Part 1: mr-1-sample2.pdf

수신일: 2023년 2월 20일, 승인일: 2023년 3월 16일, 출판일: 2023년 3월 27일, 현재 버전 날짜: 2023년 4월 3일.
디지털 객체 식별자: 10.1109/ACCESS.2023.3262138

기계 학습 운영(MLOps): 개요, 정의 및 아키텍처
도미닉 크로이츠베르거1, 니클라스 퀼1,2, 세바스찬 허슈1
1IBM, 독일 에닝겐 71139
2바이에로이트 대학교 인간 중심 인공지능 및 정보 시스템 95447, 바이에로이트, 독일
교신 저자: 니클라스 퀼 (kuehl@uni-bayreuth.de)

이 연구는 독일 연구재단(DFG)과 바이에로이트 대학교의 출판 펀드의 지원을 받았습니다.

요약:
산업 기계 학습(ML) 프로젝트의 궁극적 목표는 ML 제품을 개발해 빠르게 생산 단계로 진입시키는 것입니다. 그러나 ML 제품을 자동화하고 운영하는 것은 매우 도전적이며, 이로 인해 많은 ML 프로젝트가 기대에 미치지 못하고 실패합니다. 기계 학습 운영(MLOps) 패러다임은 이러한 문제를 해결합니다. MLOps는 모범 사례, 개념의 집합, 개발 문화를 포함합니다. 그러나 MLOps는 여전히 모호한 용어로 연구자와 전문가에게 그 영향이 불분명합니다. 이 격차를 해결하기 위해 문헌 리뷰, 도구 검토, 전문가 인터뷰를 포함한 혼합 연구 방법을 수행했습니다. 이를 통해 필요한 원칙, 구성 요소, 역할, 아키텍처 및 워크플로우에 대한 종합적인 개요를 제공하고, MLOps의 포괄적인 정의를 제시하며, 이 분야의 해결되지 않은 과제를 강조합니다. 또한 ML 연구자와 실무자들이 ML 제품을 자동화하고 운영할 수 있도록 특정 기술 세트를 안내합니다.

핵심 용어: CI/CD, DevOps, 기계 학습, MLOps, 운영, 워크플로우 오케스트레이션.

I. 서론

기계 학습(ML)은 데이터의 잠재력을 활용하여 기업의 혁신[1], 효율성[2], 지속 가능성[3]을 가능하게 하는 중요한 기술이 되었습니다. 그러나 실제 환경에서 많은 ML 응용 프로그램의 성공은 기대에 미치지 못하고[4], 많은 ML 프로젝트는 프로덕션 단계에 도달하지 못한 채 실패합니다[5]. 연구 관점에서는, ML 커뮤니티가 ML 모델 구축에 집중해왔지만, (a) 프로덕션 준비된 ML 제품 구축과 (b) 현실 세계 환경에서 ML 시스템의 구성이 필요한 역할과 인프라 조정에 집중하지 않았기 때문입니다[6]. 예를 들어, 많은 산업 애플리케이션에서 데이터 과학자들은 여전히 수작업으로 ML 워크플로를 관리하여 운영 중 많은 문제를 야기합니다[7]. 이를 해결하기 위해, 이 연구는 수동 ML 프로세스를 자동화하고 운영하여 더 많은 ML 개념 증명이 프로덕션으로 이관되도록 하는 방법을 조사합니다. 우리는 MLOps라는 새롭게 떠오르는 ML 엔지니어링 실무를 탐구하고, 생산적인 ML 설계 및 유지 관리 문제를 정확히 다룹니다. 통합된 관점에서 관련 구성 요소, 원칙, 역할 및 아키텍처를 이해합니다. 기존 연구가 MLOps의 다양한 특정 측면을 조명하지만, ML 시스템 설계에 대한 전체적 개념화, 일반화 및 명확화는 아직 부족합니다. "MLOps"라는 용어에 대한 다양한 관점과 개념은 오해와 오류를 초래할 수 있습니다. 따라서 우리는 다음 연구 질문을 던집니다: RQ: MLOps란 무엇인가?

Part 2: mr-2-sample2.pdf

D. Kreuzberger 외 "머신러닝 운영(MLOps): 개요, 정의 및 아키텍처"에서 이 질문에 답하기 위해, 우리는 혼합 방법 연구를 수행했습니다. (a) MLOps의 중요한 원칙을 식별하고, (b) 기능적 핵심 구성 요소를 조각하고, (c) MLOps를 성공적으로 구현하기 위해 필요한 역할을 강조하며, (d) ML 시스템 설계를 위한 일반적인 아키텍처를 도출합니다. 이러한 통찰력들은 MLOps의 정의로 이어지며, 이 용어와 관련된 개념에 대한 공통의 이해에 기여합니다. 따라서 명확한 지침을 제공함으로써 학문적 및 실무적 논의에 긍정적인 영향을 미치기를 바랍니다. 이러한 통찰력은 시스템 설계에서 오류를 줄이고 더 많은 개념 증명을 실제 운영으로 전환하며, 궁극적으로 실제 환경에서 더욱 견고한 예측을 가능하게 할 수 있습니다. 이 논문의 나머지 부분은 다음과 같이 구성되어 있습니다. 먼저 필요한 기초와 관련 작업을 설명하고 그 다음으로 문헌 검토, 도구 검토 및 인터뷰 연구로 구성된 방법론의 개요를 제공합니다. 그런 다음 이 방법론의 적용에서 도출된 통찰력을 제시하고 이를 통합하여 정의를 제공하며, 논문을 간단한 요약, 한계 및 전망으로 마무리합니다.

DEVOPS의 기초

과거에는 다양한 소프트웨어 프로세스 모델과 개발 방법론이 소프트웨어 엔지니어링 분야에 등장했습니다. 대표적인 예로는 폭포수 모델과 애자일 선언문이 있습니다. 이러한 방법론은 생산 준비가 된 소프트웨어 제품을 제공하는 것을 목표로 합니다. 2008/2009년에 등장한 "DevOps"는 소프트웨어 개발의 문제를 줄이는 것을 목표로 합니다. DevOps는 단순한 방법론 이상이며, 소프트웨어 개발에 참여하는 조직의 사회적, 기술적 문제를 해결하는 패러다임을 나타냅니다. DevOps는 개발과 운영 간의 격차를 해소하고 협력, 커뮤니케이션, 지식 공유를 강조합니다. DevOps는 지속적인 테스트, 품질 보증, 지속적인 모니터링, 로깅 및 피드백 루프를 보장하도록 설계되었습니다. DevOps의 상업화로 인해 Slack, Trello, GitLab wiki 등의 협업 및 지식 공유, GitHub, GitLab과 같은 소스 코드 관리, Jenkins, GitLab CI 등의 지속적 통합 도구 등 여러 가지 그룹으로 구분되는 많은 DevOps 도구가 등장하고 있습니다. 클라우드 환경은 DevOps 도구를 이용할 수 있는 상태로 점점 준비되고 있으며, 이는 가치 창출을 효율적으로 촉진합니다. 이러한 DevOps로의 새 패러다임 전환으로 개발자는 자신이 개발하는 것뿐만 아니라 운영에 대해서도 신경 써야 합니다. 경험적 결과는 DevOps가 소프트웨어 품질을 향상함을 보여 줍니다. 산업계와 학계의 사람들은 DevOps를 사용한 소프트웨어 엔지니어링에서 많은 경험을 얻었으며, 이제 이 경험은 머신러닝을 자동화하고 운영하는 데 사용되고 있습니다.

방법론

학술 지식 기반에서 통찰력을 얻고 업계 전문가의 전문 지식을 활용하기 위해 혼합 방법 접근 방식을 사용합니다. 첫 단계로, 관련 연구를 개관하기 위해 체계적인 문헌 검토를 진행합니다. 또한, MLOps 분야의 관련 도구 지원을 검토하여 기술적 구성 요소를 더 잘 이해합니다. 마지막으로 전화면접을 통해 다양한 도메인의 전문가들과의 반구조화된 인터뷰를 진행합니다. 이를 바탕으로 "MLOps" 용어를 개념화하고, 문헌 및 인터뷰를 종합하여 결과를 도출합니다.

문헌 검토

과학적 지식에 기반한 결과 확보를 위해 체계적인 문헌 검토를 Webster와 Watson, Kitchenham 등의 방법에 따라 진행합니다. 초기 탐색 이후 다음 검색 쿼리를 정의했습니다: (((‘‘DevOps’’ OR ‘‘CICD’’ OR ‘‘Continuous Integration’’ OR ‘‘Continuous Delivery’’ OR ‘‘Continuous Deployment’’) AND ‘‘Machine Learning’’) OR ‘‘MLOps’’ OR ‘‘CD4ML’’). Google Scholar, Web of Science, ScienceDirect, Scopus 및 Information Systems eLibrary의 데이터베이스를 검색했습니다. DevOps의 ML 사용, MLOps, ML과의 연속적 관행은 학문 문헌에서 비교적 새로운 분야입니다. 따라서 동료 검토 연구가 적습니다. 그러나 이 분야에서 경험을 얻기 위해 동료 검토되지 않은 문헌도 포함했습니다. 검색은 2021년 5월에 수행되었고 1,864편의 논문을 검색했습니다. 그중 194편의 논문을 자세히 검토했습니다.

Part 3: mr-3-sample2.pdf

D. Kreuzberger 외: 머신러닝 운영(MLOps): 개요, 정의 및 아키텍처
표 1. 평가된 기술 목록.
우리는 포함 및 제외 기준(예: MLOps 또는 DevOps라는 용어와 CI/CD가 ML과 함께 자세히 설명된 경우, 논문이 영어로 작성된 경우 등)에 따라 27개의 논문을 선정했습니다. 이 27개의 논문은 모두 동료 검토를 받았습니다.
B. 도구 검토
27개의 논문과 8번의 인터뷰를 통해 다양한 오픈 소스 도구, 프레임워크 및 상용 클라우드 ML 서비스가 확인되었습니다. 이러한 도구, 프레임워크 및 ML 서비스를 검토하여 2023년 11권 31868페이지에 대한 이해를 얻었습니다.

Part 4: mr-4-sample2.pdf

D. Kreuzberger 외: 머신러닝 운영(MLOps): 개요, 정의 및 아키텍처

인터뷰 파트너 목록을 기술 구성 요소와 함께 Table 2에 제시합니다. 연구 질문에 대한 실무적 통찰을 얻기 위해 Myers와 Newman의 방법에 따라 반구조화된 전문가 인터뷰를 진행했습니다. 전문가 인터뷰 연구 설계의 중요한 측면 중 하나는 적절한 표본 크기를 선택하는 것입니다. 이론적 표본 추출 방식을 적용하여, 고품질의 데이터를 확보하기 위해 경험 많은 인터뷰 파트너를 선택합니다. LinkedIn을 통해 MLOps 지식이 풍부한 글로벌 전문가를 식별하였습니다. 다양한 관점을 얻기 위해, 다른 조직, 산업, 국가, 국적 및 성별의 인터뷰 파트너를 선택했습니다. 새로운 카테고리와 개념이 더 이상 나타나지 않을 때까지 인터뷰를 진행하여, 총 8명의 전문가와 인터뷰를 진행했습니다. 인터뷰는 2021년 6월부터 8월 사이에 진행되었습니다. 인터뷰 디자인은 몇 가지 질문을 포함한 반구조화된 가이드로 준비했고, 인터뷰 스크립트로 문서화했습니다. 인터뷰 동안 "어떻게"와 "왜"라는 질문으로 심층 탐구하는 "소프트 래더링" 기법을 사용했습니다. 이 방법 덕분에 추가적인 통찰을 얻을 수 있었습니다. 모든 인터뷰는 녹음 후 전사되었습니다. 전사된 인터뷰는 개방 코딩 방식을 사용하여, 데이터를 분석적으로 분해하여 개념적으로 유사한 주제를 카테고리와 하위 카테고리로 그룹화했습니다. 이 카테고리들은 "코드"라고 부릅니다. 개념은 여러 인터뷰에서 반복적으로 나타날 때 식별되었습니다.

결과적으로, 중요한 원칙과 이 원칙이 구성 요소로 구체화된 결과, 필요한 역할, 아키텍처 및 워크플로우 제안을 제시합니다. 마지막으로 MLOps의 개념화와 정의를 도출합니다.

Part 5: mr-5-sample2.pdf

D.Kreuzberger 외: 머신러닝 운영(MLOps): 개요, 정의 및 아키텍처

그림 2. 기술 구성 요소 내 원칙의 구현

A. 원칙

원칙은 MLOps에서 기본 가이드라인으로, '모범 사례'와 밀접한 관련이 있습니다. MLOps 구현에 필요한 아홉 가지 원칙을 다음과 같이 정리했습니다.

P1 CI/CD 자동화: 연속 통합 및 배포를 통해 개발자의 생산성을 높입니다.

P2 워크플로 오케스트레이션: DAG를 기반으로 ML 워크플로의 작업을 조정합니다.

P3 재현성: 실험을 반복하여 동일한 결과를 얻을 수 있습니다.

P4 버전 관리: 데이터, 모델, 코드의 버전 관리를 통해 추적 가능성을 제공합니다.

P5 협업: 데이터, 모델, 코드 상에서 협업을 지원하며, 커뮤니케이션 강화에 중점을 둡니다.

P6 지속적 ML 학습 및 평가: 모델 품질 변화를 평가하며 재학습 주기를 결정합니다.

P7 ML 메타데이터 추적/로깅: 실험의 전체 추적 가능성을 확보합니다.

P8 지속적 모니터링: 데이터, 모델, 코드, 인프라 성능을 주기적으로 평가하여 품질 변화를 감시합니다.

P9 피드백 루프: 품질 평가 단계의 통찰을 개발 프로세스에 통합합니다.

B. 기술 구성 요소

원칙을 MLOps에 통합하기 위해 다음의 기술 구성 요소를 구현합니다.

C1 CI/CD 구성 요소 (P1, P6, P9): 빠른 피드백을 제공하여 생산성을 높입니다. Jenkins, GitHub Actions 등이 사용됩니다.

C2 소스 코드 저장소 (P4, P5): 다수의 개발자가 훈련, 추론, 애플리케이션 코드를 저장소에서 버전 관리할 수 있도록 합니다.

2023년 11권, 31870쪽.

Part 6: mr-6-sample2.pdf

D. 크로이츠베르거 외: 머신러닝 운영(MLOps): 개요, 정의, 그리고 아키텍처

코드 예시로는 Bitbucket, GitLab, GitHub, Gitea 등이 있습니다. C3 워크플로 오케스트레이션 컴포넌트는 DAGs를 통해 ML 워크플로의 작업을 조율합니다. 이러한 그래프는 워크플로의 각 단계의 실행 순서와 아티팩트 사용을 나타냅니다. 워크플로는 예를 들어, 데이터 추출, 모델 훈련, 추론 등과 같은 프로세스 단계에서 패키지 코드를 사용합니다. Apache Airflow, Kubeflow Pipelines, Watson Studio Pipelines, Luigi 등이 예시입니다. CI/CD 도구는 특정 작업을 순차적으로 트리거하는 데 사용할 수 있지만, 데이터 엔지니어링 및 ML 파이프라인 작업의 복잡성 증대로 인해 워크플로나 작업 조율에 특화된 도구가 필요합니다. 이러한 도구는 복잡한 작업 체인을 관리하기 쉽게 해줍니다.

C4 피처 스토어 시스템은 일반적으로 사용되는 피처의 중앙 저장을 보장합니다. 오프라인 및 온라인 피처 스토어로 구성되어 있습니다. Google Feast, AWS Feature Store, Tecton.ai 등이 예시입니다. 대부분의 데이터는 ML 모델 훈련에 사용됩니다. 확장성은 일반적으로 클라우드 인프라를 통해 실현됩니다.

C5 모델 훈련 인프라는 CPU, RAM, GPU 등의 컴퓨팅 자원을 제공합니다. 분산 또는 비분산 인프라를 제공하며, 대개 확장 가능한 분산 인프라가 권장됩니다. 로컬 머신 또는 클라우드 컴퓨팅, 분산 계산 등이 예로 들 수 있습니다.

C6 모델 레지스트리는 훈련된 ML 모델과 메타데이터를 중앙에 저장합니다. 주요 기능으로는 ML 아티팩트 및 메타데이터 저장이 있으며, MLflow, AWS SageMaker Model Registry 등이 있습니다.

C7 ML 메타데이터 스토어는 각 ML 워크플로 파이프라인 작업의 다양한 메타데이터를 기록합니다. Kubeflow Pipelines, AWS SageMaker Pipelines 등이 예시입니다.

C8 모델 서빙 컴포넌트는 온라인 추론이나 대량의 입력 데이터를 사용하는 배치 추론 등 다양한 목적으로 구성할 수 있습니다. 예를 들어, Kubernetes와 Docker를 사용해 모델을 컨테이너화하고 Flask와 같은 웹 프레임워크를 활용할 수 있습니다. 실제 모델 배포는 실시간, 배치, 또는 서버리스 추론으로 분류됩니다.

Part 7: mr-7-sample2.pdf

C9 모니터링 구성 요소(P8, P9). 모니터링 구성 요소는 모델 서비스 성능(Prediction 정확도 등)을 지속적으로 모니터링합니다. 또한, ML 인프라, CI/CD 및 오케스트레이션의 모니터링이 필요합니다. [7],[23],[24],[28],[32],[33],[50],[α,ζ,η,θ]. 예로 Prometheus와 Grafana[η,ζ], ELK 스택(Elasticsearch, Logstash, Kibana)[α,η,ζ], TensorBoard[θ]가 있습니다. 내장된 모니터링 기능이 포함된 예로 Kubeflow[θ], MLflow[η], AWS SageMaker 모델 모니터나 Cloudwatch[ζ]가 있습니다.

C. 역할

이제 우리는 MLOps를 구현하기 위해 필요한 역할을 살펴보겠습니다. MLOps는 여러 분야의 협업이 필요한 절차이며, 다양한 역할의 협력이 중요합니다. 각 역할과 그 목적, 관련 작업은 다음과 같습니다:

R1 사업 이해관계자: ML을 통해 성취할 사업 목표를 정의하고, 투자 수익(ROI)을 제시하는 역할을 합니다. [7],[24],[45],[α,β,δ,θ].

R2 솔루션 아키텍트: 아키텍처를 설계하고 사용할 기술을 정의하는 역할입니다. [7],[24],[α,ζ].

R3 데이터 과학자: 사업 문제를 ML 문제로 변환하고, 최적의 알고리즘과 하이퍼파라미터를 선택하는 작업을 담당합니다. [7],[25],[32],[33],[α,β,γ,δ,ε,ζ,η,θ].

R4 데이터 엔지니어: 데이터 및 기능 엔지니어링 파이프라인을 구축 및 관리하며, 데이터베이스로 적절히 데이터를 수집합니다. [25],[26],[32],[α,β,γ,δ,ε,ζ,η,θ].

R5 소프트웨어 엔지니어: 소프트웨어 설계 패턴, 코딩 지침, 모범 사례를 활용해 ML 문제를 잘 설계된 제품으로 변환합니다. [32],[α,γ].

R6 데브옵스 엔지니어: 개발과 운영 간의 간극을 메우고, CI/CD 자동화, ML 워크플로우 오케스트레이션, 모델 배포를 담당합니다. [7],[22],[25],[51],[α,β,γ,ε,ζ,η,θ].

R7 ML 엔지니어/MLOps 엔지니어: 여러 역할의 측면을 결합하여 교차 분야 지식을 갖고 있는 역할입니다. [7],[24],[25],[32],[α,β,γ,δ,ε,ζ,η,θ].

V. 아키텍처 및 워크플로우

우리는 MLOps 연구자와 실무자들에게 적절한 가이드를 제공하기 위해 일반화된 MLOps 엔드 투 엔드 아키텍처를 도출했습니다. 이 아티팩트는 기술 중립적으로 설계되어, 연구자와 실무자가 자신에게 가장 적합한 기술과 프레임워크를 선택할 수 있습니다. "최고의" 오픈 소스 도구나 기업 솔루션을 혼합하여 MLOps를 실현할 수 있습니다. 특히 빠르게 성장하는 오픈 소스 툴 시장의 최신 개발 상황을 고려하는 것이 중요합니다. API 인터페이스 연결 및 조합 시 몇 가지 제약 사항이 있을 수 있으므로 주의가 필요합니다. 우리가 새로운 애플리케이션과 도구를 통해 가능한 조합을 예시로 보여드리겠습니다.

위 그림에서 MLOps 제품 시작 단계부터 모델 서비스까지의 엔드 투 엔드 프로세스를 설명하고 있습니다. A: MLOps 제품 시작; B: 데이터 수집 및 특징 스토어; C: 실험; D: 자동화된 ML 워크플로우 및 모델 서비스.

Part 8: mr-8-sample2.pdf

D. Kreuzberger 외: 머신러닝 운영(MLOps): 개요, 정의, 및 아키텍처

그림 4. 기능 구성 요소와 역할을 포함한 MLOps의 종단 간 아키텍처 및 워크플로우.

(A) MLOps 제품 시작.
(1) 비즈니스 이해관계자(R1)는 비즈니스를 분석하고 ML을 통해 해결할 수 있는 잠재적인 비즈니스 문제를 식별합니다.
(2) 솔루션 아키텍트(R2)는 전체 ML 시스템의 아키텍처 설계를 정의하고 철저한 평가 후 사용할 기술을 결정합니다.
(3) 데이터 과학자(R3)는 비즈니스 목표로부터 회귀 또는 분류가 사용될지와 같은 ML 문제를 도출합니다.
(4) 데이터 엔지니어(R4)와 데이터 과학자(R3)는 문제 해결에 필요한 데이터를 이해하기 위해 협력합니다.
(5) 답변이 명확해지면, 데이터 엔지니어(R4)와 데이터 과학자(R3)는 초기 데이터 분석을 위한 원천 데이터를 찾기 위해 협력합니다. 그들은 데이터의 분포와 품질을 확인하고 검증 절차를 수행합니다. 또한, 데이터 소스로부터 들어오는 데이터가 레이블링되어 목표 속성이 있는지 확인합니다.

Part 9: mr-9-sample2.pdf

D. Kreuzberger 외: 머신러닝 운영(MLOps): 개요, 정의 및 아키텍처

이 예에서 데이터 소스는 이미 레이블이 있는 데이터를 사용합니다. 이는 감독 학습에 필수적입니다.

(B1) 피처 엔지니어링 파이프라인 요구사항: 모델 훈련에 필요한 관련 속성인 피처에 대한 요구사항이 설정됩니다.

(6) 데이터 엔지니어는 데이터를 유용한 형식으로 변환하기 위해 정규화, 집계, 정리 규칙을 정의합니다.

(7) 데이터 과학자와 데이터 엔지니어는 다른 피처에 기반하여 새로운 피처를 계산하는 규칙을 협력하여 정의합니다. 이러한 규칙은 모델 성능 모니터링의 피드백을 기반으로 반복적으로 조정됩니다.

(B2) 피처 엔지니어링 파이프라인: 데이터 엔지니어와 소프트웨어 엔지니어는 초기 요구사항을 바탕으로 파이프라인 프로토타입을 구축합니다. 초기 요구사항은 피드백에 따라 업데이트됩니다. 데이터 엔지니어는 CI/CD 및 작업 조정 컴포넌트를 정의합니다.

(8) 파이프라인은 스트리밍 데이터, 정적 배치 데이터 또는 클라우드 저장소와 연결됩니다.

(9) 데이터 소스에서 데이터를 추출합니다.

(10) 데이터를 유용한 형식으로 변환하고 정리합니다. 규칙은 피드백에 따라 지속적으로 개선됩니다.

(11) 피처 엔지니어링 작업은 다른 피처를 기반으로 새로운 피처를 계산합니다. 규칙은 피드백에 따라 개선됩니다.

(12) 데이터는 피처 저장 시스템으로 로드됩니다.

(C) 실험: 데이터 과학자가 주도하고 소프트웨어 엔지니어가 지원합니다.

(13) 데이터 과학자는 피처 저장 시스템에 연결하여 데이터를 분석합니다. 필요 시 피드백 루프를 통해 변경점을 보고합니다.

(14) 데이터 준비 및 검증 작업은 훈련 및 테스트 데이터를 생성합니다.

(15) 가장 성능이 좋은 알고리즘과 하이퍼파라미터를 추정하고, 훈련 데이터를 사용한 훈련이 시작됩니다.

(16) 다양한 모델 매개변수 테스트 후 성능 지표가 좋으면 최적의 매개변수를 식별합니다. 이는 모델 엔지니어링이라고 할 수 있습니다.

(17) 데이터 과학자는 모델을 내보내고 코드 저장소에 커밋합니다. DevOps 엔지니어나 ML 엔지니어는 자동화된 ML 워크플로우 파이프라인을 정의하고 커밋합니다. CI/CD 컴포넌트는 업데이트된 코드를 감지하고 자동으로 빌드, 테스트 및 배포를 수행합니다.

주피터와 같은 노트북 기반 솔루션이 실험 단계에서 자주 사용됩니다.

Part 10: mr-10-sample2.pdf

다음은 자연어 처리(NLP) 분야에서 노트북 기반 환경을 활용한 사례입니다. 감정 분석, 텍스트 요약, 명명된 엔터티 인식 등의 NLP 서비스를 제공하는 회사는 대량의 텍스트 데이터에 대한 머신러닝 실험을 Jupyter 노트북에서 수행할 수 있습니다. 데이터 과학자들은 데이터를 준비하고, 다양한 머신러닝 모델(e.g., 딥러닝 모델)을 학습, 평가, 최적화하며 결과를 테스트합니다. MLflow나 Neptune.AI 같은 솔루션을 사용해 메타데이터를 추적하고 결과 모델을 저장합니다.

자동화된 ML 워크플로 파이프라인의 관리에는 DevOps 엔지니어와 ML 엔지니어가 관여하며, 런타임 환경, 하드웨어 리소스 및 Kubernetes와 같은 프레임워크 관리를 맡습니다. 워크플로 오케스트레이션 컴포넌트는 작업들을 관리하며, 각 태스크는 이미지 레지스트리에서 아티팩트를 가져와 독립된 환경에서 실행됩니다. 이러한 과정에서 작업의 메타데이터가 수집됩니다.

자동화된 파이프라인에는 다음의 작업들이 포함됩니다:
(18) 피처 스토어 시스템에서 버전 관리된 피처를 자동으로 가져오기, (19) 데이터 준비 및 검증 자동화, (20) 새로운 데이터에 대한 최종 모델 학습 자동화, (21) 모델 평가 및 하이퍼 파라미터 조정, 그리고 성능이 만족스러우면 학습이 중단됩니다.
(22) 훈련된 모델은 모델 레지스트리에 저장됩니다. ML 메타데이터 스토어는 모델 학습 작업의 메타데이터를 기록합니다. 여기에는 모델의 계보(lineage)도 포함됩니다.

모델이 프로덕션 준비 상태가 되면 DevOps 또는 ML 엔지니어에게 전달됩니다. 이후 지속적 배포(CI/CD) 파이프라인이 시작되어 ML 모델과 코드가 프로덕션 환경에 설치됩니다.

모델 서빙 컴포넌트는 새로운 데이터에 대한 예측을 수행하며, 실시간 또는 배치 추론을 지원합니다. A/B 테스트는 서로 다른 모델을 비교하는 데 유용합니다. 예를 들어, 호텔 예약 취소 예측 시 두 가지 모델을 비교할 수 있습니다. ML 엔지니어는 서빙 인프라를 관리하며, 모니터링 컴포넌트의 성능을 실시간으로 관찰합니다. 성능이 기준 이하일 경우 피드백 루프를 통해 빠르게 정보를 전송하고, 지속적인 학습 및 개선이 이루어집니다. 피드백은 실험 단계에서 모델 개선에 활용됩니다. 개념 변화 감지와 같은 메커니즘은 실시간 애플리케이션에서 지속 학습을 가능하게 합니다.

Part 11: mr-11-sample2.pdf

D. Kreuzberger 외: 기계 학습 운영(MLOps): 개요, 정의 및 아키텍처

개념 드리프트는 특정 알고리즘을 통해 감지될 수 있습니다. 모델 모니터링 컴포넌트가 데이터의 드리프트를 감지하면 이 정보가 스케줄러에게 전달되어 자동화된 ML 워크플로 파이프라인을 재훈련하도록 트리거됩니다. 배포된 모델의 적합성 변화는 분포 비교를 통해 감지됩니다. 재훈련은 통계적 임계값 도달 시 자동으로, 또는 새로운 피처 데이터가 있을 때, 또는 주기적으로 스케줄링됨으로써 트리거될 수 있습니다. 이를 지원하는 기술로는 Apache Airflow, Kubeflow Pipelines, IBM Watson Studio Pipelines, SageMaker Pipelines 등이 있습니다. 예를 들어, 온라인 광고 분야에서는 Airflow를 사용하여 광고 타겟팅과 최적화를 위한 머신러닝 모델의 학습과 배포 과정을 자동화합니다. 이 파이프라인은 다양한 소스에서 대량의 데이터를 추출, 변환 및 로드하는 것으로 시작합니다. 그런 다음, 데이터 전처리와 피처 엔지니어링을 거쳐 여러 머신러닝 모델이 이 데이터로 학습되고 평가됩니다. 최상의 모델이 선택되어 실시간 광고 타겟팅 결정을 내리기 위해 프로덕션 환경에 배포됩니다. 이 모든 과정은 Airflow를 통해 자동화되어, 스케줄링, 모니터링, 실패한 작업 재실행까지 포함됩니다.

MLOps는 머신러닝, 소프트웨어 엔지니어링, DevOps, 그리고 데이터 엔지니어링의 교차점에 위치합니다. MLOps란 엔드 투 엔드 개념화, 구현, 모니터링, 배포 및 확장성을 포함하는 패러다임으로, 머신러닝 제품의 생산화를 위해 액면이끔 개발(Dev)과 운영(Ops)의 격차를 연결하는 것을 목표로 합니다. 중요하게도, CI/CD 자동화, 워크플로 관리, 재현성, 데이터, 모델, 코드 버전 관리, 협업, 지속적 ML 학습 및 평가, ML 메타데이터 추적 및 로깅, 지속적 모니터링, 피드백 루프를 활용합니다.

MLOps 채택을 위한 여러 도전 과제가 확인되었습니다. 조직적, ML 시스템, 운영적 도전 과제로 나뉩니다. 조직적 도전 과제로는 데이터 과학의 사고방식과 문화 전환이 필요하며, 이와 관련하여 데이터 중심 AI 트렌드는 데이터 중심의 접근을 강조합니다. 모델 생성뿐 아니라 ML 제품 구축에 필요한 기술과 컴포넌트를 학습해야 하는 필요성을 강조합니다. 데이터 과학자만으로는 MLOps 목표를 달성할 수 없으며, 다학제적 팀이 필요합니다. 이는 팀이 협력보다는 개별적으로 작업하기 때문에 종종 방해를 받습니다. 더 나은 협업을 위해서는 결정권자의 노력이 필요합니다.

Part 12: mr-12-sample2.pdf

D. Kreuzberger 외: 기계 학습 운영(MLOps): 개요, 정의 및 아키텍처

MLOps의 성숙도 증가와 제품 중심의 사고 방식이 명확한 비즈니스 개선을 가져올 것이라는 확신이 필요합니다.

주요 과제 중 하나는 수요 변동에 대한 설계로, 특히 ML 학습 과정에서 나타납니다. 이는 잠재적으로 방대한 데이터와 변동으로 인해 필요한 인프라 자원(CPU, RAM, GPU)을 정확하게 추정하기 어렵고 인프라의 확장성에 높은 유연성이 필요합니다.

제품 환경에서 ML을 수동으로 운영하기란 다양한 소프트웨어 및 하드웨어 스택과 그 상호작용 때문에 복잡하여 자동화가 필수적입니다. 또한 지속적인 데이터 증가로 인해 재학습 능력이 필요하므로 반복적 작업이 요구됩니다. 이러한 작업은 많은 아티팩트를 생성하고 강력한 관리와 데이터, 모델, 코드의 버전 관리가 필요합니다.

결론적으로 데이터 가용성과 분석 능력이 증가함에 따라 혁신의 압력이 커지고 더욱 많은 기계 학습 제품이 개발되고 있지만, 극소수만이 실제 배포 및 운영으로 이어집니다. 학계는 모델 개발과 벤치마킹에 집중했지만, 복잡한 기계 학습 시스템 운영에는 소홀했습니다. 현실에서는 데이터 과학자들이 여전히 많은 ML 작업을 수동으로 관리하고 있습니다.

MLOps 패러다임이 이러한 문제를 해결합니다. 우리는 문헌 및 도구를 분석하고 8명의 전문가와 인터뷰하여 MLOps의 네 가지 주요 측면: 원칙, 구성 요소, 역할 및 아키텍처를 밝혀 홀리스틱한 정의를 도출했습니다. 이 연구는 MLOps와 그 개념에 대한 공통 이해를 지원하여 연구자와 전문가들이 성공적인 ML 제품을 구축하는 데 도움을 줄 것입니다.

Part 14: mr-14-sample2.pdf

도미닉 크로이츠베르거는 독일 바덴뷔르템베르크 협력 주립 대학에서 경영 정보 시스템 학사 학위를, 칼스루에 공과대학교에서 정보 시스템 공학 및 관리 석사 학위를 받았으며, 디지털 서비스와 기계 학습 운영(MLOps)에 중점을 두었습니다. 현재 그는 IT 아키텍트로서 하이브리드 클라우드 컴퓨팅과 인공지능 솔루션을 전문으로 하고 있습니다. IBM에서 고객 성공에 중점을 두고 기업용 데이터 및 기계 학습 제품을 설계 및 구축하고 있습니다. IBM에 합류하기 전, 그는 글로벌 스포츠 회사인 아디다스에서 거의 10년간 근무하며 전자상거래 및 데이터 분석 분야에서 다양한 직책을 맡았습니다.

니클라스 쾰은 정보 시스템 박사 학위를 최우수 성적으로 받았으며, 응용 기계 학습에 중점을 두고 연구하고 있습니다. 그는 인공지능 제품의 개념화, 설계 및 구현을 진행하며, 조직 간 학습 및 인간-인공지능 팀 간의 공정하고 효과적인 협력에 중점을 둡니다. 현재 바이로이트 대학교의 정보 시스템 및 인간 중심 AI 전임 교수이자 프라운호퍼 FIT의 비즈니스 분석 그룹 리더이며 IBM의 AI 분야 선임 전문가입니다. 과거 IBM에서 데이터 과학 관련 관리 컨설턴트로 근무하며 이론적 지식을 실무적 통찰과 결합하였습니다. 그는 2014년부터 다양한 도메인에서 기계 학습(ML)과 AI에 관해 연구하고 있으며, 텍사스 대학교와 MIT–IBM 왓슨 AI 연구소 등 여러 기관과 국제적으로 협력해 왔습니다.

세바스찬 히르슐은 IBM에서 선임 엔지니어/아키텍트로 근무하며 독일의 기계 학습 분야 활동을 이끌고 있습니다. 그는 기계 학습과 인공지능에 대한 컴퓨터 과학 배경을 바탕으로 IBM 내에서 지난 5년간 기계 학습 공학 분야를 발전시켰으며, 모범 사례, 방법론, 역할, 도구 등을 포함합니다. 독일 및 유럽의 클라이언트를 위한 기업용 데이터 및 기계 학습 제품의 설계 및 구현을 주도합니다. 팀과 함께 IBM 데이터 사이언스 모범 사례를 출판하며, IBM 데이터 및 AI 참조 아키텍처를 설계합니다. MLOps 패러다임의 발전을 내부 및 외부적으로 추진합니다.