MLOps - Deploying ML Model with Minikube

Mallikharjuna Rao S, L00157129

Msc in AI and BDA , ATU Donegal

Abstract:
Majority of AI and ML research projects never reach the end of the development life cycle, because of failures throughout the estimating and analysis phase, team cooperation, data drift, and model drift, . MLOPS bring much-needed rigor to the ML model development process. The importance of MLOPS in a machine learning development project is explained in this study, as well as existing MLOPS techniques. MLOPS's future road map is also briefly highlighted. As part of hands-on practice, a wine quality prediction ML model is constructed and deployed using the MLOPS tools Flask, Dockers, and minikube.

Index Terms: MLOPS,Dockers, API development, Flask, Minikube, cURL .

1. Introduction

With the introduction of advanced AI algorithms and computational resources to process and analyze big data, organizations are leaning toward implementing machine learning (ML) models to process their data, putting them on the verge of operationalization to address issues like model monitoring, inefficient deployment, insufficient ML governance, insufficient security measures, and the inability to scale the model, data, and resources. MLOps (Machine Learning Model Operationalization Management) is a reproducible, testable, and evolvable machine learning development method for designing, producing, and managing ML-powered applications [1]. These techniques and concepts combine development and IT operations with the purpose of streamlining the systems development life cycle and ensuring continuous delivery of high-quality software.

Because data and business objectives are always changing, model training must be updated, and AI governance must be handled. Risk assessment is also a big consideration when adopting an AI model in a corporate setting. These motives prompted the use of MLOPS methods in a development environment, as MLOPS prioritizes business demands and establishes quantitative benchmarks for ML development. Development assignments resemble academic research projects without these guidance activities [3].

The entire MLOPS is divided into three sections: Designing ML-based applications, Development, and Operations. All of these areas are linked and influence one another. With continuous deployment, versioning, testing, and monitoring, this is an iterative and incremental model.

To match the outcomes with quickly changing business objectives, an AI-based system may require data-based, model-based, or code-based adjustments over time. The following are the principles that make up strong MLOPS practices:

- Automation of the work flow steps with minimal or no human interaction
- Continuous integration, deployment, testing and monitoring
- Versioning the ML models and data sets to track back

- Versioning the ML models and data sets to trackback
- Test for features, data, models, development and infrastructure
- Monitoring the products and performances and metrics

MLOPS also assists in tracking and selecting the best algorithm when numerous models are used to solve a problem, and it deploys the best performing model.

2. ML Pipeline

A machine learning pipeline is a way of regulating and automating the procedures required in building a machine learning model. Data extraction, pre-processing, model training, and deployment are all handled by machine learning pipelines. These stages can be carried out manually or by automation. These steps are iterative, meaning that each one is repeated to improve the model's accuracy and reach the end goal [2].

The steps involved in an ML pipeline are :

- Data Extraction
- Data Analysis
- Data Preparation
- Model Training
- Model Evaluation
- Model Validation
- Serving and Monitoring

The AI and ML project begins when one of the company partners notices a problem that can be handled using AI and ML models. The technical architect, in collaboration with data scientists and data engineers, creates the system's blueprint. They then collect data from all sources and prepare it for analysis.

The data is labeled during the data analysis phase, and all other data operations such as normalization, aggregations, and outliers are performed on the data to prepare it for training the model in a usable manner. In this stage, the model's features are defined, and they will be improved in a continual process. This step includes the processes of eliminating superfluous characteristics using ”principal component analysis. [8]”

Experimentation also refers to model training, model evaluation, and model validation. Using the information saved from the previous steps, data scientists and developers can create numerous models by modifying the hyper parameters. The code is stored in the repositories after CI/CD is implemented. The highest performing model will be chosen based on model engineering, assessment measures, and accuracy scores [5].

The pipeline's final phase is the maintenance phase, which is a critical step. The model's underpinning architecture must be maintained and updated to meet ever-changing business needs. The received feedback will be analyzed, and the models and pipelines will be adjusted accordingly.

3. Research to date on the topic ‘MLOps’ and what the future holds

3.1. Current Research Developments

Over the previous few decades, the software development process has developed quickly. From waterfall and agile approaches to MLOPS, more ideas and proof-of-concepts being materialized and deployed in production contexts.

MLOPS research intersects with a number of related software development and deployment fields, including Data Science, Database engineering, data engineer, Dev Ops engineer, and AI/ML developer. Because the goal of MLOPS is to establish coordination and communication among all of these teams, all of the critical and cross-cutting parts from each field are gathered and a set of best practices and methodologies developed [4].

Multiple tools have been developed by the MLOPS research and development community to

automate the MLOPS pipeline in order to reduce manual interactions and hence reduce errors and hazards. There are six different groups of tools that have been developed and widely used in the current industry.

The tables below show all of the categories as well as the most popular tools within each category.

Collaborate and Knowledge sharing	Source Code Management	Build Process	Continuous Integration
Slack Trello GitLab Wiki	GitHub GitLab	Maven	Jenkins

table cont..

Deployment Automation	Monitoring and Logging
kubernetes Dockers	Prometheus Logstash

Many cloud service providers, such as Google, Amazon, and others, provide infrastructure that includes MLOPS tools that are ready to use. Small and medium-sized businesses can save money and time by establishing their own MLOPS pipelines using this development.

3.2. MLOPS future road map

MLOPS' success is due to the involvement of numerous teams throughout the process. As a result, there will be a big culture shift in the AI business in the next years, from model- and data-driven development to product-driven development. MLOPS will make the process of developing AI products faster, more efficient, and more adaptable in an incremental manner, rather than relying just on research [10].

MLOPS will become a specialized expertise in the future days as it is established as a mandatory process across the industry as more than a best practice to follow, and data scientists, AI and ML developers, and researchers will be needed to have this skill as a must-know skill.

In terms of operations, there is still a lot of space for automation in areas like software and hardware maintenance and usage, data and artifact storage and processing, versioning, and reproducibility. They are improving and becoming more dependable every day, and they will continue to improve given the rate of progress.

4. Description of the Machine Learning Pipeline implemented

The constructed ML model uses simple linear regression to predict wine quality. There are 11 different characteristics in the data, such as fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, density, and so on. The quality of the red wine can be predicted using these values. There are no outliers or missing values in the data because it has been processed. There are no category values, and all of the data is in float-number format, making calculations simple.

To implement linear regression, a python file named "train.py" is constructed. The data file "winequality-red.csv" is read, with the labels X(features) and y(output) highlighted. Following that, 80% of the data was split for training and 20% for testing. The data is subjected to Linear-Regression from sklearn.linear model, and the model is stored as "WineQuality.pkl." The pkl file is produced with the Python module pickle, which allows objects to be serialized to files on the desk and then deserialized back into the original programs during execution.

To publish the model to the web app, a new file called "main.py" is produced. The input values are taken in and delivered on to the previously created pkl file using flask's POST method. The

web application prints the output, which is the predicted wine quality for the specified parameters.

All of the files are kept in a folder called ”app” on the cloud machine’s unix system. Outside of the app folder, a folder called ”Dockerfile” is generated with all of the instructions needed to construct the image. The pickle file is made by compiling the model file ”train.py” during the construction. To execute on the isolated host machine, the image is containerized. The local host URL is used to access the flask web app.

The API is tested with cURL, with the values passed as a json object. A POST request is used to send all of the argument values to the API. The dockerhub repository is updated with the functioning image. Minikube is utilized to automate the scalability of work loads and container management, and a pod, a small Kubernetes unit, is produced for the winequality prediction app.

5. Analysis of the technologies used

5.1. Dockers

Dockers is an open platform for creating and distributing apps. Dockers helps to break down apps from their infrastructure. Docker Hub is a public docker repository that enables to distribute applications over the cloud. With the automation of development workflows, applications can be deployed to production environments more quickly. Dockers’ enormous success is due to its ability to execute containers without any language or hardware dependencies. Container applications run each service in its own container with its own dependencies.

Containers are self-contained(isolated) environments with their own processes, services, and networks. They’re all connected by a kernal. Docker runs on Ubuntu, Fedora, Susi, and CentOS, to name a few. Dockers’ goal is to package and containerize applications so that they can be run anywhere, at any time, and as many times as needed.

In terms of utilization, size, and boot up, Dockers outperform standard VM machines. Containers are frequently supplied on virtual Docker hosts.

Docker images, which are made up of app.war and Dcokerfile (file with set of instructions and requirements), can be thought of as package template blueprints. The containers created from the pictures are isolated operating instances of those images with their own environments and processes.

The processes inside a container are alive as long as they are alive. Containers, unlike virtual machines, are not designed to host operating systems. The container exists once the application in it terminates.

5.2. Flask

Flask is a Python framework for building web applications for RESTful APIs. Flask is a lightweight and micro-framework that is simple to get started with for web development. No SQL databases such as MangoDB and DynamoDB can be integrated and queried with Flask.

Despite the existence of another famous micro framework, Danjo, which is also based on Python, flask is popular due to its speed, ease of usage, and language flexibility. Flask also comes with a lot of documentation and supports secure cookies [7].

The GET and POST decorators in Flask are used to handle http requests. A message is sent via the GET method, and the server returns the data. The HTTP form is sent to the server using the POST method, and the data is not cached. Because the request data is not accessible, the POST technique is more secure.

5.3. cURL

Client URL (cURL) is a command line utility that allows to transmit data from one server to another. This tool can be used to test data transfer for APIs. Because of its mobility and compatibility with all devices and operating systems, cURL is widely used. It’s good at logging issues and providing additional information about data transfers, which aids troubleshooting [6].

The four basic pieces of an API request are as follows:

- Endpoint : The URL to which the address needs to sent
- HTTP method: Mode of sending the message(GET, POST, PUT, DELETE)
- Headers : Metadata about the request
- Body :The data which needs to be transferred

POSTMAN is one of the most extensively used API testing tools, and it works effectively. POSTMAN is also a graphical user interface (GUI) utility.

5.4. Minikube

Minikube is a light-weight variant of Kubernetes that is also open source. This allows to run Kubernetes on the host system as a single node. Minikube provides the majority of Kubernetes capabilities, such as dashboards, DNS, and NodePorts, while using fewer resources.

Container orchestration is made easier with Minikube (or Kubernetes on a bigger scale). They reduce the amount of effort required to run containerized workloads and services by automating the process. These frameworks will handle load balancing, networking, scaling up and down resources, and controlling the health of the system.

Kubernetes is made up of a collection of worker machines, often known as nodes, on which containers are run. Pods are components of the workload, and these worker nodes host pods. A cloud provider API, worker nodes, and a control plane are all included in the Kubernetes cluster (made of kube apiserver, etcd, kube schedular and kube manager) [9].

6. Recommendations and Conclusions

The machine learning model developed here uses limited resources and lightweight frameworks such as cURL, flask, and minikube. It is suggested that the model be implemented with additional features such as danjo, postman, and Kubernetes. To test the APIs and handle the data, the ML model required some manual intervention. Because the API will be unit tested, there will be more automation in the improved system. The data will be processed in batches, and the flask interface of the web application will have additional UI features for accepting data in batches in a secure manner. More AI models will be tested as an enhancement, with the best performing model will be moved to production.

To summarize, MLOPS is a required practice for every ML and AI project development to minimize operational and estimation difficulties and to establish best development practices. MLOPS techniques have become more automated over time, with less human involvement to reduce errors.

GitHub files

The machine learning model was created as an API with Flask and Dockers, and container orchestration was done with Minkube. The code files are available in the below Github location.

https://github.com/MallikharjunaSakhamuri/MLOPS

Biographies

References

[1] Kreuzberger, D., Kühl, N., Hirschl, S.. (2022). Machine Learning Operations (MLOps): Overview, Definition, and Architecture.

[2] ml-ops.org. (n.d.). ml-ops.org. [online] Available at: https://ml-ops.org/content/end-to-end-ml-workflow [Accessed 2 Jun. 2022].

[3] Tyagi, H. (2021). What is MLOps — Everything You Must Know to Get Started. [online] Medium. Available at: https://towardsdatascience.com/what-is-mlops-everything-you-must-know-to-get-started-523f2d0b8bd8 [Accessed 2 Jun. 2022].

[4] Susilo, M. (2021). MLOps level 2 — CI/CD + Continous Training. [online] Medium. Available at: https://towardsdatascience.com/mlops-level-2-ci-cd-continous-training-b9b64042368 [Accessed 2 Jun. 2022].
[5] www.datacamp.com. (n.d.). Machine Learning, Pipelines, Deployment and MLOps Tutorial. [online] Available at: https://www.datacamp.com/tutorial/tutorial-machine-learning-pipelines-mlops-deployment [Accessed 2 Jun. 2022].
[6] IBM Developer. (n.d.). What is the curl command? Learning and testing APIs with cURL tools. [online] Available at: https://developer.ibm.com/articles/what-is-curl-command/.
[7] Idris, N., Foozy, C.F.M. and Shamala, P. (2020). A Generic Review of Web Technology: DJango and Flask. International Journal of Advanced Science Computing and Engineering, [online] 2(1), pp.34–40. doi:10.30630/ijasce.2.3.29.
[8] Y. Zhou, Y. Yu and B. Ding, "Towards MLOps: A Case Study of ML Pipeline Platform," 2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE), 2020, pp. 494-500, doi: 10.1109/ICAICE51518.2020.00102.
[9] Choudhary, S. (2021). Kubernetes-Based Architecture For An On-premises Machine Learning Platform (Aalto University. School of Science). http://urn.fi/URN:NBN:fi:aalto-202110249694
[10] T. Granlund, A. Kopponen, V. Stirbu, L. Myllyaho and T. Mikkonen, "MLOps Challenges in Multi-Organization Setup: Experiences from Two Real-World Cases," 2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN), 2021, pp. 82-88, doi: 10.1109/WAIN52551.2021.00019.