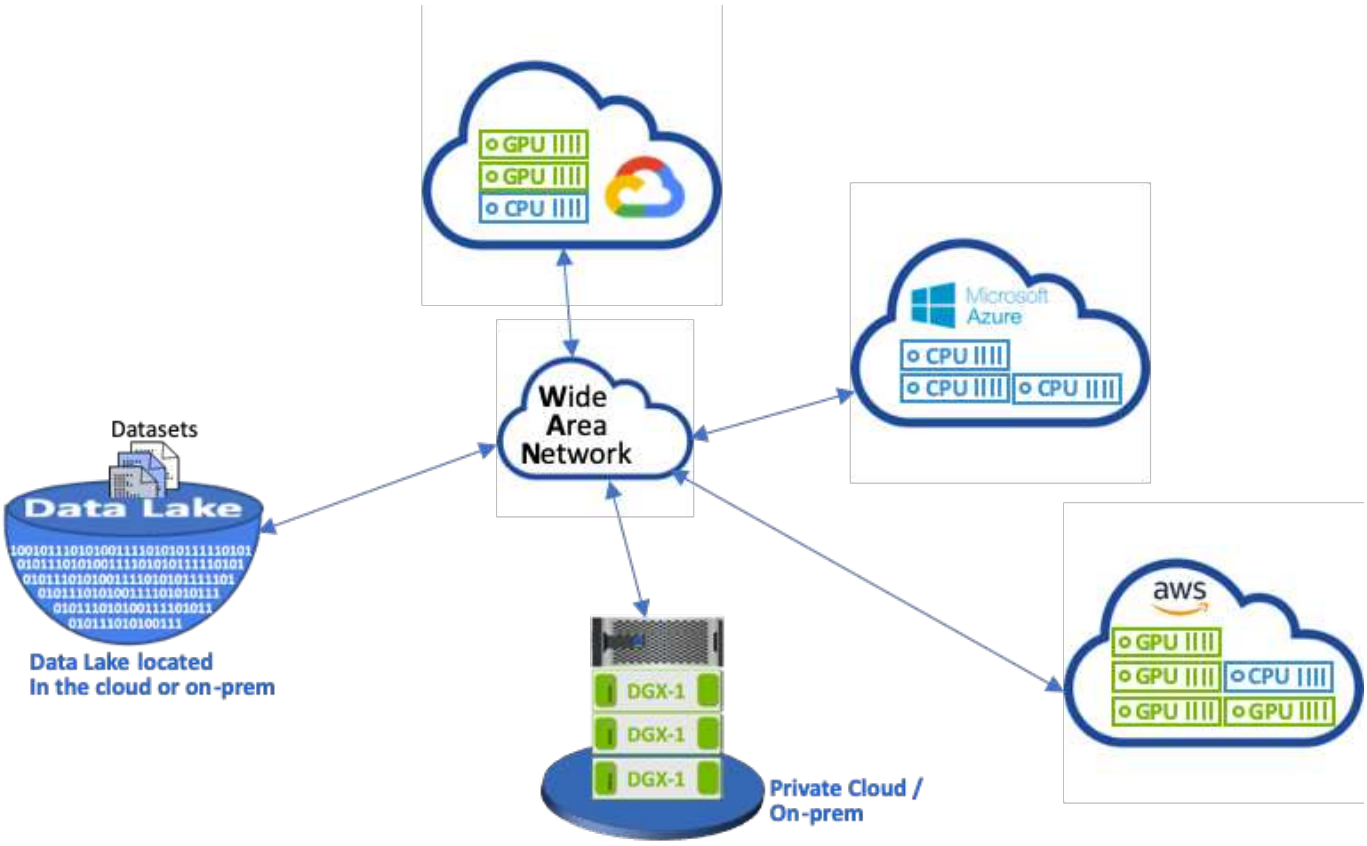
데이터의 폭발적인 증가와 ML 및 AI의 기하급수적인 성장으로 인해 고유한 개발 및 구현 과제를 가진 제타바이트 경제성이 창출되었습니다.

머신 러닝 모델은 데이터를 많이 필요로 하며 컴퓨팅 리소스 가까이에 고성능 데이터 스토리지가 필요한 것으로 널리 알려져 있지만, 실제로 하이브리드 클라우드 및 탄력적인 컴퓨팅 인스턴스 구축을 위해 이러한 모델을 구현하는 것은 그리 간단하지 않습니다. 일반적으로 대량의 데이터가 GPU와 같은 고성능 AI 컴퓨팅 리소스가 효율적으로 액세스할 수 없는 저비용 데이터 레이크에 저장됩니다. 일부 워크로드가 클라우드에서 작동하고 일부는 사내 또는 다른 HPC 환경에 완전히 있는 하이브리드 클라우드 인프라에서는 이 문제가 더욱 가중됩니다.

이 문서에서는 IT 전문가와 데이터 엔지니어가 토폴로지 인식 데이터 허브로 진정한 하이브리드 클라우드 AI 플랫폼을 구축하여 데이터 과학자가 컴퓨팅 리소스 가까이에 있는 데이터 세트의 캐시를 즉시 자동으로 생성할 수 있는 새로운 솔루션을 소개합니다. 있습니다. 그 결과, 고성능 모델 훈련을 수행할 수 있을 뿐만 아니라 데이터 세트 버전 허브 내에서 데이터 세트 캐시, 버전 및 계모델에 즉시 액세스할 수 있는 여러 AI 전문가의 협업을 비롯한 추가 이점을 얻을 수 있습니다.

사용 사례 개요 및 문제 설명

데이터 세트 및 데이터 세트 버전은 일반적으로 비용을 줄이고 기타 운영 이점을 제공하는 NetApp StorageGRID 오브젝트 기반 스토리지와 같은 데이터 레이크에 있습니다. 데이터 과학자는 이러한 데이터 세트를 가져와 다양한 단계로 엔지니어링하여 특정 모델을 사용하여 교육 준비를 합니다. 종종 여러 버전을 만들어냅니다. 다음 단계로 데이터 과학자는 모델을 실행하기 위해 최적화된 컴퓨팅 리소스(GPU, 하이엔드 CPU 인스턴스, 온프레미스 클러스터 등)를 선택해야 합니다. 다음 그림에서는 ML 컴퓨팅 환경에서 데이터 세트의 근접 위치 부족을 보여줍니다.



하지만 다양한 컴퓨팅 환경에서 여러 개의 교육 실험을 병렬로 실행해야 합니다. 각 환경에서는 데이터 레이크에서 데이터 세트를 다운로드해야 하며, 이 프로세스는 비용과 시간이 많이 소요됩니다. 데이터 세트와 컴퓨팅 환경(특히 하이브리드 클라우드의 경우)의 근접성이 보장되지는 않습니다. 또한 동일한 데이터 세트를 사용하여 자체 실험을 수행하는 다른 팀 구성원은 동일한 극한 용도의 프로세스를 거쳐야 합니다. 분명한 느린 데이터 액세스 외에도 데이터 세트 버전 추적, 데이터 세트 공유, 협업 및 재현성의 어려움 등의 문제가 있습니다.

고객 요구 사항

리소스를 효율적으로 사용하면서 고성능 ML 실행을 구현하기 위해 고객 요구사항이 달라질 수 있습니다. 예를 들어, 고객은 다음과 같은 요구사항을 충족해야 할 수 있습니다.

- 많은 비용이 드는 다운로드 및 데이터 액세스 복잡성을 발생시키지 않으면서 교육 모델을 실행하는 각 컴퓨팅 인스턴스에서 데이터 세트에 빠르게 액세스할 수 있습니다
- 데이터 세트의 위치에 관계없이 클라우드 또는 온프레미스에서 컴퓨팅 인스턴스(GPU 또는 CPU)를 사용합니다
- 불필요한 지연 시간 및 데이터 지연 시간 없이 동일한 데이터 세트에서 여러 컴퓨팅 리소스와 동시에 여러 교육

실험을 실행하여 효율성 및 생산성 향상

- 컴퓨팅 인스턴스 비용 최소화
- 데이터 세트, 계열, 버전 및 기타 메타데이터 세부 정보를 기록할 수 있는 도구를 통해 재현성이 향상되었습니다
- 공유 및 협업이 향상되어 권한이 있는 팀원 중 한 명이 데이터 세트에 액세스하여 실험을 실행할 수 있습니다

NetApp ONTAP 데이터 관리 소프트웨어를 사용하여 데이터 세트 캐싱을 구축하려면 다음과 같은 작업을 수행해야 합니다.

- 컴퓨팅 리소스에 가장 가까운 NFS 스토리지를 구성하고 설정합니다.
- 캐시할 데이터 세트 및 버전을 결정합니다.
- 캐시된 데이터 세트에 커밋된 총 메모리 용량과 추가 캐시 커밋에 사용할 수 있는 NFS 스토리지 용량(예: 캐시 관리)을 모니터링합니다.
- 특정 시간에 사용하지 않은 데이터 세트가 캐시에서 노후화되었습니다. 기본값은 1일입니다. 다른 구성 옵션을 사용할 수 있습니다.

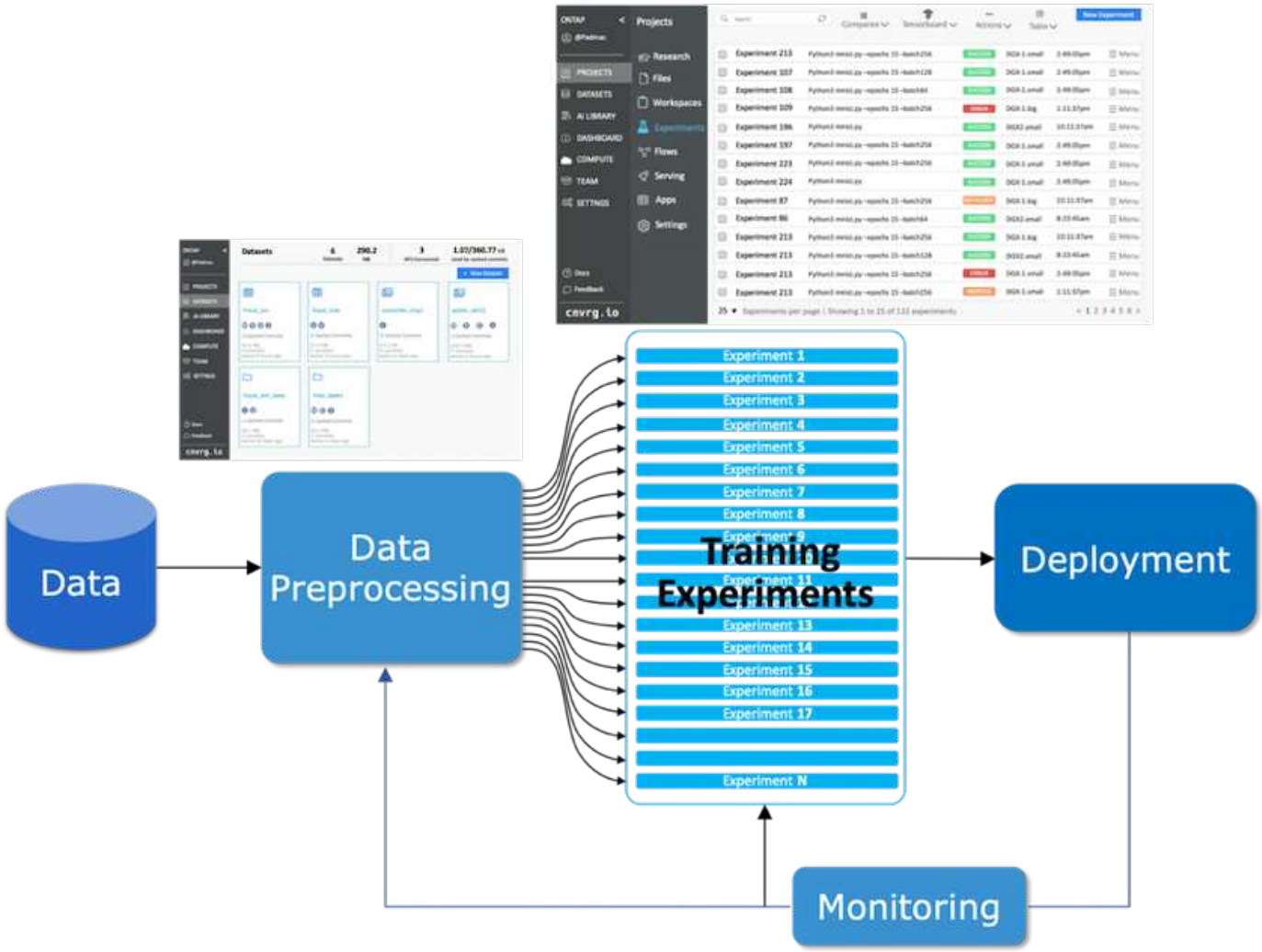
솔루션 개요

이 섹션에서는 기존의 데이터 과학 파이프라인과 그 단점을 검토합니다. 또한, 제안된 데이터 세트 캐싱 솔루션의 아키텍처도 제공합니다.

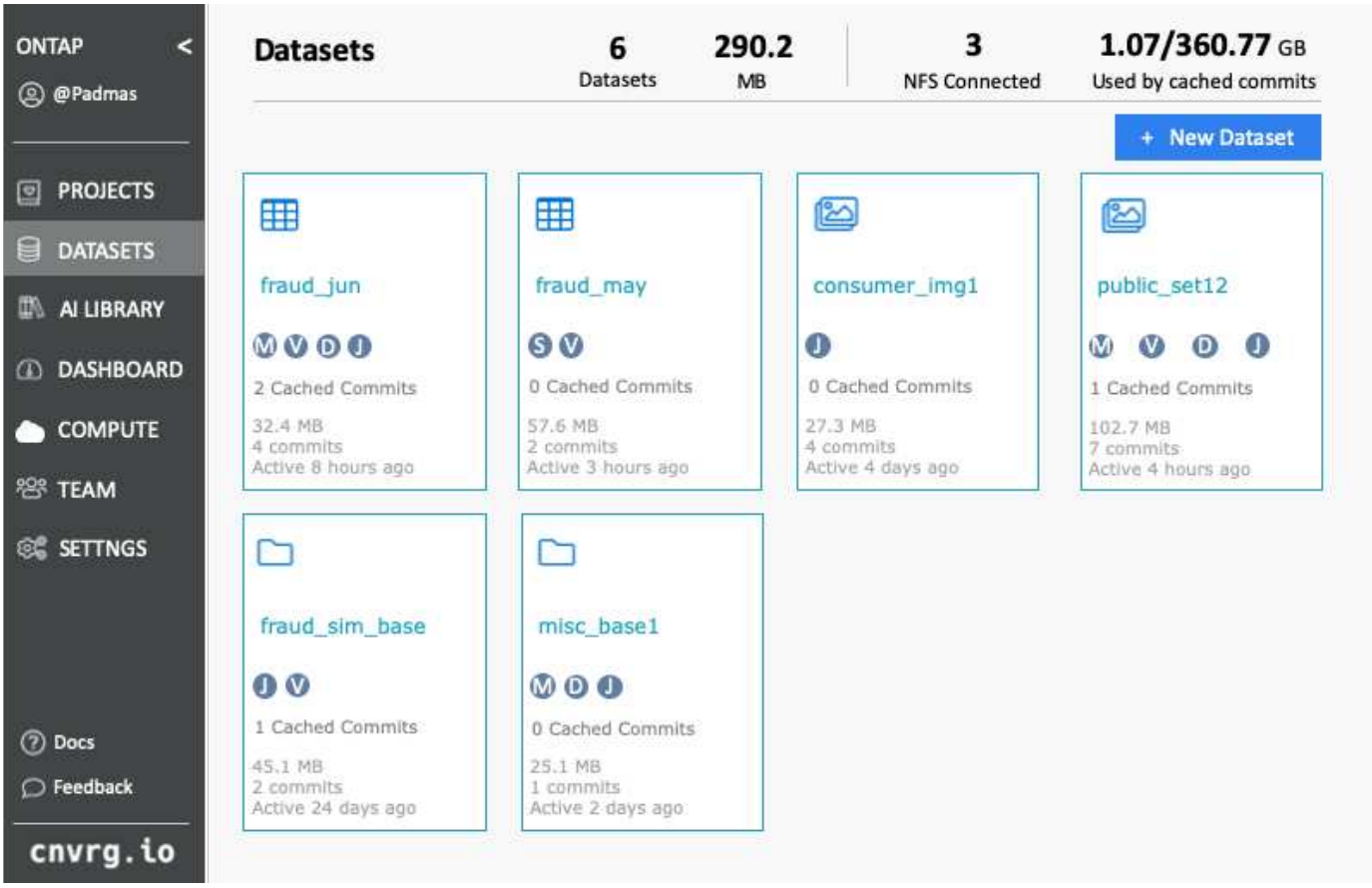
기존의 데이터 과학 파이프라인 및 결점

ML 모델 개발 및 배포의 일반적인 시퀀스에는 다음을 포함하는 반복 단계가 포함됩니다.

- 데이터 수집 중
- 데이터 사전 처리(여러 버전의 데이터 세트 생성)
- 하이퍼파라미터 최적화, 다른 모델 등과 관련된 여러 실험 실행
- 구축
- Monitoringcnvrg.io는 연구부터 배포에 이르는 모든 작업을 자동화하는 포괄적인 플랫폼을 개발했습니다. 다음 그림에서는 파이프라인과 관련된 대시보드 스크린샷의 작은 샘플을 보여 줍니다.



퍼블릭 저장소 및 프라이빗 데이터에서 여러 데이터 세트를 재생하는 것이 일반적입니다. 또한 각 데이터 세트에는 데이터 세트 정리 또는 기능 엔지니어링으로 인해 여러 버전이 있을 수 있습니다. 다음 그림과 같이 팀에서 공동 작업 및 일관성 도구를 사용할 수 있도록 데이터 세트 허브와 버전 허브를 제공하는 대시보드가 필요합니다.



파이프라인의 다음 단계에서는 각각 데이터 세트 및 특정 컴퓨팅 인스턴스와 관련된 교육 모델의 여러 병렬 인스턴스가 필요합니다. 특정 컴퓨팅 인스턴스를 사용하여 특정 실험으로 데이터 세트를 바인딩하는 것은 쉽지 않습니다. AWS(Amazon Web Services)의 GPU 인스턴스에서 일부 실험을 수행하는 동시에, DGX-1 또는 DGX-2 온프레미스 인스턴스에서 다른 실험을 수행할 수 있기 때문입니다. GCP의 CPU 서버에서 다른 실험을 실행할 수도 있지만 데이터 세트 위치가 교육을 수행하는 컴퓨팅 리소스 가까이에 있지 않습니다. 데이터 세트 스토리지에서 컴퓨팅 인스턴스까지 지연 시간이 짧은 10GbE 또는 더 많은 연결이 끊어지려면 어느 정도의 근접성이 있어야 합니다.

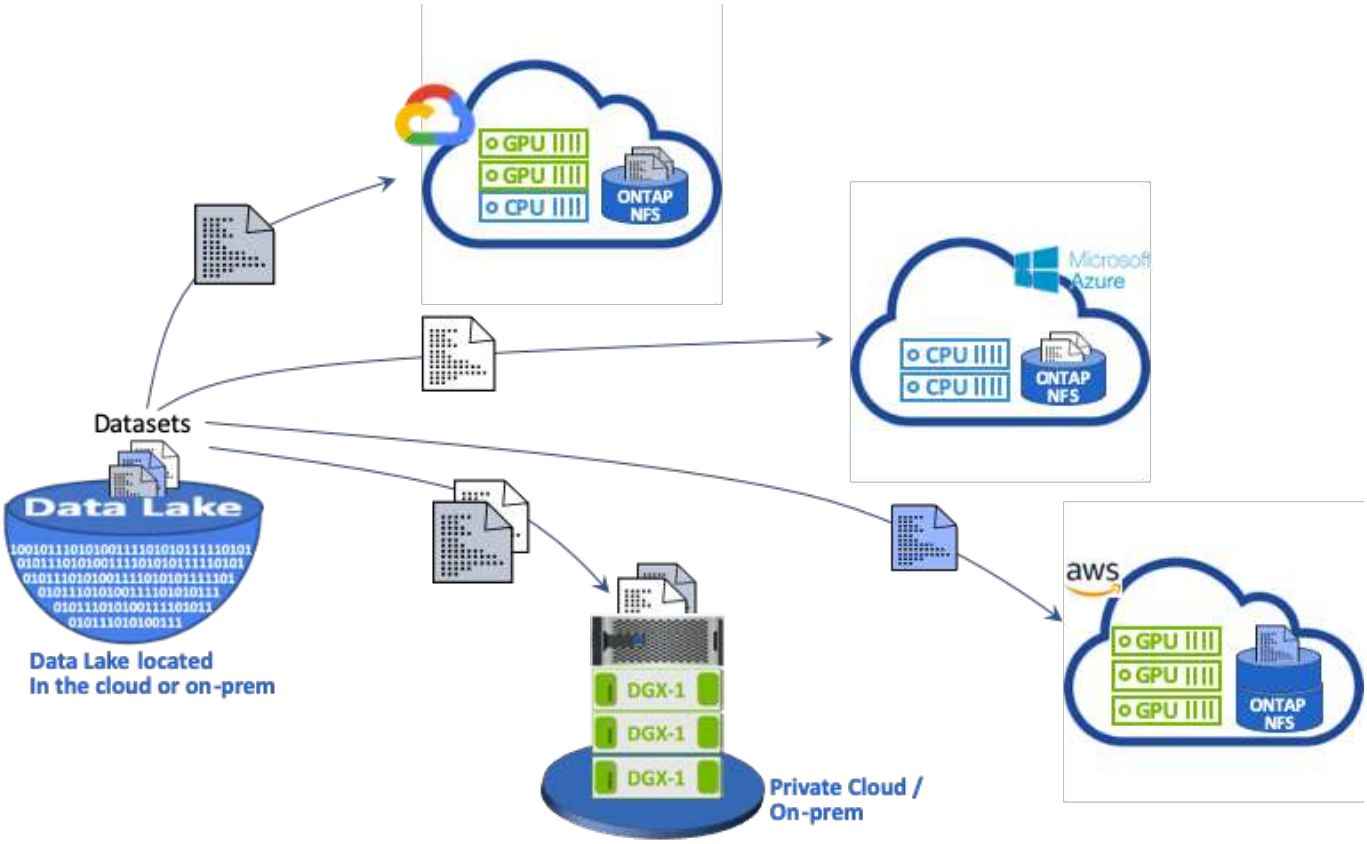
데이터 과학자는 훈련을 수행하고 실험을 실행하는 컴퓨팅 인스턴스에 데이터 세트를 다운로드하는 것이 일반적입니다. 그러나 이 접근 방식에는 몇 가지 잠재적 문제가 있습니다.

- 데이터 과학자가 데이터 세트를 컴퓨팅 인스턴스로 다운로드할 때 통합 컴퓨팅 스토리지가 고성능을 보장하는 것은 아닙니다(고성능 시스템의 예로는 ONTAP AFF A800 NVMe 솔루션이 있음).
- 다운로드한 데이터 세트가 하나의 컴퓨팅 노드에 상주하면 NetApp ONTAP 고성능 분산 스토리지와 달리 여러 노드에서 분산 모델을 실행하면 스토리지 병목 현상이 발생할 수 있습니다.
- 대기열 충돌 또는 우선순위 때문에 훈련 실험의 다음 반복을 다른 컴퓨팅 인스턴스에서 수행할 수 있으며, 데이터 세트에서 컴퓨팅 위치까지의 거리가 크게 멀어지거나
- 동일한 컴퓨팅 클러스터에서 교육 실험을 실행하는 다른 팀 구성원은 이 데이터 세트를 공유할 수 없으며, 각 팀원이 임의의 위치에서 데이터 세트의 (값비싼) 다운로드를 수행합니다.
- 후속 훈련 작업에 동일한 데이터 세트의 다른 데이터 세트 또는 버전이 필요한 경우 데이터 과학자는 training.NetApp 및 cnvrg.io를 수행하는 컴퓨팅 인스턴스에 데이터 세트의 (값비싼) 다운로드를 다시 수행해야 합니다. 그 결과, 이러한 장애 요소를 제거하는 새로운 데이터 세트 캐싱 솔루션이 만들어졌습니다. 이 솔루션은 ONTAP 고성능 스토리지 시스템에서 핫 데이터 세트를 캐싱하여 ML 파이프라인의 실행을 가속합니다. ONTAP NFS를 사용하면 NetApp에서 제공하는 Data Fabric(예: AFF A800)에서 데이터 세트를 한 번만 캐싱할 수 있으며, 이 데이터는 컴퓨팅과 함께 배치됩니다. NetApp ONTAP NFS 고속 스토리지가 여러 ML 컴퓨팅 노드를 지원할 수 있으므로 교육 모델의 성능이 최적화되어 비용 절감, 생산성 및 운영 효율성이 조직에 제공됩니다.

솔루션 아키텍처

다음 그림과 같이 NetApp 및 cnvrg.io의 이 솔루션은 데이터 세트 캐싱을 제공합니다. 데이터 세트 캐싱을 사용하면 데이터 과학자가 원하는 데이터 세트 또는 데이터 세트 버전을 선택하여 ML 컴퓨팅 클러스터 근처에 있는 ONTAP NFS 캐시로 이동할 수 있습니다. 이제 데이터 과학자는 지연 또는 다운로드를 유발하지 않고 여러 실험을 실행할 수 있습니다. 또한 모든 공동 작업 엔지니어는 데이터 레이크에서 추가로 다운로드할 필요 없이 연결된 컴퓨팅 클러스터(노드를 선택할 수 있는 자유로이)에서 동일한 데이터 세트를 사용할 수 있습니다. 데이터 과학자는 모든 데이터 세트 및 버전을 추적 및 모니터링하고 캐시된 데이터 세트를 확인하는 대시보드를 제공합니다.

cnvrg.io 플랫폼은 특정 시간 동안 사용되지 않은 오래된 데이터 세트를 자동으로 감지하여 캐시에서 데이터를 제거하므로 자주 사용하는 데이터 세트에 대해 사용 가능한 NFS 캐시 공간을 유지합니다. ONTAP의 데이터 세트 캐싱은 클라우드와 사내에서 이루어지므로 최대한의 유연성을 제공하는 것이 중요합니다.



개념 및 구성 요소

이 섹션에서는 ML 워크플로우에서 데이터 캐싱과 관련된 개념 및 구성 요소에 대해 설명합니다.

머신 러닝

ML은 전 세계 많은 기업과 조직에 빠르게 필수 요소가 되고 있습니다. 따라서, IT 및 DevOps 팀은 ML 워크로드 및 프로비저닝 클라우드, 온프레미스, 하이브리드 컴퓨팅 리소스를 표준화하여 ML 작업 및 파이프라인에 필요한 동적이고 집약적인 워크플로우를 지원해야 하는 과제에 직면해 있습니다.

컨테이너 기반 머신 러닝 및 **Kubernetes**

컨테이너는 공유 호스트 운영 체제 커널 위에서 실행되는 격리된 사용자 공간 인스턴스입니다. 컨테이너 채택이 빠르게 증가하고 있습니다. 컨테이너는 가상 머신(VM)이 제공하는 것과 동일한 애플리케이션 샌드박스(sandbox)의 많은 이점을 제공합니다. 하지만 VM이 사용하는 하이퍼바이저 및 게스트 운영 체제 계층이 없어졌기 때문에 컨테이너는 훨씬

더 가볍습니다.

또한 컨테이너를 사용하면 애플리케이션 종속성, 실행 시간 등을 애플리케이션과 직접 효율적으로 패키징할 수 있습니다. 가장 일반적으로 사용되는 컨테이너 패키징 형식은 Docker 컨테이너입니다. Docker 컨테이너 형식으로 컨테이너화된 애플리케이션은 Docker 컨테이너를 실행할 수 있는 모든 시스템에서 실행할 수 있습니다. 모든 종속성이 컨테이너 자체에 패키지되어 있기 때문에 응용 프로그램의 종속성이 컴퓨터에 없는 경우에도 마찬가지입니다. 자세한 내용은 를 참조하십시오 "Docker 웹 사이트".

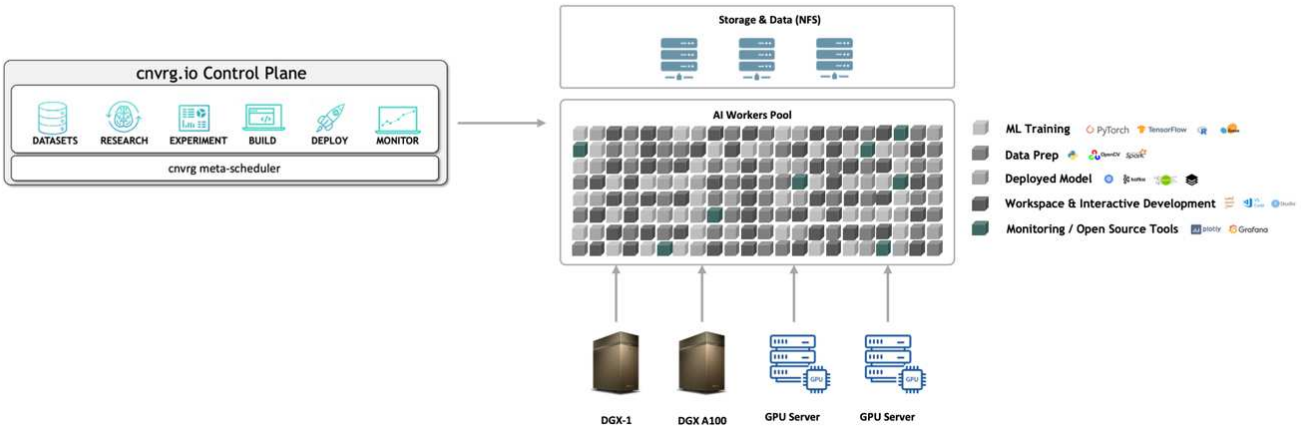
데이터 과학자는 널리 사용되는 컨테이너 오케스트레이터인 Kubernetes를 사용하여 유연한 컨테이너 기반 작업 및 파이프라인을 시작할 수 있습니다. 또한 인프라 팀이 단일 관리형 클라우드 네이티브 환경에서 ML 워크로드를 관리하고 모니터링할 수 있습니다. 자세한 내용은 를 참조하십시오 "Kubernetes 웹 사이트".

cnvrg.io

cnvrg.io는 AI 및 데이터 과학 개발의 관리, 확장 및 속도를 연구에서 운영으로 전환하는 AI 운영 체제입니다. 데이터 과학자가 코드 우선 플랫폼을 구축하고 사내 또는 클라우드에서 유연하게 실행할 수 있습니다. 모델 관리, MLOps 및 지속적인 ML 솔루션을 통해 cnvrg.IO는 데이터 과학 팀에 최고의 기술을 제공하므로 DevOps에 더 적은 시간을 할애하고 진정한 마법인 알고리즘에 집중할 수 있습니다. cnvrg.io를 사용한 이후, 여러 산업 분야의 팀들이 생산 모델에 더 많은 모델을 투입하여 비즈니스 가치를 증대하고 있습니다.

cnvrg.io 메타 스케줄러

cnvrg IO는 IT와 엔지니어가 서로 다른 컴퓨팅 리소스를 동일한 제어 평면에 연결하고 cnvrg.io를 사용하여 모든 리소스에 걸쳐 ML 작업을 관리할 수 있는 고유한 아키텍처를 가지고 있습니다. 즉, 다음 그림과 같이 여러 온프레미스 Kubernetes 클러스터, VM 서버 및 클라우드 계정을 연결하고 모든 리소스에서 ML 워크로드를 실행할 수 있습니다.



cnvrg.io 데이터 캐싱

데이터 과학자는 cnvrg.io를 사용하여 데이터 캐싱 기술을 통해 핫 데이터 세트 및 콜드 데이터 세트 버전을 정의할 수 있습니다. 기본적으로 데이터 세트는 중앙 집중식 오브젝트 스토리지 데이터베이스에 저장됩니다. 그런 다음 데이터 과학자는 선택한 컴퓨팅 리소스에 특정 데이터 버전을 캐시하여 다운로드 시간을 줄이고 ML 개발 및 생산성을 향상시킬 수 있습니다. 캐싱되고 며칠 동안 사용되지 않는 데이터 세트는 선택한 NFS에서 자동으로 지워집니다. 한 번의 클릭으로 캐시 캐싱 및 지우기를 수행할 수 있으며 코딩, IT 또는 DevOps 작업이 필요하지 않습니다.

cnvrg.io는 플로우 및 ML 파이프라인

cnvrg.IO Flows는 생산 ML 파이프라인을 구축하기 위한 도구입니다. 플로우의 각 구성 요소는 기본 Docker 이미지를 사용하여 선택한 컴퓨팅에서 실행되는 스크립트/코드입니다. 이 설계를 통해 데이터 과학자와 엔지니어가 사내 및

클라우드에서 모두 실행할 수 있는 단일 파이프라인을 구축할 수 있습니다. cnvrg.io는 데이터, 매개 변수 및 아티팩트가 서로 다른 구성 요소 간에 이동하고 있는지 확인합니다. 또한 각 흐름을 모니터링하고 추적하여 100% 재현성 있는 데이터 과학을 제공합니다.

cnvrg.io 코어

cnvrg.io core는 데이터 과학자가 DevOps에 초점을 맞추는 데 도움을 주기 위해 데이터 과학 커뮤니티를 위한 무료 플랫폼입니다. Core의 유연한 인프라를 통해 데이터 과학자는 온프레미스 또는 클라우드 등 어떤 언어, AI 프레임워크 또는 컴퓨팅 환경이라도 사용할 수 있으므로 가장 잘하는 일을 하고 알고리즘을 구축할 수 있습니다. cnvrg.io 코어는 모든 Kubernetes 클러스터에서 단일 명령으로 간편하게 설치할 수 있습니다.

NetApp ONTAP AI를 참조하십시오

ONTAP AI는 NetApp AFF 스토리지 시스템 및 NVIDIA DGX 시스템과 Tesla V100 GPU를 사용하는 ML 및 딥 러닝(DL) 워크로드를 위한 데이터 센터 참조 아키텍처입니다. ONTAP AI는 100Gb 이더넷을 통한 산업 표준 NFS 파일 프로토콜을 기반으로 하며, 표준 데이터 센터 기술을 사용하여 구현 및 관리 오버헤드를 줄이는 고성능 ML/DL 인프라를 고객에게 제공합니다. 표준화된 네트워크 및 프로토콜을 사용하여 ONTAP AI를 하이브리드 클라우드 환경에 통합하는 동시에 운영 일관성과 단순성을 유지할 수 있습니다. 사전 검증된 인프라 솔루션인 ONTAP AI를 사용하면 구축 시간과 위험을 줄이고 관리 오버헤드를 크게 줄여 고객이 투자 회수 시간을 단축할 수 있습니다.

NVIDIA DeepOps

DeepOps는 NVIDIA의 오픈 소스 프로젝트로, Ansible을 사용하여 GPU 서버 클러스터를 모범 사례에 따라 자동으로 구축합니다. DeepOps는 모듈식이며 다양한 배포 작업에 사용할 수 있습니다. 이 문서와 이 문서에서 설명하는 검증 연습에서는 GPU 서버 작업자 노드로 구성된 Kubernetes 클러스터를 배포하는 데 DeepOps를 사용합니다. 자세한 내용은 를 참조하십시오 "DeepOps 웹 사이트".

NetApp 트라이던트

Trident는 NetApp에서 개발 및 유지 관리하는 오픈 소스 스토리지 오케스트레이터로서 Kubernetes 워크로드를 위한 영구 스토리지의 생성, 관리 및 사용을 크게 단순화합니다. Trident 자체 Kubernetes 네이티브 애플리케이션 - Kubernetes 클러스터 내에서 직접 실행됩니다. Trident를 사용하면 Kubernetes 사용자(개발자, 데이터 과학자, Kubernetes 관리자 등)가 이미 익숙한 표준 Kubernetes 형식으로 영구 스토리지 볼륨을 생성, 관리 및 상호 작용할 수 있습니다. 이와 동시에 NetApp 기술에서 제공하는 NetApp 고급 데이터 관리 기능과 Data Fabric을 활용할 수 있습니다. Trident는 영구 스토리지의 복잡성을 추상화하여 사용이 간편합니다. 자세한 내용은 를 참조하십시오 "Trident 웹 사이트".

NetApp StorageGRID를 참조하십시오

NetApp StorageGRID는 사용자가 S3 프로토콜을 통해 액세스할 수 있는 간단하고 클라우드식 스토리지를 제공하여 이러한 요구를 충족하도록 설계된 소프트웨어 정의 오브젝트 스토리지 플랫폼입니다. StorageGRID는 인터넷에 연결된 사이트에서 거리에 관계없이 여러 노드를 지원하도록 설계된 스케일아웃 시스템입니다. StorageGRID의 지능형 정책 엔진을 사용하여 지리적 복원력을 위해 사이트 전체에서 오브젝트를 삭제 코딩하거나 원격 사이트 간에 오브젝트 복제를 선택하여 WAN 액세스 지연 시간을 최소화할 수 있습니다. StorageGRID은 이 솔루션에서 탁월한 프라이빗 클라우드 1차 오브젝트 스토리지 데이터 레이크를 제공합니다.

NetApp Cloud Volumes ONTAP를 참조하십시오

NetApp Cloud Volumes ONTAP 데이터 관리 소프트웨어는 AWS, Google Cloud Platform 및 Microsoft Azure를 비롯한 퍼블릭 클라우드 공급자의 유연성을 통해 사용자 데이터에 제어, 보호 및 효율성을 제공합니다. Cloud Volumes ONTAP은 NetApp ONTAP 스토리지 소프트웨어를 기반으로 하는 클라우드 네이티브 데이터 관리 소프트웨어로, 클라우드 데이터 요구사항을 해결하는 뛰어난 범용 스토리지 플랫폼을 제공합니다. 클라우드와 사내에서 동일한 스토리지 소프트웨어를 사용하는 사용자는 새로운 데이터 관리 방법을 통해 IT 직원을 교육하지 않고도 Data Fabric의

가치를 실현할 수 있습니다.

하이브리드 클라우드 구현 모델에 관심 있는 고객을 위해 Cloud Volumes ONTAP은 대부분의 퍼블릭 클라우드에서 동일한 기능과 동급 최고의 성능을 제공하여 어떠한 환경에도 일관되고 원활한 사용자 경험을 제공할 수 있습니다.

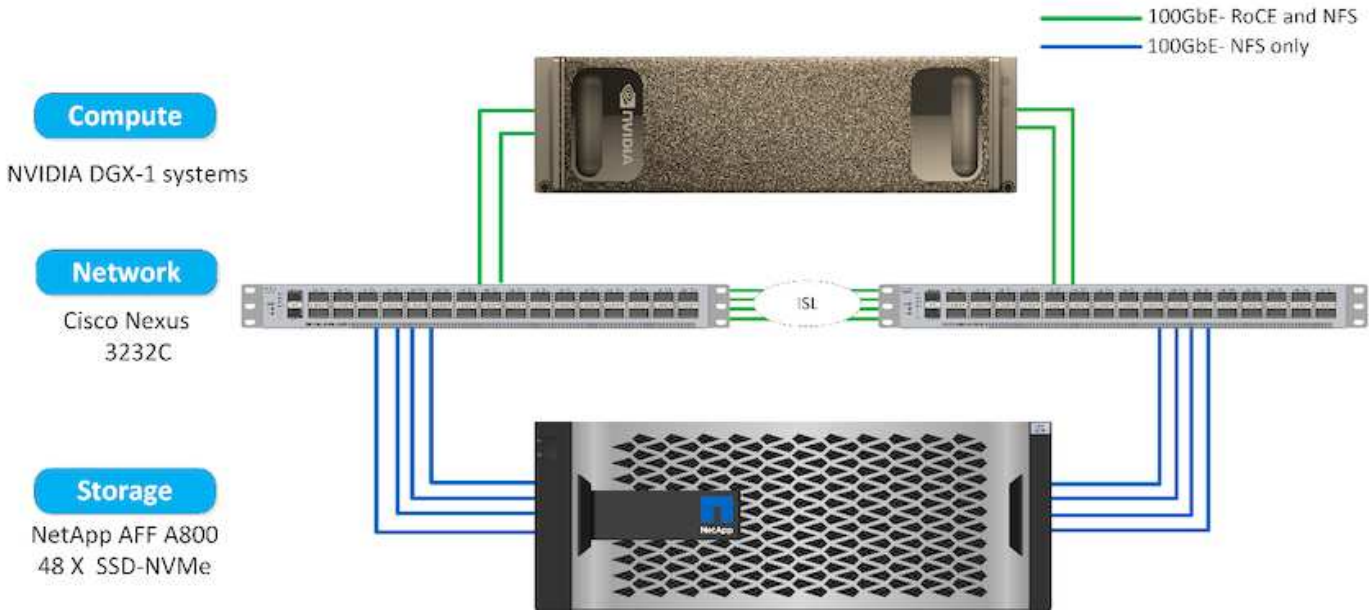
하드웨어 및 소프트웨어 요구 사항

이 섹션에서는 ONTAP AI 솔루션의 기술 요구사항을 다룹니다.

하드웨어 요구 사항

하드웨어 요구사항은 특정 고객 워크로드에 따라 다르지만, ONTAP AI는 대규모 ML/DL 작업을 위해 단일 GPU에서 랙 확장 구성까지 데이터 엔지니어링, 모델 훈련, 운영 추론을 위해 어떤 확장하고 구축할 수 있습니다. ONTAP AI에 대한 자세한 내용은 를 참조하십시오 "ONTAP AI 웹 사이트".

이 솔루션은 컴퓨팅, NetApp AFF A800 스토리지 시스템 및 Cisco Nexus 3232C 네트워크 연결을 위해 DGX-1 시스템을 사용하여 검증되었습니다. 이 검증에 사용된 AFF A800은 대부분의 ML/DL 워크로드에 대해 최대 10개의 DGX-1 시스템을 지원할 수 있습니다. 다음 그림은 이 검증에서 모델 훈련에 사용되는 ONTAP AI 토폴로지를 보여줍니다.



이 솔루션을 퍼블릭 클라우드로 확장하려면 Cloud Volumes ONTAP을 클라우드 GPU 컴퓨팅 리소스와 함께 구축하고 하이브리드 클라우드 데이터 패브릭에 통합하면 모든 워크로드에 적합한 리소스를 사용할 수 있습니다.

소프트웨어 요구 사항

다음 표는 이 솔루션 검증에 사용된 특정 소프트웨어 버전을 보여줍니다.

구성 요소	버전
우분투	18.04.4 LTS
NVIDIA DGX OS	4.4.0
NVIDIA DeepOps	20.02.1

구성 요소	버전
쿠버네티스	1.15
헬름	3.1.0
cnvrg.io	3.0.0
NetApp ONTAP를 참조하십시오	9.6P4

이 솔루션 검증에서 Kubernetes는 DGX-1 시스템에서 단일 노드 클러스터로 구축되었습니다. 대규모 배포의 경우 관리 서비스의 고가용성을 제공하고 ML 및 DL 워크로드에 대한 중요한 DGX 리소스를 예약하려면 독립 Kubernetes 마스터 노드를 구축해야 합니다.

솔루션 배포 및 검증 세부 정보

다음 섹션에서는 솔루션 구축 및 검증에 대한 세부 정보를 다룹니다.

ONTAP AI 배포

ONTAP AI를 배포하려면 네트워킹, 컴퓨팅 및 스토리지 하드웨어를 설치하고 구성해야 합니다. ONTAP AI 인프라 구축에 대한 구체적인 지침은 이 문서의 범위를 벗어납니다. 자세한 배포 정보는 를 참조하십시오 "NVA-1121-deploy: NetApp ONTAP AI, NVIDIA 구현".

이 솔루션 검증에서 단일 볼륨이 생성되어 DGX-1 시스템에 마운트되었습니다. 그런 다음 이 마운트 지점을 컨테이너에 마운트하여 데이터를 교육에 액세스할 수 있도록 했습니다. 대규모 배포의 경우 NetApp Trident는 볼륨의 생성 및 마운트를 자동화하여 관리 오버헤드를 제거하고 최종 사용자 리소스 관리를 지원합니다.

Kubernetes 구축

NVIDIA DeepOps를 사용하여 Kubernetes 클러스터를 구축하고 구성하려면 배포 점프 호스트에서 다음 작업을 수행하십시오.

1. 의 지침에 따라 NVIDIA DeepOps를 다운로드합니다 "시작 페이지" NVIDIA DeepOps GitHub 사이트에서 다운로드할 수 있습니다.
2. 의 지침에 따라 클러스터에 Kubernetes를 배포합니다 "Kubernetes 구축 가이드" NVIDIA DeepOps GitHub 사이트에서 다운로드할 수 있습니다.

> DeepOps Kubernetes 구축이 작동하려면 모든 Kubernetes 마스터 및 작업자 노드에 동일한 사용자가 있어야 합니다.

배포가 실패하면 depops/config/group_vars/k8s-cluster.yml에서 kubctl_localhost의 값을 false로 변경하고 2단계를 반복합니다. kubeck_localhost의 값이 true인 경우에만 실행되는 Ansible 호스트에 kubbeck 바이너리 복사 작업은 알려진 메모리 사용 문제가 있는 Ansible 모듈 가져오기를 사용합니다. 이러한 메모리 사용 문제로 인해 작업이 실패할 수 있습니다. 메모리 문제로 인해 작업이 실패하면 나머지 배포 작업이 성공적으로 완료되지 않습니다.

kubbctl_localhost의 값을 false로 변경한 후 배포가 성공적으로 완료되면 Kubernetes 마스터 노드에서 배포 점프 호스트로 kubbectl 바이너리를 수동으로 복사해야 합니다. 특정 마스터 노드에서 kudctl 명령을 직접 실행하여 kubctl 바이너리의 위치를 찾을 수 있습니다.

cnvrg.io 배포

제어 장치를 사용하여 cnvrg 코어를 배포합니다

Helm은 모든 클러스터, 온프레미스, Minikube 또는 모든 클라우드 클러스터(예: AKS, EKS, GKE)를 사용하여 cnvrg를 신속하게 배포하는 가장 쉬운 방법입니다. 이 섹션에서는 Kubernetes가 설치된 사내(DGX-1) 인스턴스에 cnvrg를 설치한 방법을 설명합니다.

필수 구성 요소

설치를 완료하려면 먼저 로컬 컴퓨터에 다음 종속 항목을 설치하고 준비해야 합니다.

- 쿠베틀입니다
- Helm 3.x
- Kubernetes 클러스터 1.15 이상

헬름으로 배포

1. 최신 cnvrg Helm 차트를 다운로드하려면 다음 명령을 실행합니다.

```
helm repo add cnvrg https://helm.cnvrg.io
helm repo update
```

2. cnvrg를 구축하기 전에 클러스터의 외부 IP 주소와 cnvrg를 배포할 노드의 이름이 필요합니다. 사내 Kubernetes 클러스터에 cnvrg를 배포하려면 다음 명령을 실행합니다.

```
helm install cnvrg cnvrg/cnvrg --timeout 1500s  --wait \ --set
global.external_ip=<ip_of_cluster> \ --set global.node=<name_of_node>
```

3. helm install 명령을 실행합니다. 모든 서비스 및 시스템이 클러스터에 자동으로 설치됩니다. 이 프로세스는 최대 15분 정도 소요될 수 있습니다.
4. 헬름 설치 명령은 10분 정도 걸릴 수 있습니다. 배포가 완료되면 새로 배포된 cnvrg의 URL로 이동하거나 새 클러스터를 조직 내의 리소스로 추가합니다. 'helm' 명령은 올바른 URL을 알려줍니다.

```
Thank you for installing cnvrg.io!
Your installation of cnvrg.io is now available, and can be reached via:
Talk to our team via email at
```

5. 모든 컨테이너의 상태가 실행 중 또는 완료됨이면 cnvrg가 성공적으로 배포된 것입니다. 이 결과는 다음 예제 출력과 유사해야 합니다.

```
NAME                                READY   STATUS      RESTARTS   AGE
cnvrg-app-69fbb9df98-6xrgf                  1/1     Running     0          2m
cnvrg-sidekiq-b9d54d889-5x4fc               1/1     Running     0          2m
controller-65895b47d4-s96v6                 1/1     Running     0          2m
init-app-vs-config-wv9c4                    0/1     Completed   0          9m
init-gateway-vs-config-2zbpp                0/1     Completed   0          9m
init-minio-vs-config-cd2rg                  0/1     Completed   0          9m
minio-0                                     1/1     Running     0          2m
postgres-0                                  1/1     Running     0          2m
redis-695c49c986-kcbt9                      1/1     Running     0          2m
seeder-wh655                                0/1     Completed   0          2m
speaker-5sghr                               1/1     Running     0          2m
```

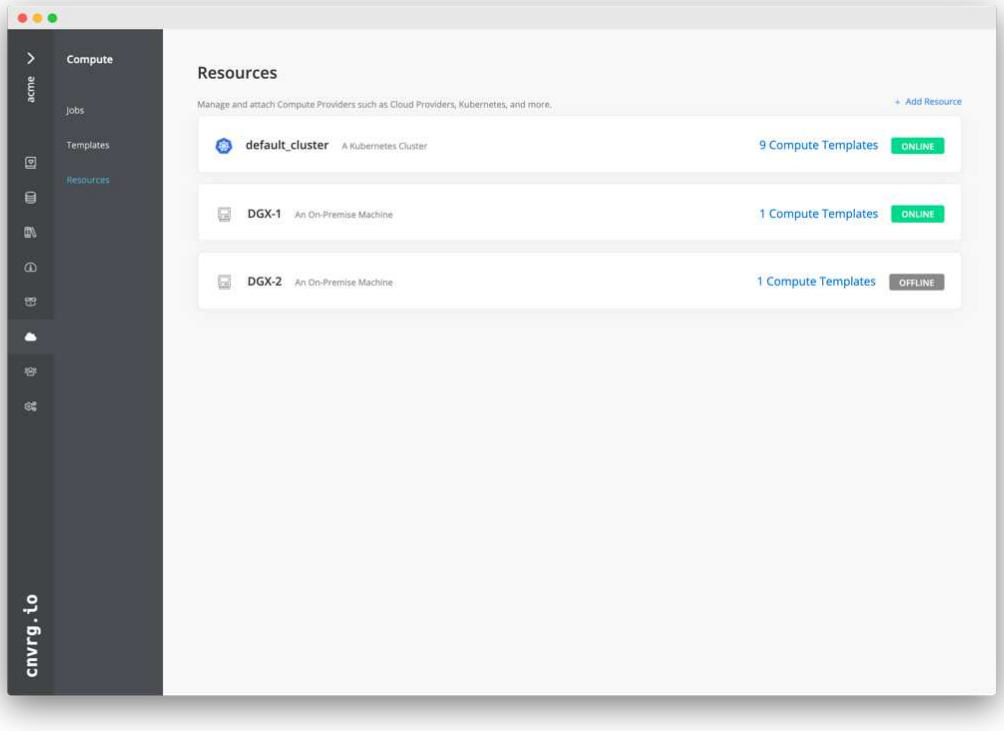
ResNet50 및 **Chest X-ray** 데이터 집합을 사용한 컴퓨터 비전 모델 교육

cnvrg.io AI OS는 NVIDIA DGX 시스템을 기반으로 하는 NetApp ONTAP AI 아키텍처를 기반으로 Kubernetes 설정에 구축했습니다. 검증을 위해 흉부 X-레이의 식별 불가 영상으로 구성된 NIH Chest X-ray 데이터 세트를 사용했습니다. 이미지는 PNG 형식이었습니다. 이 데이터는 NIH 임상 센터에서 제공했으며 을 통해 확인할 수 있습니다 "NIH 다운로드 사이트". 15개 클래스에서 627, 615의 이미지를 사용하여 250GB 데이터 샘플을 사용했습니다.

데이터 세트가 cnvrg 플랫폼으로 업로드되었으며 NetApp AFF A800 스토리지 시스템에서 NFS 엑스포트에 캐싱되었습니다.

컴퓨팅 리소스 설정

엔지니어 및 IT 전문가는 cnvrg 아키텍처 및 메타 스케줄링 기능을 사용하여 서로 다른 컴퓨팅 리소스를 단일 플랫폼에 연결할 수 있습니다. 설정에서 딥 러닝 워크로드를 실행하기 위해 배포된 것과 동일한 클러스터 cnvrg를 사용했습니다. 추가 클러스터를 연결해야 하는 경우 다음 스크린샷과 같이 GUI를 사용합니다.



데이터 로드

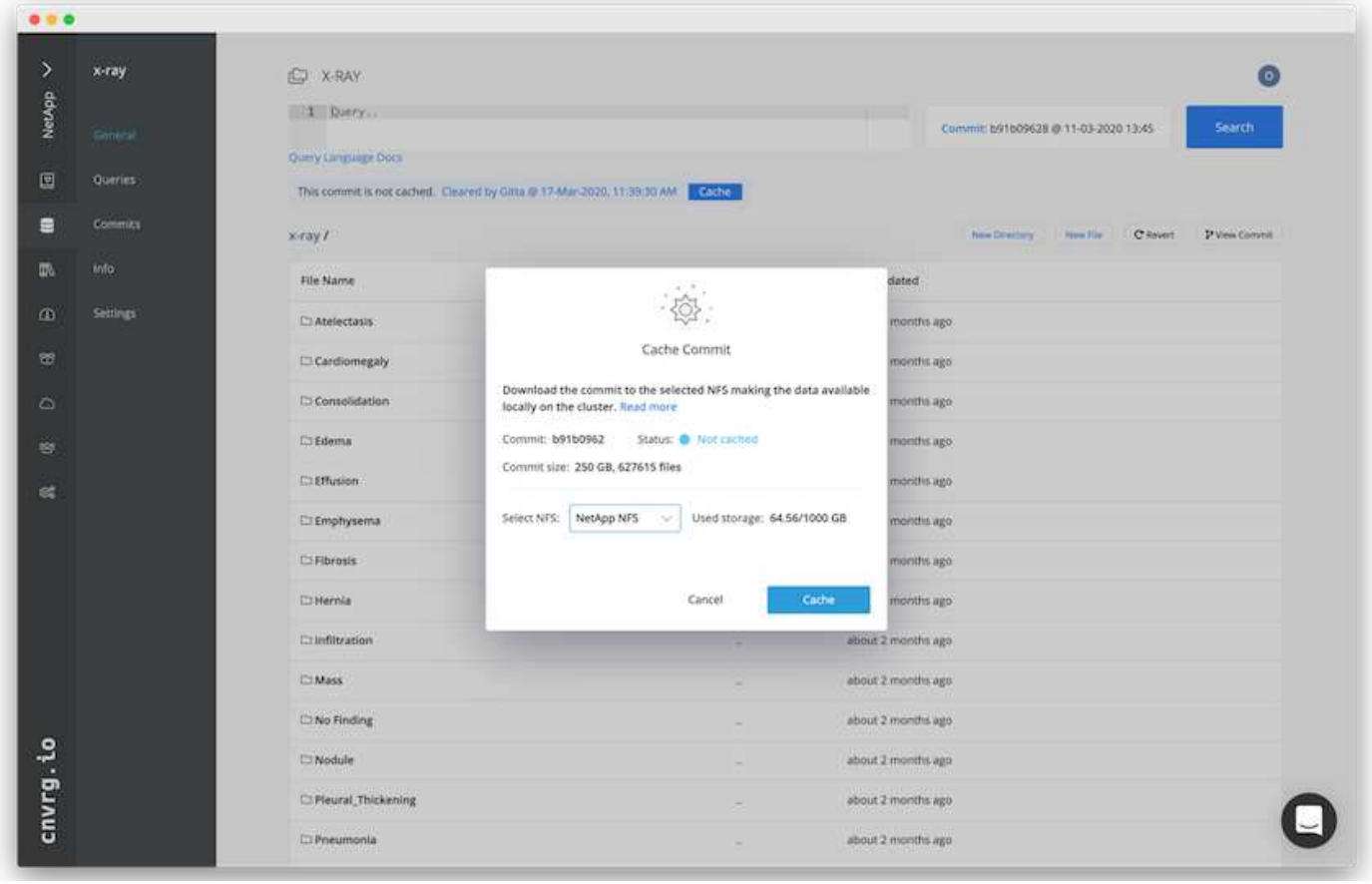
cnvrg 플랫폼에 데이터를 업로드하려면 GUI 또는 cnvrg CLI를 사용할 수 있습니다. 대규모 데이터 세트의 경우 많은 파일을 처리할 수 있는 강력하고 확장 가능하며 안정적인 툴이므로 CLI를 사용하는 것이 좋습니다.

데이터를 업로드하려면 다음 단계를 완료하십시오.

1. 를 다운로드합니다 "cnvrg CLI를 참조하십시오".
2. X-ray 디렉토리로 이동합니다.
3. 'cnvrg data init' 명령으로 플랫폼 내 데이터세트를 초기화한다.
4. 중앙 오브젝트 저장소(StorageGRID, S3 등)에 데이터를 업로드한 후 GUI로 검색할 수 있습니다. 다음 그림은 로드된 흉부 X선 섬유증 영상 PNG 파일을 보여줍니다. 또한, cnvrg는 데이터를 버전화하므로 빌드하는 모든 모델을 데이터 버전으로 복제할 수 있습니다.

Cach 데이터

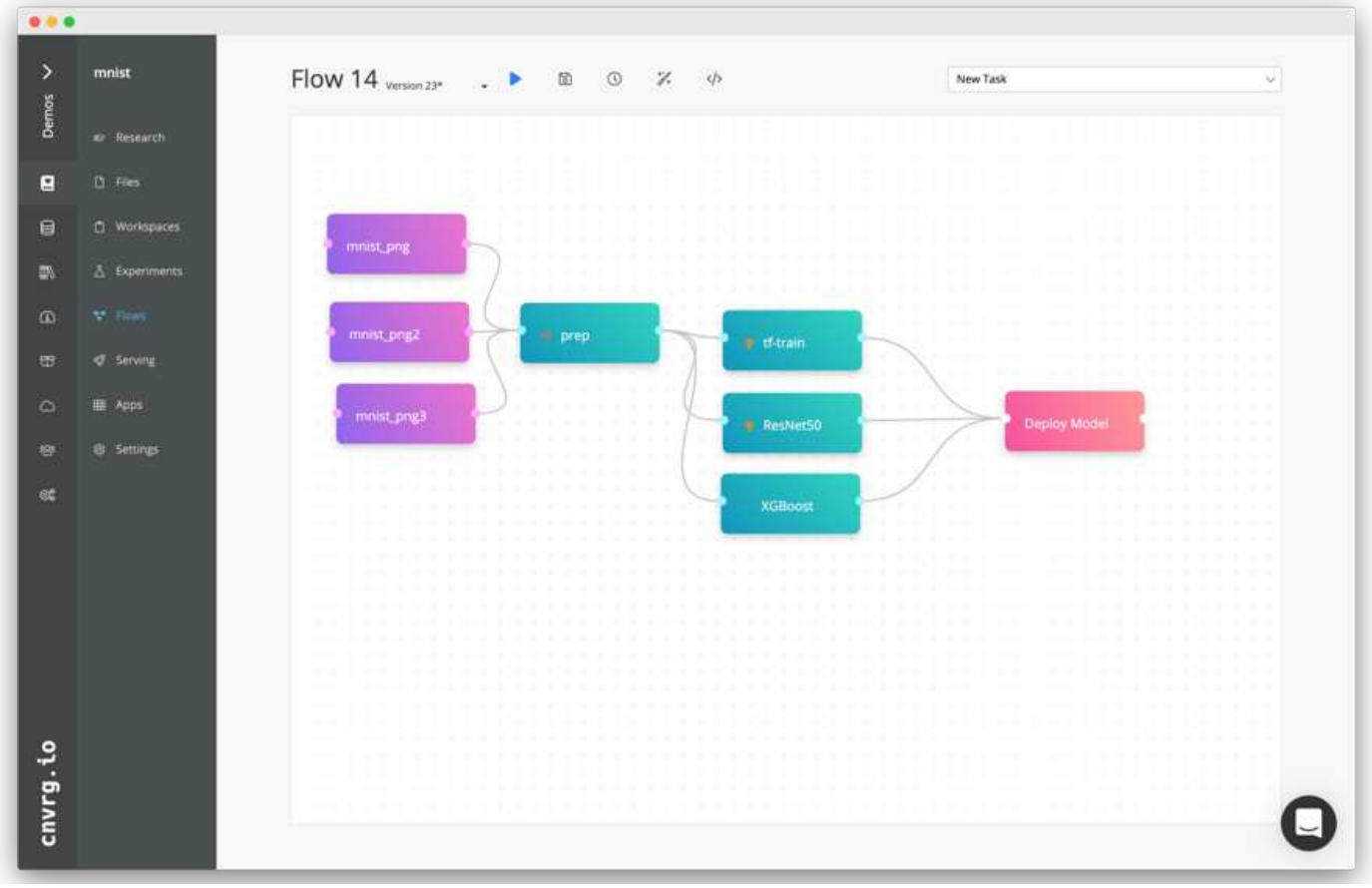
각 모델의 교육 및 실험을 위해 600k+ 파일을 다운로드하지 않고 더 빠르게 교육을 제공하기 위해 데이터를 중앙 데이터 레이크 오브젝트 저장소에 처음 업로드한 후 데이터 캐싱 기능을 사용했습니다.



사용자가 캐시를 클릭하면 cnvrg는 원격 오브젝트 저장소에서 특정 커밋에 있는 데이터를 다운로드하여 ONTAP NFS 볼륨에 캐시합니다. 완료되면 데이터를 즉시 교육에 사용할 수 있습니다. 또한 데이터가 며칠 동안 사용되지 않으면(예: 모델 교육 또는 탐색) cnvrg가 자동으로 캐시를 지웁니다.

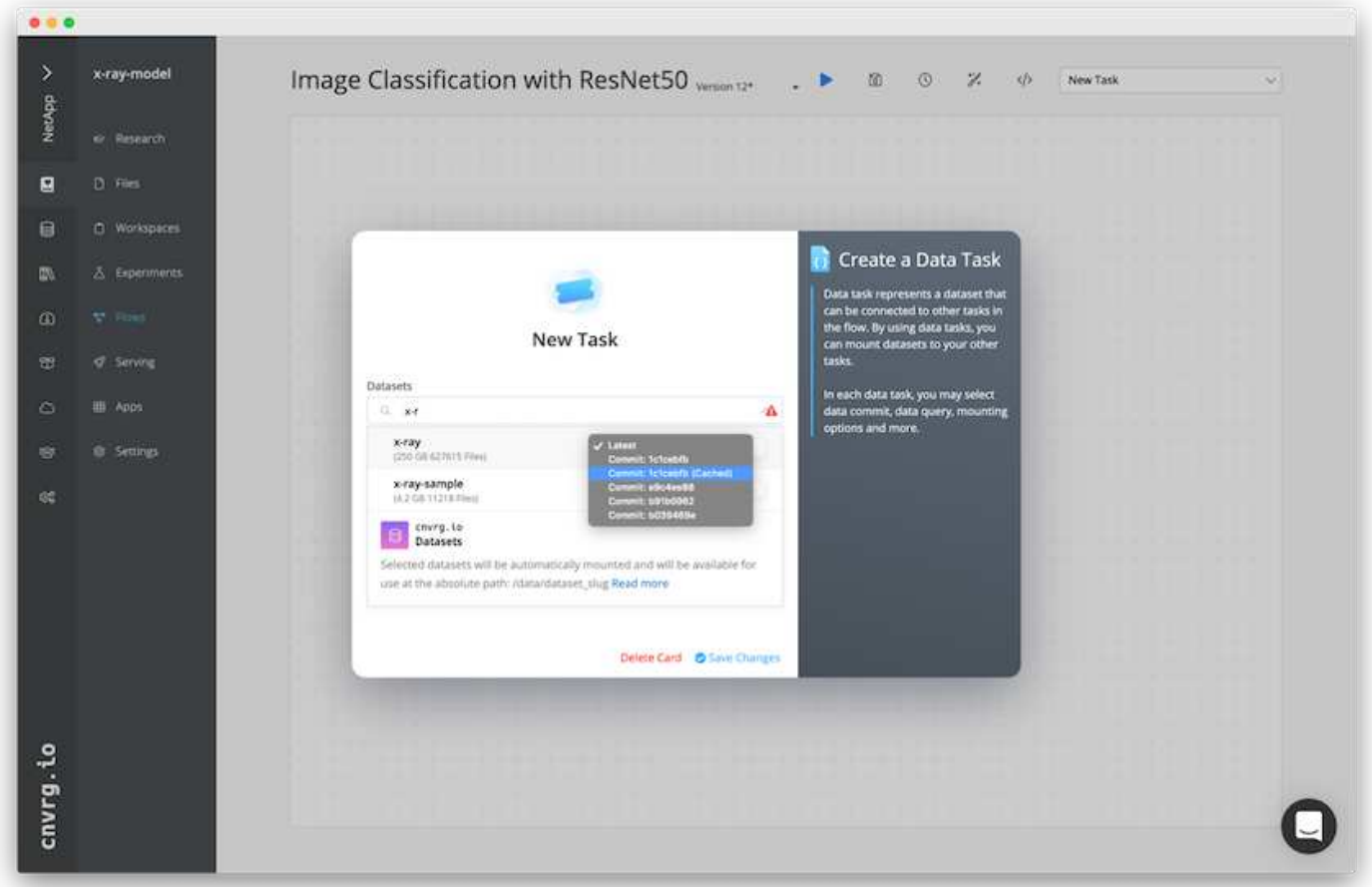
캐시된 데이터로 **ML** 파이프라인을 구축합니다

cnvrg 흐름으로 생산 ML 파이프라인을 쉽게 구축할 수 있습니다. 흐름은 유연하며 모든 종류의 ML 사용 사례에 사용할 수 있으며 GUI 또는 코드를 통해 생성할 수 있습니다. 플로우의 각 구성 요소는 다른 Docker 이미지를 사용하여 다른 컴퓨팅 리소스에서 실행될 수 있으므로 하이브리드 클라우드와 최적화된 ML 파이프라인을 구축할 수 있습니다.



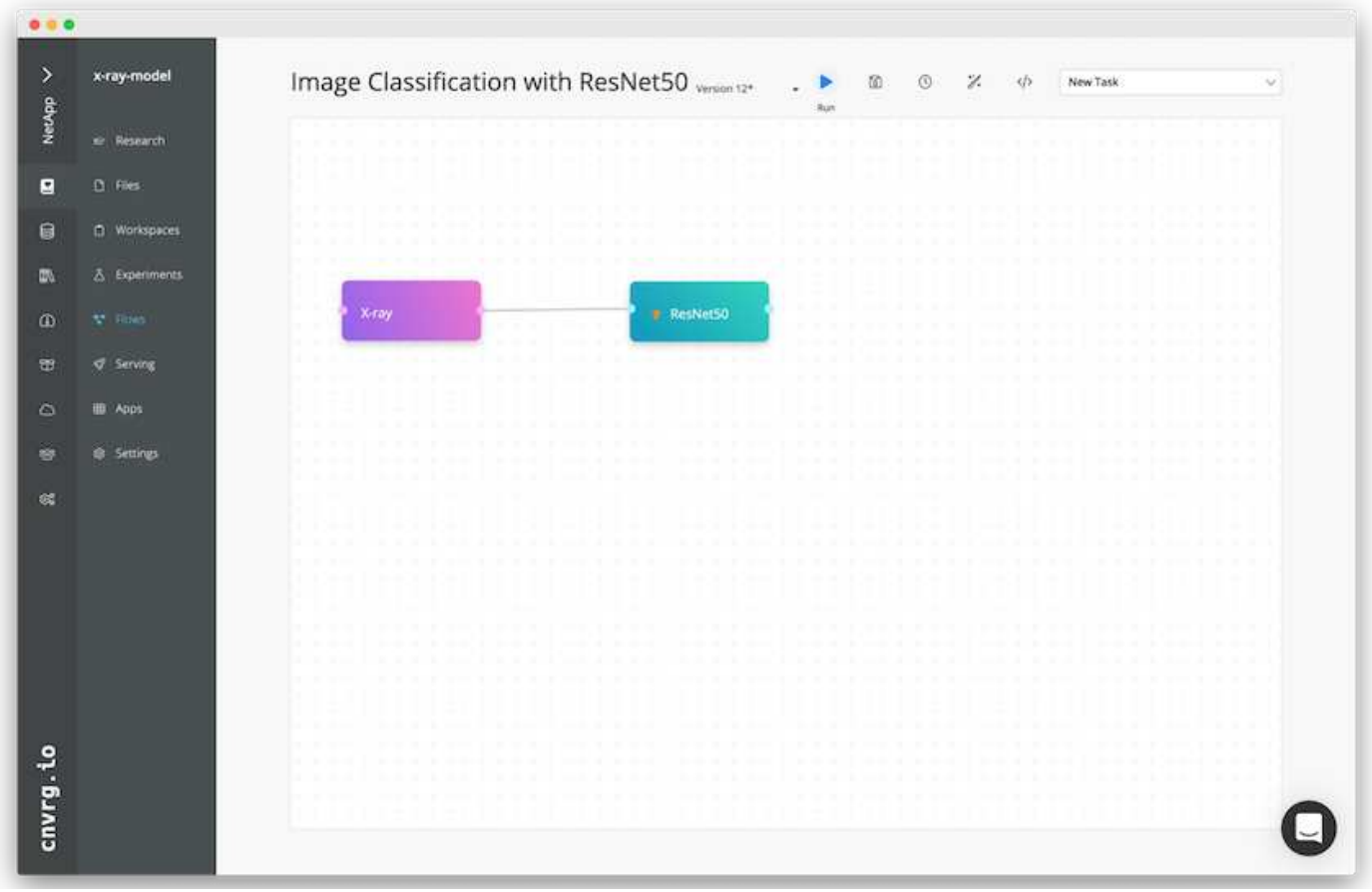
흉부 **X**선 흐름 구축**:**데이터 설정

새로 생성된 흐름에 데이터 세트를 추가했습니다. 데이터 집합을 추가할 때 특정 버전(커밋)을 선택하고 캐시된 버전을 사용할지 여부를 지정할 수 있습니다. 이 예에서는 캐시된 커밋을 선택했습니다.



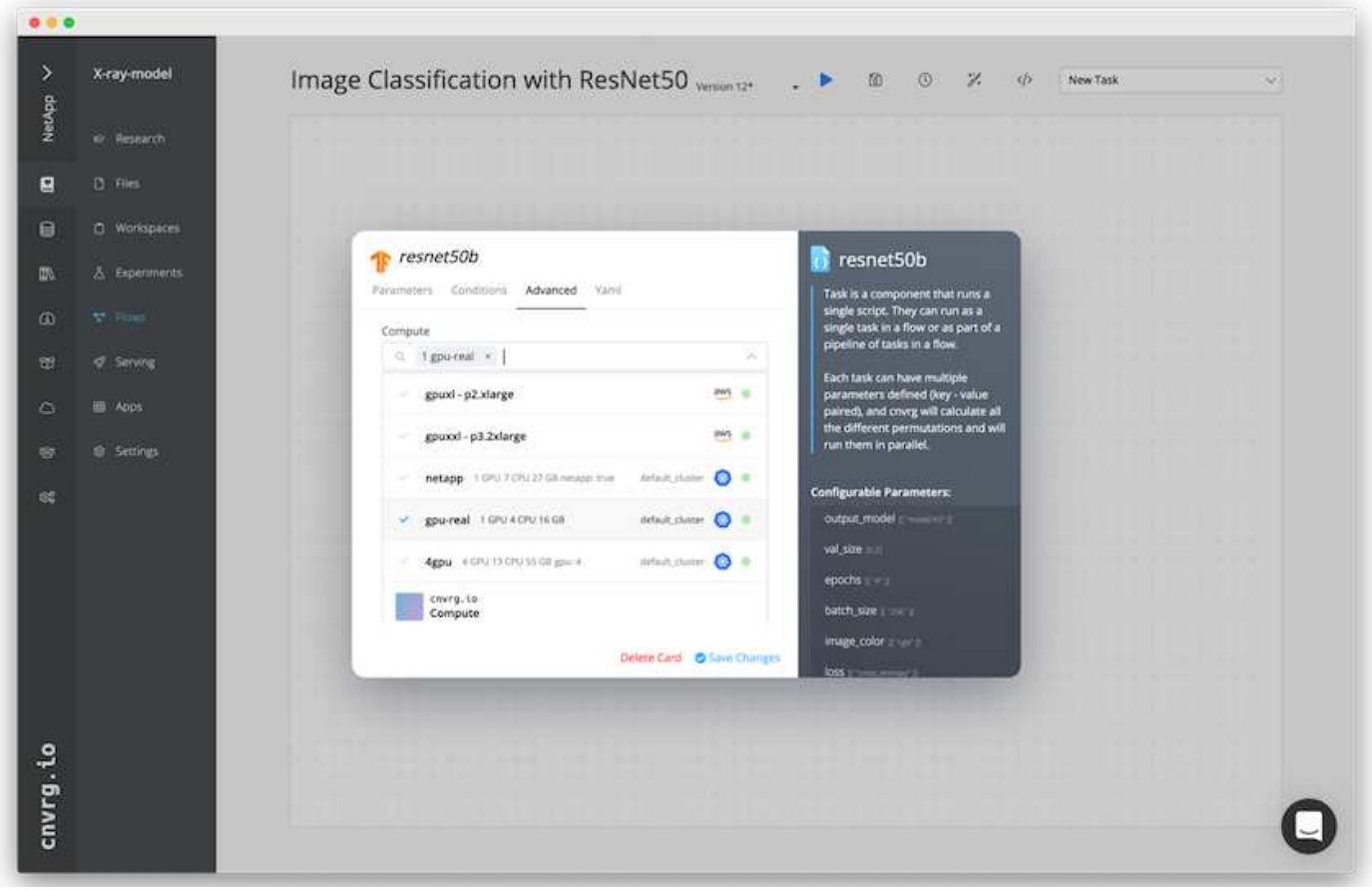
흉부 **X**선 흐름 구축**:**교육 모델 설정**:ResNet50**

파이프라인에서는 원하는 모든 종류의 사용자 지정 코드를 추가할 수 있습니다. cnvrg에는 재사용 가능한 ML 구성 요소 컬렉션인 AI 라이브러리도 있습니다. AI 라이브러리에는 모든 ML 또는 딥 러닝 플로우에 사용할 수 있는 알고리즘, 스크립트, 데이터 소스 및 기타 솔루션이 있습니다. 이 예에서는 사전 구축된 ResNet50 모듈을 선택했습니다. batch_size:128, epoch:10 등과 같은 기본 매개 변수를 사용했습니다. 이러한 매개 변수는 AI 라이브러리 문서에서 확인할 수 있습니다. 다음 스크린샷은 ResNet50에 연결된 X선 데이터 세트의 새로운 흐름을 보여줍니다.



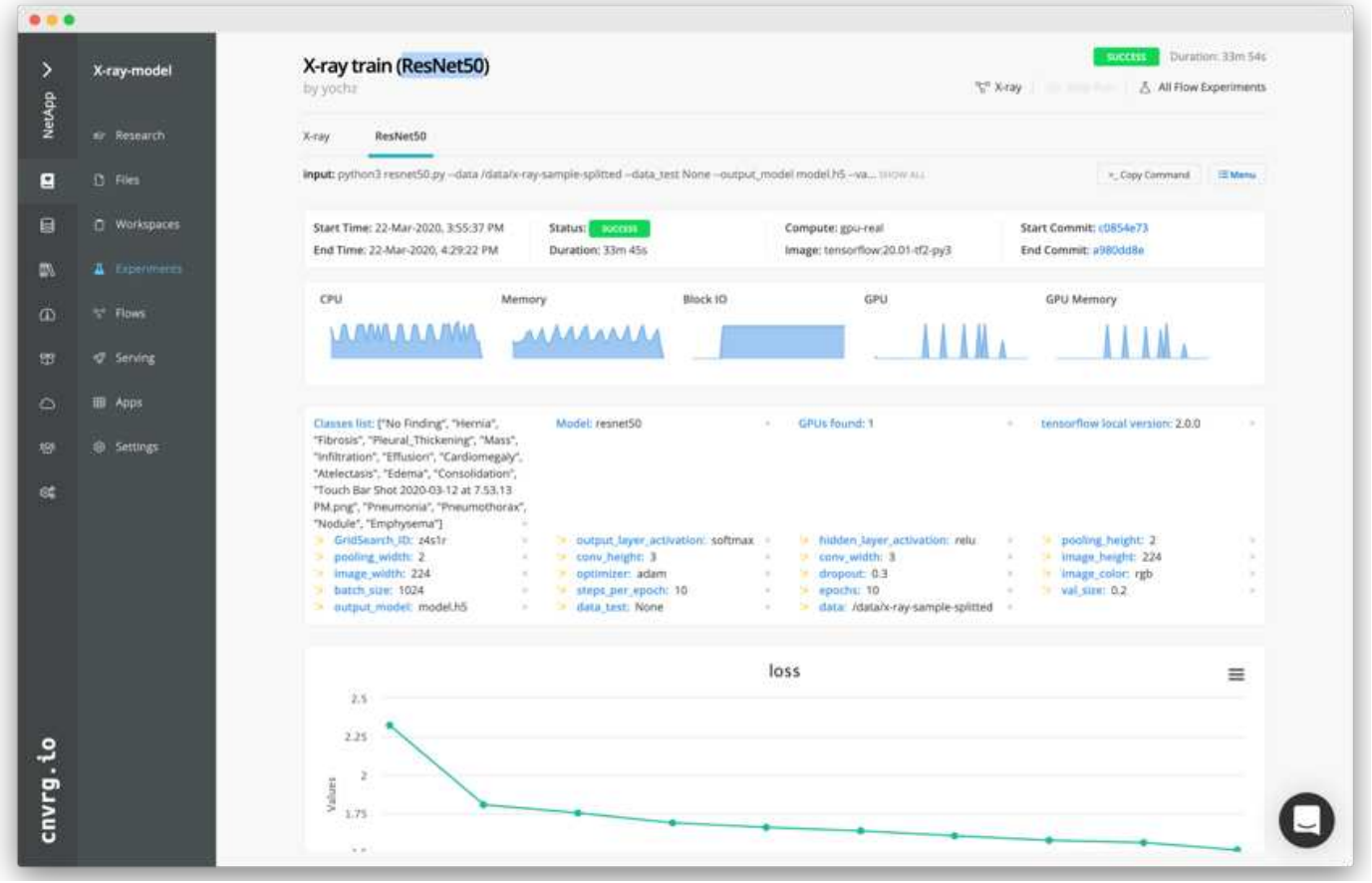
ResNet50의 컴퓨팅 리소스를 정의합니다

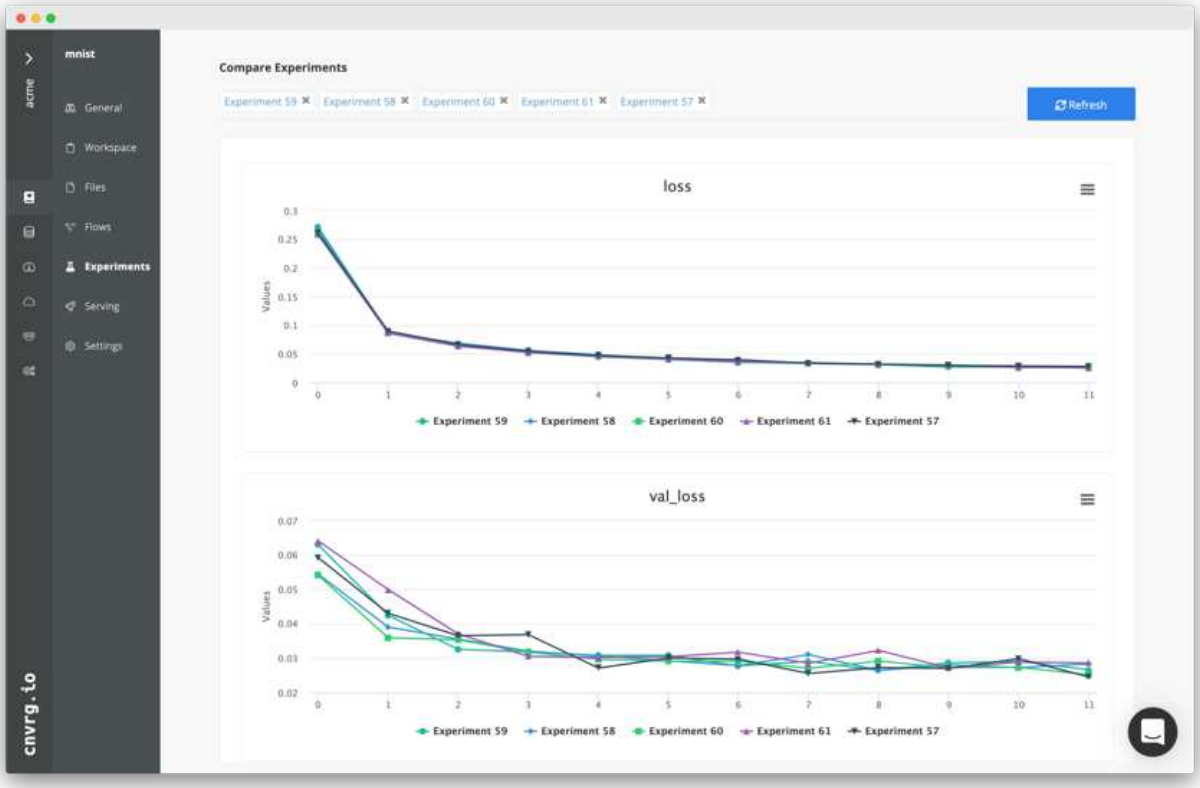
cnvrg 플로우의 각 알고리즘 또는 구성 요소는 다른 Docker 이미지와 함께 다른 컴퓨팅 인스턴스에서 실행될 수 있습니다. 저희 셋업에서는 NetApp ONTAP AI 아키텍처를 사용하여 NVIDIA DGX 시스템에 대한 훈련 알고리즘을 실행하려고 했습니다. 다음 그림에서는 사내 클러스터의 컴퓨팅 템플릿과 사양인 GPU-Real을 선택했습니다. 또한 템플릿 큐와 여러 템플릿을 선택했습니다. 이렇게 하면 'GPU-실제' 리소스를 할당할 수 없는 경우(예: 다른 데이터 과학자가 사용 중인 경우) 클라우드 공급자 템플릿을 추가하여 자동 클라우드 증가를 지원할 수 있습니다. 다음 스크린샷에서는 ResNet50의 컴퓨팅 노드로 GPU-real을 사용하는 방법을 보여 줍니다.



추적 및 모니터링 결과

흐름이 실행된 후 cnvrg가 추적 및 모니터링 엔진을 트리거합니다. 각 흐름 실행은 자동으로 문서화되고 실시간으로 업데이트됩니다. Hyperparameters, 메트릭, 리소스 사용량(GPU 활용률 등), 코드 버전, 아티팩트, 로그, 다음 두 스크린샷과 같이 실험 섹션에서 자동으로 사용할 수 있습니다.


X-ray-model
X-ray train (ResNet50)
by yochz
Research
Files
Workspaces
Experiments
Flows
Serving
Apps
Settings
cnvrg.io
NetApp
X-ray
ResNet50
CPU
Memory
Block IO
GPU
GPU Memory
loss
Values


mnist
Compare Experiments
Refresh
General
Workspace
Files
Flows
Experiments
Serving
Settings
cnvrg.io
acme
loss
val_loss
Values
Experiment 59
Experiment 58
Experiment 60
Experiment 61
Experiment 57

결론

NetApp과 cnvrg.io는 파트너십을 통해 고객에게 ML 및 DL 소프트웨어 개발을 위한 완벽한 데이터 관리 솔루션을 제공합니다. ONTAP AI는 모든 규모의 운영에 고성능 컴퓨팅 및 스토리지를 제공하며 cnvrg.io 소프트웨어는 데이터 과학 워크플로우를 간소화하고 리소스 활용률을 향상합니다.

감사의 말

- NetApp 기술 마케팅 엔지니어 Mike Olesby
- NetApp 선임 기술 담당 이사 Santosh Rao

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 리소스를 참조하십시오.

- cnvrg.io("https://cnvrg.io"):
  - cnvrg 코어(무료 ML 플랫폼)

    https://cnvrg.io/platform/core
  - cnvrg 문서

    "https://app.cnvrg.io/docs"
- NVIDIA DGX-1 서버:
  - NVIDIA DGX-1 서버

    https://www.nvidia.com/en-us/data-center/dgx-1/
  - NVIDIA Tesla V100 Tensor 코어 GPU

    https://www.nvidia.com/en-us/data-center/tesla-v100/
  - NGC(NVIDIA GPU Cloud)

    https://www.nvidia.com/en-us/gpu-cloud/
- NetApp AFF 시스템:
  - AFF 데이터시트 를 참조하십시오

    https://www.netapp.com/us/media/d-3582.pdf
  - AFF를 위한 NetApp FlashAdvantage

    https://www.netapp.com/us/media/ds-3733.pdf
  - ONTAP 9.x 설명서

    http://mysupport.netapp.com/documentation/productlibrary/index.html?productID=62286

- NetApp FlexGroup 기술 보고서

  https://www.netapp.com/us/media/tr-4557.pdf

- 컨테이너용 NetApp 영구 스토리지:
  - NetApp 트라이던트

    https://netapp.io/persistent-storage-provisioner-for-kubernetes/

- NetApp 상호 운용성 매트릭스:
  - NetApp 상호 운용성 매트릭스 툴

    http://support.netapp.com/matrix

- ONTAP AI 네트워킹:
  - Cisco Nexus 3232C 스위치

    https://www.cisco.com/c/en/us/products/switches/nexus-3232c-switch/index.html

  - Mellanox Spectrum 2000 시리즈 스위치

    http://www.mellanox.com/page/products_dyn?product_family=251&mtag=sn2000

- ML 프레임워크 및 도구:
  - 달리

    https://github.com/NVIDIA/DALI

  - TensorFlow: 모두를 위한 오픈 소스 머신 러닝 프레임워크

    https://www.tensorflow.org/

  - Horovod: Uber의 TensorFlow용 오픈 소스 분산 딥 러닝 프레임워크

    https://eng.uber.com/horovod/

  - 컨테이너 런타임 에코시스템에서 GPU 지원

    https://devblogs.nvidia.com/gpu-containers-runtime/

  - Docker 를 참조하십시오

    https://docs.docker.com

  - 쿠버네티스

    https://kubernetes.io/docs/home/

  - NVIDIA DeepOps

    https://github.com/NVIDIA/deepops

- Kubeflow

  http://www.kubeflow.org/

- Jupyter 노트북 서버

  http://www.jupyter.org/

- 데이터 세트 및 벤치마크
  - NIH 흉부 X선 데이터 세트

    https://nihcc.app.box.com/v/ChestXray-NIHCC

  - Xiaosong Wang, Yifan Peng, Le Lu, Zhivyong Lu, Mohammadadadhadi Bagheri, Ronald Summers, ChestX-ray8: 병원 스케일 흉부 X선 데이터베이스 및 약한 감독 하에 Common Thorax 질환의 분류 및 국소화에 대한 벤치마크, IEEE CVPR, pp. 3462-3471, 2017TR-4841-0620

Edge-NetApp에서 Lenovo ThinkSystem을 사용한 AI 추론 - 솔루션 설계

TR-4886: Edge-NetApp에서 Lenovo ThinkSystem - 솔루션 설계를 사용한 AI 추론

Sathish Thyagarajan, NetApp Miroslav Hodak, Lenovo

요약

ADAS(Advanced Driver Assistance Systems), Industry 4.0, 스마트 시티 및 IoT(Internet of Things)와 같은 몇 가지 새로운 애플리케이션 시나리오에서는 지연 시간이 거의 없이 지속적인 데이터 스트림을 처리해야 합니다. 이 문서에서는 이러한 요구사항을 충족하는 에지 환경에서 NetApp 스토리지 컨트롤러 및 Lenovo ThinkSystem 서버에 GPU 기반 인공 지능(AI) 추론을 배포하기 위한 컴퓨팅 및 스토리지 아키텍처에 대해 설명합니다. 또한, NVIDIA T4 GPU가 장착된 에지 서버에서 다양한 추론 작업을 평가하여 업계 표준 MLPerf Inference 벤치마크의 성능 데이터도 제공합니다. 오프라인, 단일 스트림 및 다중 스트림 추론 시나리오의 성능을 조사한 결과, 비용 효율적인 공유 네트워크 스토리지 시스템이 포함된 아키텍처의 성능이 매우 뛰어나며 여러 에지 서버에 대한 데이터 및 모델 관리의 중앙 지점을 제공하는 것으로 나타났습니다.

소개

기업들은 네트워크 에지에 대량의 데이터를 생성하고 있습니다. 스마트 센서 및 IoT 데이터를 활용하여 최대의 가치를 실현하기 위해 조직은 에지 컴퓨팅을 지원하는 실시간 이벤트 스트리밍 솔루션을 찾고 있습니다. 따라서 데이터 센터 외부의 에지에서는 컴퓨팅 작업이 점점 더 많이 수행됩니다. AI 추론을 이러한 트렌드에 동인으로 이끄는 요인 중 하나입니다. 에지 서버는 특히 가속기를 사용할 때 이러한 워크로드에 충분한 연산 능력을 제공하지만 제한된 스토리지는 종종 문제가 됩니다. 특히 다중 서버 환경에서는 더욱 그렇습니다. 이 문서에서는 에지 환경에서 공유 스토리지 시스템을 구축하는 방법과 성능 저하 없이 AI 추론 워크로드의 이점을 활용하는 방법을 설명합니다.

이 문서에서는 에지의 AI 추론을 위한 참조 아키텍처에 대해 설명합니다. 여러 Lenovo ThinkSystem 에지 서버를 NetApp 스토리지 시스템과 결합하여 간편하게 구축 및 관리할 수 있는 솔루션을 구축합니다. 이 가이드는 여러 대의 카메라와 산업용 센서가 장착된 공장 바닥, 소매 거래의 POS(Point-of-Sale) 시스템 또는 자율 차량의 시각적 이상을 식별하는 FSD(Full Self-Driving) 시스템 등 다양한 상황에서 실제 배포를 위한 기본 안내서입니다.

이 문서에서는 Lenovo ThinkSystem SE350 Edge Server와 엔트리 레벨 NetApp AFF 및 EF-Series 스토리지 시스템으로 구성된 컴퓨팅 및 스토리지 구성의 테스트 및 검증을 다룹니다. 참조 아키텍처는 AI 배포를 위한 효율적이고 비용 효율적인 솔루션을 제공하는 동시에 NetApp ONTAP 및 NetApp SANtricity 데이터 관리 소프트웨어를 통해

포괄적인 데이터 서비스, 통합 데이터 보호, 원활한 확장성 및 클라우드 연결 데이터 스토리지를 제공합니다.

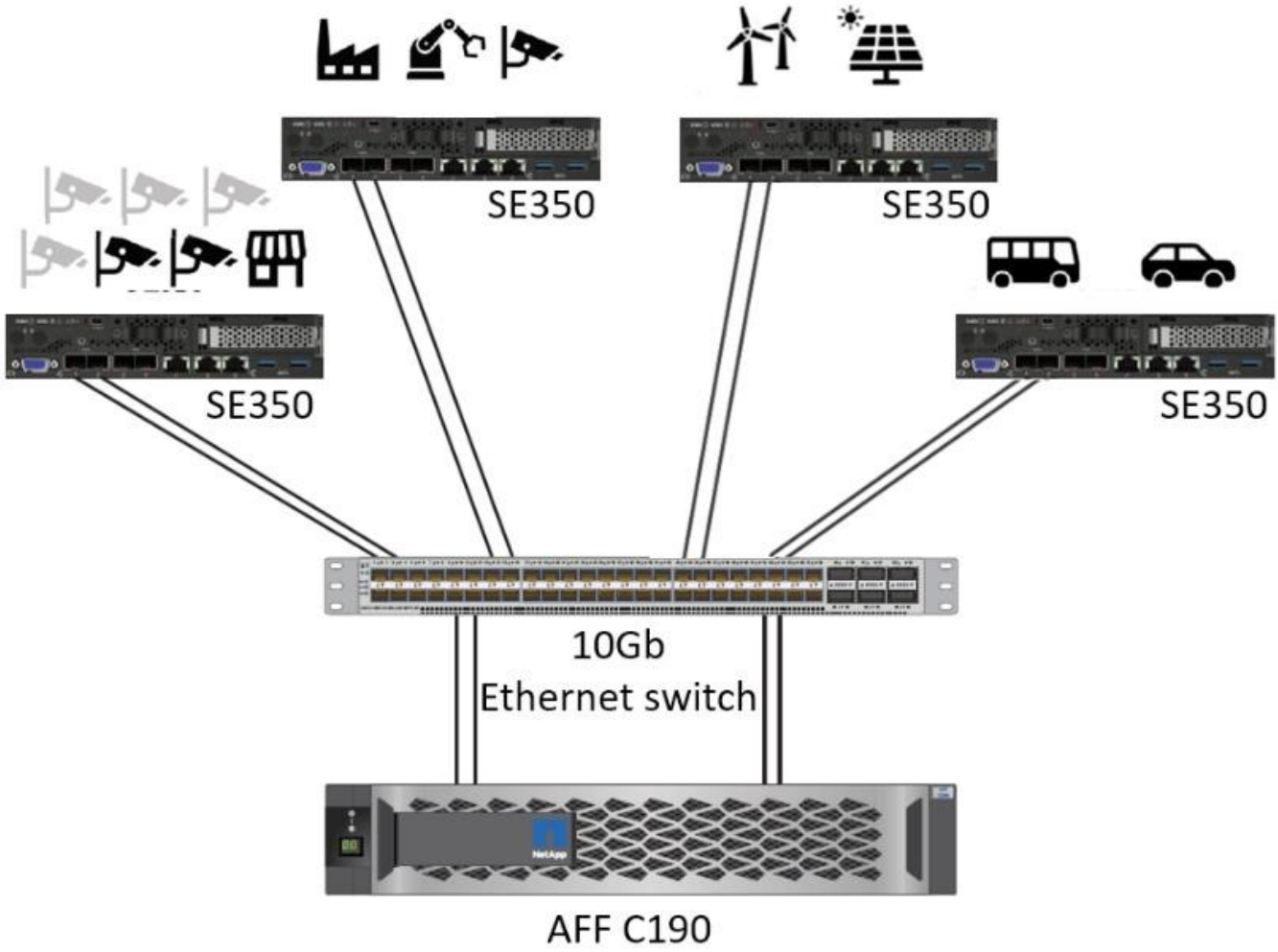
대상

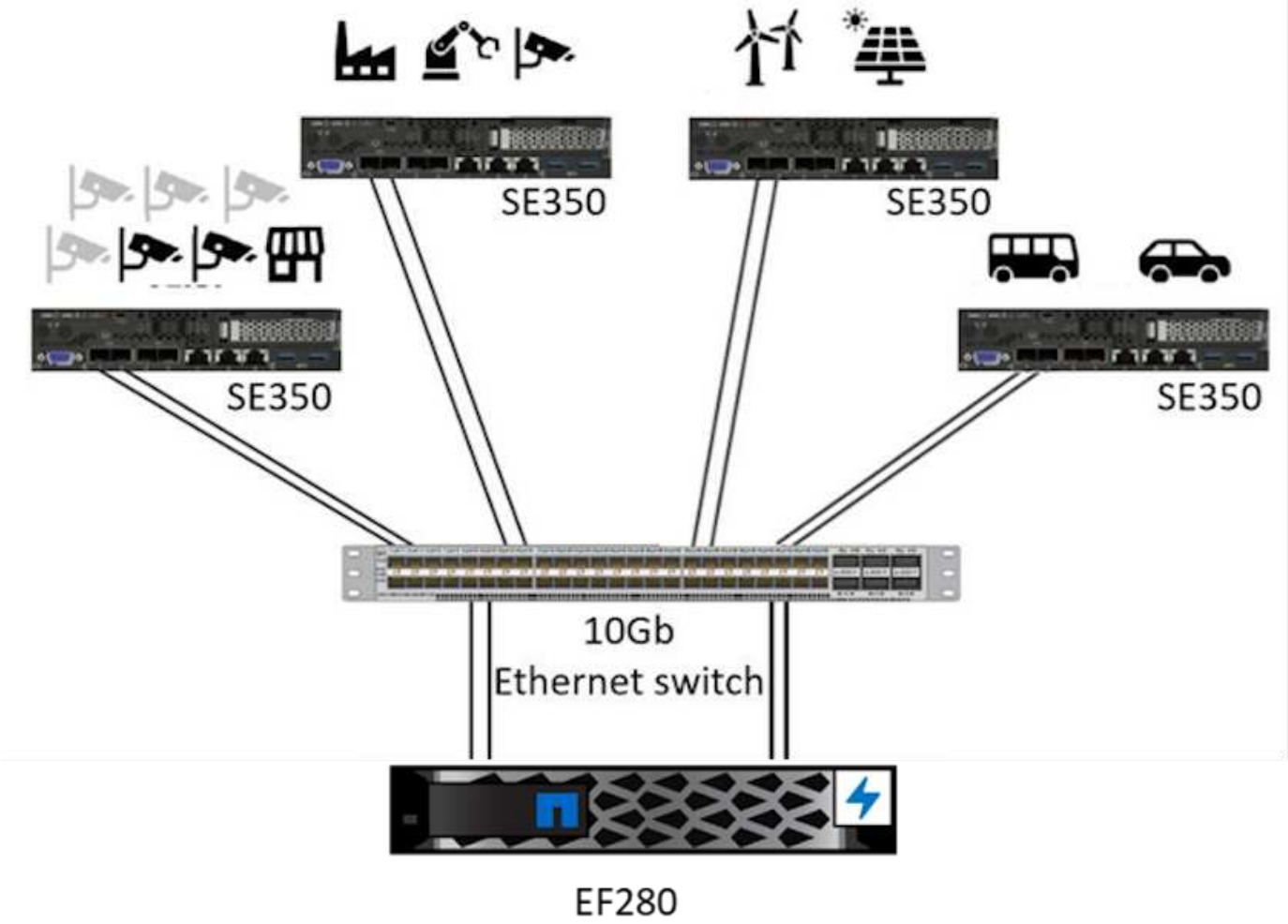
이 문서는 다음 사용자를 대상으로 합니다.

- 에지의 AI를 제품화하려는 비즈니스 리더 및 엔터프라이즈 설계자
- 데이터 과학자, 데이터 엔지니어, AI/기계 학습(ML) 연구원 및 AI 시스템 개발자.
- AI/ML 모델 및 애플리케이션 개발을 위한 솔루션을 설계하는 엔터프라이즈 설계자
- 딥 러닝(DL) 및 ML 모델을 구축하는 효율적인 방법을 찾고 있는 데이터 과학자 및 AI 엔지니어
- 에지 장치 관리자 및 에지 서버 관리자는 에지 추론 모델의 구축과 관리를 담당합니다.

솔루션 아키텍처

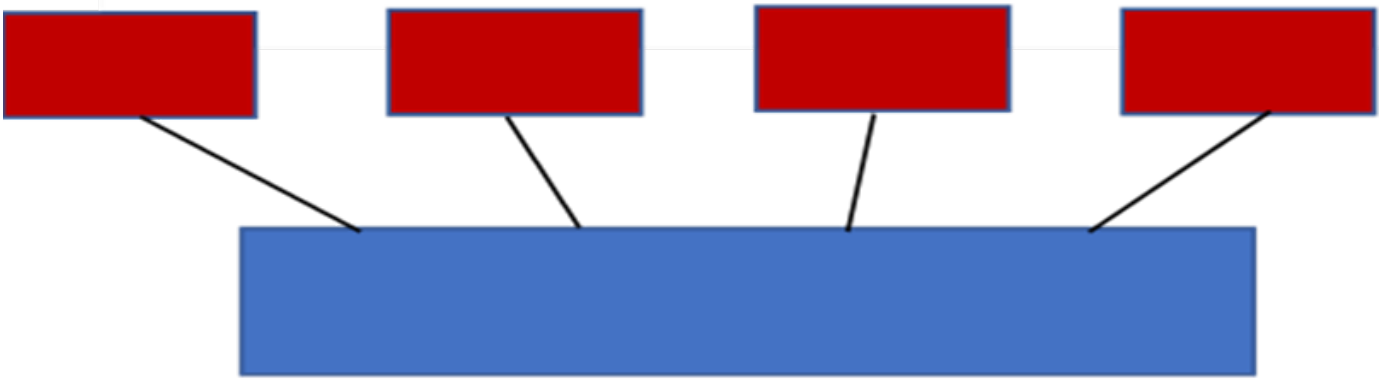
이 Lenovo ThinkSystem 서버 및 NetApp ONTAP 또는 NetApp SANtricity 스토리지 솔루션은 기존 CPU와 함께 GPU의 처리 능력을 사용하여 대규모 데이터 세트에서 AI 추론을 처리하도록 설계되었습니다. 이 검증 방식은 다음 두 그림에 표시된 대로 단일 NetApp AFF 스토리지 시스템과 상호 연결된 단일 또는 다중 Lenovo SR350 에지 서버를 사용하는 아키텍처로 고성능 및 최적의 데이터 관리를 수행하는 것입니다.





다음 그림의 논리적 아키텍처 개요에서는 이 아키텍처의 컴퓨팅 및 스토리지 요소 역할을 보여 줍니다. 특히 다음과 같은 사항이 표시됩니다.

- 에지 컴퓨팅 장치가 카메라, 센서 등의 데이터를 기반으로 추론을 수행합니다.
- 다양한 용도로 사용되는 공유 스토리지 요소:
  - 추론 모델과 추론을 수행하는 데 필요한 다른 데이터를 위한 중심 위치를 제공합니다. 컴퓨팅 서버는 스토리지를 직접 액세스하고 로컬에서 복사할 필요 없이 네트워크 전체에서 추론 모델을 사용합니다.
  - 업데이트된 모델이 여기에 푸시됩니다.
  - 에지 서버가 나중에 분석할 수 있도록 수신하는 입력 데이터를 보관합니다. 예를 들어, 에지 장치가 카메라에 연결된 경우 저장소 요소는 카메라에서 캡처한 비디오를 유지합니다.



빨간색	파란색
Lenovo 컴퓨팅 시스템	NetApp AFF 스토리지 시스템
카메라, 센서 등의 입력에서 추론을 수행하는 에지 장치	추측 모델과 에지 디바이스의 데이터를 저장하는 공유 스토리지로, 추후 분석 지원

이 NetApp 및 Lenovo 솔루션은 다음과 같은 주요 이점을 제공합니다.

- 소규모 지사 또는 부서에서의 GPU 가속 컴퓨팅.
- 공유 스토리지에서 백업 및 관리되는 다중 에지 서버 배포
- 데이터 손실 없이 낮은 RPO(복구 시점 목표) 및 RTO(복구 시간 목표)를 충족하는 강력한 데이터 보호
- NetApp Snapshot 복사본 및 클론을 통해 데이터 관리를 최적화하여 개발 워크플로우를 간소화합니다.

이 아키텍처를 사용하는 방법

이 문서에서는 제안된 아키텍처의 설계 및 성능을 검증합니다. 하지만 NetApp은 특정 소프트웨어 수준의 컨테이너, 워크로드, 모델 관리, 클라우드 또는 온프레미스의 데이터 센터 등과 같은 특정 소프트웨어 레벨 구성 요소를 테스트하지 않았습니다. 이러한 소프트웨어 레벨 구성 요소가 배포 시나리오에 한정되어 있기 때문입니다. 여기에는 여러 개의 선택 사항이 있습니다.

컨테이너 관리 수준에서 Kubernetes 컨테이너 관리는 좋은 선택이며 전체 업스트림 버전(Canonical) 또는 엔터프라이즈 배포에 적합한 수정 버전(Red Hat)에서 지원됩니다. 를 클릭합니다 "NetApp AI Control Plane" NetApp Trident 및 새로 추가된 Trident를 사용합니다 "NetApp DataOps 툴킷" 데이터 과학자 및 데이터 엔지니어가 NetApp 스토리지와 통합할 수 있도록 추적 가능성, 데이터 관리 기능, 인터페이스 및 툴을 기본으로 제공합니다. Kubernetes용 ML 툴킷인 Kubeflow는 추가 AI 기능을 제공하는 동시에 TensorFlow Serving 또는 NVIDIA Triton Inference Server와 같은 여러 플랫폼에서 모델 버전 관리 및 KFServing을 지원합니다. 또 다른 옵션은 NVIDIA EGX 플랫폼으로, GPU 지원 AI 추론 컨테이너 카탈로그에 액세스하여 워크로드 관리를 제공합니다. 그러나 이러한 옵션을 사용하려면 운영 환경에 투입하기 위해 상당한 노력과 전문 지식이 필요할 수 있으며 타사 ISV(독립 소프트웨어 공급업체) 또는 컨설턴트의 도움이 필요할 수 있습니다.

솔루션 영역

AI 추론 및 에지 컴퓨팅의 주요 이점은 지연 시간 없이 높은 수준의 품질로 데이터를 컴퓨팅, 처리 및 분석할 수 있는 장치의 기능입니다. 이 문서에서 설명하는 에지 컴퓨팅 사용 사례는 매우 많지만 다음과 같은 몇 가지 대표적인 사례가 있습니다.

자동차: 자율주행 차량

전형적인 에지 컴퓨팅 일러스트는 자율주행 차량(AV)의 첨단 운전자 지원 시스템(ADAS)에 포함되어 있습니다. 무인 자동차의 AI는 안전하고 성공적인 운전자가 되려면 카메라와 센서의 많은 데이터를 신속하게 처리해야 합니다. 물체와 사람 사이의 해석에 너무 많은 시간이 걸릴경우 생명 또는 사망이 발생할 수 있으므로 데이터를 최대한 차량과 가깝게 처리할 수 있어야 합니다. 이 경우 하나 이상의 에지 컴퓨팅 서버가 카메라, 레이더, LiDAR 및 기타 센서의 입력을 처리하는 동시에 공유 스토리지에는 추론 모델이 저장되고 센서의 입력 데이터가 저장됩니다.

의료: 환자 모니터링

AI 및 에지 컴퓨팅이 미치는 가장 큰 영향 중 하나는 가정 및 중환자실(ICU) 모두에서 만성 질환 환자를 지속적으로 모니터링할 수 있는 기능입니다. 인슐린 수치, 호흡, 신경학적 활동, 심장 리듬 및 위장관 기능을 모니터링하는 에지 장치에서 얻은 데이터는 다른 사람의 생명을 구하기 위한 제한된 시간이 있기 때문에 즉시 실행되어야 하는 데이터에 대한 즉각적인 분석이 필요합니다.

소매**:** 계산원 없는 지불

에지 컴퓨팅은 유통업체가 계산 시간을 단축하고 발트 트래픽을 늘릴 수 있도록 AI 및 ML을 지원합니다. 계산원이 필요 없는 시스템은 다음과 같은 다양한 구성 요소를 지원합니다.

- 인증 및 액세스. 물리적 쇼핑객을 검증된 계정에 연결하고 소매 공간에 대한 액세스를 허용합니다.
- 인벤토리 모니터링. 센서, RFID 태그 및 컴퓨터 비전 시스템을 사용하여 쇼핑객의 아이템 선택 또는 선택 취소를 확인할 수 있습니다.

  여기서 각 에지 서버는 각 계산 카운터를 처리하며 공유 스토리지 시스템은 중앙 동기화 지점으로 사용됩니다.

금융 서비스**:** 키오스크의 인적 안전 및 사기 방지

은행 조직에서는 AI 및 에지 컴퓨팅을 사용하여 혁신을 진행하고 맞춤형 뱅킹 경험을 만들고 있습니다. 실시간 데이터 분석 및 AI 추론을 사용하는 대화형 키오스크는 이제 ATM을 통해 고객이 돈을 인출할 수 있도록 지원할 뿐만 아니라 카메라에서 캡처한 이미지를 통해 키오스크를 사전 예방적으로 모니터링하여 사람의 안전 또는 사기 행위 위험을 식별할 수 있습니다. 이 시나리오에서는 에지 컴퓨팅 서버 및 공유 스토리지 시스템이 대화형 키오스크 및 카메라에 연결되어 은행이 AI 추론 모델로 데이터를 수집하고 처리할 수 있도록 도와줍니다.

제조**: Industry 4.0**

4차 산업혁명(Industry 4.0)은 Smart Factory 및 3D 프린팅과 같은 새로운 트렌드와 함께 시작되었습니다. 데이터 중심의 미래에 대비하기 위해 대규모 M2M(Machine-to-Machine) 통신 및 IoT가 통합되어 사람의 개입 없이 자동화 수준을 높일 수 있습니다. 제조는 이미 고도로 자동화되어 있으며 AI 기능을 추가하는 것은 장기적인 추세를 자연스럽게 이어주는 것입니다. AI를 사용하면 컴퓨터 비전 및 기타 AI 기능을 활용하여 자동화할 수 있는 운영을 자동화할 수 있습니다. 제조 공장이 안전 및 품질 관리에 필요한 ISO 표준을 충족할 수 있도록 제조 공장의 조립 라인에서 자재를 더 빠르게 분석하는 데 있어 인간의 시각이나 의사 결정에 의존하는 품질 관리 또는 작업을 자동화할 수 있습니다. 여기서 각 컴퓨팅 에지 서버는 제조 프로세스를 모니터링하는 센서 배열에 연결되고 필요에 따라 업데이트된 추론 모델이 공유 스토리지로 푸시됩니다.

통신**: Rust** 감지**,** 타워 검사 및 네트워크 최적화

통신 업계에서는 컴퓨터 비전과 AI 기술을 사용하여 녹을 자동으로 탐지하고 부식된 셀 타워를 식별하는 이미지를 처리하여 추가적인 검사가 필요합니다. 드론 이미지와 AI 모델을 사용하여 타워의 특정 영역을 식별하고 녹, 표면 균열 및 부식을 분석하는 일이 최근 몇 년 사이에 증가했습니다. 통신 인프라와 셀 타워를 효율적으로 검사하고, 정기적으로 성능 저하를 평가하며, 필요할 때 신속하게 수리할 수 있는 AI 기술에 대한 수요가 지속적으로 증가하고 있습니다.

또한, 데이터 트래픽 패턴을 예측하고 5G 지원 장치를 감지하고 MIMO(다중 입력 및 다중 출력) 에너지 관리를 자동화 및 보강하기 위해 AI 및 ML 알고리즘을 사용하는 것도 통신 업계의 새로운 사용 사례입니다. MIMO 하드웨어는 무선 타워에서 네트워크 용량을 늘리기 위해 사용되지만, 추가 에너지 비용이 필요합니다. 셀 사이트에 배치된 “MIMO 절전 모드”용 ML 모델은 무전기의 효율적인 사용을 예측하고 모바일 네트워크 사업자(MNO)의 에너지 소비 비용을 줄이는 데 도움이 됩니다. AI 추론 및 에지 컴퓨팅 솔루션은 MNO가 데이터 센터로 주고받는 데이터 양을 줄이고, TCO를 낮추고, 네트워크 운영을 최적화하고, 최종 사용자의 전반적인 성능을 개선하는 데 도움이 됩니다.

기술 개요

이 섹션에서는 AI 솔루션의 기술 기반에 대해 설명합니다.

NetApp AFF 시스템

최첨단 NetApp AFF 스토리지 시스템을 사용하면 AI 추론 구축을 통해 에지에서 업계 최고 수준의 성능, 탁월한 유연성,

클라우드 통합, 동급 최고의 데이터 관리로 엔터프라이즈 스토리지 요구사항을 충족할 수 있습니다. 플래시 전용으로 설계된 NetApp AFF 시스템은 비즈니스 크리티컬 데이터를 더 빠르게 처리하고 관리, 보호할 수 있도록 지원합니다.

- 엔트리 레벨 NetApp AFF 스토리지 시스템은 FAS2750 하드웨어 및 SSD 플래시 미디어를 기반으로 합니다
- HA 구성의 컨트롤러 2개

NetApp 엔트리 레벨 AFF C190 스토리지 시스템은 다음 기능을 지원합니다.

- 최대 드라이브 수는 24x 960GB SSD입니다
- 두 가지 가능한 구성:
  - 이더넷(10GbE): 10GBASE-T(RJ-45) 포트 4개
  - 유니파이드(16Gb FC 또는 10GbE): 4x UTA2(Unified Target Adapter 2) 포트
- 최대 50.5TB의 유효 용량

> NAS 워크로드의 경우, 단일 엔트리 레벨 AFF C190 시스템은 연속 읽기의 경우 4.4GBps의 처리량과 작은 랜덤 읽기의 경우 1ms 이하의 지연 시간으로 230K IOPS를 지원합니다.

NetApp AFF A220을 참조하십시오

또한, NetApp은 대규모 구축을 위해 더 뛰어난 성능과 확장성을 제공하는 다른 엔트리급 스토리지 시스템을 제공합니다. NAS 워크로드의 경우 단일 엔트리 레벨 AFF A220 시스템이 다음을 지원합니다.

- 순차적 읽기의 경우 6.2GBps의 처리량
- 375K IOPS, 1ms 미만의 지연 시간으로 소규모 랜덤 읽기 지원
- 최대 드라이브 수는 144x 960GB, 3.8TB 또는 7.6TB SSD입니다
- AFF A220은 1PB 이상의 실제 용량으로 확장됩니다

NetApp AFF A250

- 최대 실제 용량은 35PB이며 최대 스케일아웃 2-24개 노드(HA 쌍 12개)를 지원하는 경우
- AFF A220보다 45% 이상 높은 성능 향상을 제공합니다
- 440k IOPS 랜덤 읽기 @ 1ms
- 최신 NetApp ONTAP 릴리스 ONTAP 9.8을 기반으로 구축
- HA 및 클러스터 인터커넥트에 2개의 25GB 이더넷을 활용합니다

NetApp E-Series EF 시스템

EF-Series는 엔트리 레벨 및 미드레인지 All-Flash SAN 스토리지 어레이 제품군으로, NetApp SANtricity 소프트웨어를 사용하여 데이터에 더 빠르게 액세스하고 가치를 더 빠르게 창출할 수 있습니다. 이러한 시스템은 SAS 및 NVMe 플래시 스토리지를 모두 제공하며 경제적인 가격으로 최고 수준의 IOPS, 100마이크로초 미만의 응답 시간, 최대 44GBps의 대역폭을 제공하므로 AI 추론 및 고성능 컴퓨팅(HPC)과 같은 까다로운 애플리케이션과 혼합 워크로드에 적합합니다.

다음 그림에서는 NetApp EF280 스토리지 시스템을 보여 줍니다.

NetApp EF280

- 32Gb/16Gb FC, 25Gb/10Gb iSCSI 및 12Gb SAS 지원
- 최대 실제 용량은 총 1.5PB의 96개 드라이브입니다
- 10GBps 처리량(순차적 읽기)
- 300K IOPS(랜덤 읽기)
- NetApp EF280은 NetApp 포트폴리오에서 가장 경제적인 All-Flash 어레이(AFA)입니다

NetApp EF300

- 24x NVMe SSD 드라이브로 총 367TB 용량 지원
- 총 240x NL-SAS HDD, 96x SAS SSD 또는 그 조합 확장 옵션
- 100Gb NVMe/IB, NVMe/RoCE, iSER/IB 및 SRP/IB
- 32Gb NVMe/FC, FCP

- 25GB iSCSI
- 20GBps(순차적 읽기)
- 670K IOPS(랜덤 읽기)

> 자세한 내용은 를 참조하십시오 "NetApp EF-Series NetApp EF-Series All-Flash 어레이 EF600, F300, EF570, EF280 데이터시트".

NetApp ONTAP 9

NetApp의 최신 세대 스토리지 관리 소프트웨어인 ONTAP 9.8.1을 통해 기업은 인프라를 현대화하고 클라우드 지원 데이터 센터로 전환할 수 있습니다. ONTAP는 업계 최고 수준의 데이터 관리 기능을 활용하여 데이터가 상주하는 위치와 상관없이 단일 툴셋으로 데이터를 관리하고 보호할 수 있습니다. 필요에 따라 에지, 코어, 클라우드 등 어느 위치로도 데이터를 자유롭게 이동할 수 있습니다. ONTAP 9.8.1에는 데이터 관리를 단순화하고, 중요 데이터를 더 빨리 처리하고 보호하고, 하이브리드 클라우드 아키텍처 전반에서 차세대 인프라 기능을 지원하는 다양한 기능이 포함되어 있습니다.

데이터 관리를 단순화하십시오

애플리케이션 및 데이터 세트에 적절한 리소스가 사용될 수 있도록 데이터 관리는 엔터프라이즈 IT 운영에 매우 중요합니다. ONTAP에는 운영을 간소화 및 단순화하고 총 운영 비용을 절감할 수 있는 다음과 같은 기능이 포함되어 있습니다.

- * 인라인 데이터 컴팩션 및 확대된 중복제거. * 데이터 컴팩션은 스토리지 블록 내부의 낭비되는 공간을 줄이고, 중복제거는 실제 용량을 크게 증가시킵니다. 이는 로컬에 저장된 데이터와 클라우드로 계층화된 데이터에 적용됩니다.
- * 최소, 최대 및 적응형 서비스 품질(AQoS). * 세분화된 서비스 품질(QoS) 제어는 공유 수준이 높은 환경에서 중요 애플리케이션의 성능 수준을 유지하는 데 도움이 됩니다.
- * NetApp FabricPool. * 이 기능은 콜드 데이터를 AWS(Amazon Web Services), Azure, NetApp StorageGRID 스토리지 솔루션을 포함한 퍼블릭 및 프라이빗 클라우드 스토리지 옵션으로 자동 계층화합니다. FabricPool에 대한 자세한 내용은 를 참조하십시오 "TR-4598".

데이터 가속화 및 보호

ONTAP 9은 탁월한 수준의 성능과 데이터 보호를 제공하며 다음과 같은 방법으로 이러한 기능을 확장합니다.

- * 성능 및 낮은 지연 시간 * ONTAP는 가장 짧은 지연 시간으로 가장 높은 처리량을 제공합니다.
- * 데이터 보호. * ONTAP는 모든 플랫폼에서 공통 관리를 지원하는 내장 데이터 보호 기능을 제공합니다.
- * NVE(NetApp 볼륨 암호화). * ONTAP는 온보드 및 외부 키 관리를 모두 지원하여 네이티브 볼륨 레벨 암호화를 제공합니다.
- * 멀티테넌시 및 다단계 인증 * ONTAP를 통해 인프라 리소스를 최고 수준의 보안으로 공유할 수 있습니다.

미래 지향형 인프라

ONTAP 9은 다음과 같은 기능을 통해 지속적으로 변화하는 까다로운 비즈니스 요구사항을 충족할 수 있도록 지원합니다.

- * 원활한 확장 및 무중단 운영 * ONTAP은 기존 컨트롤러 및 스케일아웃 클러스터에 무중단으로 용량을 추가할 수 있도록 지원합니다. 고객은 고비용이 따르는 데이터 마이그레이션이나 운영 중단 없이 NVMe 및 32Gb FC와 같은 최신 기술로 업그레이드할 수 있습니다.

- * 클라우드 연결. * ONTAP은 클라우드에 가장 많이 연결되는 스토리지 관리 소프트웨어로, 모든 퍼블릭 클라우드에서 소프트웨어 정의 스토리지(ONTAP Select) 및 클라우드 네이티브 인스턴스(NetApp Cloud Volumes Service) 옵션이 제공됩니다.
- 새로운 애플리케이션과의 통합 * ONTAP는 기존 엔터프라이즈 앱을 지원하는 인프라와 동일한 인프라를 사용하여 자율주행 차량, 스마트 시티, Industry 4.0과 같은 차세대 플랫폼 및 애플리케이션을 위한 엔터프라이즈급 데이터 서비스를 제공합니다.

NetApp SANtricity를 참조하십시오

NetApp SANtricity는 E-Series 하이브리드 플래시 및 EF-Series All-Flash 어레이에 업계 최고의 성능, 안정성, 단순성을 제공하도록 설계되었습니다. 데이터 분석, 비디오 감시, 백업 및 복구 등 워크로드가 많은 애플리케이션에서 E-Series 하이브리드 플래시 및 EF-Series All-Flash 어레이의 성능과 활용률을 극대화합니다. SANtricity를 사용하면 스토리지를 온라인 상태로 유지하면서 구성 조정, 유지 관리, 용량 확장 및 기타 작업을 완료할 수 있습니다. 또한 SANtricity는 사용하기 쉬운 온박스형 시스템 관리자 인터페이스를 통해 뛰어난 데이터 보호, 사전 예방 모니터링 및 인증 보안을 제공합니다. 자세한 내용은 를 참조하십시오 "NetApp E-Series SANtricity 소프트웨어 데이터시트 를 참조하십시오".

최적의 성능

성능에 최적화된 SANtricity 소프트웨어는 모든 데이터 분석, 비디오 감시 및 백업 앱에 높은 IOPS 및 처리량과 짧은 지연 시간으로 데이터를 제공합니다. IOPS가 높고 지연 시간이 짧은 애플리케이션과 대역폭과 처리량이 높은 애플리케이션의 성능을 더욱 높이십시오.

가동 시간 극대화

스토리지가 온라인 상태일 때 모든 관리 작업을 완료하십시오. I/O를 중단하지 않고 구성을 변경하거나, 유지보수를 수행하거나, 용량을 확장할 수 있습니다 자동화된 기능, 온라인 구성, 최첨단 DPP(Dynamic Disk Pool) 기술 등을 통해 동급 최고의 안정성을 실현합니다.

편안한 휴식

SANtricity 소프트웨어는 사용이 간편한 온박스형 시스템 관리자 인터페이스를 통해 뛰어난 데이터 보호, 사전 예방 모니터링 및 인증 보안을 제공합니다. 스토리지 관리 업무를 간소화합니다. 모든 E-Series 스토리지 시스템의 고급 튜닝에 필요한 유연성 확보 언제 어디서나 NetApp E-Series 시스템을 관리할 수 있습니다. NetApp의 온박스 웹 기반 인터페이스는 관리 워크플로우를 간소화합니다.

NetApp 트라이던트

"트라이던트" NetApp은 Docker 및 Kubernetes용 오픈 소스 동적 스토리지 오케스트레이터로서 영구 스토리지의 생성, 관리 및 사용을 단순화합니다. Kubernetes 네이티브 애플리케이션인 Trident는 Kubernetes 클러스터 내에서 직접 실행됩니다. Trident를 사용하면 고객이 DL 컨테이너 이미지를 NetApp 스토리지에 원활하게 배포하고 AI 컨테이너 배포를 위한 엔터프라이즈급 경험을 제공할 수 있습니다. Kubernetes 사용자(예: ML 개발자 및 데이터 과학자)는 오케스트레이션 및 클론 복제를 생성, 관리 및 자동화하여 NetApp 기술이 제공하는 NetApp 고급 데이터 관리 기능을 활용할 수 있습니다.

NetApp BlueXP 복사 및 동기화

"BlueXP 복사 및 동기화" 는 빠르고 안전한 데이터 동기화를 제공하는 NetApp 서비스입니다. 온프레미스 NFS 또는 SMB 파일 공유 간에 파일을 전송해야 하는 경우, NetApp StorageGRID, NetApp ONTAP S3, NetApp Cloud Volumes Service, Azure NetApp Files, Amazon Simple Storage Service(Amazon S3), Amazon Elastic File System(Amazon EFS), Azure Blob, Google Cloud Storage, 또는 IBM Cloud Object Storage인 BlueXP Copy and Sync는 필요한 파일을 빠르고 안전하게 이동합니다. 데이터가 전송되면 소스와 타겟 모두에서 사용할 수 있습니다.

BlueXP 복사 및 동기화는 미리 정의된 일정에 따라 데이터를 지속적으로 동기화하므로 변경된 부분만 이동하므로 데이터 복제에 소비되는 시간과 비용이 최소화됩니다. BlueXP Copy and Sync는 매우 간단하게 설정하고 사용할 수 있는 서비스형 소프트웨어(SaaS) 툴입니다. BlueXP Copy 및 Sync에 의해 트리거되는 데이터 전송은 데이터 브로커에 의해 수행됩니다. AWS, Azure, Google Cloud Platform 또는 사내에 BlueXP Copy 및 Sync 데이터 브로커를 배포할 수 있습니다.

Lenovo ThinkSystem 서버

Lenovo ThinkSystem 서버는 현재 고객의 과제를 해결하고 미래의 과제를 해결할 수 있는 혁신적인 모듈식 설계 접근 방식을 제공하는 혁신적인 하드웨어, 소프트웨어 및 서비스를 갖추고 있습니다. 이러한 서버는 동급 최강의 업계 표준 기술과 차별화된 Lenovo의 혁신적인 기술을 결합하여 x86 서버에서 최대한의 유연성을 제공합니다.

Lenovo ThinkSystem 서버 배포의 주요 이점은 다음과 같습니다.

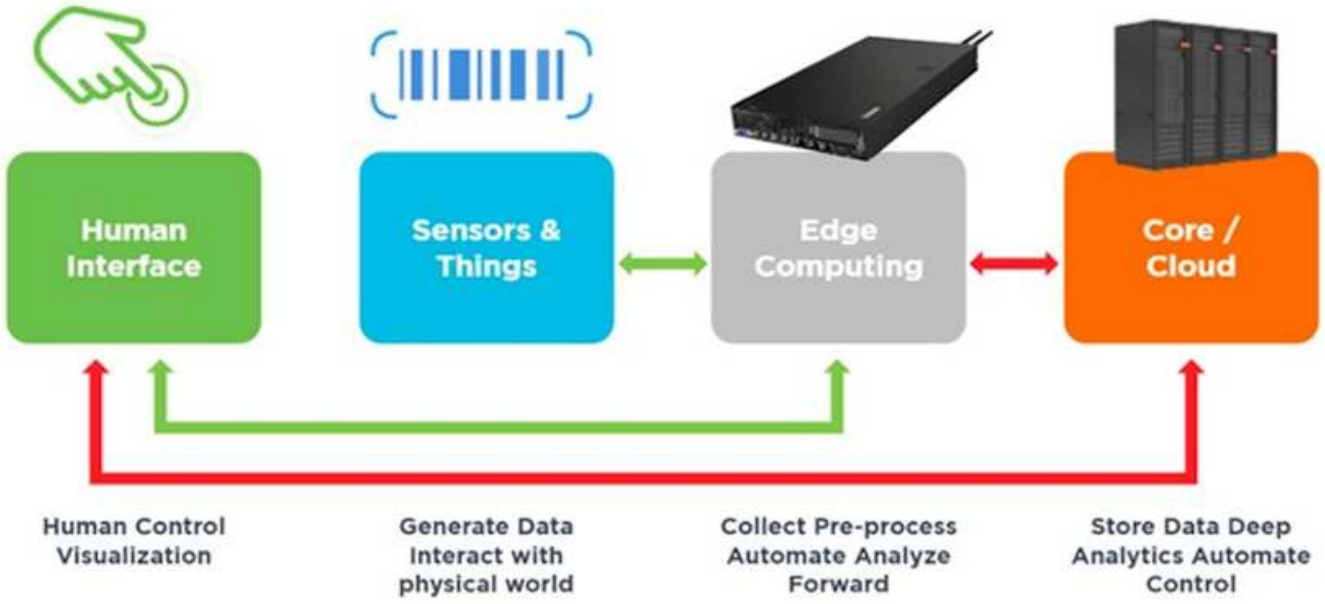
- 비즈니스 성장에 맞춰 확장할 수 있는 모듈식 설계
- 업계 최고 수준의 복원력으로 예기치 못한 가동 중지의 비용이 많이 드는 시간을 절약할 수 있습니다
- 빠른 플래시 기술을 통해 지연 시간을 단축하고, 응답 시간을 단축하며, 데이터 관리를 실시간으로 수행할 수 있습니다

AI 분야에서 Lenovo는 기업들이 워크로드에 대한 ML 및 AI의 이점을 이해하고 적용할 수 있도록 실질적인 접근 방식을 취하고 있습니다. Lenovo 고객은 Lenovo AI Innovation Center의 Lenovo AI 제품을 살펴보고 평가하여 해당 사용 사례의 가치를 완벽하게 파악할 수 있습니다. 가치 창출 시간을 단축하기 위해 이 고객 중심 접근 방식은 AI에 사용하고 최적화할 수 있는 솔루션 개발 플랫폼에 대한 개념 증명을 고객에게 제공합니다.

Lenovo ThinkSystem SE350 Edge 서버

에지 컴퓨팅을 사용하면 데이터 센터 또는 클라우드로 전송되기 전에 네트워크 에지에서 IoT 장치의 데이터를 분석할 수 있습니다. 아래 그림과 같이 Lenovo ThinkSystem SE350은 견고하며 환경 친화적인 소형 폼 팩터에서 유연성, 연결, 보안 및 원격 관리 기능에 중점을 두고 엣지에서의 배포를 위한 고유한 요구 사항을 충족하도록 설계되었습니다.

에지 AI 워크로드에 대한 가속화를 지원할 수 있는 유연성을 갖춘 인텔 제온 D 프로세서를 장착한 SE350은 데이터 센터 외부의 다양한 환경에서 서버 배포의 과제를 해결하기 위해 특별히 제작되었습니다.



MLPerf

MLPerf는 AI 성능 평가를 위한 업계 최고의 벤치마크 제품군입니다. 여기에는 영상 분류, 물체 감지, 의료 영상 및 NLP(자연어 처리)를 비롯한 다양한 적용 AI 영역을 다룹니다. 이 검증에서는 이 검증이 완료될 때 MLPerf 추론의 최신 반복인 Inference v0.7 워크로드를 사용했습니다. 를 클릭합니다 "MLPerf Inference v0.7" 데이터 센터 및 에지 시스템을 위한 새로운 벤치마크 4개가 포함된 제품군:

- * BERT. * Transformers(BERT)의 양방향 Encoder Representation은 Squad 데이터 세트를 사용하여 질문 답변에 맞게 미세 조정되었습니다.
- * DLRM. * DLRM(Deep Learning Recommendation Model)은 CTR(Click-Through Rates)을 최적화하도록 교육받은 개인 설정 및 권장 모델입니다.
- * 3D U-Net. * 3D U-Net 아키텍처는 Brain Tumor Segmentation(뇌종양 분할) 데이터 세트에 대한 교육을 받습니다.
- * RNN-T * Recurrent Neural Network Transducer(RNN-T)는 LibriSpeech의 하위 집합에 대한 교육을 받은 자동 음성 인식(ASR) 모델입니다. MLPerf Inference 결과 및 코드는 공개적으로 사용할 수 있으며 Apache 라이센스에 따라 릴리스됩니다. MLPerf Inference에는 다음과 같은 시나리오를 지원하는 Edge 분산이 있습니다.
- * 단일 스트림. * 이 시나리오는 스마트폰에서 실행되는 오프라인 AI 쿼리와 같이 응답성이 중요한 요소인 시스템을 모방합니다. 개별 쿼리가 시스템으로 전송되고 응답 시간이 기록됩니다. 모든 응답의 90번째 백분위수 지연 시간이 결과로 보고됩니다.
- * 멀티스트림. * 이 벤치마크는 여러 센서의 입력을 처리하는 시스템을 위한 것입니다. 테스트 중에 쿼리는 고정된 시간 간격으로 전송됩니다. QoS 제약(허용되는 최대 지연 시간)이 적용됩니다. QoS 제한을 충족하는 동안 시스템에서 처리할 수 있는 스트림의 수를 보고합니다.
- * Offline. * 배치 처리 응용 프로그램을 다루는 가장 간단한 시나리오이며 메트릭은 초당 샘플 처리량입니다. 모든 데이터를 시스템에서 사용할 수 있으며 벤치마크는 모든 샘플을 처리하는 데 걸리는 시간을 측정합니다.

Lenovo는 이 문서에 사용된 서버인 T4가 포함된 SE350에 대한 MLPerf Inference 점수를 게시했습니다. 의 결과를 참조하십시오 "https://mlperf.org/inference-results-0-7/" 입력 #0.7-145의 "Edge, Closed Division" 섹션에 있습니다.

테스트 계획

이 문서는 MLPerf Inference v0.7을 따릅니다 "코드", MLPerf Inference v1.1 "코드", 및 "규칙". 아래 표에 정의된 대로 에지에서 추론을 위해 설계된 MLPerf 벤치마크를 실행했습니다.

영역	작업	모델	데이터 세트	**QSL** 크기	품질	멀티스트림 지연 제한
비전	영상 분류	Resnet50v1.5	ImageNet(224 x224)	1024	FP32의 99%	50ms
비전	물체 감지(대형)	SSD-ResNet34	코코(1200x1200)	64	FP32의 99%	66ms
비전	물체 감지(소형)	SSD - MobileNetsv1	코코(300x300)	256	FP32의 99%	50ms
비전	의료 영상 분할	3D UNET	2019 (224x224x160)	16	FP32의 99% 및 99.9%	해당 없음
음성	텍스트 음성 변환	RNNT	리브리스페흐(LiBrispeech) 개발 - 청소	2513	FP32의 99%	해당 없음
언어	언어 처리	베르	스쿼드 v1.1	10833	FP32의 99%	해당 없음

다음 표에는 Edge 벤치마크 시나리오가 나와 있습니다.

영역	작업	시나리오
비전	영상 분류	단일 스트림, 오프라인, 멀티스트림
비전	물체 감지(대형)	단일 스트림, 오프라인, 멀티스트림
비전	물체 감지(소형)	단일 스트림, 오프라인, 멀티스트림
비전	의료 영상 분할	단일 스트림, 오프라인
음성	텍스트 음성 변환	단일 스트림, 오프라인
언어	언어 처리	단일 스트림, 오프라인

이 검증에서 개발된 네트워크 스토리지 아키텍처를 사용하여 이러한 벤치마크를 수행한 결과 및 이전에 MLPerf에 제출한 에지 서버에서 로컬 실행의 결과를 비교했습니다. 이와 비교하여 공유 스토리지가 추론 성능에 미치는 영향을 확인합니다.

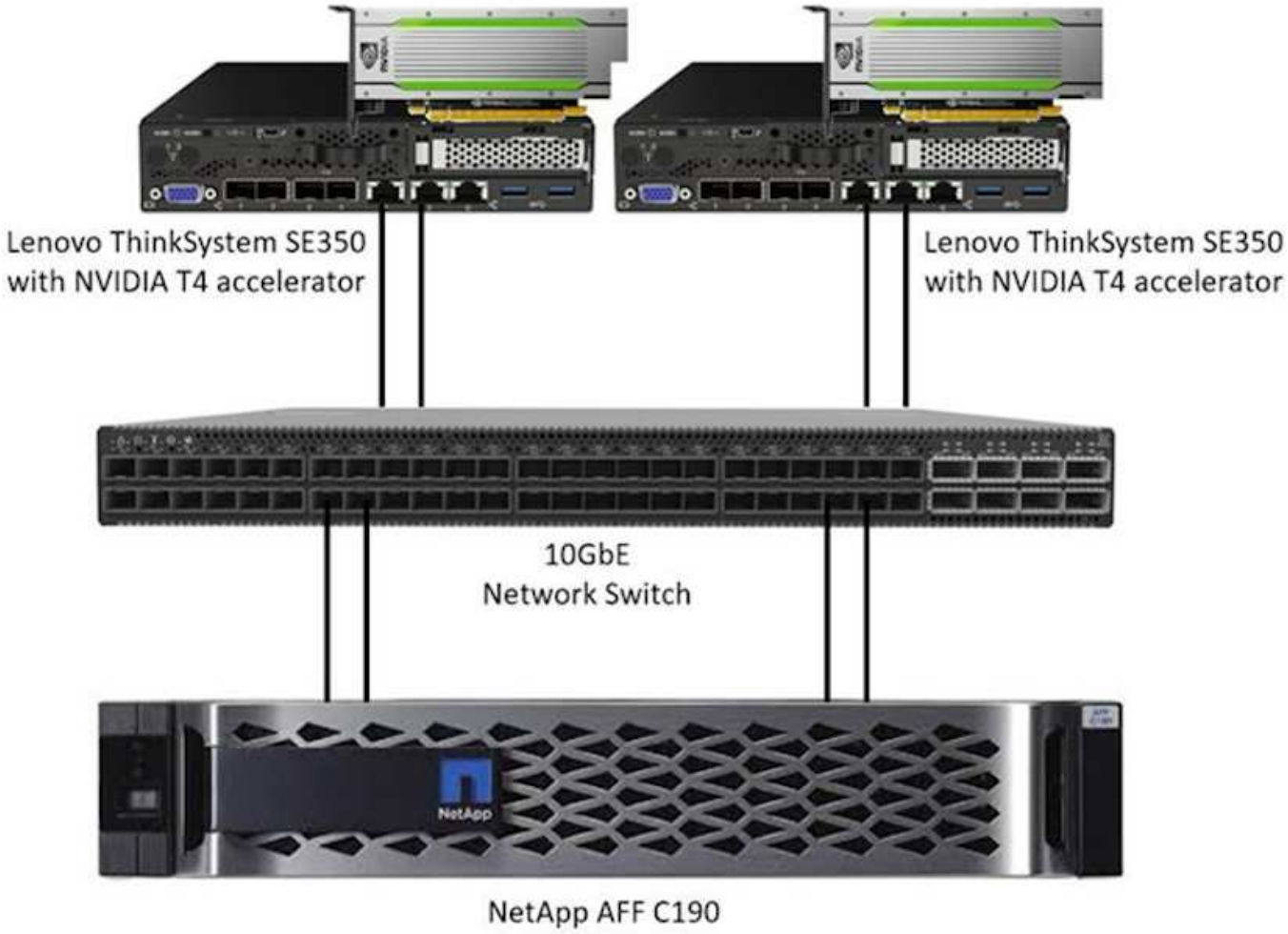
구성을 테스트합니다

다음 그림은 테스트 구성을 보여 줍니다. NetApp AFF C190 스토리지 시스템과 Lenovo ThinkSystem SE350 서버 2대(각각 NVIDIA T4 가속기 1대)를 사용했습니다. 이러한 구성요소는 10GbE 네트워크 스위치를 통해 연결됩니다. 네트워크 스토리지는 검증/테스트 데이터 세트와 사전 교육 모델을 보유하고 있습니다. 서버는 컴퓨팅 기능을 제공하며 스토리지는 NFS 프로토콜을 통해 액세스됩니다.

이 섹션에서는 테스트된 구성, 네트워크 인프라, SE350 서버 및 스토리지 프로비저닝 세부 정보에 대해 설명합니다.

다음 표에서는 솔루션 아키텍처의 기본 구성 요소를 보여 줍니다.

솔루션 구성 요소	세부 정보
Lenovo ThinkSystem 서버	• 각각 NVIDIA T4 GPU 카드 1개가 장착된 SE350 서버 2대
	• 각 서버에는 2.20GHz 및 128GB RAM에서 4개의 물리적 코어가 실행되는 Intel Xeon D-2123IT CPU 1개가 포함되어 있습니다
엔트리 레벨 NetApp AFF 스토리지 시스템(HA 쌍,	• NetApp ONTAP 9 소프트웨어 • 24x 960GB SSD • NFS 프로토콜 • 컨트롤러당 1개의 인터페이스 그룹으로, 마운트 지점에 4개의 논리 IP 주소를 사용합니다



다음 표에는 스토리지 구성이 2RU, 24개 드라이브 슬롯이 포함된 AFF C190에 나와 있습니다.

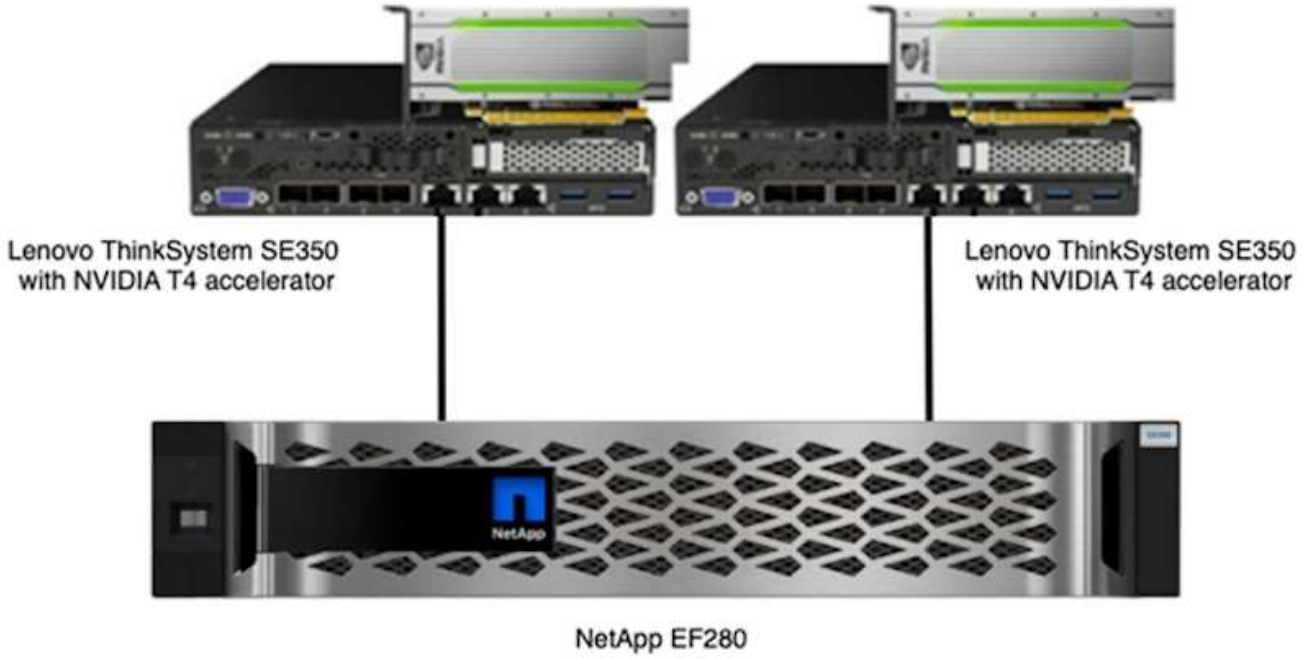
컨트롤러	집계	**FlexGroup** 볼륨	애그리게이트 크기	볼륨 크기	운영 체제 마운트 지점
컨트롤러1	집계1	/netapplenovo_AI_FG	8.42TiB	15TB	/NetApp_Lenovo_FG입니다
컨트롤러 2	집계2		8.42TiB		

/netappLenovo_AI_FG 폴더에는 모델 검증에 사용된 데이터 세트가 포함되어 있습니다.

아래 그림은 테스트 구성을 보여 줍니다. NetApp EF280 스토리지 시스템과 두 개의 Lenovo ThinkSystem SE350 서버(각각 NVIDIA T4 가속기 1개 포함)를 사용했습니다. 이러한 구성요소는 10GbE 네트워크 스위치를 통해 연결됩니다. 네트워크 스토리지는 검증/테스트 데이터 세트와 사전 교육 모델을 보유하고 있습니다. 서버는 컴퓨팅 기능을 제공하며 스토리지는 NFS 프로토콜을 통해 액세스됩니다.

다음 표에는 EF280에 대한 스토리지 구성이 나와 있습니다.

컨트롤러	볼륨 그룹	볼륨	볼륨 크기	**DDPsize**를 참조하십시오	연결 방법
컨트롤러1	DDP1	볼륨 1	8.42TiB	16TB	SE350-1에서 iSCSI LUN 0으로
컨트롤러 2		볼륨 2	8.42TiB		SE350-2를 iSCSI LUN 1로 설정합니다



테스트 절차

이 섹션에서는 이 솔루션을 검증하는 데 사용되는 테스트 절차를 설명합니다.

운영 체제 및 **AI** 추론 설정

AFF C190의 경우 NVIDIA GPU를 지원하고 MLPerf를 사용하는 NVIDIA 드라이버 및 Docker와 함께 Ubuntu 18.04를 사용했습니다 "코드" MLPerf Inference v0.7에 대한 Lenovo 제출의 일부로 사용할 수 있습니다.

EF280의 경우 NVIDIA GPU 및 MLPerf를 지원하는 Ubuntu 20.04와 NVIDIA 드라이버 및 Docker를 사용했습니다 "코드" MLPerf Inference v1.1에 대한 Lenovo 제출의 일부로 제공됩니다.

AI 추론을 설정하려면 다음 단계를 수행하십시오.

1. 등록이 필요한 데이터 세트, ImageNet 2012 검증 세트, Critio Terabyte 데이터 세트 및 브라츠 2019 교육 세트를 다운로드한 다음 파일의 압축을 풉니다.
2. 최소 1TB의 작업 디렉토리를 생성하고 디렉토리를 참조하는 환경 변수 MLPERF_Scratch_path를 정의합니다.

   네트워크 스토리지 활용 사례나 로컬 데이터로 테스트할 때 로컬 디스크에 대해 공유 스토리지에서 이 디렉토리를 공유해야 합니다.

3. make "prebuild" 명령을 실행하여 필요한 추론 작업을 위해 Docker 컨테이너를 빌드하고 실행합니다.

   > 다음 명령은 실행 중인 Docker 컨테이너 내에서 모두 실행됩니다.

   - MLPerf Inference 태스크에 대한 사전 교육 AI 모델 'MAKE download_model'을 다운로드합니다
   - 무료로 다운로드할 수 있는 추가 데이터셋 'make download_data'를 다운로드하세요
   - 데이터 사전 처리: preprocess_data를 만든다
   - 러닝: 메이크 빌드.
   - 컴퓨팅 서버의 GPU에 최적화된 추론 엔진 'make generate_gservers'를 구축합니다
   - 추론 워크로드를 실행하려면 다음 명령을 실행합니다(하나의 명령).

```
make run_harness RUN_ARGS="--benchmarks=<BENCHMARKS>
--scenarios=<SCENARIOS>"
```

AI 추론 실행

세 가지 유형의 실행이 실행되었습니다.

- 로컬 스토리지를 사용하는 단일 서버 AI 추론
- 네트워크 스토리지를 사용하여 단일 서버 AI 추론
- 네트워크 스토리지를 사용하여 다중 서버 AI 추론

테스트 결과

제안된 아키텍처의 성능을 평가하기 위해 다수의 테스트를 실행했습니다.

6가지 워크로드(영상 분류, 물체 감지[소형], 물체 감지[대형], 의료 영상, 텍스트 음성 변환, 및 NLP(Natural Language Processing))를 사용하여 오프라인, 단일 스트림 및 멀티스트림의 세 가지 시나리오에서 실행할 수 있습니다.

마지막 시나리오는 영상 분류 및 물체 감지에 대해서만 구현됩니다.

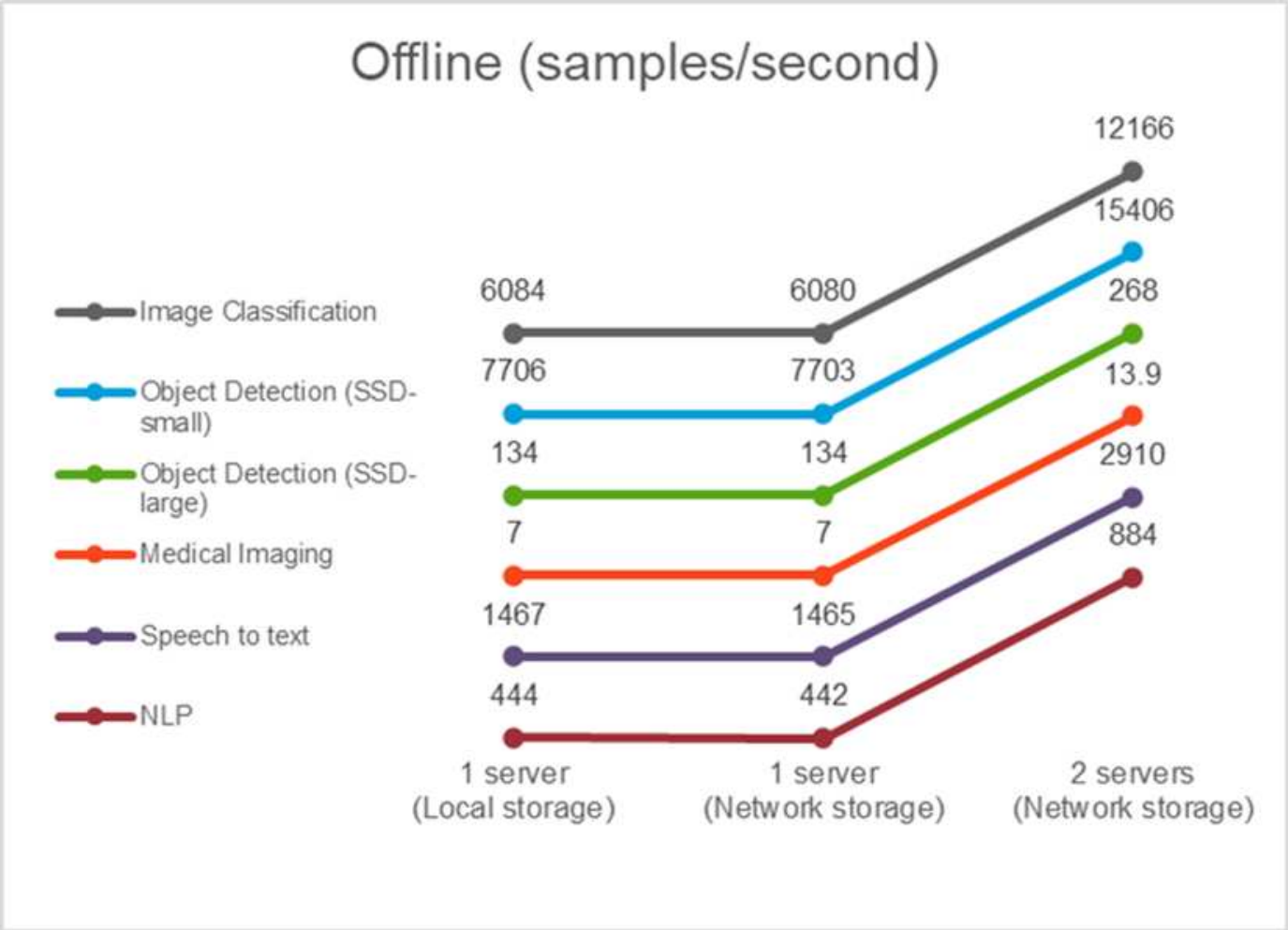
이렇게 하면 다음과 같은 세 가지 다른 설정 하에서 모두 테스트한 15가지 가능한 워크로드가 제공됩니다.

- 단일 서버/로컬 스토리지
- 단일 서버/네트워크 스토리지
- 멀티 서버/네트워크 스토리지

결과는 다음 섹션에 설명되어 있습니다.

AFF의 오프라인 시나리오에서 AI 추론을 사용합니다

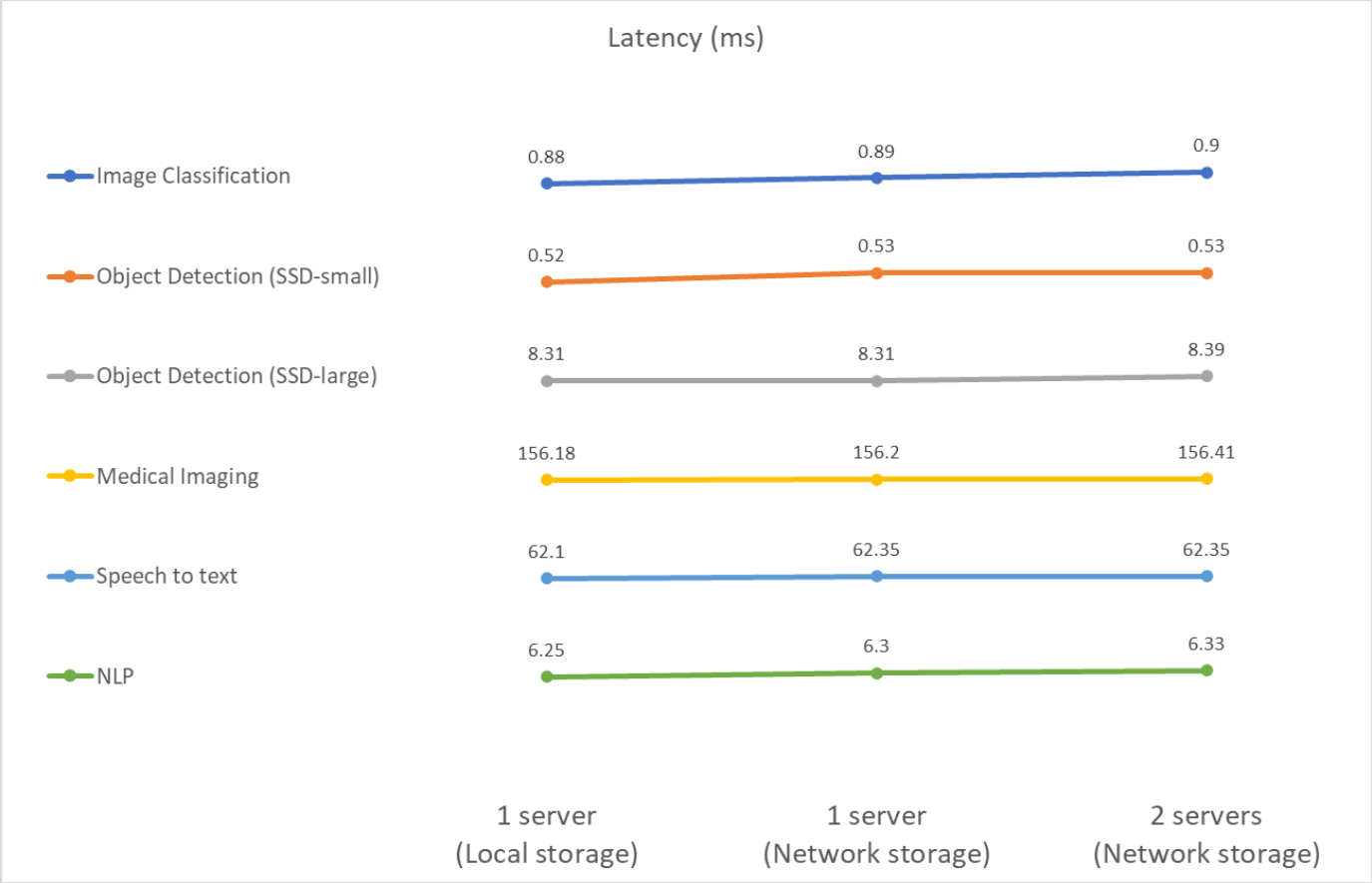
이 시나리오에서는 서버에서 모든 데이터를 사용할 수 있었고 모든 샘플을 처리하는 데 걸린 시간이 측정되었습니다. 테스트 결과로 초당 샘플에 대역폭이 보고됩니다. 두 개 이상의 컴퓨팅 서버를 사용한 경우 모든 서버에 대한 총 대역폭을 합산한 것으로 보고합니다. 아래 그림에서는 세 가지 사용 사례 모두의 결과를 보여 줍니다. 2서버 사례에서는 두 서버의 결합된 대역폭을 보고합니다.



결과에 따르면 네트워크 스토리지는 성능에 부정적인 영향을 주지 않습니다. 변경 사항은 최소이며 일부 작업의 경우 아무것도 발견되지 않습니다. 두 번째 서버를 추가할 때 총 대역폭이 정확히 두 배 또는 최악의 경우 변경률이 1% 미만입니다.

AFF의 단일 스트림 시나리오에서 **AI** 추론을 사용합니다

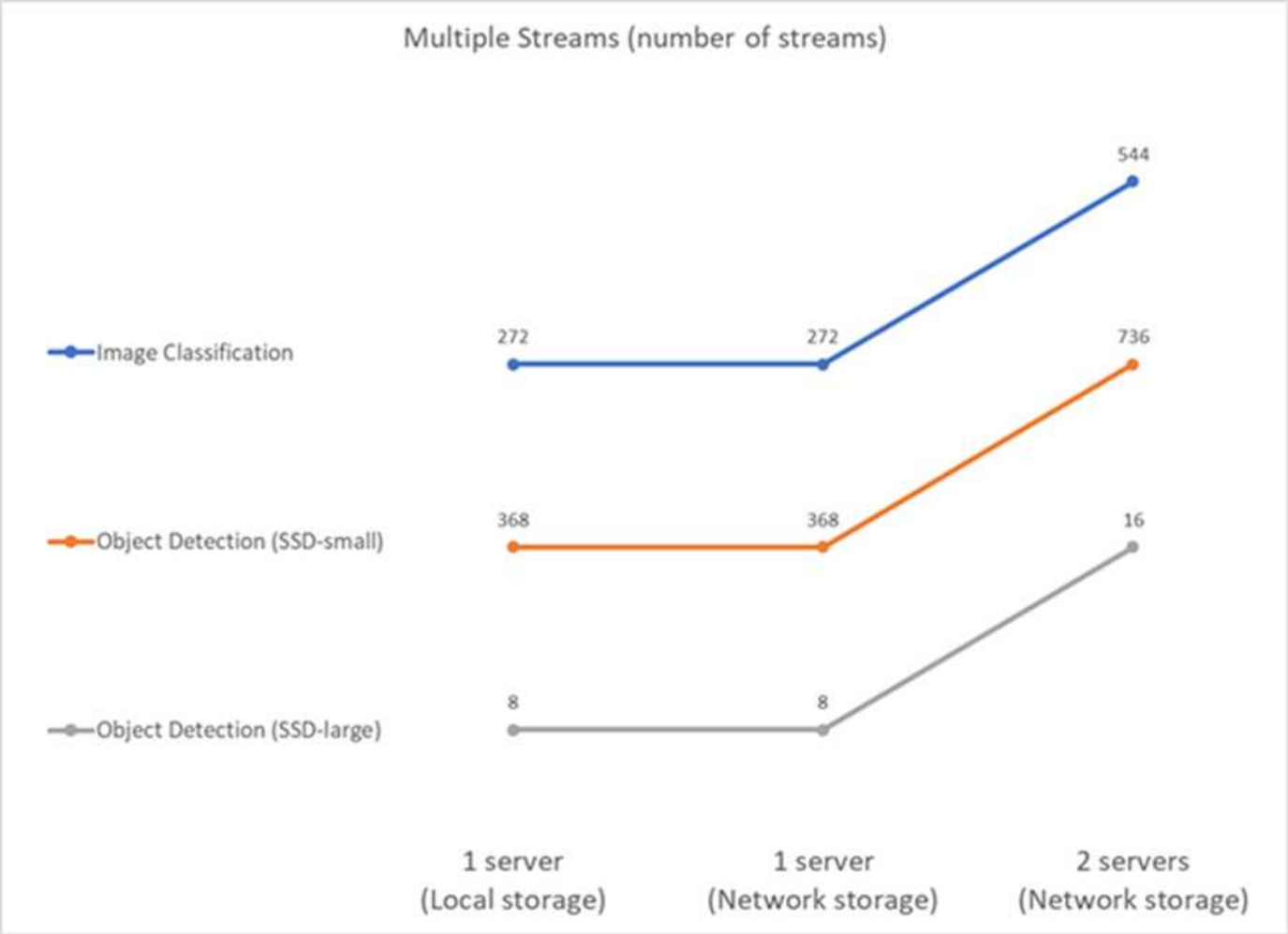
이 벤치마크는 지연 시간을 측정합니다. 여러 계산 서버 사례에서는 평균 지연 시간을 보고합니다. 작업 세트의 결과는 아래 그림에 나와 있습니다. 2서버 사례에서는 두 서버 모두의 평균 지연 시간을 보고합니다.



결과는 네트워크 스토리지가 작업을 처리하기에 충분하다는 것을 다시 한 번 보여 줍니다. 한 서버 케이스에서 로컬 스토리지와 네트워크 스토리지의 차이는 최소 또는 없음입니다. 마찬가지로 두 서버가 동일한 스토리지를 사용하는 경우 두 서버의 지연 시간은 동일하게 유지되거나 매우 적은 양의 변경 사항이 적용됩니다.

AFF의 다중 스트림 시나리오에서 **AI** 추론을 사용합니다

이 경우 결과적으로 QoS 제약 조건을 만족하면서 시스템에서 처리할 수 있는 스트림의 수가 됩니다. 따라서 결과는 항상 정수입니다. 둘 이상의 서버에 대해 모든 서버에 대해 집계된 총 스트림 수를 보고합니다. 모든 워크로드가 이 시나리오를 지원하는 것은 아니지만 이를 실행했습니다. 테스트 결과는 아래 그림에 요약되어 있습니다. 2서버 사례에서는 두 서버 모두에서 스트림 수가 결합된 것으로 보고합니다.



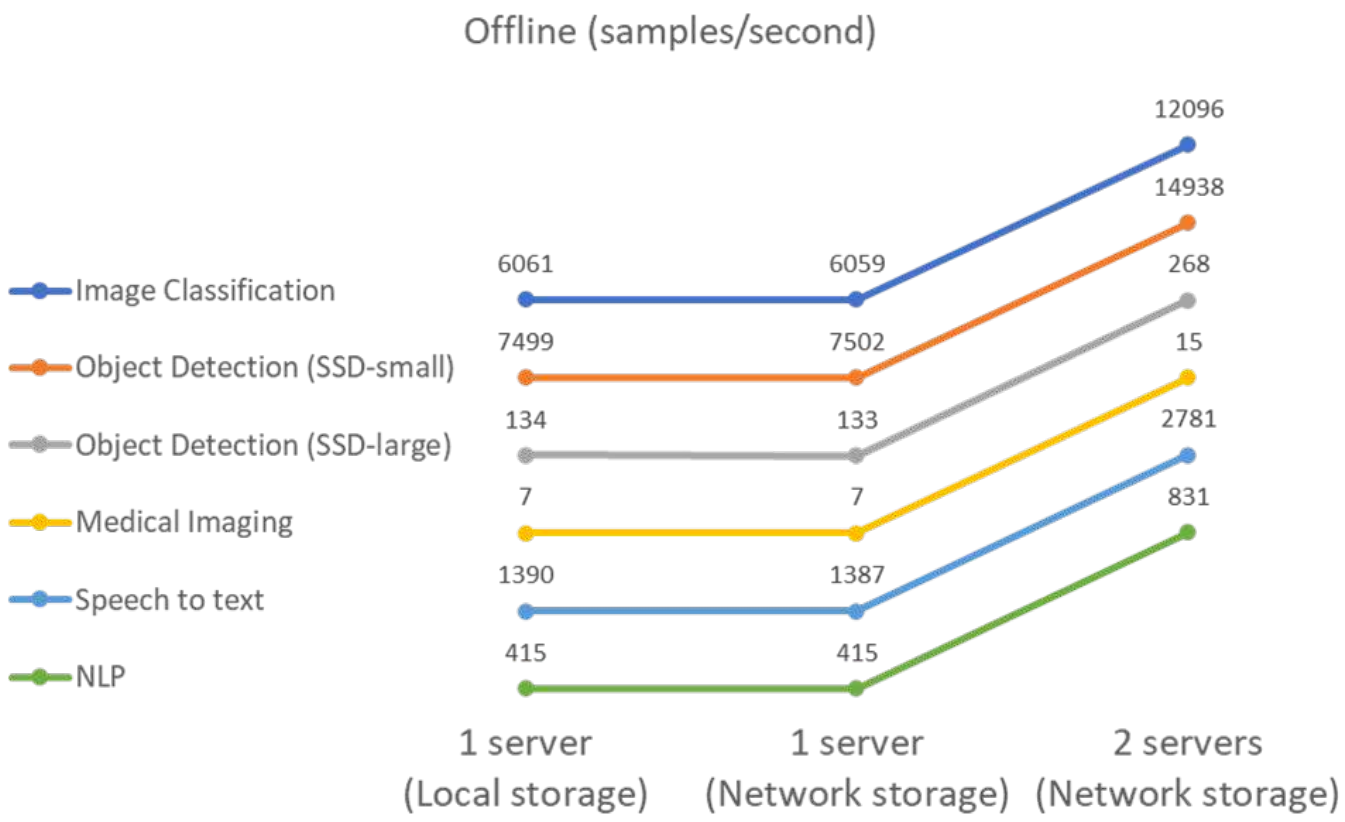
결과는 설정의 완벽한 성능을 보여줍니다. 로컬 및 네트워킹 스토리지는 동일한 결과를 제공하며 두 번째 서버를 추가하면 제안된 설정에서 처리할 수 있는 스트림 수가 두 배가 됩니다.

EF 테스트 결과

제안된 아키텍처의 성능을 평가하기 위해 다수의 테스트를 실행했습니다. 6가지 워크로드(영상 분류, 물체 감지[소형], 물체 감지[대형], 의료 영상, 텍스트 음성 변환, 두 가지 시나리오(오프라인 및 단일 스트림)에서 실행된 자연어 처리[NLP])를 들 수 있습니다. 결과는 다음 섹션에 설명되어 있습니다.

EF의 오프라인 시나리오에서 AI 추론을 사용합니다

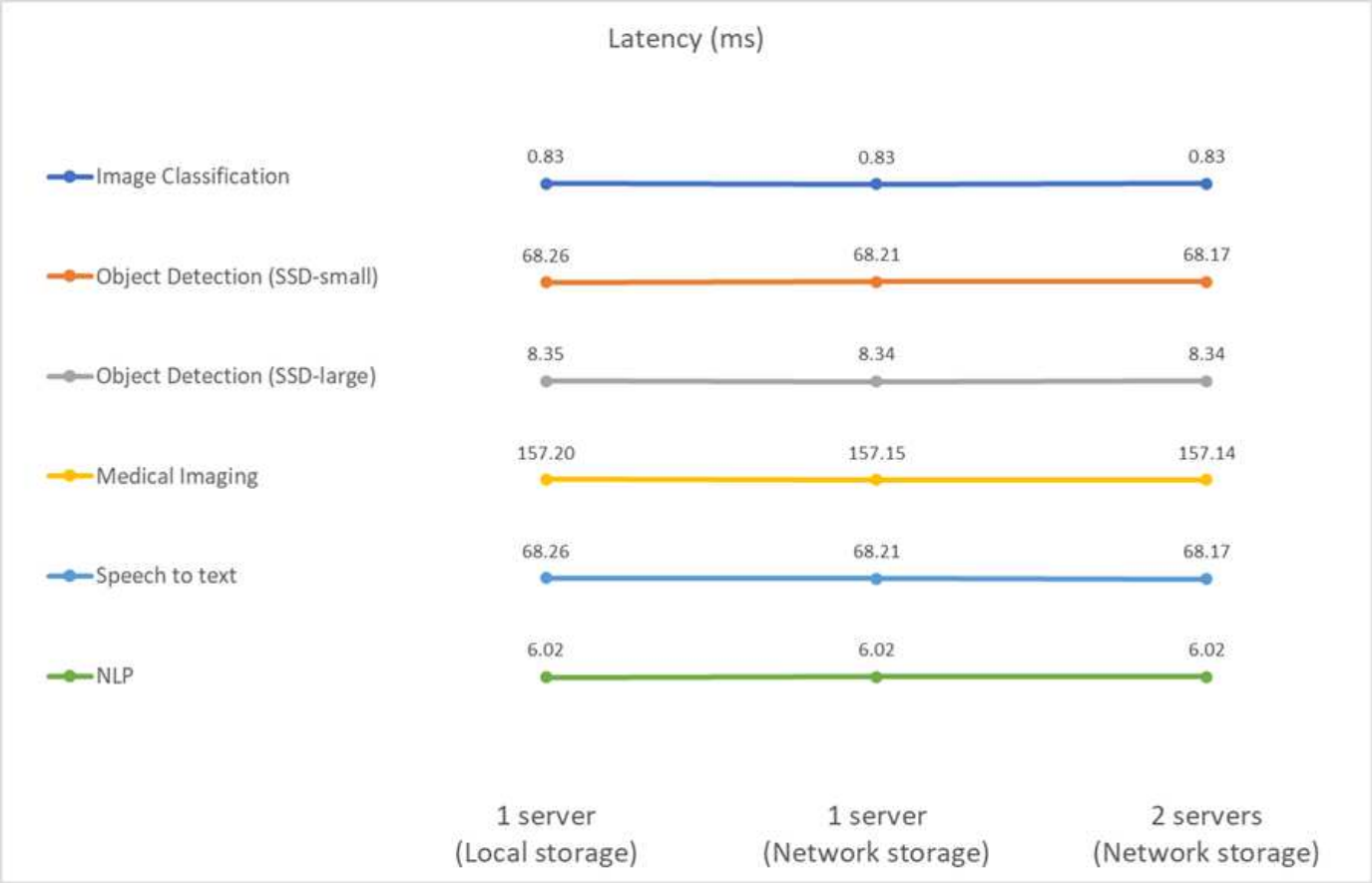
이 시나리오에서는 서버에서 모든 데이터를 사용할 수 있었고 모든 샘플을 처리하는 데 걸린 시간이 측정되었습니다. 테스트 결과로 초당 샘플에 대역폭이 보고됩니다. 단일 노드 실행의 경우 두 서버 모두에서 평균을 보고하며, 두 서버 실행 시 모든 서버에 대해 총 대역폭을 집계합니다. 사용 사례에 대한 결과는 아래 그림에 나와 있습니다.



결과에 따르면 네트워크 스토리지는 성능에 부정적인 영향을 주지 않습니다. 변경 사항은 최소이며 일부 작업의 경우 아무것도 발견되지 않습니다. 두 번째 서버를 추가할 때 총 대역폭이 정확히 두 배 또는 최악의 경우 변경률이 1% 미만입니다.

EF의 단일 스트림 시나리오에서 AI 추론을 사용합니다

이 벤치마크는 지연 시간을 측정합니다. 모든 경우에 대해 실행에 관련된 모든 서버의 평균 지연 시간을 보고합니다. 작업 세트의 결과가 제공됩니다.



결과는 네트워크 스토리지가 작업을 처리하기에 충분하다는 것을 다시 보여줍니다. 한 서버 케이스에서 로컬 스토리지와 네트워크 스토리지의 차이는 Minimal(최소) 또는 None(없음)입니다. 마찬가지로 두 서버가 동일한 스토리지를 사용하는 경우 두 서버의 지연 시간은 동일하게 유지되거나 매우 적은 양의 변경 사항이 적용됩니다.

아키텍처 사이징 옵션

다른 사용 사례에 맞게 검증에 사용된 설정을 조정할 수 있습니다.

컴퓨팅 서버

우리는 SE350에서 지원되는 최저 수준의 CPU인 Intel Xeon D-2123IT CPU를 4개의 물리적 코어와 60W TDP로 사용했습니다. 서버는 CPU 교체를 지원하지 않지만 보다 강력한 CPU로 주문할 수 있습니다. 지원되는 최상위 CPU는 16개의 코어가 있는 Intel Xeon D-2183IT, 2.20GHz에서 실행되는 100W입니다. 이렇게 하면 CPU 계산 기능이 크게 향상됩니다. CPU는 추론 워크로드 자체를 실행하는 데 병목 지점이 되지 않지만, 데이터 처리와 추론과 관련된 다른 작업에 도움이 됩니다. 현재 NVIDIA T4는 에지 사용 사례에 사용할 수 있는 유일한 GPU이므로, 현재는 GPU를 업그레이드하거나 다운그레이드할 수 없습니다.

공유 스토리지

테스트 및 검증을 위해 최대 스토리지 용량이 50.5TB, 순차적 읽기의 경우 처리량 4.4GBps, 소규모 랜덤 읽기의 경우 230K IOPS를 지원하는 NetApp AFF C190 시스템이 이 문서의 목적에 사용되었으며 에지 추론 워크로드에 적합한 것으로 입증되었습니다.

그러나 스토리지 용량 또는 더 빠른 네트워킹 속도가 필요한 경우 NetApp AFF A220 또는 을 사용해야 합니다 "NetApp AFF A250" 기술을 자세히 소개합니다. 또한 최대 1.5PB의 용량을 가진 NetApp EF280 시스템도 이 솔루션 검증을 위해 10Gbps 대역폭 사용이 사용되었습니다. 더 높은 대역폭으로 더 많은 스토리지 용량을 원하는 경우, "NetApp

EF300" 사용할 수 있습니다.

결론

AI 기반 자동화 및 에지 컴퓨팅은 비즈니스 조직이 디지털 혁신을 달성하고 운영 효율성과 안전을 극대화할 수 있도록 지원하는 선도적인 접근 방식입니다. 에지 컴퓨팅은 데이터 센터와 데이터를 전송할 필요가 없기 때문에 훨씬 더 빠르게 처리됩니다. 따라서 데이터를 데이터 센터 또는 클라우드로 전송하는 데 따른 비용이 절감됩니다. 에지에 구축된 AI 추론 모델을 사용하여 거의 실시간으로 의사 결정을 내려야 하는 경우 지연 시간이 단축되고 속도가 빨라질 수 있습니다.

NetApp 스토리지 시스템은 로컬 SSD 스토리지와 동일하거나 더 우수한 성능을 제공하여 데이터 과학자, 데이터 엔지니어, AI/ML 개발자 및 비즈니스 또는 IT 의사 결정자에게 다음과 같은 이점을 제공합니다.

- AI 시스템, 분석 및 기타 중요한 비즈니스 시스템 간에 데이터를 손쉽게 공유 이러한 데이터 공유는 인프라 오버헤드를 줄이고 성능을 향상하며 기업 전체에서 데이터 관리를 간소화합니다.
- 컴퓨팅과 스토리지를 독립적으로 확장하므로 비용을 최소화하고 리소스 사용량을 높일 수 있습니다.
- 즉각적이고 공간 효율적인 사용자 작업 공간, 통합 버전 제어 및 자동화된 구축을 위해 통합 Snapshot 복사본과 클론을 사용하여 개발 및 구축 워크플로우를 간소화했습니다.
- 재해 복구 및 비즈니스 연속성을 위한 엔터프라이즈급 데이터 보호 기능 이 문서에 제공된 NetApp 및 Lenovo 솔루션은 에지에서 엔터프라이즈급 AI 추론 배포에 이상적인 유연한 스케일아웃 아키텍처입니다.

감사의 말

- J.J. Falkanger, 선임 Lenovo, HPC 및 AI 솔루션 매니저
- Dave Arnette, NetApp 기술 마케팅 엔지니어
- Joey Parnell, 기술 팀장 E-Series AI 솔루션, NetApp
- Cody Harryman, NetApp QA 엔지니어

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 문서 및/또는 웹 사이트를 참조하십시오.

- NetApp AFF A-Series 어레이 제품 페이지 를 참조하십시오

  "https://www.netapp.com/data-storage/aff-a-series/"

- NetApp ONTAP 데이터 관리 소프트웨어 - ONTAP 9 정보 라이브러리

  http://mysupport.netapp.com/documentation/productlibrary/index.html?productID=62286

- TR-4727: NetApp EF-Series 소개

  https://www.netapp.com/pdf.html?item=/media/17179-tr4727pdf.pdf

- NetApp E-Series SANtricity 소프트웨어 데이터시트 를 참조하십시오

  https://www.netapp.com/pdf.html?item=/media/19775-ds-3171-66862.pdf

- 컨테이너용 NetApp 영구 스토리지 - NetApp Trident

  "https://netapp.io/persistent-storage-provisioner-for-kubernetes/"

- MLPerf
  - "https://mlcommons.org/en/"
  - "http://www.image-net.org/"
  - "https://mlcommons.org/en/news/mlperf-inference-v11/"
- NetApp BlueXP 복사 및 동기화

  "https://docs.netapp.com/us-en/occm/concept_cloud_sync.html#how-cloud-sync-works"

- TensorFlow 벤치마크

  "https://github.com/tensorflow/benchmarks"

- Lenovo ThinkSystem SE350 Edge 서버

  "https://lenovopress.com/lp1168"

- Lenovo ThinkSystem DM5100F 유니파이드 플래시 스토리지 어레이

  "https://lenovopress.com/lp1365-thinksystem-dm5100f-unified-flash-storage-array"

WP-7328: NVIDIA Jarvis를 사용하는 NetApp 대화형 AI

Rick Huang, Sung-Han Lin, NetApp Davide Onfrio, NVIDIA

NVIDIA DGX 시스템 제품군은 엔터프라이즈 AI용으로 특별 제작된 세계 최초의 통합 인공 지능(AI) 기반 시스템으로 구성되어 있습니다. NetApp AFF 스토리지 시스템은 탁월한 성능과 업계 최고 수준의 하이브리드 클라우드 데이터 관리 기능을 제공합니다. NetApp과 NVIDIA는 협력 관계를 맺고 엔터프라이즈급 성능, 안정성 및 지원을 제공하는 AI 및 머신 러닝(ML) 워크로드를 위한 턴키 솔루션인 NetApp ONTAP AI 참조 아키텍처를 구축했습니다.

이 백서에서는 다양한 산업 분야의 다양한 사용 사례를 지원하는 대화 AI 시스템을 구축하는 고객에게 직접 지침을 제공합니다. NVIDIA Jarvis를 사용하는 시스템 구축에 대한 정보를 제공합니다. 테스트는 NVIDIA DGX Station 및 NetApp AFF A220 스토리지 시스템을 사용하여 수행되었습니다.

이 솔루션의 대상 고객은 다음과 같은 그룹을 포함합니다.

- 가상 소매 비서 등의 대화형 AI 사용 사례를 위한 AI 모델 및 소프트웨어 개발을 위한 솔루션을 설계하는 엔터프라이즈 설계자
- 언어 모델링 개발 목표를 달성하기 위한 효율적인 방법을 찾고 있는 데이터 과학자
- 고객 질문 및 대화 내용 등 텍스트 데이터의 유지 관리 및 처리를 담당하는 데이터 엔지니어
- 대화형 AI 경험을 바꾸고 AI 이니셔티브를 통해 출시 시기를 앞당기고자 하는 경영진 및 IT 의사 결정자 및 비즈니스 리더

솔루션 개요

NetApp ONTAP AI 및 BlueXP Copy and Sync 를 참조하십시오

NVIDIA DGX 시스템 및 NetApp 클라우드 연결형 스토리지 시스템을 기반으로 하는 NetApp ONTAP AI 아키텍처는 NetApp과 NVIDIA가 개발 및 검증했습니다. 이 참조 아키텍처는 IT 조직이 다음과 같은 이점을 얻을 수 있도록 해 줍니다.

- 설계 복잡성 제거
- 컴퓨팅과 스토리지의 독립적인 확장 지원
- 고객이 작은 규모로 시작한 후 원활하게 확장할 수 있도록 지원
- 다양한 성능 및 비용 관련 다양한 스토리지 옵션을 제공합니다.NetApp ONTAP AI는 DGX 시스템과 NetApp AFF A220 스토리지 시스템을 최첨단 네트워킹과 완벽하게 통합합니다. NetApp ONTAP AI 및 DGX 시스템은 설계 복잡성과 추측을 제거함으로써 AI 배포를 단순화합니다. 고객은 작은 규모로 시작한 후 에지에서 코어 및 클라우드까지 포괄하여 데이터를 지능적으로 관리하면서 중단 없이 시스템을 확장할 수 있습니다.

NetApp BlueXP 복사 및 동기화를 사용하면 2개의 NFS 공유, 2개의 CIFS 공유 또는 1개의 파일 공유와 Amazon S3, EFS(Amazon Elastic File System) 또는 Azure Blob 스토리지 간에 다양한 프로토콜 간에 데이터를 쉽게 이동할 수 있습니다. 액티브-액티브 작업은 소스와 타겟을 동시에 계속 사용하여 필요한 경우 데이터 변경 사항을 점진적으로 동기화할 수 있음을 의미합니다. BlueXP Copy 및 Sync는 온프레미스 또는 클라우드 기반의 모든 소스 및 타겟 시스템 간에 데이터를 이동하고 증분 동기화할 수 있게 지원하므로 데이터를 사용할 수 있는 다양한 새로운 방법을 제시합니다. 사내 시스템, 클라우드 온보딩, 클라우드 마이그레이션, 협업 및 데이터 분석 간에 데이터를 마이그레이션하는 작업을 모두 쉽게 수행할 수 있게 되었습니다. 아래 그림은 사용 가능한 소스 및 대상을 보여줍니다.

대화형 AI 시스템에서 개발자는 BlueXP Copy and Sync를 활용하여 클라우드에서 데이터 센터로 대화 기록을 아카이브함으로써 자연어 처리(NLP) 모델의 오프라인 훈련을 지원할 수 있습니다. 더 많은 연고를 인식하는 교육 모델을 통해 대화형 AI 시스템은 최종 사용자의 더 복잡한 질문을 더 효과적으로 관리할 수 있습니다.

NVIDIA Jarvis 다중 모드 프레임워크

"NVIDIA Jarvis" 대화형 AI 서비스를 구축하기 위한 엔드 투 엔드 프레임워크입니다. 다음과 같은 GPU 최적화 서비스가 포함됩니다.

- 자동 음성 인식(ASR)
- 자연어 이해(NLU)
- 도메인별 이행 서비스와 통합
- TTS(Text-to-Speech)
- 컴퓨터 비전(CV) Jarvis 기반 서비스는 최첨단 딥 러닝 모델을 사용하여 실시간 대화 AI의 복잡하고 까다로운 작업을 해결합니다. 최종 사용자와의 자연스러운 실시간 상호 작용을 지원하기 위해 모델은 300밀리초 이내에 계산을 완료해야 합니다. 자연스러운 상호작용은 어려운 과제이며, 다중 모드 감각 통합이 필요합니다. 또한 모델 파이프라인은 복잡하며 위 서비스 간의 조정이 필요합니다.

Jarvis는 엔드 투 엔드 딥 러닝 파이프라인을 사용하는 다중 모달 대화 AI 서비스를 구축하기 위한 애플리케이션 프레임워크입니다. Jarvis 프레임워크는 음성, 비전 및 NLU 작업을 위한 사전 교육 대화 AI 모델, 도구 및 최적화된 엔드 투 엔드 서비스를 포함합니다. Jarvis를 사용하면 AI 서비스 외에도 비전, 오디오 및 기타 센서 입력을 동시에 결합하여 가상 보조자, 다중 사용자 양극화 및 콜센터 보조자와 같은 애플리케이션에서 다중 사용자, 다중 컨텍스트 대화 등의 기능을 제공할 수 있습니다.

NVIDIA 니모

"NVIDIA 니모" 사용하기 쉬운 API(애플리케이션 프로그래밍 인터페이스)를 사용하여 GPU 가속 첨단 대화 AI 모델을 구축, 교육 및 미세 조정하는 오픈 소스 Python 툴킷입니다. Nemo는 NVIDIA GPU에서 Tensor Core를 사용하여 혼합 정밀 컴퓨팅을 실행하며 여러 GPU로 손쉽게 확장하여 가능한 최고의 교육 성능을 제공할 수 있습니다. Nemo는 의료, 재무, 소매 및 통신 등 다양한 산업 분야에서 화상 통화 기록, 지능형 비디오 비서, 자동화된 콜 센터 지원 등의 실시간 ASR, NLP 및 TTS 애플리케이션을 위한 모델을 구축하는 데 사용됩니다.

Nemo를 사용하여 아카이빙된 대화 기록의 사용자 질문에서 복잡한 인텐트를 인식하는 모델을 교육했습니다. 이 교육은 소매 가상 보조자의 능력을 Jarvis가 제공하는 지원 범위를 넘어 확장합니다.

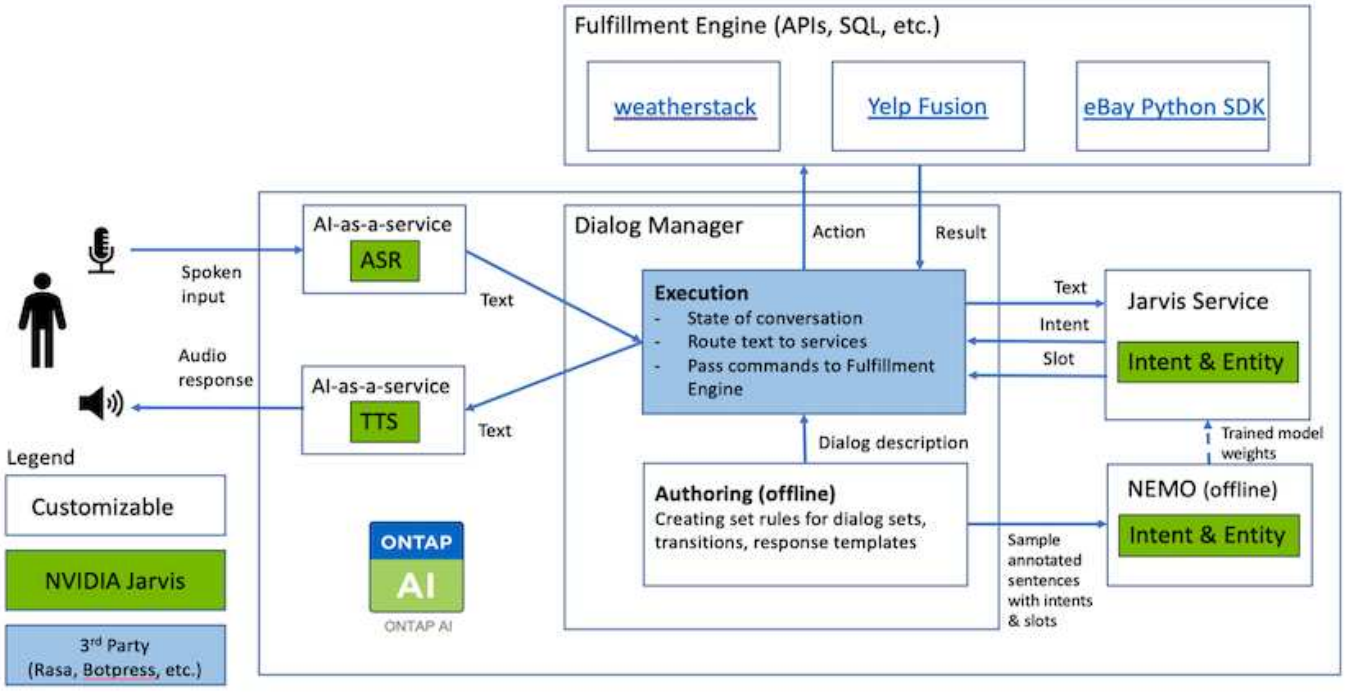
소매 사용 사례 요약

NVIDIA Jarvis를 사용하여 음성 또는 텍스트 입력을 수용하는 가상 소매 도우미를 구축하고 날씨, 관심 지점 및 재고 가격 관련 질문에 답변했습니다. 대화형 AI 시스템은 예를 들어, 사용자가 날씨 또는 관심 지점을 지정하지 않은 경우 후속 질문을 하여 대화 흐름을 기억할 수 있습니다. 또한 이 시스템은 "태국식 음식" 또는 "노트북 메모리"와 같은 복잡한 엔터티도 인식합니다. 그것은 “다음주 로스앤젤레스에서 비가 올까요?”와 같은 자연어 질문을 이해합니다. 소매 가상 비서의 데모는 에서 확인할 수 있습니다 "소매 사용 사례에 대한 상태 및 흐름을 사용자 지정합니다".

솔루션 기술

다음 그림에서는 제안한 대화형 AI 시스템 아키텍처를 보여 줍니다. 음성 신호 또는 텍스트 입력으로 시스템과 상호 작용할 수 있습니다. 음성 입력이 감지되면 Jarvis AIaaS(AI-as-service)가 ASR을 수행하여 Dialog Manager에 대한 텍스트를 생성합니다. 대화 관리자는 대화 상태를 기억하고, 텍스트를 해당 서비스로 라우팅하고, 명령을 이행 엔진에 전달합니다. Jarvis NLP 서비스는 텍스트를 가져와 인텐트와 엔터티를 인식하고 이러한 인텐트와 엔터티 슬롯을 다시 대화 상자 매니저로 출력한 다음 작업을 이행 엔진에 보냅니다. 이행 엔진은 사용자 쿼리에 응답하는 타사 API 또는 SQL 데이터베이스로 구성됩니다. 이행 엔진에서 결과를 수신한 후 대화 상자 관리자는 텍스트를 Jarvis TTS AIaaS로 라우팅하여 최종 사용자에 대한 오디오 응답을 생성합니다. 대화 기록을 보관하고, intents와 nemo 교육 슬롯을 사용해 문장에 주석을 달 수 있습니다. 그러면 NLP 서비스가 시스템과 상호 작용하는 사용자가 많아질수록 성능이

향상됩니다.



하드웨어 요구 사항

이 솔루션은 하나의 DGX Station과 하나의 AFF A220 스토리지 시스템을 사용하여 검증되었습니다. Jarvis는 딥 신경 네트워크 계산을 수행하려면 T4 또는 V100 GPU가 필요합니다.

다음 표에는 솔루션을 테스트하는 데 필요한 하드웨어 구성요소가 나와 있습니다.

하드웨어	수량
T4 또는 V100 GPU	1
NVIDIA DGX 스테이션	1

소프트웨어 요구 사항

다음 표에는 테스트를 통해 솔루션을 구현하는 데 필요한 소프트웨어 구성요소가 나와 있습니다.

소프트웨어	버전 또는 기타 정보
NetApp ONTAP 데이터 관리 소프트웨어	9.6
Cisco NX-OS 스위치 펌웨어	7.0(3) I6(1)
NVIDIA DGX OS	4.0.4 - Ubuntu 18.04 LTS
NVIDIA Jarvis 프레임워크	EA v0.2
NVIDIA 니모	nvcr.io/nvidia/nemo:v0.10
Docker 컨테이너 플랫폼	18.06.1-CE[e68fc7a]

개요

이 섹션에서는 Virtual Retail Assistant 구현에 대해 자세히 설명합니다.

Jarvis 배포

에 등록할 수 있습니다 "Jarvis Early Access 프로그램" NGC(NVIDIA GPU Cloud)에서 Jarvis 컨테이너에 액세스 NVIDIA로부터 자격 증명을 받은 후 다음 단계를 사용하여 Jarvis를 배포할 수 있습니다.

1. NGC에 로그인합니다.
2. NGC를 통해 조직을 "ea-2-Jarvis"로 설정합니다.
3. Jarvis EA v0.2 자산 찾기: Jarvis 컨테이너는 '개인 레지스트리'>'조직 컨테이너'에 있습니다.
4. 자비스(Jarvis) 를 선택하고 모델 스크립트(Model Scripts) 로 이동한 다음 자비스 빠른 시작(Jarvis Quick Start) 을 클릭합니다
5. 모든 자산이 제대로 작동하는지 확인합니다.
6. PDF는 모델 스크립트 > Jarvis Documentation > File Browser에서 찾을 수 있습니다.

소매 사용 사례에 대한 상태 및 흐름을 사용자 지정합니다

특정 사용 사례에 맞게 Dialog Manager의 상태 및 흐름을 사용자 지정할 수 있습니다. 당사의 소매 예시에서 다음과 같은 네 가지 YAML 파일이 다양한 인도에 따라 대화를 유도합니다.

다음 파일 이름 목록과 각 파일에 대한 설명을 따르십시오.

- main_flow.yml: 주요 대화의 흐름과 상태를 정의하고 필요에 따라 다른 3개의 YAML 파일로 흐름을 안내합니다.
- RETail_flow.yml: 소매 또는 관심 지점 질문과 관련된 주가 포함되어 있습니다. 시스템은 가장 가까운 매장의 정보 또는 지정된 품목의 가격을 제공합니다.
- 날씨 흐름.yml: 날씨 문제와 관련된 주가 포함되어 있습니다. 위치를 확인할 수 없는 경우 시스템은 추가 질문을 통해 명확히 합니다.
- error_flow.yml: 위의 3가지 YAML 파일에 포함되지 않는 경우를 처리합니다. 오류 메시지를 표시한 후 시스템은 사용자 질문 수락으로 다시 라우팅합니다. 다음 섹션에는 이러한 YAML 파일에 대한 자세한 정의가 나와 있습니다.

Main_flow.yml

```
name: JarvisRetail
intent_transitions:
  jarvis_error: error
  price_check: retail_price_check
  inventory_check: retail_inventory_check
  store_location: retail_store_location
  weather.weather: weather
  weather.temperature: temperature
  weather.sunny: sunny
  weather.cloudy: cloudy
```

```
  weather.snow: snow
  weather.rainfall: rain
  weather.snow_yes_no: snowfall
  weather.rainfall_yes_no: rainfall
  weather.temperature_yes_no: tempyesno
  weather.humidity: humidity
  weather.humidity_yes_no: humidity
  navigation.startnavigationpoi: retail # Transitions should be context
and slot based. Redirecting for now.
  navigation.geteta: retail
  navigation.showdirection: retail
  navigation.showmappoi: idk_what_you_talkin_about
  nomatch.none: idk_what_you_talkin_about
states:
  init:
    type: message_text
    properties:
      text: "Hi, welcome to NARA retail and weather service. How can I
help you?"
  input_intent:
    type: input_context
    properties:
      nlp_type: jarvis
      entities:
        intent: dontcare
# This state is executed if the intent was not understood
  dont_get_the intent:
    type: message_text_random
    properties:
      responses:
        - "Sorry I didn't get that! Please come again."
        - "I beg your pardon! Say that again?"
        - "Are we talking about weather? What would you like to know?"
        - "Sorry I know only about the weather"
        - "You can ask me about the weather, the rainfall, the
temperature, I don't know much more"
      delay: 0
    transitions:
      next_state: input_intent
  idk_what_you_talkin_about:
    type: message_text_random
    properties:
      responses:
        - "Sorry I didn't get that! Please come again."
        - "I beg your pardon! Say that again?"
        - "Are we talking about retail or weather? What would you like to
```

```
know?"
        - "Sorry I know only about retail and the weather"
        - "You can ask me about retail information or the weather, the
rainfall, the temperature. I don't know much more."
      delay: 0
    transitions:
      next_state: input_intent
  error:
    type: change_context
    properties:
        update_keys:
           intent: 'error'
    transitions:
        flow: error_flow
  retail_inventory_check:
    type: change_context
    properties:
        update_keys:
           intent: 'retail_inventory_check'
    transitions:
        flow: retail_flow
  retail_price_check:
    type: change_context
    properties:
        update_keys:
           intent: 'check_item_price'
    transitions:
        flow: retail_flow
  retail_store_location:
    type: change_context
    properties:
        update_keys:
           intent: 'find_the_store'
    transitions:
        flow: retail_flow
  weather:
    type: change_context
    properties:
        update_keys:
           intent: 'weather'
    transitions:
        flow: weather_flow
  temperature:
    type: change_context
    properties:
        update_keys:
```

```
            intent: 'temperature'
      transitions:
          flow: weather_flow
    rainfall:
      type: change_context
      properties:
          update_keys:
             intent: 'rainfall'
      transitions:
          flow: weather_flow
    sunny:
      type: change_context
      properties:
          update_keys:
             intent: 'sunny'
      transitions:
          flow: weather_flow
    cloudy:
      type: change_context
      properties:
          update_keys:
             intent: 'cloudy'
      transitions:
          flow: weather_flow
    snow:
      type: change_context
      properties:
          update_keys:
             intent: 'snow'
      transitions:
          flow: weather_flow
    rain:
      type: change_context
      properties:
          update_keys:
             intent: 'rain'
      transitions:
          flow: weather_flow
    snowfall:
        type: change_context
        properties:
            update_keys:
               intent: 'snowfall'
        transitions:
            flow: weather_flow
    tempyesno:
```

```
      type: change_context
      properties:
          update_keys:
             intent: 'tempyesno'
      transitions:
          flow: weather_flow
  humidity:
      type: change_context
      properties:
          update_keys:
             intent: 'humidity'
      transitions:
          flow: weather_flow
  end_state:
    type: reset
    transitions:
      next_state: init
```

retail_flow.yml

```
name: retail_flow
states:
  store_location:
    type: conditional_exists
    properties:
      key: '{{location}}'
    transitions:
      exists: retail_state
      notexists: ask_retail_location
  retail_state:
    type: Retail
    properties:
    transitions:
      next_state: output_retail
  output_retail:
      type: message_text
      properties:
        text: '{{retail_status}}'
      transitions:
        next_state: input_intent
  ask_retail_location:
    type: message_text
    properties:
      text: "For which location? I can find the closest store near you."
    transitions:
```

```
      next_state: input_retail_location
  input_retail_location:
    type: input_user
    properties:
      nlp_type: jarvis
      entities:
        slot: location
      require_match: true
    transitions:
      match: retail_state
      notmatch: check_retail_jarvis_error
  output_retail_acknowledge:
    type: message_text_random
    properties:
      responses:
        - 'ok in {{location}}'
        - 'the store in {{location}}'
        - 'I always wanted to shop in {{location}}'
      delay: 0
    transitions:
      next_state: retail_state
  output_retail_notlocation:
    type: message_text
    properties:
      text: "I did not understand the location. Can you please repeat?"
    transitions:
      next_state: input_intent
  check_rerail_jarvis_error:
    type: conditional_exists
    properties:
      key: '{{jarvis_error}}'
    transitions:
      exists: show_retail_jarvis_api_error
      notexists: output_retail_notlocation
  show_retail_jarvis_api_error:
    type: message_text
    properties:
      text: "I am having troubled understanding right now. Come again on
that?"
    transitions:
      next_state: input_intent
```

WATEER_flow.yml

```
name: weather_flow
```

```
states:
  check_weather_location:
    type: conditional_exists
    properties:
      key: '{{location}}'
    transitions:
      exists: weather_state
      notexists: ask_weather_location
  weather_state:
    type: Weather
    properties:
    transitions:
      next_state: output_weather
  output_weather:
      type: message_text
      properties:
        text: '{{weather_status}}'
      transitions:
        next_state: input_intent
  ask_weather_location:
    type: message_text
    properties:
      text: "For which location?"
    transitions:
      next_state: input_weather_location
  input_weather_location:
    type: input_user
    properties:
      nlp_type: jarvis
      entities:
        slot: location
      require_match: true
    transitions:
      match: weather_state
      notmatch: check_jarvis_error
  output_weather_acknowledge:
    type: message_text_random
    properties:
      responses:
        - 'ok in {{location}}'
        - 'the weather in {{location}}'
        - 'I always wanted to go in {{location}}'
      delay: 0
    transitions:
      next_state: weather_state
  output_weather_notlocation:
```

```
    type: message_text
    properties:
      text: "I did not understand the location, can you please repeat?"
    transitions:
      next_state: input_intent
  check_jarvis_error:
    type: conditional_exists
    properties:
      key: '{{jarvis_error}}'
    transitions:
      exists: show_jarvis_api_error
      notexists: output_weather_notlocation
  show_jarvis_api_error:
    type: message_text
    properties:
      text: "I am having troubled understanding right now. Come again on
that, else check jarvis services?"
    transitions:
      next_state: input_intent
```

ERROR_flow.yml

```
name: error_flow
states:
  error_state:
    type: message_text_random
    properties:
      responses:
        - "Sorry I didn't get that!"
        - "Are we talking about retail or weather? What would you like to
know?"
        - "Sorry I know only about retail information or the weather"
        - "You can ask me about retail information or the weather, the
rainfall, the temperature. I don't know much more"
        - "Let's talk about retail or the weather!"
      delay: 0
    transitions:
      next_state: input_intent
```

타사 **API**에 이행 엔진으로 연결합니다

다음 타사 API를 이행 엔진으로 연결하여 질문에 답변했습니다.

- "WeatherStack API를 참조하십시오": 지정된 위치에 날씨, 온도, 강우량 및 눈을 반환합니다.

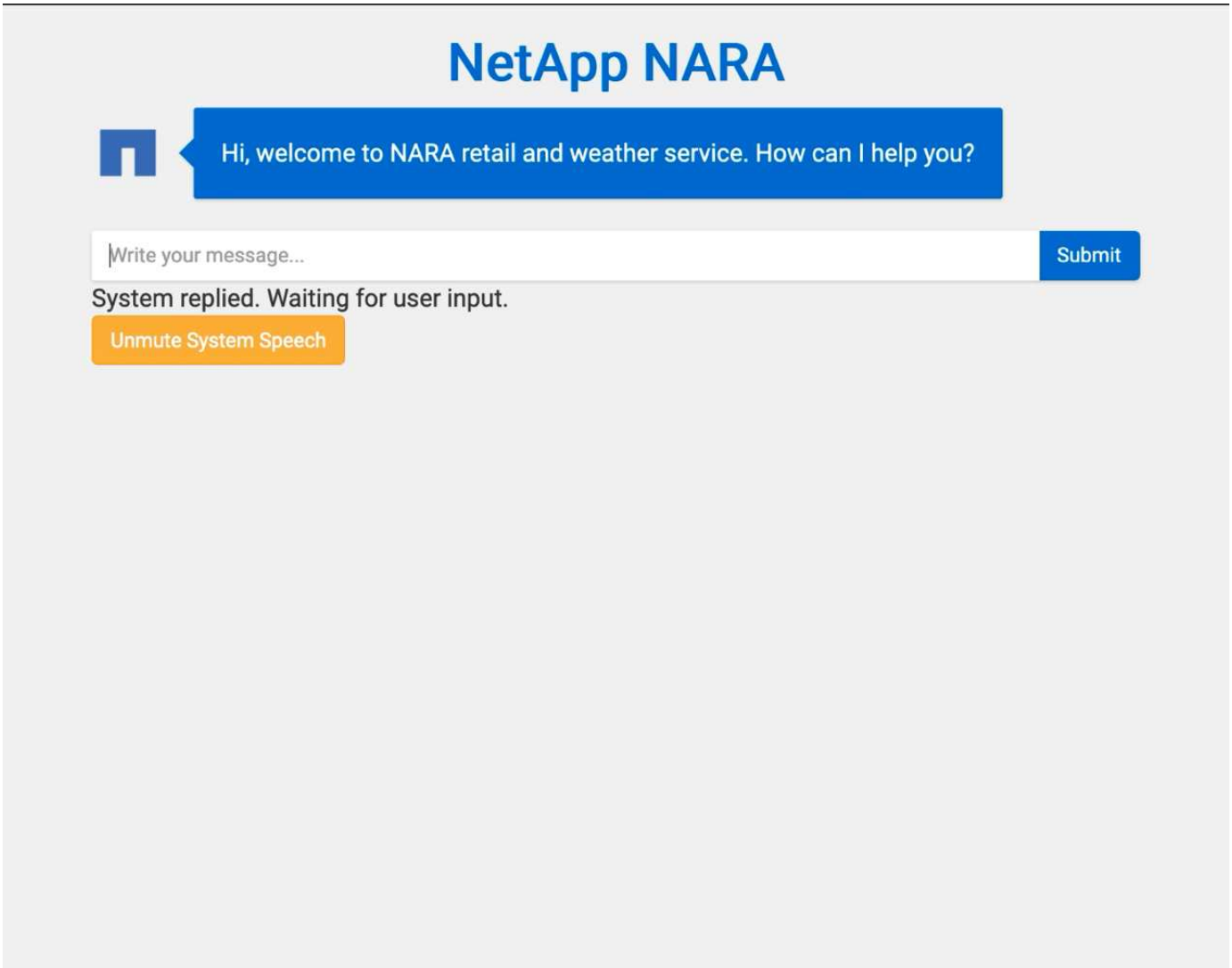
- "Yelp Fusion API를 참조하십시오": 지정된 위치에서 가장 가까운 매장 정보를 반환합니다.
- "eBay Python SDK": 지정된 항목의 가격을 반환합니다.

NetApp Retail Assistant 데모

NetApp Retail Assistant(Nara)의 데모 비디오를 녹화했습니다.

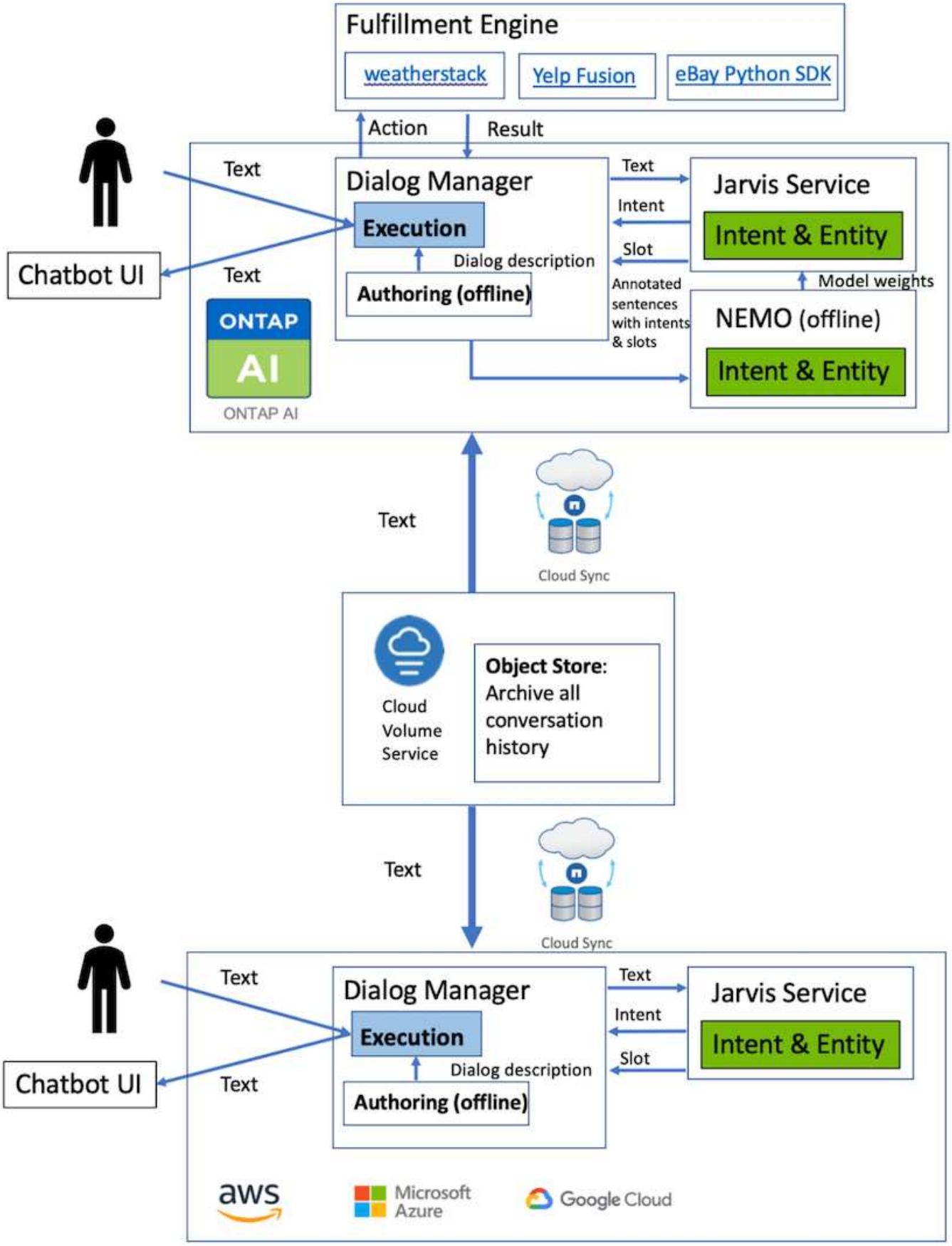
나라 동영상 데모

나라 동영상 데모



NetApp BlueXP 복사 및 동기화를 사용하여 대화 기록을 아카이브합니다

하루에 한 번 대화 기록을 CSV 파일에 덤프하면 BlueXP 복사본 및 동기화 를 활용하여 로그 파일을 로컬 스토리지에 다운로드할 수 있습니다. 다음 그림에서는 Jarvis가 온프레미스 및 퍼블릭 클라우드에 구축하면서 BlueXP Copy 및 Sync를 사용하여 Nemo 교육을 위한 대화 기록을 전송하는 아키텍처를 보여 줍니다. Nemo 교육에 대한 자세한 내용은 섹션을 참조하십시오 "Nemo Training을 사용하여 Intent 모델을 확장합니다".


Fulfillment Engine
weatherstack
Yelp Fusion
eBay Python SDK
Action
Result
Text
Chatbot UI
Text
ONTAP
AI
ONTAP AI
Dialog Manager
Execution
Dialog description
Authoring (offline)
Text
Intent
Slot
Annotated sentences with intents & slots
Jarvis Service
Intent & Entity
Model weights
NEMO (offline)
Intent & Entity
Text
Cloud Sync
Cloud Volume Service
Object Store: Archive all conversation history
Text
Cloud Sync
Text
Chatbot UI
Text
Dialog Manager
Execution
Dialog description
Authoring (offline)
Text
Intent
Slot
Jarvis Service
Intent & Entity
aws
Microsoft Azure
Google Cloud

Nemo Training을 사용하여 **Intent** 모델을 확장합니다

NVIDIA Nemo는 NVIDIA에서 대화형 AI 애플리케이션을 만들기 위해 만든 툴킷입니다. 이 툴킷에는 ASR, NLP 및 TTS에 대한 사전 교육 모듈 모음이 포함되어 있어 연구자와 데이터 과학자가 복잡한 신경망 아키텍처를 쉽게 구성하고 자체 애플리케이션을 설계하는 데 더 많은 노력을 집중할 수 있습니다.

앞의 예에서와 같이 나라에서는 제한된 질문 유형만 처리할 수 있습니다. 사전 교육 받은 NLP 모델은 이러한 유형의 질문에만 교육을 제공하기 때문입니다. Nara가 보다 광범위한 질문을 처리하도록 하려면 자체 데이터세트를 사용하여 재교육해야 합니다. 따라서 여기서는 Nemo를 사용하여 NLP 모델을 확장하여 요구 사항을 충족하는 방법을 보여 줍니다. 우선 Nara에서 수집한 로그를 Nemo 형식으로 변환한 다음 NLP 모델을 향상시키기 위해 데이터 세트를 사용하여 훈련합니다.

모델

우리의 목표는 Nara가 사용자 선호도에 따라 항목을 정렬할 수 있도록 하는 것입니다. 예를 들어, 나라에게 최고 등급의 스시 레스토랑을 추천하거나 나라(Nara)가 가장 낮은 가격으로 청바지를 찾아보길 원할 수도 있습니다. 이를 위해 Nemo에 제공된 intent detection 및 slot filling 모델을 실습 모델로 사용한다. 이 모델을 통해 Nara는 선호하는 검색의 의도를 이해할 수 있습니다.

데이터 준비

모델을 학습하기 위해 이 유형의 질문에 대한 데이터 세트를 수집하고 이를 Nemo 형식으로 변환합니다. 여기서는 모델을 훈련하는 데 사용하는 파일을 나열했습니다.

dict.intents.csv

이 파일에는 Nemo가 이해할 수 있는 모든 인텐트가 나열되어 있습니다. 여기서는 일차 연고 2개와 일차 연고 중 하나에 적합하지 않은 질문을 분류하는 데만 사용되는 의도로 1개를 사용합니다.

```
price_check
find_the_store
unknown
```

dict.slots.csv

이 파일에는 교육 질문에 표시할 수 있는 모든 슬롯이 나열되어 있습니다.

```
B-store.type
B-store.name
B-store.status
B-store.hour.start
B-store.hour.end
B-store.hour.day
B-item.type
B-item.name
B-item.color
B-item.size
```

```
B-item.quantity
B-location
B-cost.high
B-cost.average
B-cost.low
B-time.period_of_time
B-rating.high
B-rating.average
B-rating.low
B-interrogative.location
B-interrogative.manner
B-interrogative.time
B-interrogative.personal
B-interrogative
B-verb
B-article
I-store.type
I-store.name
I-store.status
I-store.hour.start
I-store.hour.end
I-store.hour.day
I-item.type
I-item.name
I-item.color
I-item.size
I-item.quantity
I-location
I-cost.high
I-cost.average
I-cost.low
I-time.period_of_time
I-rating.high
I-rating.average
I-rating.low
I-interrogative.location
I-interrogative.manner
I-interrogative.time
I-interrogative.personal
I-interrogative
I-verb
I-article
O
```

훈련.**TSV**

이 데이터 세트는 주요 교육 데이터 세트입니다. 각 줄은 dict.intent.csv 파일의 의도 범주 목록에 따라 질문으로 시작합니다. 레이블은 0부터 열거됩니다.

기차_슬롯.**TSV**

```
20 46 24 25 6 32 6
52 52 24 6
23 52 14 40 52 25 6 32 6
…
```

모델 훈련

```
docker pull nvcr.io/nvidia/nemo:v0.10
```

그런 다음 다음 다음 다음 명령을 사용하여 컨테이너를 시작합니다. 이 명령은 간단한 교육 연습이므로 컨테이너가 단일 GPU(GPU ID=1)를 사용하도록 제한합니다. 또한 로컬 작업 공간/작업 공간/Nemo/를 컨테이너/Nemo 내부의 폴더에 매핑합니다.

```
NV_GPU='1' docker run --runtime=nvidia -it --shm-size=16g \
                        --network=host --ulimit memlock=-1 --ulimit
stack=67108864 \
                        -v /workspace/nemo:/nemo\
                        --rm nvcr.io/nvidia/nemo:v0.10
```

컨테이너 내부에서 사전 훈련된 원래 BERT 모델에서 시작하려면 다음 명령을 사용하여 교육 절차를 시작할 수 있습니다. data_dir은 교육 데이터의 경로를 설정하기 위한 인수입니다. Work_dir 체크포인트 파일을 저장할 위치를 구성할 수 있습니다.

```
cd examples/nlp/intent_detection_slot_tagging/
python joint_intent_slot_with_bert.py \
    --data_dir /nemo/training_data\
    --work_dir /nemo/log
```

새로운 교육 데이터 세트가 있고 이전 모델을 개선하려는 경우 다음 명령을 사용하여 중지한 시점부터 계속 진행할 수 있습니다. checkpoint_dir 은 경로를 이전 체크포인트 폴더로 가져갑니다.

```
cd examples/nlp/intent_detection_slot_tagging/
python joint_intent_slot_infer.py \
    --data_dir /nemo/training_data \
    --checkpoint_dir /nemo/log/2020-05-04_18-34-20/checkpoints/ \
    --eval_file_prefix test
```

모델을 추론합니다

특정 수의 Epoch 후에 교육 이수 모델의 성능을 검증해야 합니다. 다음 명령을 사용하여 쿼리를 하나씩 테스트할 수 있습니다. 예를 들어, 이 명령에서 모델이 '최고의 파스타를 어디서 얻을 수 있는지'라는 질의의 의도를 제대로 파악할 수 있는지 확인해야 합니다.

```
cd examples/nlp/intent_detection_slot_tagging/
python joint_intent_slot_infer_b1.py \
--checkpoint_dir /nemo/log/2020-05-29_23-50-58/checkpoints/ \
--query "where can i get the best pasta" \
--data_dir /nemo/training_data/ \
--num_epochs=50
```

그런 다음, 추론의 출력입니다. 출력물에서는 숙련된 모델이 find_the_store의 의도를 적절히 예측하고 관심 있는 키워드를 반환할 수 있습니다. 이러한 키워드를 사용하여 Nara는 사용자가 원하는 것을 검색하고 보다 정확한 검색을 수행할 수 있습니다.

```
[NeMo I 2020-05-30 00:06:54 actions:728] Evaluating batch 0 out of 1
[NeMo I 2020-05-30 00:06:55 inference_utils:34] Query: where can i get the
best pasta
[NeMo I 2020-05-30 00:06:55 inference_utils:36] Predicted intent:       1
find_the_store
[NeMo I 2020-05-30 00:06:55 inference_utils:50] where   B-
interrogative.location
[NeMo I 2020-05-30 00:06:55 inference_utils:50] can     O
[NeMo I 2020-05-30 00:06:55 inference_utils:50] i       O
[NeMo I 2020-05-30 00:06:55 inference_utils:50] get     B-verb
[NeMo I 2020-05-30 00:06:55 inference_utils:50] the     B-article
[NeMo I 2020-05-30 00:06:55 inference_utils:50] best    B-rating.high
[NeMo I 2020-05-30 00:06:55 inference_utils:50] pasta   B-item.type
```

결론

진정한 대화형 AI 시스템은 인간과 같은 대화에 참여하고, 컨텍스트를 이해하고, 지능적인 응답을 제공합니다. 이러한 AI 모델은 대개 규모가 크고 매우 복잡합니다. NVIDIA GPU 및 NetApp 스토리지를 사용하면 최첨단 대용량 언어 모델을 훈련 및 최적화하여 추론을 신속하게 실행할 수 있습니다. 빠르게, 크고 복잡한 AI 모델 간에 이루어지는 거래를 끝내기 위한 주요 발걸음을

내딛습니다. 의료, 소매 및 금융 서비스 등의 산업을 위해 GPU에 최적화된 언어 이해 모델을 AI 애플리케이션에 통합하여 스마트 스피커 및 고객 서비스 분야에서 고급 디지털 음성 지원 기능을 제공할 수 있습니다. 이러한 고품질 대화형 AI 시스템을 통해 수직 시장에 있는 기업들은 고객과 교류할 때 이전에는 불가능했던 맞춤형 서비스를 제공할 수 있습니다.

Jarvis를 사용하면 가상 보조자, 디지털 아바타, 다중 모드 센서 Fusion(CV와 ASR/NLP/TTS 결합) 또는 변환 등의 ASR/NLP/TTS/CV 독립 실행형 사용 사례를 구축할 수 있습니다. 날씨, 관심 지점 및 재고 가격 관련 질문에 답할 수 있는 가상 소매 도우미를 구축했습니다. 또한 BlueXP Copy and Sync 를 사용하여 대화 기록을 보관하고 새로운 데이터에 대한 Nemo 모델을 훈련하여 대화형 AI 시스템의 자연어 이해 기능을 개선하는 방법을 입증했습니다.

감사의 말

저자는 NVIDIA의 존경받는 동료 Davide Onofrio, Alex Qi, Sicong Ji, Marty Jain 및 Robert Sohigian이 이 백서에 기여한 바를 진심으로 인정합니다. 또한 NetApp의 주요 팀원 중 Santosh Rao, David Arnette, Michael Oglesby, Brent Davis, Andy Sayare의 기여에 대해 인정하고자 합니다. Erik Mulder와 Mike McNamara가 함께 합니다.

본 문서를 작성하는 데 큰 도움이 되는 통찰과 전문 지식을 제공해주신 모든 분에게 진심으로 감사합니다.

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 리소스를 참조하십시오.

- NVIDIA DGX Station, V100 GPU, GPU Cloud
  - NVIDIA DGX 스테이션https://www.nvidia.com/en-us/data-center/dgx-station/["https://www.nvidia.com/en-us/data-center/dgx-station/"^]
  - NVIDIA V100 Tensor 코어 GPUhttps://www.nvidia.com/en-us/data-center/tesla-v100/["https://www.nvidia.com/en-us/data-center/tesla-v100/"^]
  - NGChttps://www.nvidia.com/en-us/gpu-cloud/["https://www.nvidia.com/en-us/gpu-cloud/"^]
- NVIDIA Jarvis 다중 모드 프레임워크
  - NVIDIA Jarvishttps://developer.nvidia.com/nvidia-jarvis["https://developer.nvidia.com/nvidia-jarvis"^]
  - NVIDIA Jarvis 조기 액세스https://developer.nvidia.com/nvidia-jarvis-early-access["https://developer.nvidia.com/nvidia-jarvis-early-access"^]
- NVIDIA 니모
  - NVIDIA 니모https://developer.nvidia.com/nvidia-nemo["https://developer.nvidia.com/nvidia-nemo"^]
  - 개발자 가이드 를 참조하십시오https://nvidia.github.io/NeMo/["https://nvidia.github.io/NeMo/"^]
- NetApp AFF 시스템
  - NetApp AFF A 시리즈 데이터시트https://www.netapp.com/us/media/ds-3582.pdf["https://www.netapp.com/us/media/ds-3582.pdf"^]
  - All Flash FAS에서 NetApp 플래시의 이점https://www.netapp.com/us/media/ds-3733.pdf["https://www.netapp.com/us/media/ds-3733.pdf"^]
  - ONTAP 9 정보 라이브러리http://mysupport.netapp.com/documentation/productlibrary/index.html?productID=62286["http://mysupport.netapp.com/documentation/productlibrary/index.html?productID=62286"^]

  - NetApp ONTAP FlexGroup 볼륨 기술 보고서https://www.netapp.com/us/media/tr-4557.pdf["https://www.netapp.com/us/media/tr-4557.pdf"^]
- NetApp ONTAP AI를 참조하십시오
  - DGX-1 및 Cisco 네트워킹 기반 ONTAP AI 설계 가이드https://www.netapp.com/us/media/nva-1121-design.pdf["https://www.netapp.com/us/media/nva-1121-design.pdf"^]
  - DGX-1 및 Cisco 네트워킹 지원 ONTAP AI 배포 가이드https://www.netapp.com/us/media/nva-1121-deploy.pdf["https://www.netapp.com/us/media/nva-1121-deploy.pdf"^]
  - DGX-1 및 Mellanox 네트워킹 설계 가이드를 지원하는 ONTAP AIhttp://www.netapp.com/us/media/nva-1138-design.pdf["http://www.netapp.com/us/media/nva-1138-design.pdf"^]
  - DGX-2 기반 ONTAP AI 설계 가이드https://www.netapp.com/us/media/nva-1135-design.pdf["https://www.netapp.com/us/media/nva-1135-design.pdf"^]

TR-4858: 실행 시 NetApp 오케스트레이션 솔루션: AI

Rick Huang, David Arnette, Sung-Han Lin, NetApp Yaron Goldberg, Run: AI

NetApp AFF 스토리지 시스템은 탁월한 성능과 업계 최고 수준의 하이브리드 클라우드 데이터 관리 기능을 제공합니다. NetApp 및 Run: AI는 엔터프라이즈급 성능, 안정성 및 지원을 제공하는 인공 지능(AI) 및 머신 러닝(ML) 워크로드용 NetApp ONTAP AI 솔루션의 고유한 기능을 시연하기 위해 파트너 계약을 체결했습니다. 실행: AI 워크로드 오케스트레이션에서 Kubernetes 기반 스케줄링 및 리소스 활용률 플랫폼을 추가하여 연구원이 GPU 활용률을 관리하고 최적화할 수 있도록 지원합니다. NVIDIA DGX 시스템과 NetApp, NVIDIA, Run의 통합 솔루션: AI는 엔터프라이즈 AI 워크로드를 위해 특별 제작된 인프라 스택을 제공합니다. 이 기술 보고서는 다양한 사용 사례와 산업 수직 분야를 지원하기 위해 대화형 AI 시스템을 구축하는 고객에게 직접적인 지침을 제공합니다. 이 이니셔티브에는 Run:AI 및 NetApp AFF A800 스토리지 시스템 구축에 대한 정보가 포함되며, AI 이니셔티브를 빠르고 성공적으로 구현하는 가장 간단한 방법을 위한 참조 아키텍처 역할을 합니다.

이 솔루션의 대상 고객은 다음과 같은 그룹을 포함합니다.

- 컨테이너 마이크로서비스와 같은 Kubernetes 기반 사용 사례에 대한 AI 모델 및 소프트웨어 개발을 위한 솔루션을 설계하는 엔터프라이즈 설계자
- 여러 팀 및 프로젝트가 있는 클러스터 환경에서 효율적인 모델 개발 목표를 달성할 수 있는 방법을 찾는 데이터 과학자
- 운영 모델의 유지 관리 및 실행을 담당하는 데이터 엔지니어
- 최적의 Kubernetes 클러스터 리소스 활용률 경험을 만들고 AI 이니셔티브를 통한 시장 출시 기간을 단축하려는 경영진 및 IT 의사 결정자 및 비즈니스 리더

솔루션 개요

NetApp ONTAP AI 및 AI 제어 플레인

NetApp과 NVIDIA가 개발 및 검증한 NetApp ONTAP AI 아키텍처는 NVIDIA DGX 시스템과 NetApp 클라우드 연결형 스토리지 시스템을 기반으로 합니다. 이 참조 아키텍처는 IT 조직이 다음과 같은 이점을 얻을 수 있도록 해 줍니다.

- 설계 복잡성 제거
- 컴퓨팅과 스토리지의 독립적인 확장 지원
- 고객이 작은 규모로 시작한 후 원활하게 확장할 수 있도록 지원
- 다양한 성능 및 비용 요소에 부합하는 폭넓은 스토리지 옵션을 제공합니다

NetApp ONTAP AI는 DGX 시스템과 NetApp AFF A800 스토리지 시스템을 최첨단 네트워킹과 긴밀하게 통합합니다. NetApp ONTAP AI 및 DGX 시스템은 설계 복잡성과 추측을 제거함으로써 AI 배포를 단순화합니다. 고객은 작은 규모로 시작한 후 에지에서 코어 및 클라우드까지 포괄하여 데이터를 지능적으로 관리하면서 중단 없이 시스템을 확장할 수 있습니다.

NetApp AI Control Plane은 데이터 과학자 및 데이터 엔지니어를 위한 전체 스택 AI, ML 및 딥 러닝(DL) 데이터 및 실험 관리 솔루션입니다. 조직이 AI를 더 많이 사용함에 따라 워크로드 확장성 및 데이터 가용성을 비롯한 여러 과제에 직면하게 됩니다. NetApp AI Control Plane은 Git repo와 마찬가지로 데이터 네임스페이스를 신속하게 클론 복제하여 추적 및 버전 관리를 위한 데이터 및 모델 기준을 거의 즉각적으로 생성하는 AI 교육 워크플로우를 정의 및 구현하는 등의 기능을 통해 이러한 문제를 해결합니다. NetApp AI Control Plane을 사용하면 사이트 및 지역 간에 데이터를 원활하게 복제하고 대규모 데이터 세트에 액세스할 수 있는 Jupyter Notebook 작업 공간을 신속하게 프로비저닝할 수 있습니다.

실행**: AI** 워크로드 오케스트레이션에 **AI** 플랫폼 사용

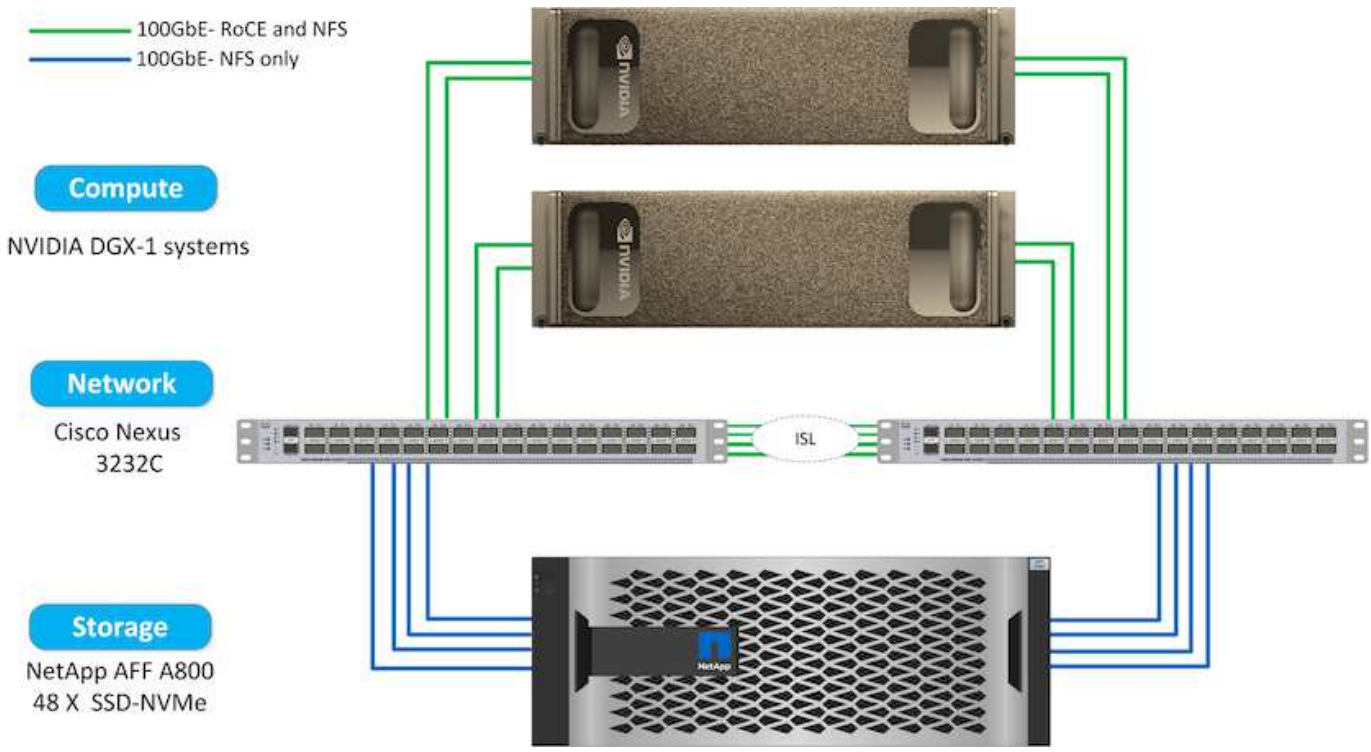
실행: AI는 AI 인프라를 위한 세계 최초의 오케스트레이션 및 가상화 플랫폼을 구축했습니다. 실행: AI는 기본 하드웨어에서 워크로드를 추상화하여 동적으로 프로비저닝할 수 있는 GPU 리소스 공유 풀을 만들어 AI 워크로드를 효율적으로 조정하고 GPU를 최적화된 상태로 사용할 수 있도록 지원합니다. 데이터 과학자는 대용량 GPU 전력을 원활하게 소비하여 연구 결과를 개선하고 가속화하는 동시에, IT 팀이 리소스 프로비저닝, 대기 및 활용률에 대한 중앙 집중식 교차 사이트 제어 및 실시간 가시성을 유지할 수 있습니다. 실행: AI 플랫폼은 Kubernetes를 기반으로 구축되므로 기존 IT 및 데이터 과학 워크플로우와의 간편한 통합이 가능합니다.

Run:AI 플랫폼은 다음과 같은 이점을 제공합니다.

- * 혁신을 위한 더 빠른 시간. * Run을 사용하면 AI 리소스 풀링, 큐 처리 및 우선순위 지정 메커니즘을 NetApp 스토리지 시스템과 함께 사용하여 연구원들은 인프라 관리 문제와 관련된 문제를 해결할 수 있으며 데이터 과학에만 집중할 수 있습니다. 실행: AI 및 NetApp 고객은 컴퓨팅 또는 데이터 파이프라인 병목 현상 없이 필요한 만큼 워크로드를 실행하여 생산성을 향상할 수 있습니다.
- * 팀 생산성 향상. * 실행: AI 공정성 알고리즘은 모든 사용자와 팀이 적절한 리소스 공유를 확보할 수 있도록 보장합니다. 우선 순위 프로젝트와 관련된 정책을 미리 설정할 수 있으며, 플랫폼을 통해 사용자 또는 팀 간에 리소스를 동적으로 할당할 수 있으므로 사용자가 원하는 GPU 리소스에 적시에 액세스할 수 있습니다.
- * GPU 사용률이 개선되었습니다. * 실행: AI 스케줄러를 사용하면 Kubernetes에서 분산된 훈련을 위해 소수점 GPU, 정수 GPU 및 여러 GPU 노드를 쉽게 사용할 수 있습니다. 이런 식으로 AI 워크로드는 용량이 아닌 사용자의 요구사항을 기반으로 실행됩니다. 데이터 과학 팀은 동일한 인프라에서 더 많은 AI 실험을 실행할 수 있습니다.

솔루션 기술

이 솔루션은 NetApp AFF A800 시스템 1대, DGX-1 서버 2대, Cisco Nexus 3232C 100GbE 스위치 2개를 사용하여 구축했습니다. RoCE(RDMA over Converged Ethernet)를 통한 원격 직접 메모리 액세스(RDMA)를 사용하여 각 DGX-1 서버는 GPU 간 통신에 사용되는 4개의 100GbE 연결을 통해 Nexus 스위치에 연결합니다. NFS 스토리지 액세스를 위한 기존 IP 통신도 이 링크에서 발생합니다. 4개의 100GbE 링크를 사용하여 각 스토리지 컨트롤러를 네트워크 스위치에 연결합니다. 다음 그림은 모든 테스트 시나리오에 대해 이 기술 보고서에 사용된 ONTAP AI 솔루션 아키텍처를 보여줍니다.



이 솔루션에 사용된 하드웨어

이 솔루션은 ONTAP AI 참조 아키텍처 2개의 DGX-1 노드 및 1개의 AFF A800 스토리지 시스템을 사용하여 검증되었습니다. 을 참조하십시오 "NVA-1121" 이 검증에 사용된 인프라에 대한 자세한 내용은 를 참조하십시오.

다음 표에는 솔루션을 테스트하는 데 필요한 하드웨어 구성요소가 나와 있습니다.

하드웨어	수량
DGX-1 시스템	2
AFF A800	1
Nexus 3232C 스위치	2

소프트웨어 요구 사항

이 솔루션은 Run:AI 운영자가 설치된 기본 Kubernetes 구축을 사용하여 검증되었습니다. Kubernetes는 를 사용하여 구축했습니다 "NVIDIA DeepOps" 구축 엔진: 즉시 프로덕션할 수 있는 환경에 필요한 모든 구성 요소를 배포합니다. DeepOps가 자동으로 배포됩니다 "NetApp 트라이던트" 스토리지를 K8s 환경과 지속적으로 통합하고 기본 스토리지 클래스를 만들어 컨테이너가 AFF A800 스토리지 시스템에서 스토리지를 활용하도록 했습니다. ONTAP AI 기반 Kubernetes를 사용하는 Trident에 대한 자세한 내용은 를 참조하십시오 "TR-4798".

다음 표에는 테스트를 통해 솔루션을 구현하는 데 필요한 소프트웨어 구성요소가 나와 있습니다.

소프트웨어	버전 또는 기타 정보
NetApp ONTAP 데이터 관리 소프트웨어	9.6p4
Cisco NX-OS 스위치 펌웨어	7.0(3) I6(1)
NVIDIA DGX OS	4.0.4 - Ubuntu 18.04 LTS

소프트웨어	버전 또는 기타 정보
Kubernetes 버전	1.17
Trident 버전	20.04.0
실행: AI CLI	v2.1.13
실행: AI Orchestration Kubernetes Operator version	1.0.39
Docker 컨테이너 플랫폼	18.06.1-CE[e68fc7a]

Run:AI에 대한 추가 소프트웨어 요구사항은 에서 확인할 수 있습니다 "AI GPU 클러스터의 사전 요구사항 을 실행하십시오".

실행 시 클러스터 및 **GPU** 활용률 최적화**: AI**

다음 섹션에서는 AI 설치, 테스트 시나리오, 그리고 이 검증에서 수행된 결과에 대해 자세히 설명합니다.

TensorFlow 벤치마크를 비롯하여 업계 표준 벤치마크 툴을 사용하여 이 시스템의 운영 및 성능을 검증했습니다. ImageNet 데이터 세트는 이미지 분류를 위해 유명한 CNN(Convolutional Neural Network) DL 모델인 ResNet-50을 교육하는 데 사용되었습니다. ResNet-50은 더욱 빠른 처리 시간으로 정확한 교육 결과를 제공하므로 스토리지에서 충분한 수요를 창출할 수 있습니다.

AI 설치 를 실행하십시오

Run:AI를 설치하려면 다음 단계를 완료하십시오.

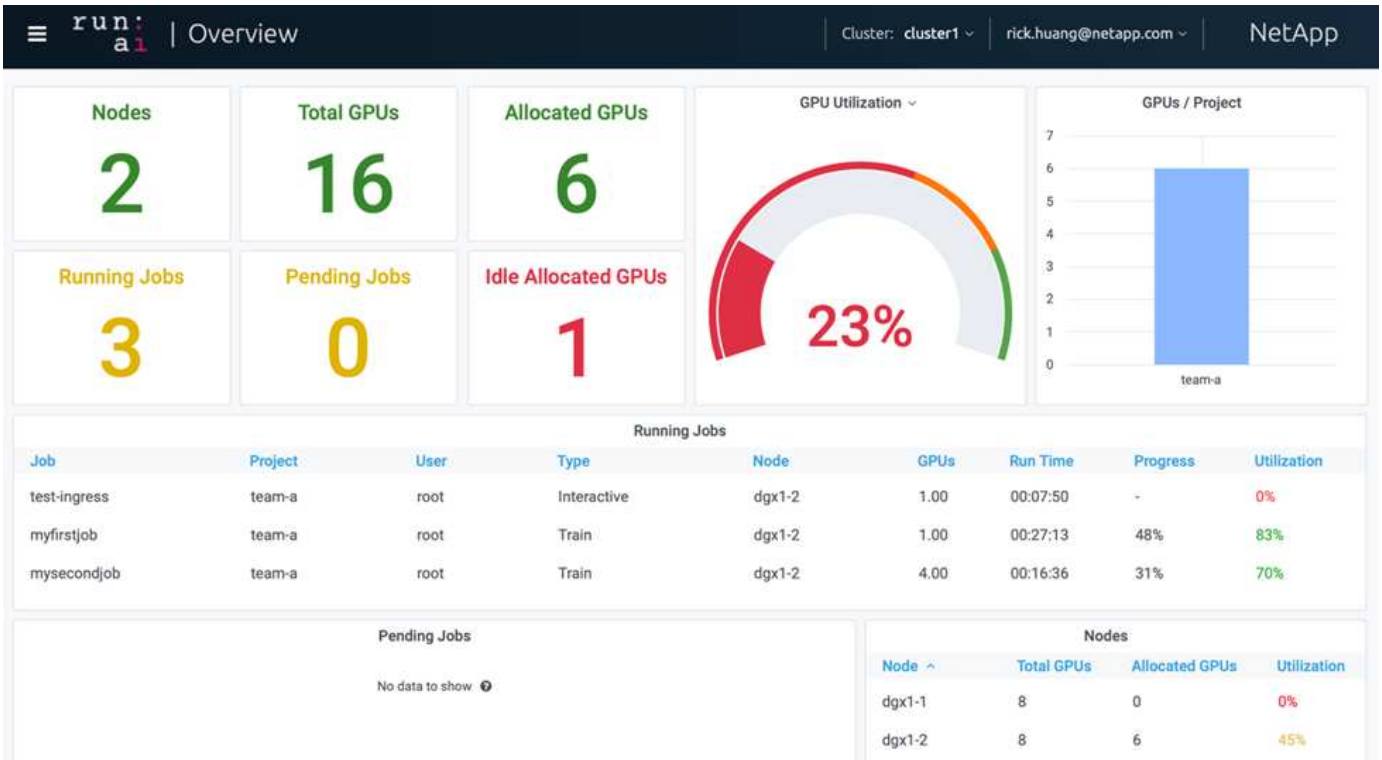
1. DeepOps를 사용하여 Kubernetes 클러스터를 설치하고 NetApp 기본 스토리지 클래스를 구성합니다.
2. GPU 노드 준비:
   a. NVIDIA 드라이버가 GPU 노드에 설치되었는지 확인합니다.
   b. nVidia-docker가 기본 Docker 런타임으로 설치 및 구성되어 있는지 확인합니다.
3. 러닝 설치: AI:
   a. 에 로그인합니다 "AI 관리자 UI를 실행합니다" 클러스터를 생성합니다.
   b. 생성된 'runai-operator-<clustername>.yAML' 파일을 다운로드합니다.
   c. Kubernetes 클러스터에 운영자 구성을 적용하십시오.

```
kubectl apply -f runai-operator-<clustername>.yaml
```

4. 설치를 확인합니다.
   a. 로 이동합니다 "https://app.run.ai/".
   b. 개요 대시보드로 이동합니다.
   c. 오른쪽 위에 있는 GPU 수가 예상 GPU 수를 반영하고 GPU 노드가 모두 서버 목록에 있는지 확인합니다. 실행:AI 배포에 대한 자세한 내용은 을 참조하십시오 "Run 설치: 사내 Kubernetes 클러스터에 AI를 설치합니다" 및 "Run:AI CLI 설치".

실행: **AI** 대시보드 및 뷰

Run:AI를 Kubernetes 클러스터에 설치하고 컨테이너를 올바르게 구성하면 에 다음과 같은 대시보드와 뷰가 표시됩니다 "https://app.run.ai" 다음 그림과 같이 브라우저에서



2개의 DGX-1 노드에서 제공하는 클러스터에 총 16개의 GPU가 있습니다. 노드 수, 총 사용 가능한 GPU, 워크로드와 할당된 할당된 할당된 GPU, 총 실행 중인 작업 수, 보류 중인 작업 및 유휴 할당 GPU를 볼 수 있습니다. 오른쪽의 막대 다이어그램에서는 프로젝트당 GPU를 보여 주며, 각 팀이 클러스터 리소스를 사용하는 방법을 요약합니다. 가운데는 작업 이름, 프로젝트, 사용자, 작업 유형, 각 작업이 실행 중인 노드, 해당 작업에 할당된 GPU 수, 작업의 현재 실행 시간, 작업의 작업 진행 상태, 해당 작업의 GPU 사용률 단일 팀('team-A')이 제출한 실행 중인 작업이 3개뿐이므로 클러스터 사용률이 낮은 상태(GPU 사용률이 23%)입니다.

다음 섹션에서는 프로젝트 탭에서 여러 팀을 생성하고 각 팀에 GPU를 할당하여 클러스터당 사용자가 많을 때 클러스터 사용을 최대화하고 리소스를 관리하는 방법을 보여줍니다. 테스트 시나리오는 훈련, 추론 및 대화형 워크로드 간에 메모리 및 GPU 리소스가 공유되는 엔터프라이즈 환경을 모방합니다.

데이터 과학 팀을 위한 프로젝트 생성 및 **GPU** 할당

연구원들은 Run:AI CLI, Kubeflow 또는 유사한 프로세스를 통해 워크로드를 제출할 수 있습니다. 리소스 할당을 간소화하고 우선 순위를 만들기 위해 Run:AI에는 프로젝트의 개념이 도입되었습니다. 프로젝트는 프로젝트 이름을 GPU 할당 및 기본 설정과 연결하는 할당량 요소입니다. 여러 데이터 과학 팀을 관리할 수 있는 간단하고 편리한 방법입니다.

워크로드를 제출하는 연구원은 프로젝트를 워크로드 요청과 연계해야 합니다. Run:AI 스케줄러는 요청을 현재 할당 및 프로젝트와 비교하여 워크로드에 리소스를 할당할 수 있는지 또는 보류 중 상태를 유지해야 하는지 여부를 결정합니다.

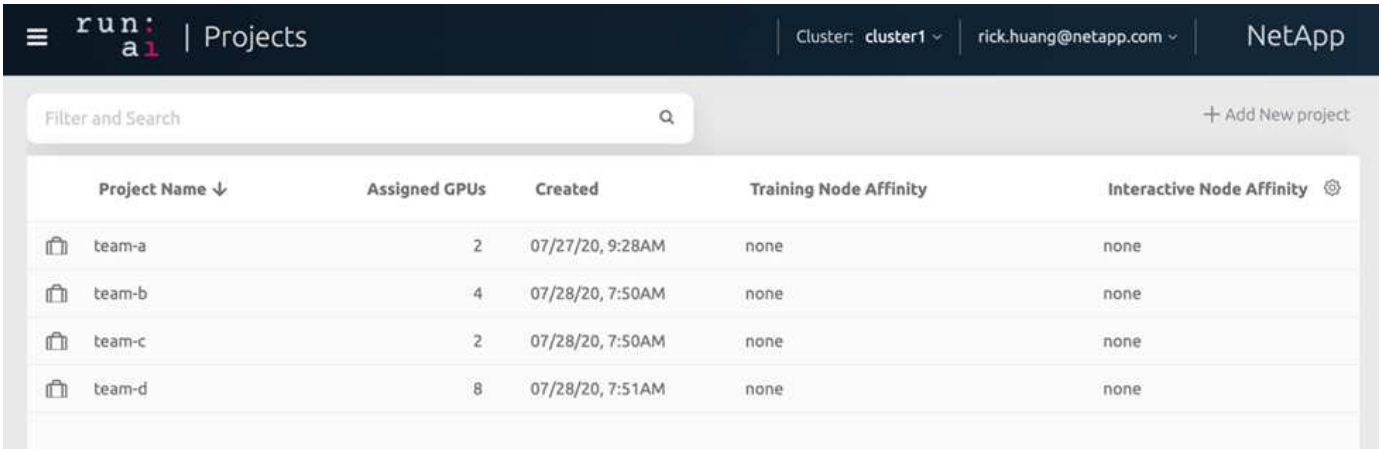
시스템 관리자는 실행: AI 프로젝트 탭에서 다음 매개 변수를 설정할 수 있습니다.

- * 모델 프로젝트 * 사용자별 프로젝트를 설정하고, 사용자 팀별로 프로젝트를 설정하고, 실제 조직 프로젝트별로

프로젝트를 설정합니다.

- * 프로젝트 할당량 * 각 프로젝트는 이 프로젝트에 동시에 할당할 수 있는 GPU 할당량과 연관됩니다. 이 프로젝트는 클러스터의 상태에 관계없이 이 프로젝트를 사용하는 연구원이 GPU 수를 확보할 수 있다는 점에서 보장된 할당량입니다. 일반적으로 프로젝트 할당의 합계는 클러스터에 있는 GPU 수와 같아야 합니다. 이 외에도 이 프로젝트의 사용자는 초과 할당량을 받을 수 있습니다. GPU를 사용하지 않는 한, 이 프로젝트를 사용하는 연구자는 더 많은 GPU를 얻을 수 있습니다. 에서는 할당량 초과 테스트 시나리오와 공정성 고려 사항을 보여 줍니다 "할당량 초과 GPU 할당을 통한 높은 클러스터 사용률 달성", "기본 자원 할당 공정성", 및 "할당량 초과 공정성".
- 새 프로젝트를 만들고, 기존 프로젝트를 업데이트하고, 기존 프로젝트를 삭제합니다.
- * 특정 노드 그룹에서 실행할 작업 제한 *. 특정 노드에서만 실행되도록 특정 프로젝트를 할당할 수 있습니다. 이 기능은 프로젝트 팀이 충분한 메모리를 갖춘 특수 하드웨어가 필요한 경우에 유용합니다. 또는 프로젝트 팀은 전문 예산으로 구입한 특정 하드웨어의 소유자가 되거나, 더 약한 하드웨어에서 작동하고 더 긴 훈련이나 무인 워크로드를 더 빠른 노드로 직접 처리하기 위해 직접 빌드하거나 대화형 워크로드를 실행해야 할 수도 있습니다. 노드를 그룹화하고 특정 프로젝트에 대한 선호도를 설정하는 명령은 을 참조하십시오 "AI 문서 를 실행하십시오".
- * 대화형 작업 기간 제한 *. 연구자들은 종종 대화식 작업을 종결하는 것을 잊어버립니다. 이로 인해 리소스가 낭비될 수 있습니다. 일부 조직에서는 대화형 작업의 기간을 제한하고 자동으로 작업을 종결하는 것을 선호합니다.

다음 그림에서는 네 개의 팀이 생성된 프로젝트 보기를 보여 줍니다. 각 팀에는 서로 다른 워크로드를 처리할 수 있는 서로 다른 수의 GPU가 할당되며, 총 GPU 수는 2개의 DGX-1로 구성된 클러스터에서 사용 가능한 총 GPU 수와 같습니다.



실행 중인 작업 제출**: AI CLI**

이 섹션에서는 Kubernetes 작업 실행에 사용할 수 있는 기본 Run:AI 명령에 대한 자세한 정보를 제공합니다. 워크로드 유형에 따라 3개 부분으로 나뉩니다. AI/ML/DL 워크로드는 다음과 같은 두 가지 일반 유형으로 나눌 수 있습니다.

- * 무인 교육 세션 *. 이러한 유형의 워크로드를 사용하여 데이터 과학자는 자체 실행 워크로드를 준비하여 실행을 위해 보냅니다. 실행 중에 고객은 결과를 검토할 수 있습니다. 이러한 유형의 워크로드는 생산 또는 인물 개발에 사람의 개입이 필요 없는 단계에 있을 때 주로 사용됩니다.
- * 대화형 빌드 세션 *. 이러한 유형의 워크로드를 사용하여 데이터 과학자는 Bash, Jupyter Notebook, remote PyCharm 또는 유사한 IDE를 사용한 대화형 세션을 열고 GPU 리소스에 직접 액세스합니다. 연결된 포트를 사용하여 대화형 워크로드를 실행하는 세 번째 시나리오가 포함되어 컨테이너 사용자에게 내부 포트를 제공합니다.

무인 교육 워크로드

프로젝트를 설정하고 GPU를 지정한 후 명령줄에서 다음 명령을 사용하여 모든 Kubernetes 워크로드를 실행할 수

있습니다.

```
$ runai project set team-a runai submit hyper1 -i gcr.io/run-ai-
demo/quickstart -g 1
```

이 명령은 단일 GPU를 할당하여 팀 A의 무인 교육 작업을 시작합니다. 이 작업은 샘플 Docker 이미지 'GCR.IO/RUN-AI-DEMO/QuickStart'를 기반으로 합니다. 우리는 그 일을 하이퍼1이라고 명명했다. 그런 다음 다음 다음 명령을 실행하여 작업의 진행률을 모니터링할 수 있습니다.

```
$ runai list
```

다음 그림은 루나이 리스트 명령의 결과를 보여준다. 표시되는 일반적인 상태는 다음과 같습니다.

- 'ContainerCreating' Docker 컨테이너를 클라우드 저장소에서 다운로드하고 있습니다.
- '보류 중'. 작업이 예약될 때까지 대기 중입니다.
- '러닝'입니다. 작업이 실행 중입니다.

```
~> runai list
Showing jobs for project team-a
NAME    STATUS   AGE  NODE                                    IMAGE                        TYPE        PROJECT  USER   GPUs
hyper1  Running  11s  gke-dev-yaron1-gpu-4-pool-154f511d-5nk5  gcr.io/run-ai-demo/quickstart  Train       team-a   yaron  1
```

작업에 대한 추가 상태를 가져오려면 다음 명령을 실행합니다.

```
$ runai get hyper1
```

작업의 로그를 보려면 "runai logs <job-name>" 명령을 실행합니다.

```
$ runai logs hyper1
```

이 예에서는 각 단계에 대해 현재 교육 Epoch, ETA, 손실 함수 값, 정확도 및 경과 시간을 포함하여 실행 중인 DL 세션의 로그를 확인해야 합니다.

의 Run:AI UI에서 클러스터 상태를 볼 수 있습니다 "https://app.run.ai/". 대시보드 > 개요 에서 GPU 사용률을 모니터링할 수 있습니다.

이 워크로드를 중지하려면 다음 명령을 실행합니다.

```
$ runai delte hyper1
```

이 명령은 교육 워크로드를 중지합니다. 이 작업은 'runai list'를 다시 실행하여 확인할 수 있습니다. 자세한 내용은 을 참조하십시오 "무인 교육 워크로드 실행".

대화형 빌드 워크로드

프로젝트를 설정하고 GPU를 할당하면 명령줄에서 다음 명령을 사용하여 대화형 빌드 워크로드를 실행할 수 있습니다.

```
$ runai submit build1 -i python -g 1 --interactive --command sleep --args
infinity
```

이 작업은 샘플 Docker 이미지 Python을 기반으로 합니다. 우리는 작업 구축1이라는 이름을 붙였습니다.

> 인터액티브 플래그는 작업이 시작이나 끝이 없다는 뜻입니다 이 일을 마무리하는 것은 연구자의 책임입니다. 관리자는 시스템에 의해 종료된 후 대화형 작업에 대한 시간 제한을 정의할 수 있습니다.

이 작업에는 '--g 1' 플래그가 GPU를 하나만 할당합니다. 명령어와 논리는 '--명령 슬립—args 무한대'입니다. 명령을 제공해야 합니다. 그렇지 않고 컨테이너가 시작되고 즉시 종료됩니다.

다음 명령은 에 설명된 명령과 유사하게 작동합니다 무인 교육 워크로드:

- 'runai list': 이름, 상태, 나이, 노드, 이미지, 작업을 위해 프로젝트, 사용자 및 GPU를 지원합니다.
- runai get build1: 작업 build1에 추가 상태를 표시합니다.
- 'runai delete build1': 대화형 워크로드 빌드 중지1. bash 셸을 컨테이너에 가져오려면 다음 명령을 사용합니다.

```
$ runai bash build1
```

그러면 컴퓨터에 직접 셸이 제공됩니다. 그런 다음 데이터 과학자는 컨테이너 내에서 모델을 개발 또는 미세 조정할 수 있습니다.

의 Run:AI UI에서 클러스터 상태를 볼 수 있습니다 "https://app.run.ai". 자세한 내용은 을 참조하십시오 "대화형 빌드 워크로드 시작 및 사용".

연결된 포트를 사용하는 대화형 작업 부하

대화형 빌드 워크로드의 확장으로 Run:AI CLI로 컨테이너를 시작할 때 컨테이너 사용자에게 내부 포트를 표시할 수 있습니다. 이 기능은 Jupyter Notebooks와 함께 작업하거나 다른 마이크로서비스에 연결하는 클라우드 환경에 유용합니다. "침투" Kubernetes 클러스터 외부에서 Kubernetes 서비스에 액세스할 수 있습니다. 어떤 인바운드 연결이 어떤 서비스에 연결할지 정의하는 규칙 모음을 만들어 액세스를 구성할 수 있습니다.

클러스터의 서비스에 대한 외부 액세스를 보다 효율적으로 관리하기 위해 클러스터 관리자를 설치하는 것이 좋습니다 "침투" 및 로드 밸런서를 구성합니다.

서비스 유형으로 수신을 사용하려면 다음 명령을 실행하여 워크로드를 제출할 때 메서드 유형과 포트를 설정합니다.

```
$ runai submit test-ingress -i jupyter/base-notebook -g 1 \
  --interactive --service-type=ingress --port 8888 \
  --args="--NotebookApp.base_url=test-ingress" --command=start-notebook.sh
```

컨테이너가 성공적으로 시작된 후 runai list를 실행하여 Jupyter Notebook에 액세스할 수 있는 Service URL(S)을

확인합니다. URL은 수신 엔드포인트, 작업 이름 및 포트로 구성됩니다. 예를 들어 를 참조하십시오 https://10.255.174.13/test-ingress-8888.

자세한 내용은 을 참조하십시오 "연결된 포트를 사용하여 대화형 빌드 워크로드 시작".

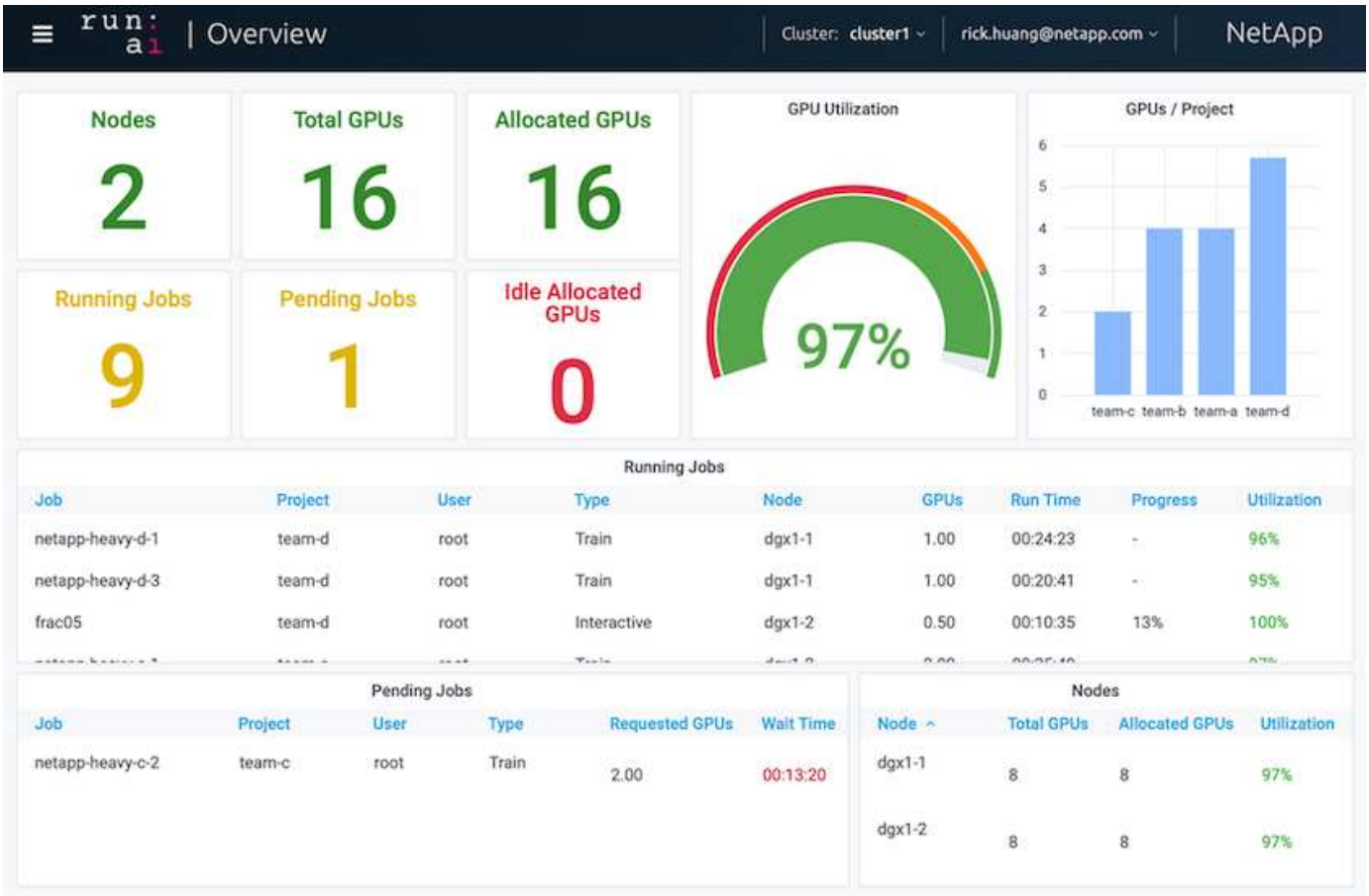
높은 클러스터 사용률 달성

이 섹션에서는 GPU 리소스의 우선순위 지정 및 밸런싱을 유지하면서 높은 클러스터 사용률을 달성하는 Run:AI 오케스트레이션 솔루션을 시연하기 위해 4개의 데이터 과학 팀이 각자 고유의 워크로드를 제출하는 실제 시나리오를 에뮬레이트합니다. 먼저 섹션에 설명된 ResNet-50 벤치마크를 사용합니다 "ImageNet 데이터 세트 벤치마크 요약을 통한 ResNet-50":

```
$ runai submit netapp1 -i netapp/tensorflow-tf1-py3:20.01.0 --local-image
--large-shm  -v /mnt:/mnt -v /tmp:/tmp --command python --args
"/netapp/scripts/run.py" --args "--
dataset_dir=/mnt/mount_0/dataset/imagenet/imagenet_original/" --args "--
num_mounts=2"  --args "--dgx_version=dgx1" --args "--num_devices=1" -g 1
```

에서와 같이 ResNet-50 벤치마크를 실행했습니다 "NVA-1121". 우리는 공공 Docker 리포지토리에 없는 컨테이너에 '--local-image' 플래그를 사용했습니다. 호스트 DGX-1 노드의 /mnt와 /tmp 디렉토리를 각각 컨테이너에 `/mnt', '/tmp' 디렉토리에 마운트했습니다. 데이터 세트는 디렉토리를 가리키는 dataset_dir와 함께 NetApp AFFA800에 있습니다. '--num_devices=1'과 '-g 1'은 이 작업에 하나의 GPU를 할당한다는 것을 의미합니다. 전자는 run.py 스크립트의 주장이고 후자는 runai submit 명령의 플래그입니다.

다음 그림은 97%의 GPU 사용률과 16개의 사용 가능한 GPU가 할당된 시스템 개요 대시보드를 보여 줍니다. GPU/프로젝트 막대 차트에서 각 팀에 할당된 GPU 수를 쉽게 확인할 수 있습니다. 실행 중인 작업 창에는 현재 실행 중인 작업 이름, 프로젝트, 사용자, 유형, 노드, GPU 사용량, 실행 시간, 진행률 및 활용률 세부 정보 대기 시간이 있는 대기열의 워크로드 목록이 보류 중인 작업에 표시됩니다. 마지막으로, 노드 상자는 클러스터의 개별 DGX-1 노드에 대한 GPU 수와 활용률을 제공합니다.



덜 까다로운 워크로드 또는 대화형 워크로드에 대한 부분 **GPU** 할당

연구자와 개발자가 개발, 고매개 변수 조정 또는 디버깅 단계에서 자신의 모델을 작업할 때 이러한 워크로드는 일반적으로 컴퓨팅 리소스를 적게 사용합니다. 따라서 동일한 GPU를 다른 워크로드에 동시에 할당할 수 있도록 소수점 GPU 및 메모리를 프로비저닝하는 것이 더 효율적입니다. 실행: AI의 오케스트레이션 솔루션은 Kubernetes에서 컨테이너화된 워크로드를 위한 분할 GPU 공유 시스템을 제공합니다. 이 시스템은 CUDA 프로그램을 실행하는 워크로드를 지원하며 추론과 모델 구축과 같은 가벼운 AI 작업에 특히 적합합니다. 소수점 GPU 시스템은 데이터 과학과 AI 엔지니어링 팀에게 단일 GPU에서 동시에 여러 워크로드를 실행할 수 있는 기능을 투명하게 제공합니다. 이를 통해 기업은 컴퓨터 비전, 음성 인식 및 자연어 처리와 같은 더 많은 워크로드를 동일한 하드웨어에서 실행할 수 있으므로 비용이 절감됩니다.

실행: AI의 분할 GPU 시스템은 컨테이너가 자급식 프로세서인 것처럼 사용하고 액세스할 수 있는 자체 메모리 및 컴퓨팅 공간을 사용하여 가상화된 논리 GPU를 효과적으로 생성합니다. 따라서 여러 워크로드가 서로 간섭하지 않고 동일한 GPU의 컨테이너에서 나란히 실행될 수 있습니다. 이 솔루션은 투명하고 단순하며 이식 가능하며 컨테이너 자체를 변경할 필요가 없습니다.

일반적인 UseCase는 동일한 GPU에서 실행되는 작업을 2~8개 볼 수 있으며, 이는 동일한 하드웨어에서 8배 더 많은 작업을 수행할 수 있음을 의미합니다.

다음 그림에서 PROJECT 팀 d에 속한 Frac05 작업에 대해 할당된 GPU 수가 0.50인 것을 알 수 있다. 이는 컨테이너에 사용 가능한 GPU 메모리가 DGX-1 노드의 V100 GPU당 32GB의 절반 인 16,255MB임을 보여 주는 'NVIDIA-SMI' 명령으로 더욱 검증되었습니다.

```
root@run-deploy:~# runai bash frac05 -p team-d
root@frac05-0:/workload# nvidia-smi
Tue Jul 28 15:17:03 2020
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 450.51.05    Driver Version: 450.51.05    CUDA Version: 11.0     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  Tesla V100-SXM2...  On   | 00000000:07:00.0 Off |                    0 |
| N/A   57C    P0   240W / 300W |  15525MiB / 16255MiB |    100%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A       156      C   python3                         15525MiB |
+-----------------------------------------------------------------------------+
```

할당량 초과 **GPU** 할당을 통한 높은 클러스터 사용률 달성

섹션을 참조하십시오 "기본 자원 할당 공정성", 및 "할당량 초과 공정성"복잡한 워크로드 관리, 자동 사전 예방 예약 및 초과 할당량 GPU 프로비저닝을 위한 Run:AI 조정 기능을 시연하기 위해 고급 테스트 시나리오를 고안했습니다. 이를 통해 ONTAP AI 환경에서 클러스터 리소스를 많이 사용하고 엔터프라이즈급 데이터 과학 팀 생산성을 최적화할 수 있었습니다.

이 세 섹션에서는 다음 프로젝트 및 할당량을 설정합니다.

프로젝트	할당량
팀-A	4
팀-b	2
팀 - c	2
팀 d	8

또한 다음 세 개의 단원에 다음과 같은 컨테이너를 사용합니다.

- Jupyter Notebook: jupyter/base-notebook
- Run:AI QuickStart:'GCR.IO/RUN-AI-DEMO/QuickStart'를 실행하십시오

이 테스트 시나리오에 대해 다음과 같은 목표를 설정했습니다.

- 리소스 프로비저닝의 간편성 및 리소스를 사용자로부터 추상화한 방법을 보여줍니다

- GPU의 분수와 GPU의 정수 수를 간편하게 프로비저닝하는 방법을 보여줍니다
- 클러스터에 무료 GPU가 있을 경우 팀 또는 사용자가 리소스 할당량을 처리할 수 있으므로 시스템에서 컴퓨팅 병목 현상이 해소되는 방법을 보여줍니다
- NetApp 컨테이너와 같은 컴퓨팅 집약적인 작업을 실행할 때 NetApp 솔루션을 사용하여 데이터 파이프라인의 병목 현상을 제거하는 방법을 보여줍니다
- 시스템을 사용하여 여러 유형의 컨테이너를 실행하는 방법을 보여 줍니다
  - Jupyter 노트북
  - 실행: AI 컨테이너
- 클러스터가 가득 찼을 때 높은 사용률을 표시합니다

테스트 중에 실행된 실제 명령 시퀀스에 대한 자세한 내용은 을 참조하십시오 "섹션 4.8의 테스트 세부 사항".

13개의 워크로드를 모두 제출하면 다음 그림과 같이 할당된 컨테이너 이름 및 GPU 목록을 볼 수 있습니다. NetApp은 7개의 교육 및 6개의 대화식 작업을 통해 4개의 데이터 과학 팀을 시뮬레이션하며 각 팀은 개발 또는 자체 모델을 실행하고 있습니다. 대화형 작업의 경우, 개별 개발자는 Jupyter Notebooks를 사용하여 코드를 작성하거나 디버깅합니다. 따라서 클러스터 리소스를 너무 많이 사용하지 않고 GPU 분할을 프로비저닝하는 것이 좋습니다.

```
root@run-deploy:~# runai list -A
NAME            STATUS   AGE  NODE    IMAGE                          TYPE        PROJECT  USER  GPUs  CREATED BY CLI  SERVICE URL(S)
b-4-gg          Running  2m   dgx1-2  gcr.io/run-ai-demo/quickstart  Train       team-b   root  2     true
c-5-g           Running  2m   dgx1-2  gcr.io/run-ai-demo/quickstart  Train       team-c   root  1     true
c-4-gg          Running  2m   dgx1-1  gcr.io/run-ai-demo/quickstart  Train       team-c   root  2     true
b-3-g           Running  2m   dgx1-1  gcr.io/run-ai-demo/quickstart  Train       team-b   root  1     true
c-3-g02         Running  2m   dgx1-1  gcr.io/run-ai-demo/quickstart  Interactive team-c   root  0.2   true
d-1-gggg        Running  2m   dgx1-2  gcr.io/run-ai-demo/quickstart  Train       team-d   root  4     true
c-2-g03         Running  2m   dgx1-1  gcr.io/run-ai-demo/quickstart  Interactive team-c   root  0.3   true
c-1-g05         Running  2m   dgx1-1  gcr.io/run-ai-demo/quickstart  Interactive team-c   root  0.5   true
a-2-gg          Running  3m   dgx1-1  gcr.io/run-ai-demo/quickstart  Train       team-a   root  2     true
b-2-g04         Running  3m   dgx1-2  gcr.io/run-ai-demo/quickstart  Interactive team-b   root  0.4   true
a-1-g           Running  3m   dgx1-1  gcr.io/run-ai-demo/quickstart  Train       team-a   root  1     true
b-1-g06         Running  3m   dgx1-2  gcr.io/run-ai-demo/quickstart  Interactive team-b   root  0.6   true
a-1-1-jupyter   Running  3m   dgx1-1  jupyter/base-notebook          Interactive team-a   root  1     true            http://10.61.218.134/a-1-1-jupyter,
https://10.61.218.134/a-1-1-jupyter
```

이 테스트 시나리오의 결과는 다음과 같습니다.

- 클러스터가 꽉 찼어야 합니다. 16/16개의 GPU를 사용했습니다.
- 높은 클러스터 사용률.
- 부분 할당으로 인해 GPU보다 더 많은 실험
- 팀 d는 쿼터를 모두 사용하지 않으므로 팀 b와 팀 c는 실험에 추가 GPU를 사용할 수 있어 혁신의 시간을 단축할 수 있습니다.

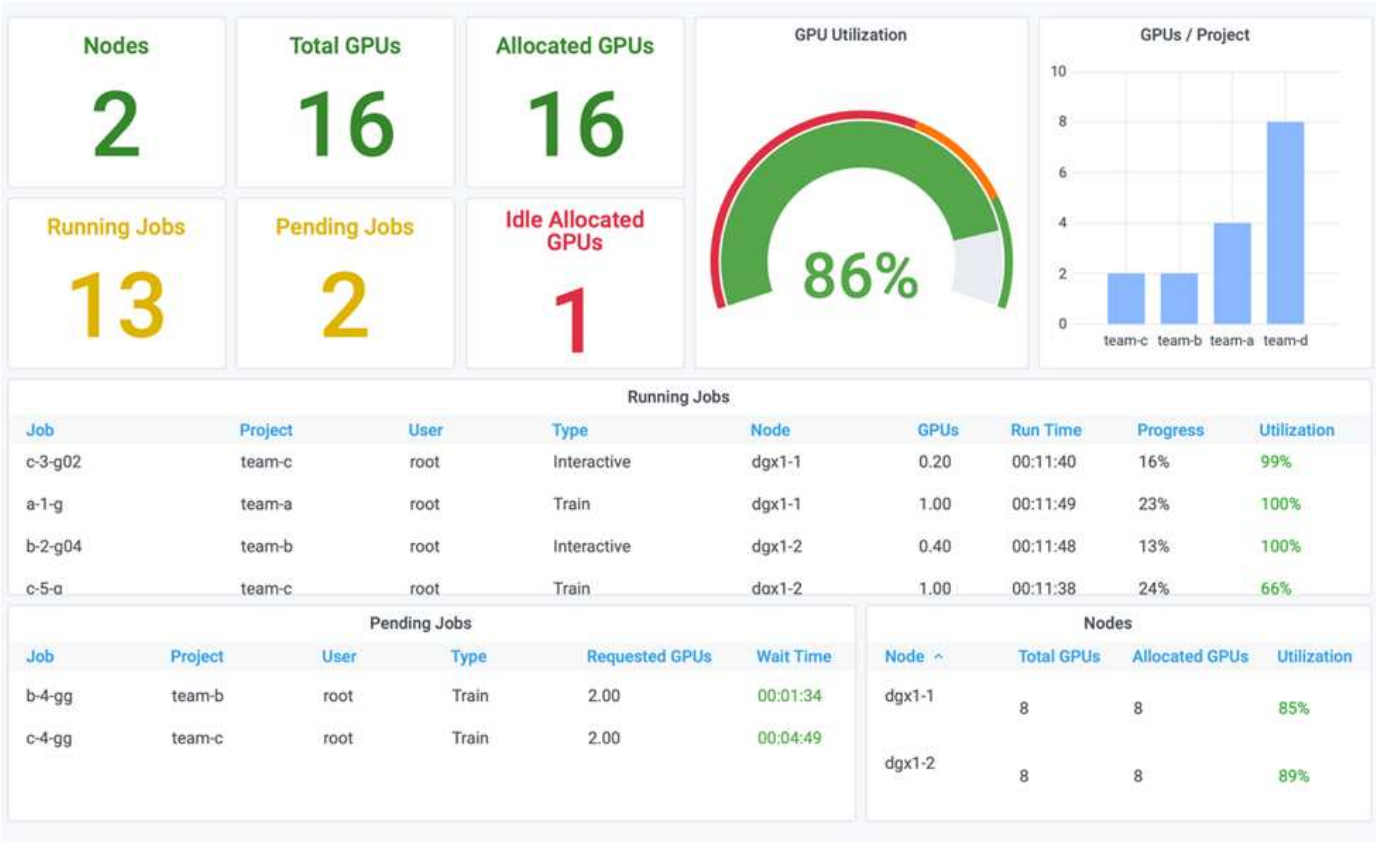
기본 자원 할당 공정성

이 섹션에서는 'team-d'가 더 많은 GPU(할당량 아래)를 요청할 때 시스템이 'team-b'와 'team-c'의 워크로드를 일시 중지하고 공평한 분배 방식으로 보류 중인 상태로 전환한다는 것을 보여줍니다.

작업 제출, 사용된 컨테이너 이미지 및 실행된 명령 시퀀스를 포함한 자세한 내용은 섹션을 참조하십시오 "섹션 4.9의 테스트 세부 정보".

다음 그림은 자동 로드 밸런싱 및 사전 예방 예약 기능으로 인해 발생하는 클러스터 활용률, 팀당 할당된 GPU 및 보류 중인 작업을 보여줍니다. 모든 팀 작업 부하에 의해 요청된 총 GPU 수가 클러스터에서 사용 가능한 총 GPU 수를 초과할 때 Run:AI의 내부 공정성 알고리즘은 프로젝트 할당량을 충족했기 때문에 "team-b"와 "team-c"에 대해 각각

하나의 작업을 일시 중지한다는 것을 알 수 있습니다. 따라서 전반적인 높은 클러스터 활용률이 제공되지만 데이터 과학 팀은 관리자가 설정한 리소스 제약 조건에서 작업을 계속 수행할 수 있습니다.



이 테스트 시나리오의 결과는 다음과 같습니다.

- * 자동 로드 밸런싱. * 시스템은 GPU의 할당량을 자동으로 조정하여 각 팀에서 현재 할당량을 사용하고 있습니다. 일시 중지된 워크로드는 할당량이 초과된 팀에 속합니다.
- * 공정한 공유 일시 중지. * 시스템이 할당량이 초과된 팀의 작업 부하를 중지하도록 선택한 다음 다른 팀의 작업 부하를 중지시킵니다. 실행: AI에는 내부 공정성 알고리즘이 있습니다.

할당량 초과 공정성

이 섹션에서는 여러 팀에서 워크로드를 제출하고 할당량을 초과하는 시나리오를 확장합니다. 이 방법으로 Run:AI의 Fairness 알고리즘이 사전 설정된 할당량의 비율에 따라 클러스터 리소스를 할당하는 방법을 보여 줍니다.

이 테스트 시나리오의 목표:

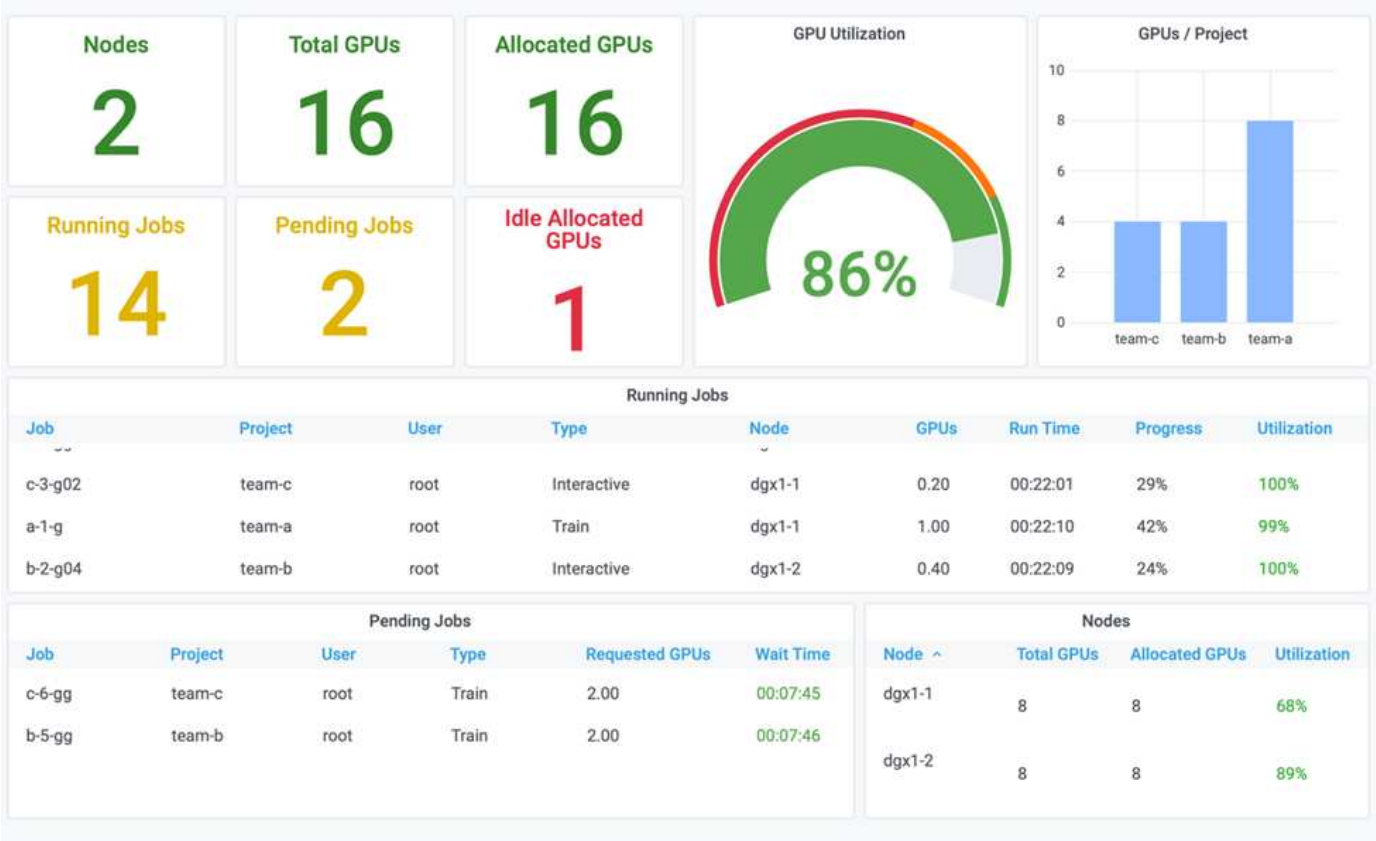
- 여러 팀에서 할당량을 통해 GPU를 요청할 때 큐 메커니즘을 표시합니다.
- 할당량 비율에 따라 할당량이 초과된 여러 팀 간에 클러스터가 공평하게 분배되어 할당량이 더 큰 팀이 여유 용량을 더 많이 점유하도록 하는 방법을 보여 줍니다.

의 끝에 있습니다 "기본 자원 할당 공정성"팀-b, 팀-c의 두 가지 워크로드가 대기 중입니다. 이 섹션에서는 추가 워크로드를 전담합니다.

작업 제출, 사용된 컨테이너 이미지 및 실행된 명령 시퀀스를 포함한 자세한 내용은 을 참조하십시오 "섹션 4.10의

테스트 세부 정보".

섹션에 따라 모든 작업이 제출되는 경우 "섹션 4.10의 테스트 세부 정보"시스템 대시보드에는 팀-A, 팀-b, 팀-c가 모두 미리 설정된 할당량보다 더 많은 GPU를 가지고 있는 것으로 표시됩니다. 팀 A는 미리 설정된 소프트 쿼터보다 4개의 GPU를 더 점유하고, 팀 b와 팀 c는 각각 소프트 할당량(2개)보다 2개의 GPU를 더 점유합니다. 할당된 초과 할당량 GPU의 비율은 사전 설정된 할당량의 비율과 동일합니다. 이는 시스템이 사전 설정 할당량을 우선 순위에 따라 사용하고 여러 팀에서 더 많은 GPU를 요청하고 할당량을 초과할 경우 적절히 프로비저닝하기 때문입니다. 이러한 자동 로드 밸런싱은 엔터프라이즈 데이터 과학 팀이 AI 모델 개발 및 생산에 적극적으로 참여할 때 공정성과 우선순위를 제공합니다.



이 테스트 시나리오의 결과는 다음과 같습니다.

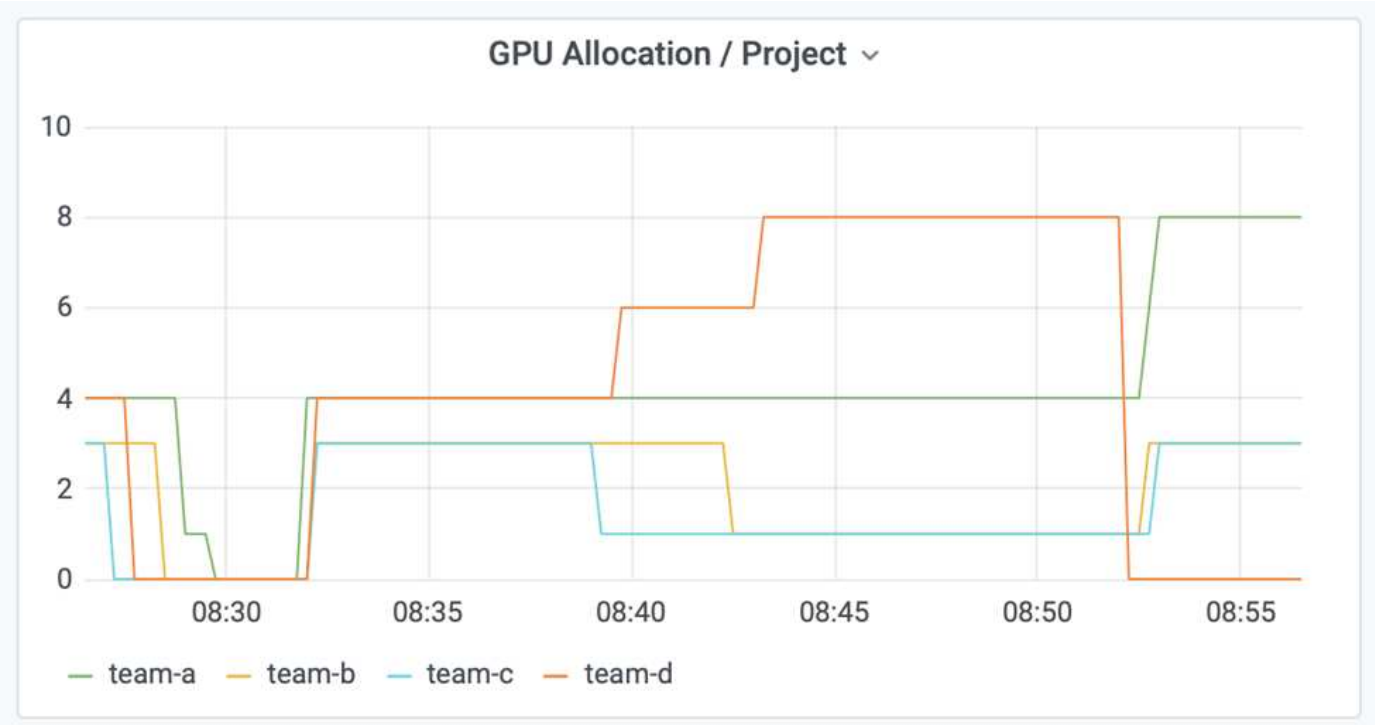
- 시스템이 다른 팀의 작업 부하를 취소하기 시작합니다.
- 대기열은 공정성 알고리즘에 따라 결정되며, 팀-b와 팀-c는 할당량 초과 GPU(할당량이 비슷하므로)와 동일한 양을 할당받습니다. 또 팀-A는 쿼터보다 2배 많은 양의 GPU를 갖게 된다. 팀-b, 팀-c의 쿼터보다 쿼터량이 2배 더 높기 때문이다.
- 모든 할당이 자동으로 수행됩니다.

따라서 시스템은 다음 상태에서 안정되어야 합니다.

프로젝트	**GPU**가 할당되었습니다	설명
팀-A	8월 4일	할당량에 4개의 GPU가 사용됩니다. 대기열이 비어 있습니다.
팀-b	4월 2일	할당량을 통해 2개의 GPU가 제공됩니다. 하나의 워크로드가 대기 중입니다.

프로젝트	**GPU**가 할당되었습니다	설명
팀 - c	4월 2일	할당량을 통해 2개의 GPU가 제공됩니다. 하나의 워크로드가 대기 중입니다.
팀 d	0/8	GPU를 전혀 사용하지 않고, 대기 중인 워크로드가 없습니다.

다음 그림에서는 섹션에 대한 Run:AI Analytics 대시보드의 시간별 프로젝트당 GPU 할당을 보여 줍니다 "할당량 초과 GPU 할당을 통한 높은 클러스터 사용률 달성", "기본 자원 할당 공정성", 및 "할당량 초과 공정성". 그림의 각 줄은 언제든지 특정 데이터 과학 팀에 프로비저닝된 GPU 수를 나타냅니다. 시스템이 제출된 워크로드에 따라 GPU를 동적으로 할당하는지 확인할 수 있습니다. 따라서 클러스터에 사용 가능한 GPU가 있을 때 팀은 할당량을 초과하고, 공정성에 따라 작업을 사전에 미분하여 4팀 모두에 대해 안정적인 상태가 될 수 있습니다.



Trident에서 프로비저닝한 **PersistentVolume**에 데이터 저장

NetApp Trident는 컨테이너화된 애플리케이션의 정교한 지속성 요구사항을 충족하도록 설계된 완전 지원되는 오픈 소스 프로젝트입니다. 데이터 계층화, 암호화, NetApp Snapshot 기술, 규정 준수 및 NetApp ONTAP 데이터 관리 소프트웨어에서 제공하는 우수한 성능의 이점을 추가로 활용하여 Trident에서 프로비저닝한 Kubernetes PersistentVolume(PV)에 데이터를 읽고 쓸 수 있습니다.

기존 네임스페이스에서 **PVC** 재사용

대규모 AI 프로젝트의 경우, 여러 컨테이너가 동일한 Kubernetes PV에서 데이터를 읽고 쓰는 것이 더 효율적일 수 있습니다. Kubernetes PVC(Persistent Volume Claim)를 재사용하려면 이미 PVC를 생성해야 합니다. 를 참조하십시오 "NetApp Trident 문서" PVC 작성에 대한 자세한 내용은. 다음은 기존 PVC를 재사용하는 예입니다.

```
$ runai submit pvc-test -p team-a --pvc test:/tmp/pvc1mount -i gcr.io/run-
ai-demo/quickstart -g 1
```

다음 명령어를 실행해 프로젝트 팀 A에 대한 작업 PVC-TEST의 상태를 확인할 수 있다.

```
$ runai get pvc-test -p team-a
```

team-A job 'PVC-test'에 PV/tmp/pvc1mount가 마운트된 것을 볼 수 있습니다. 이렇게 하면 여러 컨테이너를 동일한 볼륨에서 읽을 수 있으므로 개발 또는 운영 중인 경쟁 모델이 여러 개 있을 때 유용합니다. 데이터 과학자는 모델의 앙상블을 만든 다음 대부분의 투표 또는 기타 기술을 통해 예측 결과를 결합할 수 있습니다.

다음을 사용하여 컨테이너 셸에 액세스합니다.

```
$ runai bash pvc-test -p team-a
```

그런 다음 마운트된 볼륨을 확인하고 컨테이너 내의 데이터에 액세스할 수 있습니다.

PVC를 재사용할 수 있는 이 기능은 NetApp FlexVol 볼륨 및 NetApp ONTAP FlexGroup 볼륨과 함께 작동하여 데이터 엔지니어가 보다 유연하고 강력한 데이터 관리 옵션을 통해 NetApp이 제공하는 Data Fabric을 활용할 수 있도록 지원합니다.

결론

NetApp 및 Run: AI는 이 기술 보고서에서 AI 워크로드 오케스트레이션을 단순화하기 위한 Run:AI 플랫폼과 함께 NetApp ONTAP AI 솔루션의 고유한 기능을 시연했습니다. 이전 단계에서는 딥 러닝을 위한 데이터 파이프라인 및 워크로드 오케스트레이션의 프로세스를 간소화하는 참조 아키텍처를 제공합니다. 이러한 솔루션을 구현하려는 고객은 NetApp 및 Run:AI에 자세한 내용을 문의하도록 권장합니다.

섹션 **4.8**의 테스트 세부 사항

이 섹션에서는 섹션의 테스트 세부 정보를 다룹니다 "할당량 초과 GPU 할당을 통한 높은 클러스터 사용률 달성".

다음 순서로 작업을 제출합니다.

프로젝트	이미지	**GPU** 수	합계	설명
팀-A	Jupyter를 선택합니다	1	1/4	–
팀-A	넷엡	1	2월 4일	–
팀-A	실행: AI	2	4월 4일	모든 할당량을 사용합니다
팀-b	실행: AI	0.6	0.6/2	소수점 GPU

프로젝트	이미지	**GPU** 수	합계	설명
팀-b	실행: AI	0.4	1/2로	소수점 GPU
팀-b	넷엡	1	2월 2일	–
팀-b	넷엡	2	4월 2일	2개 초과 할당량
팀 - c	실행: AI	0.5	0.5/2	소수점 GPU
팀 - c	실행: AI	0.3	2월 8일	소수점 GPU
팀 - c	실행: AI	0.2	1/2로	소수점 GPU
팀 - c	넷엡	2	3월 2일	1개 초과 할당
팀 - c	넷엡	1	4월 2일	2개 초과 할당량
팀 d	넷엡	4	8월 4일	할당량의 절반을 사용합니다

명령 구조:

```
$ runai submit <job-name> -p <project-name> -g <#GPUs> -i <image-name>
```

테스트에 사용된 실제 명령 시퀀스:

```
$ runai submit a-1-1-jupyter -i jupyter/base-notebook -g 1 \
  --interactive --service-type=ingress --port 8888 \
  --args="--NotebookApp.base_url=team-a-test-ingress" --command=start
-notebook.sh -p team-a
$ runai submit a-1-g -i gcr.io/run-ai-demo/quickstart -g 1 -p team-a
$ runai submit a-2-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-a
$ runai submit b-1-g06 -i gcr.io/run-ai-demo/quickstart -g 0.6
--interactive -p team-b
$ runai submit b-2-g04 -i gcr.io/run-ai-demo/quickstart -g 0.4
--interactive -p team-b
$ runai submit b-3-g -i gcr.io/run-ai-demo/quickstart -g 1 -p team-b
$ runai submit b-4-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-b
$ runai submit c-1-g05 -i gcr.io/run-ai-demo/quickstart -g 0.5
--interactive -p team-c
$ runai submit c-2-g03 -i gcr.io/run-ai-demo/quickstart -g 0.3
--interactive -p team-c
$ runai submit c-3-g02 -i gcr.io/run-ai-demo/quickstart -g 0.2
--interactive -p team-c
$ runai submit c-4-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-c
$ runai submit c-5-g -i gcr.io/run-ai-demo/quickstart -g 1 -p team-c
$ runai submit d-1-gggg -i gcr.io/run-ai-demo/quickstart -g 4 -p team-d
```

이때 다음과 같은 상태가 있어야 합니다.

프로젝트	GPU가 할당되었습니다	작업 로드 대기 중
팀-A	4/4(소프트 할당량/실제 할당)	없음
팀-b	4월 2일	없음
팀 - c	4월 2일	없음
팀 d	8월 4일	없음

섹션을 참조하십시오 "Over-uota GPU 할당을 통한 높은 클러스터 활용률 달성" 진행 중인 테스트 시나리오에 대한 논의.

섹션 4.9의 테스트 세부 정보

이 섹션에서는 섹션에 대한 테스트 세부 정보를 다룹니다 "기본 자원 할당 공정성".

다음 순서로 작업을 제출합니다.

프로젝트	GPU 수	합계	설명
팀 d	2	6/8	팀-b/c 워크로드가 일시 중지되고 "보류 중"으로 이동합니다.
팀 d	2	8월 8일	다른 팀(b/c)의 워크로드는 일시 중지되고 "보류 중"으로 이동합니다.

다음 실행된 명령 시퀀스를 참조하십시오.

```
$ runai submit d-2-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-d$
runai submit d-3-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-d
```

이때 다음과 같은 상태가 있어야 합니다.

프로젝트	GPU가 할당되었습니다	작업 로드 대기 중
팀-A	4월 4일	없음
팀-b	2월 2일	없음
팀 - c	2월 2일	없음
팀 d	8월 8일	없음

섹션을 참조하십시오 "기본 자원 할당 공정성" 진행 중인 테스트 시나리오에 대한 논의.

섹션 4.10에 대한 테스트 세부 정보

이 섹션에서는 섹션에 대한 테스트 세부 정보를 다룹니다 "할당량 초과 공정성".

팀-A, 팀-b, 팀-c의 순서로 작업을 제출합니다.

프로젝트	GPU 수	합계	설명
팀-A	2	4월 4일	1개의 워크로드가 대기열에 있습니다
팀-A	2	4월 4일	2개의 작업 부하가 대기 중입니다
팀-b	2	2월 2일	2개의 작업 부하가 대기 중입니다
팀 - c	2	2월 2일	2개의 작업 부하가 대기 중입니다

다음 실행된 명령 시퀀스를 참조하십시오.

```
$ runai submit a-3-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-a$
runai submit a-4-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-a$ runai
submit b-5-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-b$ runai
submit c-6-gg -i gcr.io/run-ai-demo/quickstart -g 2 -p team-c
```

이때 다음과 같은 상태가 있어야 합니다.

프로젝트	GPU가 할당되었습니다	작업 로드 대기 중
팀-A	4월 4일	각각 GPU를 묻는 2개의 워크로드
팀-b	2월 2일	각각 2개의 GPU를 요구하는 2개의 워크로드
팀 - c	2월 2일	각각 2개의 GPU를 요구하는 2개의 워크로드
팀 d	8월 8일	없음

그런 다음 'team-d'에 대한 모든 워크로드를 삭제합니다.

```
$ runai delete -p team-d d-1-gggg d-2-gg d-3-gg
```

섹션을 참조하십시오 "할당량 초과 공정성"를 참조하십시오.

추가 정보를 찾을 수 있는 위치

이 문서에 설명된 정보에 대한 자세한 내용은 다음 리소스를 참조하십시오.

- NVIDIA DGX 시스템
  - NVIDIA DGX-1 시스템https://www.nvidia.com/en-us/data-center/dgx-1/[]
  - NVIDIA V100 Tensor 코어 GPUhttps://www.nvidia.com/en-us/data-center/tesla-v100/[]
  - NGChttps://www.nvidia.com/en-us/gpu-cloud/[]
- 실행: AI 컨테이너 오케스트레이션 솔루션

  - 실행: AI 제품 소개https://docs.run.ai/home/components/[]
  - AI 설치 설명서를 실행하십시오https://docs.run.ai/Administrator/Cluster-Setup/Installing-Run-AI-on-an-on-premise-Kubernetes-Cluster/[] https://docs.run.ai/Administrator/Researcher-Setup/Installing-the-Run-AI-Command-Line-Interface/
  - Run:AI CLI에서 작업 제출https://docs.run.ai/Researcher/Walkthroughs/Walkthrough-Launch-Unattended-Training-Workloads-/[] https://docs.run.ai/Researcher/Walkthroughs/Walkthrough-Start-and-Use-Interactive-Build-Workloads-/
  - 실행 시 GPU 분할 할당: AI CLIhttps://docs.run.ai/Researcher/Walkthroughs/Walkthrough-Using-GPU-Fractions/[]
- NetApp AI Control Plane
  - 기술 보고서https://www.netapp.com/us/media/tr-4798.pdf[]
  - 간단한 데모https://youtu.be/gfr_sO27Rvo[]
  - GitHub 리포지토리https://github.com/NetApp/kubeflow_jupyter_pipeline[]
- NetApp AFF 시스템
  - NetApp AFF A 시리즈 데이터시트https://www.netapp.com/us/media/ds-3582.pdf[]
  - All Flash FAS에서 NetApp 플래시의 이점https://www.netapp.com/us/media/ds-3733.pdf[]
  - ONTAP 9 정보 라이브러리http://mysupport.netapp.com/documentation/productlibrary/index.html?productID=62286[]
  - NetApp ONTAP FlexGroup 볼륨 기술 보고서https://www.netapp.com/us/media/tr-4557.pdf[]
- NetApp ONTAP AI를 참조하십시오
  - DGX-1 및 Cisco 네트워킹 기반 ONTAP AI 설계 가이드https://www.netapp.com/us/media/nva-1121-design.pdf[]
  - DGX-1 및 Cisco 네트워킹 지원 ONTAP AI 배포 가이드https://www.netapp.com/us/media/nva-1121-deploy.pdf[]
  - DGX-1 및 Mellanox 네트워킹 설계 가이드를 지원하는 ONTAP AIhttp://www.netapp.com/us/media/nva-1138-design.pdf[]
  - DGX-2 기반 ONTAP AI 설계 가이드https://www.netapp.com/us/media/nva-1135-design.pdf[]

TR-4799 - 설계: 자율 주행 워크로드를 위한 NetApp ONTAP AI 참조 아키텍처

David Arnette 및 Sung-Han Lin, NetApp

NVIDIA DGX 시스템 제품군은 엔터프라이즈 AI용으로 특별 제작된 세계 최초의 통합 인공지능(AI) 플랫폼입니다. NetApp AFF 스토리지 시스템은 탁월한 성능과 업계 최고 수준의 하이브리드 클라우드 데이터 관리 기능을 제공합니다. NetApp과 NVIDIA는 협력 관계를 맺고 엔터프라이즈급 지원, 안정성 및 성능으로 AI 및 머신 러닝(ML) 워크로드를 지원하는 턴키 솔루션을 고객에게 제공할 수 있는 NetApp ONTAP AI 참조 아키텍처를 구축했습니다.

"TR-4799 - 설계: 자율 주행 워크로드를 위한 NetApp ONTAP AI 참조 아키텍처"

TR-4811: 의료 서비스를 위한 NetApp ONTAP AI 참조 아키텍처: 진단 이미징 - 솔루션 설계

Rick Huang, Sung-Han Lin, Sathish Thyagarajan, NetApp Jacci Cenci, NVIDIA

이 참조 아키텍처는 NVIDIA DGX-2 시스템과 의료 사용 사례용 NetApp AFF 스토리지를 사용하는 인공 지능(AI) 인프라를 구축하는 고객을 위한 지침을 제공합니다. 이 영상에는 의료 진단 영상, 검증된 테스트 사례 및 결과를 위한 딥 러닝(DL) 모델 개발에 사용되는 고급 워크플로우에 대한 정보가 포함되어 있습니다. 또한 고객 구축을 위한 사이징 권장사항도 포함되어 있습니다.

"TR-4811: 의료 서비스를 위한 NetApp ONTAP AI 참조 아키텍처: 진단 이미징 - 솔루션 설계"

TR-4807: 금융 서비스 워크로드를 위한 NetApp ONTAP AI 참조 아키텍처 - 솔루션 설계

Karthikeyan Nagalingam, Sung-Han Lin, NetApp Jacci Cenci, NVIDIA

이 참조 아키텍처는 NVIDIA DGX-1 시스템과 NetApp AFF 스토리지를 사용하여 금융 부문 사용 사례를 위한 인공 지능 인프라를 구축하려는 고객을 위한 지침을 제공합니다. 여기에는 재무 서비스 테스트 사례 및 결과를 위한 딥 러닝 모델 개발에 사용되는 상위 레벨의 워크플로우에 대한 정보가 포함됩니다. 또한 고객 구축을 위한 사이징 권장사항도 포함되어 있습니다.

"TR-4807: 금융 서비스 워크로드를 위한 NetApp ONTAP AI 참조 아키텍처 - 솔루션 설계"

Generative AI 및 NetApp 가치

저자: 사티시 Thyagarajan, NetApp

추상화

세대 인공 지능(AI)에 대한 수요로 인해 산업 전반에 걸쳐 변혁이 일어나고 있으며, 비즈니스 창의성과 제품 혁신이 증진되고 있습니다. 많은 조직에서 세대 AI를 사용하여 새로운 제품 기능을 구축하고, 엔지니어링 생산성을 개선하고, 더 나은 결과와 소비자 경험을 제공하는 AI 기반 애플리케이션을 프로토타입을 제작하고 있습니다. GPT(Generative Pre-training Transformers)와 같은 생성 AI는 신경망을 사용하여 텍스트, 오디오 및 비디오와 같은 다양한 새로운 콘텐츠를 만듭니다. 대규모 언어 모델(LLM)과 관련된 극단적인 규모와 대규모 데이터 세트를 고려할 때, 사내, 하이브리드 및 멀티 클라우드 구축 옵션의 강력한 데이터 스토리지 기능을 활용하고 데이터 이동성과 관련된 위험을 줄여주는 강력한 AI 인프라를 구축하는 것이 매우 중요합니다. 기업이 AI 솔루션을 설계할 수 있기에 앞서 데이터 보호 및 거버넌스: 이 백서에서는 이러한 고려 사항과 더불어 훈련, 재훈련, 미세 조정 및 추론 생성 AI 모델을 위한 AI 데이터 파이프라인 전반에서 원활한 데이터 관리와 데이터 이동을 지원하는 NetApp ® AI 기능에 대해 설명합니다.

핵심 요약

2022년 11월 GPT-3의 분할인 ChatGPT가 출시된 이후 가장 최근에는 사용자 지시에 대응하여 텍스트, 코드, 이미지 또는 치료 단백질을 생성하는 데 사용되는 새로운 AI 도구가 유명해졌습니다. 이는 사용자가 자연어를 사용하여 요청을 할 수 있음을 의미하며, AI는 사용자 요청을 반영한 뉴스 기사 또는 제품 설명과 같은 텍스트를 해석하고 생성하며 기존 데이터에 대해 훈련된 알고리즘을 사용하여 코드, 음악, 음성, 시각 효과 및 3D 자산을 생성합니다. 그 결과 AI 시스템의

설계에서 안정적인 확산, 환각, 신속한 엔지니어링 및 가치 조정과 같은 문구가 빠르게 등장하고 있습니다. 다양한 다운스트림 NLP(자연어 처리) 작업을 위해 다양한 업종의 비즈니스 시설에서 채택하고 있는 클라우드 서비스 공급자 및 기타 AI 파트너를 통해 사전 훈련된 기반 모델(FM)으로 이러한 셀프 감독 또는 반감독 머신 러닝(ML) 모델을 널리 사용하고 있습니다. McKinsey와 같은 연구 분석 기관 에서 "Generative AI가 생산성에 미치는 영향은 글로벌 경제에 수조 달러의 가치를 더할 수 있습니다." 기업들이 AI를 인간에 대한 사려 깊은 파트너로 재해석하고 있으며, FMS는 기업과 기관이 생성 AI를 통해 수행할 수 있는 기능으로 동시에 확장되고 있지만, 대량의 데이터를 관리할 수 있는 기회는 지속적으로 증가할 것입니다. 이 문서에서는 온프레미스 환경과 하이브리드 또는 멀티 클라우드 환경에서 NetApp 고객에게 가치를 제공하는 NetApp 기능과 관련하여 생성 AI 및 설계 개념을 소개합니다.

- 이제 고객이 AI 환경에서 NetApp를 사용하는 데 필요한 IT 조직은 누구입니까? * NetApp를 통해 빠른 데이터 및 클라우드 성장, 멀티 클라우드 관리 및 AI 같은 차세대 기술 채택으로 인해 발생하는 복잡성을 충족할 수 있습니다. NetApp은 다양한 기능을 지능형 데이터 관리 소프트웨어 및 스토리지 인프라에 결합했으며 AI 워크로드에 최적화된 고성능과 잘 균형을 이루었습니다. LLM과 같은 발전적인 AI 솔루션은 인텔리전스를 증진하기 위해 스토리지에서 메모리로 여러 번 소스 데이터 세트를 읽고 처리해야 합니다. NetApp은 에지-코어-클라우드 에코시스템 전반에서 데이터 이동성, 데이터 거버넌스, 데이터 보안 기술 분야의 선두업체로서 기업 고객에게 규모에 맞는 AI 솔루션을 구축할 수 있도록 지원해 왔습니다. NetApp은 강력한 파트너 네트워크를 통해 최고 데이터 책임자, AI 엔지니어, 엔터프라이즈 설계자 및 데이터 과학자를 지원하여 데이터 준비, 데이터 보호, AI 모델 훈련 및 추론의 전략적 데이터 관리 책임을 맡아 AI/ML 라이프사이클의 성능 및 확장성을 최적화할 수 있습니다. 딥 러닝 데이터 파이프라인을 위한 ® ONTAP AI ®, 스토리지 엔드포인트 간에 데이터를 원활하고 효율적으로 전송하기 위한 NetApp NetApp ® SnapMirror ® 등의 NetApp 데이터 기술 및 기능 실시간 렌더링을 위한 NetApp ® FlexCache ® 를 활용하면 데이터 흐름이 배치에서 실시간으로 이동하고 데이터 엔지니어링이 즉각적으로 이루어질 때 실시간 생성 AI 모델 구축의 가치를 제공합니다. 모든 유형의 기업이 새로운 AI 툴을 도입할 때는 에지, 데이터 센터 및 클라우드에서 확장 가능하고 책임적이며 설명할 수 있는 AI 솔루션을 요구하는 데이터 문제가 발생합니다. 하이브리드 및 멀티 클라우드에 대한 데이터 관련 최고의 권위자인 NetApp은 생성 AI 모델 훈련(사전 교육), 미세 조정, 컨텍스트 기반 추론 및 LLM의 모델 붕괴 모니터링을 위해 데이터 파이프라인 및 데이터 레이크 구성의 모든 측면을 지원할 수 있는 파트너 네트워크와 공동 솔루션 구축을 위해 노력하고 있습니다.

Generative AI란?

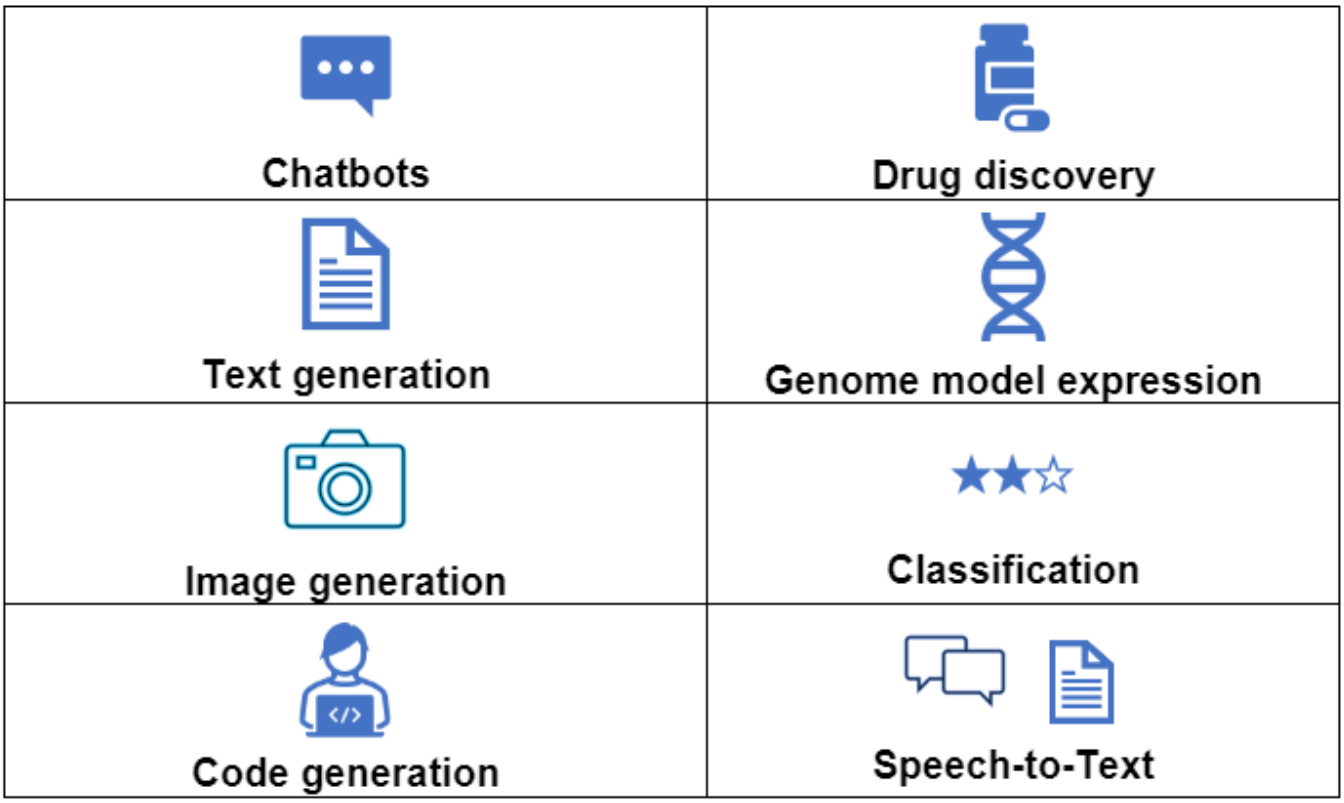
Generative AI는 콘텐츠 제작, 새로운 설계 개념 생성, 새로운 구성 탐색 방법을 변화시키고 있습니다. 텍스트, 코드, 이미지, 오디오, 비디오 등의 새로운 콘텐츠를 생성할 수 있는 GAN(Generative Adversarial Network), VAE(Variational Autoencoder) 및 GPT(Generative Pre-training Transformers)와 같은 신경망 프레임워크를 보여줍니다. 및 합성 데이터를 관리합니다. OpenAI의 Chat-GPT, Google’s Bard, 포옹하는 얼굴의 Bloom, Meta’s Llama와 같은 변압기 기반 모델은 대규모 언어 모델의 많은 발전을 뒷받침하는 기초 기술로 부상했습니다. 마찬가지로 OpenAI의 Dall-E, Meta의 CM3leon 및 Google의 Imagen도 텍스트-이미지 확산 모델의 예입니다. 이 모델은 고객에게 전례 없는 수준의 사진주의를 제공하여 처음부터 복잡한 새 이미지를 만들거나 기존 이미지를 편집함으로써 데이터 세트 증강과 텍스트-이미지 합성 텍스트를 연결하는 고품질 컨텍스트 인식 이미지를 생성합니다. 디지털 아티스트들은 Nerf(Neural Radiance Field)와 생성 AI와 같은 렌더링 기술을 결합하여 정적 2D 이미지를 몰입형 3D 장면으로 변환하기 시작했습니다. 일반적으로 LLM은 (1) 모델 크기(일반적으로 수십억 개의 매개 변수), (2) 교육 데이터 세트 크기, (3) 교육 비용, (4) 교육 후 성능 모델의 4가지 매개 변수로 구성됩니다. 또한 LLM은 주로 세 가지 변압기 아키텍처에 속합니다. (i) 인코더 전용 모델. 예: BERT(Google, 2018), (ii) 인코더-디코더(예: BART(Meta, 2020) 및 (iii) 디코더 전용 모델 예: Llama(Meta, 2023), Palm-E(Google, 2023). 일반적으로 비즈니스 요구사항에 따라 회사가 모델 매개 변수 수(N)와 교육 데이터 세트의 토큰 수(D)를 선택하는 아키텍처와 관계없이 교육(사전 교육) 또는 LLM의 세부 조정 비용을 결정합니다.

엔터프라이즈 사용 사례 및 다운스트림 **NLP** 작업

업종에 상관없이 AI가 비즈니스 운영, 영업, 마케팅, 법률 서비스를 위해 기존 데이터에서 새로운 형태의 가치를 추출하고 창출할 수 있는 가능성이 점점 더 커지고 있습니다. IDC(International Data Corporation)의 글로벌 생성 AI 사용 사례 및 투자에 대한 시장 인텔리전스에 따르면 소프트웨어 개발 및 제품 설계에 대한 지식 관리가 가장 큰 영향을 받으며, 개발자를 위한 마케팅 및 코드 생성을 위한 스토리라인 생성이 그 뒤를 이라고 합니다. 의료 분야에서는 임상 연구 조직이 의료 분야의 새로운 지평을 열고 있습니다. ProteinBERT와 같은 사전 훈련된 모델은 Gene Ontology(GO) 주석을 통합하여 약물의 단백질 구조를 신속하게 설계함으로써 약물 발견, 생물정보학 및 분자 생물학 분야에서 중요한 이정표를 제시합니다. 생명공학 기업들은 폐조직의 비가역적 흉터를 유발하는 폐섬유증(IPF)과 같은 질병을 치료하기

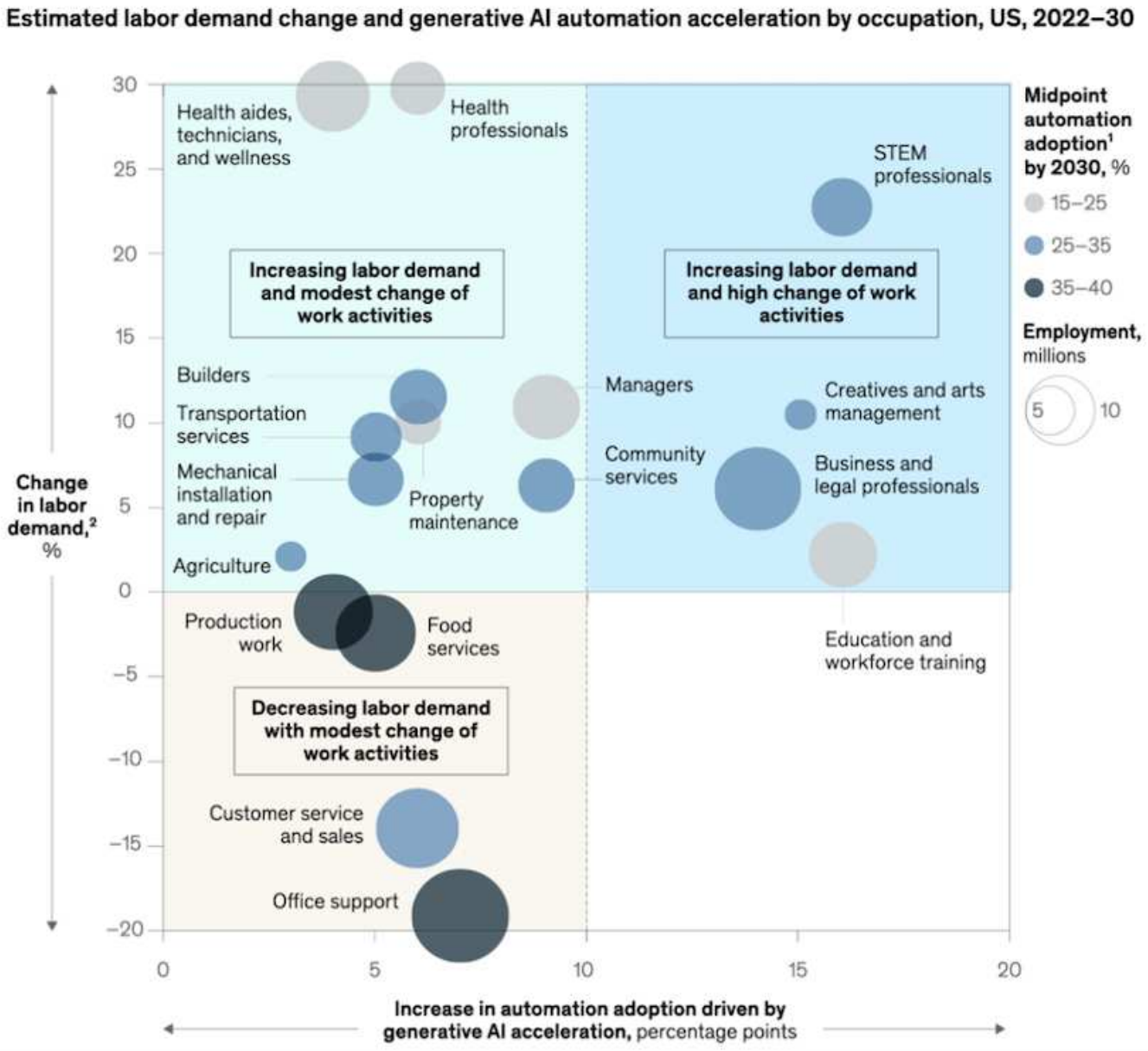
위한 목적으로 AI에서 발견된 신약 개발에 대한 인간 실험을 시작했습니다.

그림 1: 생성 AI를 이끄는 사용 사례



생성 AI를 통한 자동화 채택의 증가는 많은 직업의 업무 활동의 공급과 수요도 변화시키고 있습니다. McKinsey에 따르면 미국 노동 시장(아래 다이어그램)은 AI의 영향을 고려할 때만 지속되는 급속한 전환을 겪었습니다.

출처: McKinsey & Company



생성 **AI**에서 스토리지의 역할

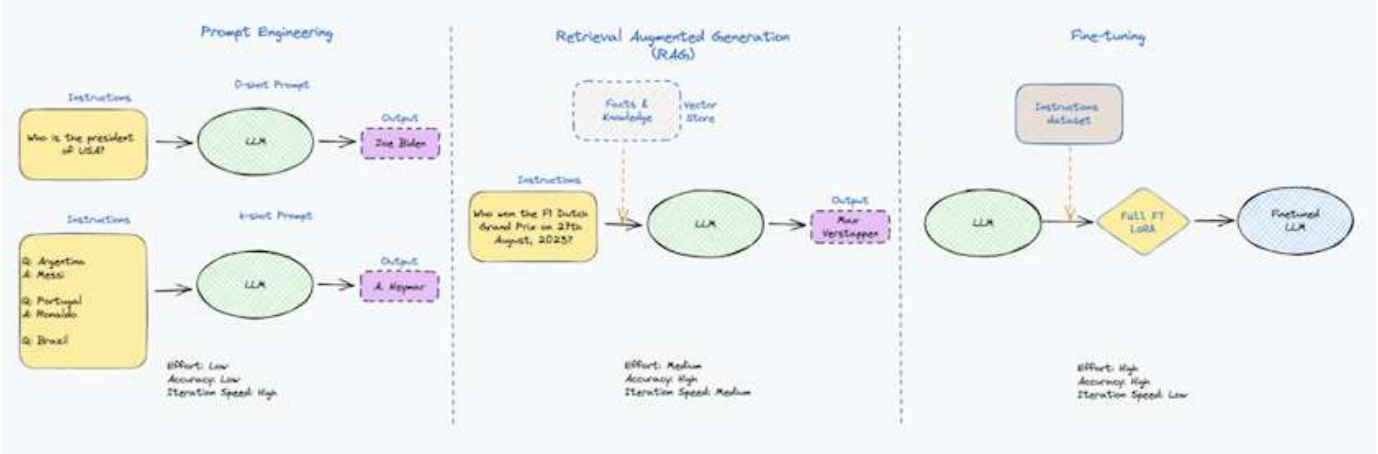
LLM은 주로 딥 러닝, GPU 및 컴퓨팅에 의존하고 있습니다. 하지만 GPU 버퍼가 가득 차면 데이터를 스토리지에 빠르게 기록해야 합니다. 일부 AI 모델은 메모리에서 실행될 수 있을 만큼 작지만, LLM에는 대규모 데이터 세트에 신속하게 액세스할 수 있도록 높은 IOPS와 높은 처리량 스토리지가 필요합니다. 특히 수십억 개의 토큰 또는 수백만 개의 이미지가 필요한 경우 그렇습니다. LLM의 일반적인 GPU 메모리 요구 사항에서 10억 개의 매개 변수를 가진 모델을 훈련하는 데 필요한 메모리는 최대 80GB @ 32비트 정밀도로 작동할 수 있습니다. 이 경우 70억 - 700억 개 매개 변수의 LLM 제품군인 Meta의 Llama 2는 약 70x80이 필요할 수 있습니다 5600GB 또는 5.6TB의 GPU RAM 또한 필요한 메모리 양은 생성하려는 최대 토큰 수에 직접 비례합니다. 예를 들어, 최대 512개의 토큰(약 380 단어)의 출력을 생성하려는 경우 이 필요합니다 "512MB". 이것은 비순차적인 것처럼 보일 수 있지만, 더 큰 배치를 실행하려는 경우 누적되기 시작합니다. 따라서 기업에서는 메모리의 LLM을 훈련하거나 미세 조정하는 데 비용이 매우 많이 들기 때문에 스토리지가 생성 AI의 토대가 되고 있습니다.

LLM에 대한 세 가지 기본 접근 방식

대부분의 기업에서 최신 동향을 바탕으로 LLM을 구축하는 접근 방식을 세 가지 기본 시나리오로 요약할 수 있습니다. 최근 에 설명된 대로 ""하버드 비즈니스 리뷰"" 기사: (1) LLM을 처음부터 완전히 교육(사전 교육) - 비용이 많이 들고

전문 AI/ML 기술이 필요함, (2) 복잡하고 실현 가능한 엔터프라이즈 데이터로 기반 모델 미세 조정, (3) 검색 증강 생성(RAG)을 사용하여 회사 데이터가 포함된 문서 저장소, API 및 벡터 데이터베이스를 쿼리합니다. 이러한 각 요소에는 서로 다른 유형의 문제를 해결하는 데 사용되는 노력, 반복 속도, 비용 효율성 및 모델 정확성이 서로 상충됩니다(아래 다이어그램).

그림 3: 문제 유형



기초 모델

기본 모델이라고도 하는 기초 모델(FM)은 광범위한 다운스트림 NLP 작업에 맞게 일반적으로 조정되는 방대한 양의 레이블 없는 데이터에 대해 훈련된 대규모 AI 모델(LLM)입니다. 훈련 데이터는 인간이 라벨링하지 않기 때문에 모델이 명시적으로 인코딩되는 것이 아니라 등장한다. 즉, 모델은 명시적으로 프로그래밍되지 않고 자신의 이야기 또는 설명을 생성할 수 있습니다. 따라서 FM의 중요한 특징은 여러 도메인에서 동일한 방법이 사용된다는 것을 의미하는 균질화입니다. 그러나 개인화 및 미세 조정 기술을 통해 오늘날 등장하는 제품에 통합된 FMS는 텍스트, 텍스트-이미지 및 텍스트-코드 생성 기능뿐만 아니라 도메인별 작업 또는 디버깅 코드를 설명하는 데도 유용합니다. 예를 들어, OpenAI의 Codex 또는 Meta의 Code Llama와 같은 FMS는 프로그래밍 작업의 자연어 설명을 기반으로 여러 프로그래밍 언어로 코드를 생성할 수 있습니다. 이러한 모델은 파이썬, C#, 자바스크립트, 펄, 루비, 및 SQL. 이들은 사용자의 의도를 이해하고 소프트웨어 개발, 코드 최적화 및 프로그래밍 작업의 자동화에 유용한 원하는 작업을 수행하는 특정 코드를 생성합니다.

미세 조정, 영역 특이성 및 재교육

데이터 준비 및 데이터 사전 처리 이후 LLM 구축의 일반적인 사례 중 하나는 크고 다양한 데이터 세트에 대해 교육을 받은 사전 훈련된 모델을 선택하는 것입니다. 세부 조정이라는 맥락에서, 과 같은 오픈 소스 대형 언어 모델이 될 수 있습니다 "메타의 라마 2" 700억 개의 매개 변수와 2조 개의 토큰에 대한 교육을 받았습니다. 사전 학습된 모델을 선택한 후 다음 단계는 도메인별 데이터에 맞게 세부 조정하는 것입니다. 이를 위해서는 모델의 매개 변수를 조정하고 새로운 데이터에 대해 훈련하여 특정 도메인 및 작업에 적응해야 합니다. 예를 들어, 금융 업계를 지원하는 광범위한 금융 데이터에 대한 교육을 받은 독점 LLM인 BloombergGPT가 있습니다. 특정 작업을 위해 설계 및 훈련된 도메인별 모델은 일반적으로 범위 내에서 정확성과 성능이 높지만 다른 작업 또는 도메인 간 전송 가능성은 낮습니다. 일정 기간 동안 비즈니스 환경과 데이터가 변경될 경우 테스트 중 FM의 예측 정확도가 성능에 비해 떨어지기 시작할 수 있습니다. 이 경우 모델을 재훈련하거나 미세 조정하는 것이 중요합니다. 기존 AI/ML에서 모델 재훈련은 배포된 ML 모델을 새 데이터로 업데이트하는 것을 의미하며, 일반적으로 두 가지 유형의 드리프트를 없애기 위해 수행됩니다. (1) 컨셉 드리프트 – 입력 변수와 목표 변수 사이의 링크가 시간에 따라 변경되면, 우리가 변화를 예측하고자 하는 것에 대한 설명 이후 모델은 부정확한 예측을 생성할 수 있습니다. (2) 데이터 드리프트 – 시간이 지남에 따라 고객 습관 또는 행동의 변화와 같이 입력 데이터의 특성이 변화하여 모델이 이러한 변화에 대응하지 못하는 경우에 발생합니다. 마찬가지로 재교육은 FMS/LLM에도 적용되지만 비용이 많이 들기 때문에(수백만 달러) 대부분의 조직이 고려할 만한 것은 아닙니다. 현재 활발한 연구 중에 있으며, LLMOps 영역에서 여전히 나타나고 있습니다. 따라서 미세 조정된 FMS에서 모델이 붕괴될 경우 재교육을 받는 대신 기업은 새로운 데이터 세트를 사용하여 다시 미세 조정을 선택할 수 있습니다 (훨씬 저렴함). 비용 측면에서 아래에 나열된 것은 Azure-OpenAI Services의 모델 가격 표의 예입니다. 고객은 각 작업 범주에 대해 특정 데이터 세트에서 모델을 미세 조정하고 평가할 수 있습니다.

출처: Microsoft Azure

Model	Per 1000 token
Text-Ada	$0.0001
GPT-3.5 Turbo	$0.003
GPT-4	$0.06
Text-Davinci	$0.02
Model	**Per 100 images**
Dall-E	$2

신속한 엔지니어링 및 추론

신속한 엔지니어링은 모델 가중치를 업데이트하지 않고 원하는 작업을 수행하기 위해 LLM과 통신하는 효과적인 방법을 의미합니다. AI 모델 훈련 및 미세 조정이 NLP 애플리케이션에 중요합니다. 하지만 추론도 마찬가지로 중요합니다. 훈련된 모델이 사용자 프롬프트에 응답합니다. 추론을 위한 시스템 요구사항은 일반적으로 수십억 개의 저장된 모델 매개 변수를 적용하여 최상의 응답을 이끌어낼 수 있어야 하기 때문에 LLM에서 GPU에 데이터를 제공하는 AI 스토리지 시스템의 읽기 성능에 훨씬 더 큰 영향을 줍니다.

LLMOps, 모델 모니터링 및 벡터스토어

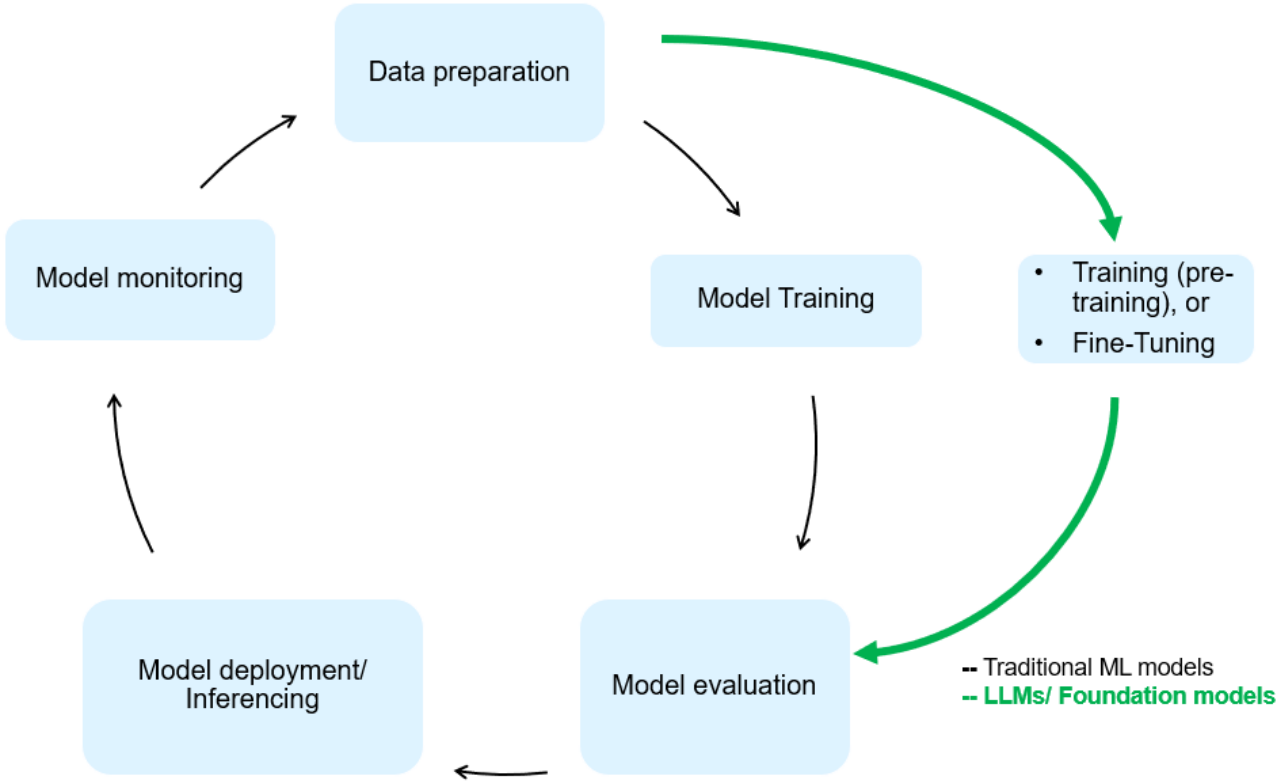
기존의 MLOps(Machine Learning Ops)와 마찬가지로 LLMOps(Large Language Model Operations)도 데이터 과학자와 DevOps 엔지니어의 협업이 필요하며, 생산 환경에서 LLM 관리를 위한 도구와 모범 사례가 필요합니다. 그러나 LLM에 대한 워크플로와 기술 스택은 어떤 면에서 다를 수 있습니다. 예를 들어, LangChain 문자열과 같은 프레임워크를 사용하여 구축된 LLM 파이프라인은 벡터스토어 또는 벡터 데이터베이스와 같은 외부 임베디드 엔드포인트에 대한 여러 LLM API 호출을 함께 통합합니다. 벡터 데이터베이스와 같은 다운스트림 커넥터에 임베드된 끝점 및 벡터스토어를 사용하는 것은 데이터를 저장하고 액세스하는 방식에 있어 상당한 발전을 나타냅니다. 처음부터 개발된 기존의 ML 모델과 달리 LLM은 전송 학습에 의존하는 경우가 많습니다. 이러한 모델은 보다 구체적인 영역에서 성능을 향상시키기 위해 새로운 데이터로 미세 조정된 FMS로 시작되기 때문입니다. 따라서 LLMOps는 위험 관리 및 모델 붕괴 모니터링 기능을 제공하는 것이 중요합니다.

발생 **AI** 시대의 위험과 윤리

“ChatGPT – 그것은 매끈하지만 여전히 무의미한.” – MIT 기술 리뷰. 가비지 입력 – 가비지 유출은 항상 컴퓨팅 측면에서 어려운 문제였습니다. Generative AI의 유일한 차이점은 쓰레기를 매우 신뢰할 수 있게 만들어 부정확한 결과를 도출하는 데 탁월하다는 것입니다. LLM은 자신이 만든 이야기에 맞게 사실을 발명한 경향이 있습니다. 따라서 세대 AI를 AI 등가물로 비용을 낮출 수 있는 좋은 기회로 간주하는 기업은 시스템을 정직하고 윤리적으로 유지하기 위해 심층적인 추측을 효율적으로 탐지하고 편견을 줄이며 위험을 낮춰야 합니다. 엔드 투 엔드 암호화 및 AI 가드레일을 통한 데이터 이동성, 데이터 품질, 데이터 거버넌스 및 데이터 보호를 지원하는 강력한 AI 인프라를 통해 유입되는 데이터 파이프라인은 책임지고 설명 가능한 생성 AI 모델의 설계에 포함되어 있습니다.

고객 시나리오 및 **NetApp**

그림 3: 기계 학습/대규모 언어 모델 워크플로



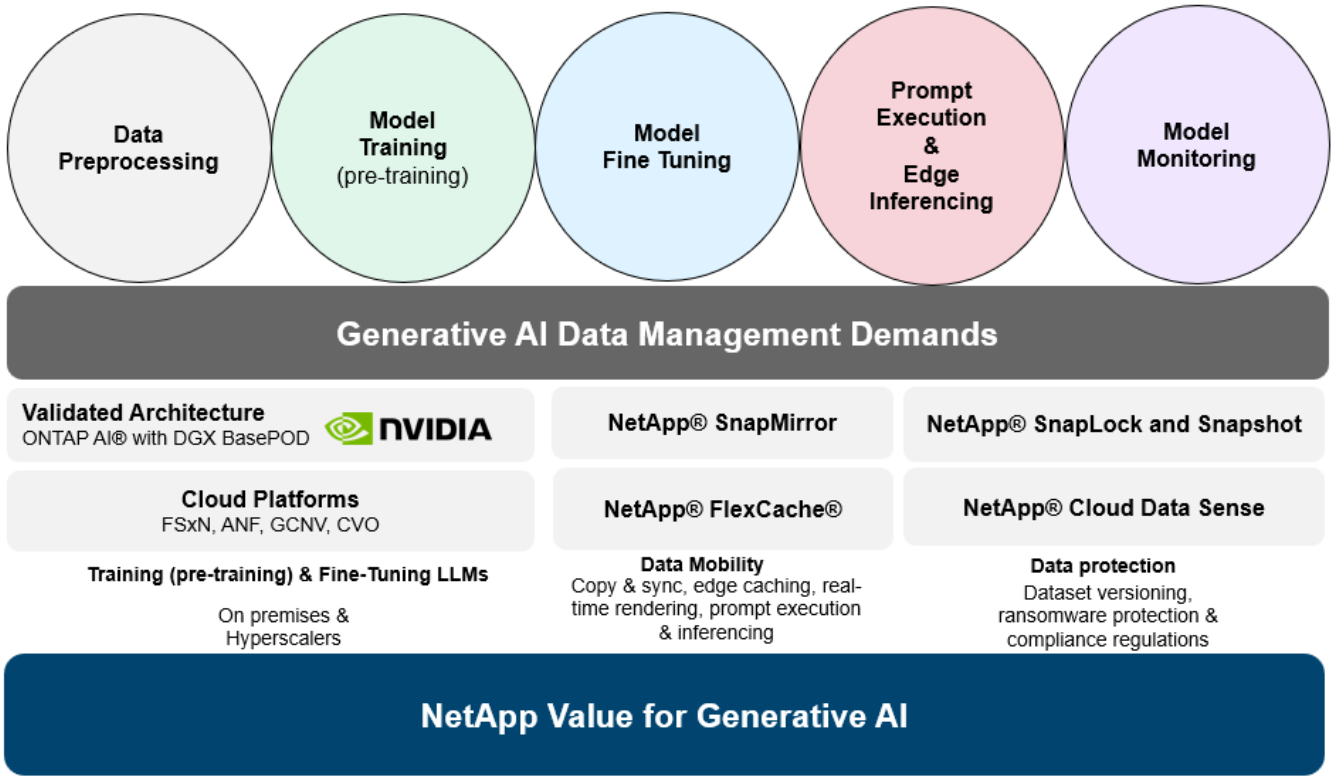
- 우리는 교육 또는 미세 조정 중입니까? * (a) LLM 모델을 처음부터 교육하거나, 사전 훈련된 FM을 미세 조정하거나, RAG를 사용하여 기초 모델 외부의 문서 저장소에서 데이터를 검색하고, 메시지를 보강할 수 있는지 여부 (b) 오픈 소스 LLM(예: Llama 2) 또는 독점 FMS(예: ChatGPT, Bard, AWS Bedrock)를 활용하는 것은 조직의 전략적 결정입니다. 각 접근 방식에는 비용 효율성, 데이터 부담, 운영, 모델 정확도 및 LLM 관리 간의 절충이 있습니다.

기업으로서 NetApp은 업무 문화와 제품 설계 및 엔지니어링 활동에 대한 접근 방식에 내부적으로 AI를 수용합니다. 예를 들어, NetApp의 자율적 랜섬웨어 방어는 AI와 머신 러닝을 사용하여 구축됩니다. 파일 시스템 이상 징후를 조기에 감지하여 운영에 영향을 미치기 전에 위협을 식별하는 데 도움이 됩니다. 둘째, NetApp는 판매 및 재고 예측, 챗봇과 같은 비즈니스 운영에 예측 AI를 사용하여 콜센터 제품 지원 서비스, 기술 사양, 보증, 서비스 매뉴얼 등과 같은 고객을 지원합니다. 셋째, NetApp은 수요 예측, 의료 영상, 감정 분석, 심리 분석, 능동적 AI 솔루션과 같은 예측 AI 솔루션을 구축하는 고객에게 제공하는 제품 및 솔루션을 통해 AI 데이터 파이프라인 및 ML/LLM 워크플로에 고객 가치를 제공합니다. Gans와 같은 차세대 AI 솔루션은 NetApp ® ONTAP AI ®, NetApp ® SnapMirror ®, NetApp ® FlexCache ® 와 같은 NetApp 제품을 사용하여 제조 부문의 이상 징후 탐지와 금융 및 금융 서비스의 자금 세탁 방지 및 사기 범죄를 탐지합니다.

NetApp 역량

챗봇, 코드 생성, 이미지 생성 또는 게놈 모델 표현과 같은 생성 AI 애플리케이션에서 데이터의 이동과 관리는 에지, 프라이빗 데이터 센터 및 하이브리드 멀티 클라우드 에코시스템에 걸쳐 있을 수 있습니다. 예를 들어, ChatGPT와 같은 사전 훈련된 모델의 API를 통해 노출된 최종 사용자 앱에서 승객의 항공권을 비즈니스 클래스로 업그레이드하는 데 도움을 주는 실시간 AI 봇은 인터넷에서 승객 정보를 공개하지 않기 때문에 그 자체로 해당 작업을 수행할 수 없습니다. API는 하이브리드 또는 멀티 클라우드 에코시스템에 존재할 수 있는 항공사의 승객의 개인 정보 및 티켓 정보에 액세스해야 합니다. LLM을 사용하여 일대다 바이오 의료 연구 기관과 관련된 약물 발견 시 임상 시험을 수행하는 최종 사용자 애플리케이션을 통해 약물 분자 및 환자 데이터를 공유하는 과학자에게도 이와 유사한 시나리오가 적용될 수 있습니다. FMS 또는 LLM에 전달되는 중요한 데이터에는 PII, 재무 정보, 건강 정보, 생체 데이터, 위치 데이터, 통신 데이터, 온라인 행동 및 법률 정보 실시간 렌더링, 프롬프트 실행 및 에지 추론의 경우, 오픈 소스 또는 독점 LLM 모델을 통해 최종 사용자 앱에서 스토리지 엔드포인트로 데이터가 이동하고 사내 또는 퍼블릭 클라우드 플랫폼의 데이터 센터로 이동합니다. 이 모든 시나리오에서 데이터 이동성과 데이터 보호는 대규모 훈련 데이터 세트와 이러한 데이터의 이동에 의존하는 LLM과 관련된 AI 운영에 매우 중요합니다.

그림 4: Generative AI-LLM Data Pipeline



지능형 데이터 관리 소프트웨어를 기반으로 하는 NetApp의 스토리지 인프라, 데이터 및 클라우드 서비스 포트폴리오입니다.

- Data Preparation *: LLM 기술 스택의 첫 번째 기둥은 기존의 ML 스택에서 거의 영향을 받지 않습니다. 훈련 또는 세부 조정 전에 AI 파이프라인의 데이터 사전 처리가 데이터를 정규화하고 정리해야 합니다. 이 단계에는 Amazon S3 계층 형태의 모든 위치나 파일 저장소 또는 NetApp StorageGRID와 같은 오브젝트 저장소와 같은 온프레미스 스토리지 시스템에서 데이터를 수집하는 커넥터가 포함됩니다.
- NetApp ® ONTAP * 는 데이터 센터와 클라우드에서 NetApp의 중요 스토리지 솔루션을 뒷받침하는 기초 기술입니다. ONTAP에는 사이버 공격에 대한 자동 랜섬웨어 보호, 내장 데이터 전송 기능, 사내, 하이브리드, NAS의 멀티 클라우드, SAN, 오브젝트, 등 다양한 아키텍처에 대한 스토리지 효율성 기능을 포함한 다양한 데이터 관리 및 보호 기능이 포함되어 있습니다. 소프트웨어 정의 스토리지(SDS)가 필요합니다.
- NetApp ® ONTAP AI ® *: 딥 러닝 모델 훈련. NetApp ® ONTAP ® 는 ONTAP 스토리지 클러스터와 NVIDIA DGX 컴퓨팅 노드를 통해 NetApp 고객을 위해 RDMA 기반 NFS를 사용하여 NVIDIA GPU Direct Storage ™ 를 지원합니다. 또한 스토리지에서 메모리로 소스 데이터 세트를 여러 번 읽고 처리할 수 있는 비용 효율적인 성능을 제공하므로 인텔리전스를 강화할 수 있어 조직이 LLM에 대한 교육, 미세 조정 및 확장 액세스를 수행할 수 있습니다.
- NetApp ® FlexCache ® * 는 파일 배포를 간소화하고 읽기 빈도가 높은 데이터만 캐시하는 원격 캐싱 기능입니다. 이 기능은 LLM 교육, 재교육 및 미세 조정에 유용하며 실시간 렌더링 및 LLM 추론과 같은 비즈니스 요구사항에 따라 고객에게 가치를 제공합니다.
- NetApp ® SnapMirror * 는 두 ONTAP 시스템 간에 볼륨 스냅샷을 복제하는 ONTAP 기능입니다. 이 기능은 에지의 데이터를 사내 데이터 센터 또는 클라우드로 최적으로 전송합니다. SnapMirror를 사용하여 온프레미스와 하이퍼스케일 클라우드 간에 데이터를 안전하고 효율적으로 이동할 수 있습니다. 고객이 엔터프라이즈 데이터가 포함된 RAG로 클라우드에서 생성 가능한 AI를 개발하려는 경우 SnapMirror를 사용할 수 있습니다. 변경 사항만 효율적으로 전송하여 대역폭을 절약하고 복제 속도를 높임으로써 FMS 또는 LLM의 훈련, 재교육 및 미세 조정 작업 중에 필수 데이터 이동성 기능을 제공합니다.
- NetApp ® SnapLock * 는 데이터 세트 버전 관리를 위해 ONTAP 기반 스토리지 시스템에서 변경 불가능한 디스크

기능을 제공합니다. 마이크로코어 아키텍처는 FPolicy ™ 제로 트러스트 엔진을 사용하여 고객 데이터를 보호하도록 설계되었습니다. NetApp는 공격자가 특히 리소스를 많이 사용하는 방식으로 LLM과 상호 작용할 때 DoS(Denial-of-Service) 공격을 차단하여 고객 데이터를 사용할 수 있도록 보장합니다.

- NetApp ® Cloud Data Sense * 는 엔터프라이즈 데이터 세트에 있는 개인 정보를 식별, 매핑 및 분류하고, 정책을 수립하고, 온프레미스 또는 클라우드의 개인 정보 보호 요구사항을 충족하고, 보안 태세를 개선하고, 규정을 준수하는 데 도움이 됩니다.

Cloud Data Sense 기반 * NetApp ® BlueXP ™ * 분류 고객은 데이터 자산 전체에서 데이터를 자동으로 스캔, 분석, 분류, 조치하고, 보안 위험을 감지하고, 스토리지를 최적화하고, 클라우드 구축을 가속화할 수 있습니다. 통합 제어 플레인을 통해 스토리지와 데이터 서비스가 결합되어 고객은 GPU 인스턴스를 계산에 사용하고 하이브리드 멀티 클라우드 환경을 사용하여 콜드 스토리지 계층화와 아카이브 및 백업을 수행할 수 있습니다.

- NetApp 파일 - 객체 이중성 *. NetApp ONTAP는 NFS 및 S3에 대한 이중 프로토콜 액세스를 지원합니다. 이 솔루션을 통해 고객은 NetApp Cloud Volumes ONTAP의 S3 버킷을 통해 Amazon AWS SageMaker 노트북의 NFS 데이터에 액세스할 수 있습니다. 따라서 NFS와 S3 모두에서 데이터를 공유할 수 있어야 하는 이기종 데이터 소스에 쉽게 액세스해야 하는 고객에게 유연성이 제공됩니다. 예를 들어, 파일 객체 버킷에 대한 액세스를 통해 SageMaker에서 Meta의 Llama 2 텍스트 생성 모델과 같은 FMS를 미세 조정합니다.
- NetApp ® Cloud Sync * 서비스는 데이터를 클라우드 또는 온프레미스의 모든 대상으로 마이그레이션하는 간단하고 안전한 방법을 제공합니다. Cloud Sync은 사내 또는 클라우드 스토리지, NAS 및 오브젝트 저장소 간에 데이터를 원활하게 전송 및 동기화합니다.
- NetApp XCP * 는 NetApp 환경 간 및 NetApp 환경 간 데이터 마이그레이션을 빠르고 안정적으로 지원하는 클라이언트 소프트웨어입니다. 또한 XCP는 대용량 데이터를 Hadoop HDFS 파일 시스템에서 ONTAP NFS, S3 또는 StorageGRID 및 XCP 파일 분석으로 효율적으로 이동할 수 있는 기능을 제공하여 파일 시스템에 대한 가시성을 제공합니다.
- NetApp ® DataOps Toolkit * 은 데이터 과학자, DevOps 및 데이터 엔지니어가 고성능 스케일 아웃 NetApp 스토리지를 통해 지원되는 데이터 볼륨 또는 JupyterLab 작업 공간의 즉각적인 프로비저닝, 복제, 스냅샷 생성 등의 다양한 데이터 관리 작업을 간편하게 수행할 수 있는 Python 라이브러리입니다.
- NetApp 제품 보안 *. LLM은 실수로 기밀 데이터를 응답에 노출시킬 수 있기 때문에 LLM을 활용하는 AI 응용 프로그램과 관련된 취약점을 연구하는 CISO에게 우려를 안겨 줍니다. OWASP(Open Worldwide Application Security Project)에서 설명한 바와 같이 데이터 손상, 데이터 유출, 서비스 거부 및 LLM 내 즉각적인 주입과 같은 보안 문제는 무단 액세스 서비스 공격자에 대한 데이터 노출로부터 기업에 영향을 미칠 수 있습니다. 데이터 스토리지 요구 사항에는 정형, 반정형 및 비정형 데이터에 대한 무결성 검사 및 변경 불가능한 스냅샷이 포함되어야 합니다. NetApp 스냅샷과 SnapLock가 데이터 세트 버전 관리에 사용됩니다. RBAC(역할 기반 액세스 제어)와 보안 프로토콜, 사용되지 않는 데이터와 전송 중인 데이터를 모두 보호하기 위한 업계 표준 암호화 기능을 제공합니다. Cloud Insights와 Cloud Data Sense는 함께 제공 기능을 통해 위협의 출처를 포렌적으로 식별하고 복원할 데이터의 우선순위를 지정할 수 있습니다.

*** ONTAP AI 및 DGX BasePOD ***

NVIDIA DGX BasePOD가 포함된 NetApp ® ONTAP ® AI 참조 아키텍처는 머신 러닝(ML) 및 인공 지능(AI) 워크로드를 위한 확장 가능한 아키텍처입니다. 일반적으로 LLM의 중요 교육 단계에서는 데이터 스토리지에서 교육 클러스터로 데이터가 정기적으로 복사됩니다. 이 단계에 사용되는 서버는 GPU를 사용해 컴퓨팅을 병렬화하여 방대한 양의 데이터를 수용합니다. 물리적 I/O 대역폭 요구사항을 충족하는 것은 높은 GPU 활용률을 유지하는 데 매우 중요합니다.

*** ONTAP AI 및 NVIDIA AI Enterprise ***

NVIDIA AI Enterprise는 NVIDIA 인증 시스템과 함께 VMware vSphere에서 실행하도록 NVIDIA에서 최적화, 인증 및 지원하는 엔드 투 엔드 클라우드 네이티브 AI 및 데이터 분석 소프트웨어 제품군입니다. 이 소프트웨어를 사용하면 최신 하이브리드 클라우드 환경에서 AI 워크로드를 쉽고 빠르게 구축, 관리, 확장할 수 있습니다. NetApp 및 VMware를 기반으로 하는 NVIDIA AI Enterprise는 단순하고 친숙한 패키지로 엔터프라이즈급 AI 워크로드 및 데이터 관리를

제공합니다.

*** 1P** 클라우드 플랫폼 *****

완전 관리형 클라우드 스토리지 오퍼링은 Microsoft Azure As ANF(Azure NetApp Files), AWS FSxN(Amazon FSx for NetApp ONTAP), Google GNCV(Google Cloud NetApp Volumes)로 기본 제공됩니다. 1P는 고객이 퍼블릭 클라우드의 향상된 데이터 보안으로 고가용성 AI 워크로드를 실행하고 AWS SageMaker, Azure-OpenAI Services, Google의 Vertex AI와 같은 클라우드 네이티브 ML 플랫폼으로 LLM/FMS를 미세 조정할 수 있도록 지원하는 고성능 파일 관리 시스템입니다.

NetApp 파트너 솔루션 제품군

NetApp은 핵심 데이터 제품, 기술 및 기능 외에도 강력한 AI 파트너 네트워크와 긴밀하게 협력하여 고객에게 부가 가치를 제공합니다.

- AI 시스템의 NVIDIA Guardrails * 는 AI 기술의 윤리적이고 책임 있는 사용을 보장하기 위한 보호 조치의 역할을 합니다. AI 개발자는 특정 주제에 대한 LLM 기반 애플리케이션의 동작을 정의하고 원치 않는 주제에 대한 토론에 참여하지 못하도록 선택할 수 있습니다. 오픈 소스 툴킷인 Guardrail은 LLM을 다른 서비스에 원활하고 안전하게 연결할 수 있는 기능을 제공하여 신뢰할 수 있고 안전하며 안전한 LLM 대화 시스템을 구축할 수 있습니다.
- Domino Data Lab * 은 AI 전환 과정에서 언제 어디서나 Generative AI를 빠르고 안전하며 경제적으로 구축 및 생산할 수 있는 다양한 엔터프라이즈급 도구를 제공합니다. Domino의 Enterprise MLOps Platform을 사용하면 데이터 과학자가 선호하는 도구와 모든 데이터를 사용하고, 어디에서든 모델을 쉽게 교육 및 배포하고, 위험 및 비용 효율적으로 관리할 수 있습니다. 이 모든 것이 하나의 제어 센터에서 가능합니다.
- Edge AI용 Modzy *. NetApp ® 과 Modzy는 파트너십을 통해 이미지, 오디오, 텍스트, 표를 비롯한 모든 유형의 데이터에 적합한 AI를 제공합니다. Modzy는 AI 모델을 배포, 통합 및 실행하기 위한 MLOps 플랫폼으로, 데이터 과학자에게 원활한 LLM 추론을 위한 통합 솔루션을 통해 모델 모니터링, 드리프트 감지 및 설명 기능을 제공합니다.
- Run:AI * 와 NetApp은 AI 워크로드 오케스트레이션을 간소화하는 Run:AI 클러스터 관리 플랫폼을 통해 NetApp ONTAP AI 솔루션의 고유한 기능을 입증하기 위해 파트너십을 체결했습니다. Spark, Ray, Dask 및 RAPIDS용 통합 프레임워크를 통해 데이터 처리 파이프라인을 수백 개의 머신으로 확장하도록 설계된 GPU 리소스를 자동으로 분할하고 결합합니다.

결론

Generative AI는 모델이 고품질 데이터에 대해 훈련될 때만 효과적인 결과를 생성할 수 있습니다. LLM은 놀라운 이정표를 달성했지만 데이터 이동성과 데이터 품질과 관련된 한계, 설계 과제 및 위험을 인식하는 것이 중요합니다. LLM은 이질적인 데이터 소스의 대규모의 이질적인 훈련 데이터 세트를 사용합니다. 모델에 의해 생성된 부정확한 결과 또는 편향된 결과는 기업과 소비자 모두에게 위기의 원인이 될 수 있습니다. 이러한 위험은 데이터 품질, 데이터 보안 및 데이터 이동성과 관련된 데이터 관리 문제로 인해 발생할 수 있는 LLM의 제약과 일치할 수 있습니다. NetApp를 사용하는 조직은 빠른 데이터 성장, 데이터 이동성, 멀티 클라우드 관리 및 AI 채택으로 인해 발생하는 복잡성을 충족할 수 있습니다. 대규모 AI 인프라와 효율적인 데이터 관리는 생성 AI와 같은 AI 애플리케이션의 성공을 정의하는 데 매우 중요합니다. 고객이 비용 효율성, 데이터 거버넌스 및 윤리적인 AI 관행을 제어하면서 엔터프라이즈에 필요한 확장 기능을 그대로 유지하면서 모든 구축 시나리오를 다룰 수 있어야 합니다. NetApp은 고객이 AI 구축을 단순화하고 가속할 수 있도록 돕기 위해 지속적으로 노력하고 있습니다.

TR-4785: NetApp E-Series 및 BeeGFS를 통해 AI 구축

Nagalakshmi Raju, Daniel Landes, Nathan Swartz, amine Bennani, NetApp

인공 지능(AI), 머신 러닝(ML) 및 딥 러닝(DL) 애플리케이션에는 대규모 데이터 세트 및 고계산이 포함됩니다. 이러한 워크로드를 성공적으로 실행하려면 스토리지 및 컴퓨팅 노드를 원활하게

확장할 수 있는 민첩한 인프라가 필요합니다. 이 보고서에는 컴퓨팅 및 스토리지 노드를 원활하게 확장할 수 있는 AI 교육 모델을 분산 모드에서 실행하는 단계가 포함됩니다. 또한, 이 보고서에는 NetApp E-Series 스토리지를 BeeGFS 병렬 파일 시스템과 결합하여 AI 워크로드를 위한 유연하고 비용 효율적이며 단순한 솔루션을 제공하는 방법을 보여주는 다양한 성능 메트릭이 포함되어 있습니다.

"TR-4785: NetApp E-Series 및 BeeGFS를 통해 AI 구축"

NVA-1150-design:Quantum StorNext with NetApp E-Series 시스템 설계 가이드

NetApp, Ryan Rodine

이 문서에서는 NetApp E-Series 스토리지 시스템을 사용하여 StorNext 병렬 파일 시스템 솔루션을 설계하는 방법에 대해 자세히 설명합니다. 이 솔루션은 NetApp EF280 All-Flash 어레이, NetApp EF300 All-Flash NVMe 어레이, EF600 All-Flash NVMe 어레이, NetApp E57760 하이브리드 시스템에 적용됩니다. 미디어 및 엔터테인먼트 업계에서 테스트에 널리 사용되는 도구인 프레임 벤치마킹(Frametest Benchmarking)을 기반으로 성능 특성을 제공합니다.

"NVA-1150-design:Quantum StorNext with NetApp E-Series 시스템 설계 가이드"

NVA-1150-Deploy:Quantum StorNext with NetApp E-Series 시스템 구축 가이드

NetApp, Ryan Rodine

이 문서에서는 NetApp E-Series 스토리지 시스템을 통해 StorNext 병렬 파일 시스템 솔루션을 구축하는 방법에 대해 자세히 설명합니다. 이 솔루션은 NetApp EF280 All-Flash 어레이, NetApp EF300 All-Flash NVMe 어레이, NetApp EF600 All-Flash NVMe 어레이, NetApp E57760 하이브리드 시스템에 적용됩니다. 미디어 및 엔터테인먼트 업계에서 테스트에 널리 사용되는 도구인 프레임 벤치마킹(Frametest Benchmarking)을 기반으로 성능 특성을 제공합니다.

"NVA-1150-Deploy:Quantum StorNext with NetApp E-Series 시스템 구축 가이드"