# Lab Report: MLflow

## Student information


- Student name: Xander Van der Linden
- Student code: 202292316


## Assignment description

In deze opdracht moest je een machine learning workflow opzetten en beheren met behulp van Prefect en MLFlow, en het model vervolgens deployen naar een Azure Managed Endpoint.

Notebook uitvoeren: voerde een Jupyter notebook uit met een eenvoudig ML-model dat appels en sinaasappels classificeert. Het doel hiervan was om bekend te raken met het model en de data.

Virtuele Omgeving Instellen: Ik maakte een virtuele omgeving voor het project en installeerde benodigde dependencies om een geïsoleerde omgeving te creëren.

Prefect Pipeline Opzetten: Ik creëerde een ML pipeline met Prefect, bestaande uit vier taken:

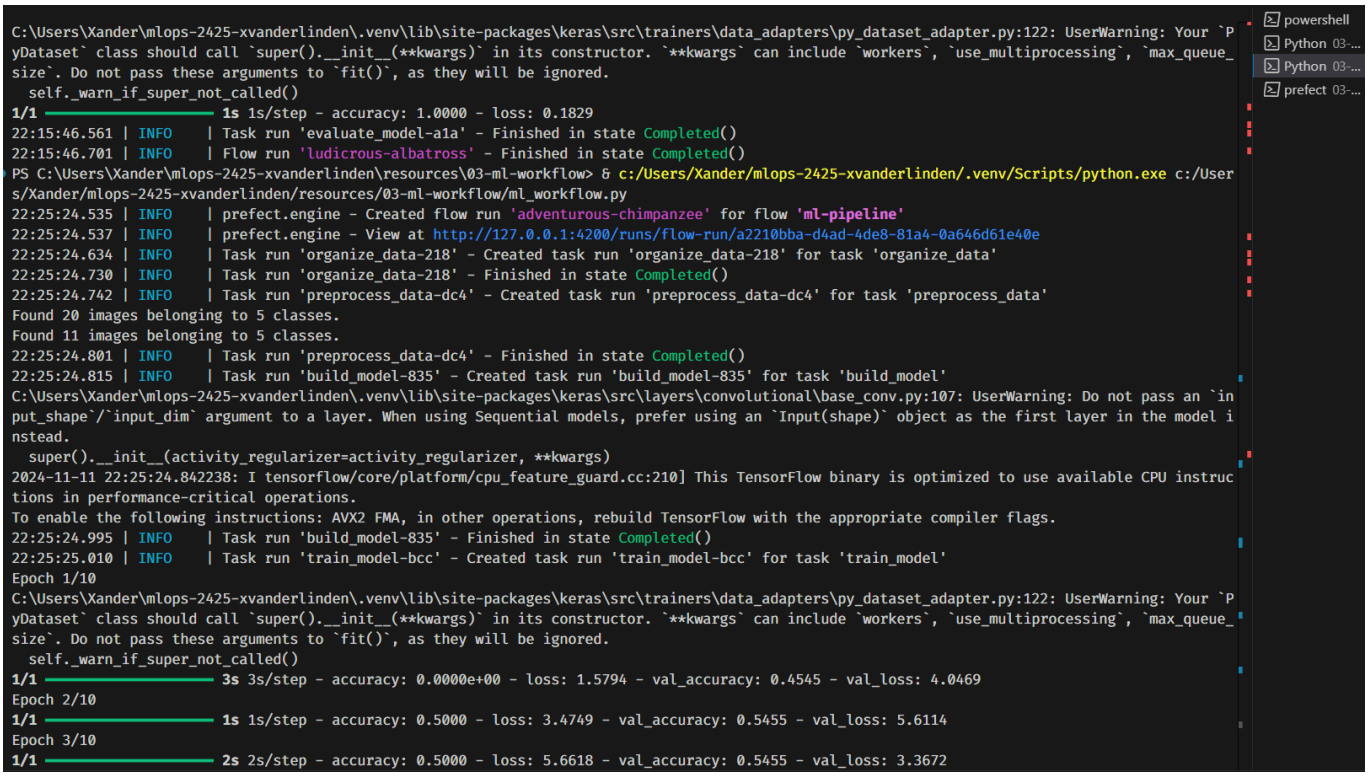
Data downloaden Data preprocessen en splitsen Model trainen Model evalueren Elke stap werd als een afzonderlijke task gedefinieerd en gecombineerd in een flow.

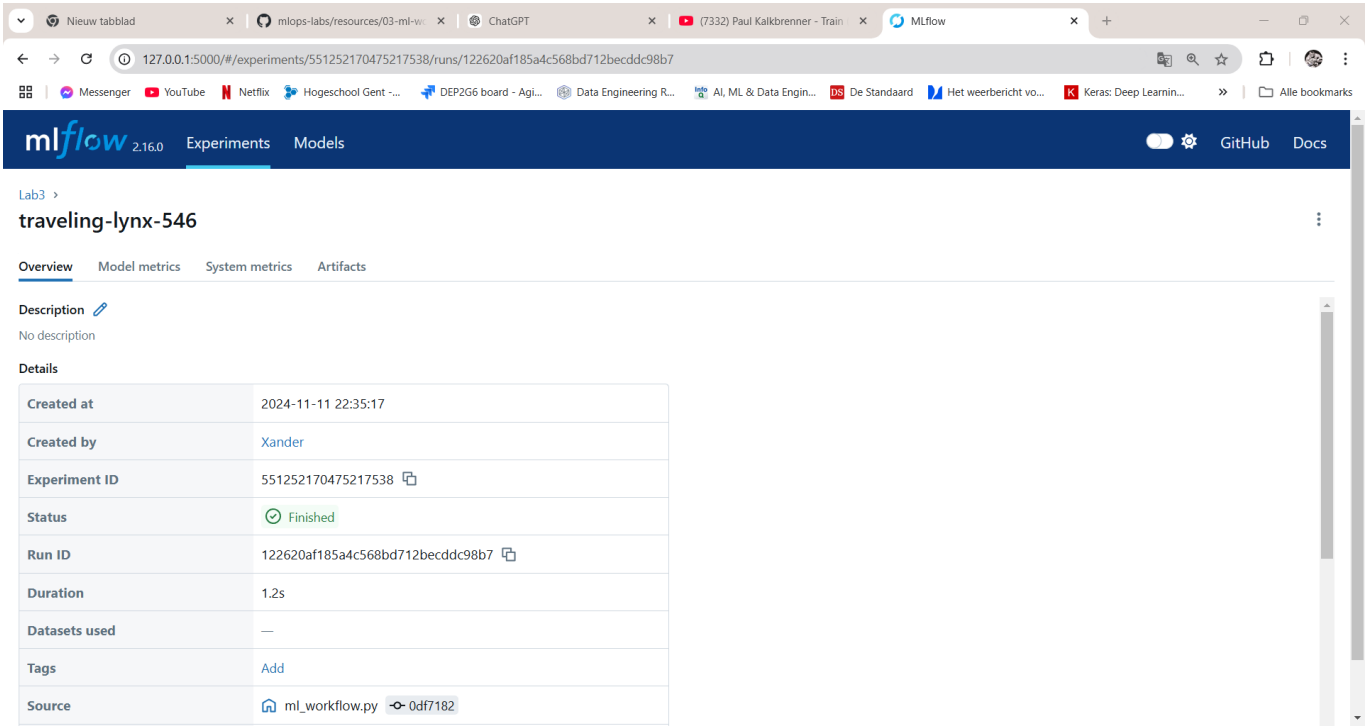
MLFlow Integratie: Ik integreerde MLFlow om experimenten te loggen en modeltrainingen te monitoren. Ik gebruikte autologging voor metrics en het loggen van model-artefacten zoals het aantal epochs en batchgrootte.

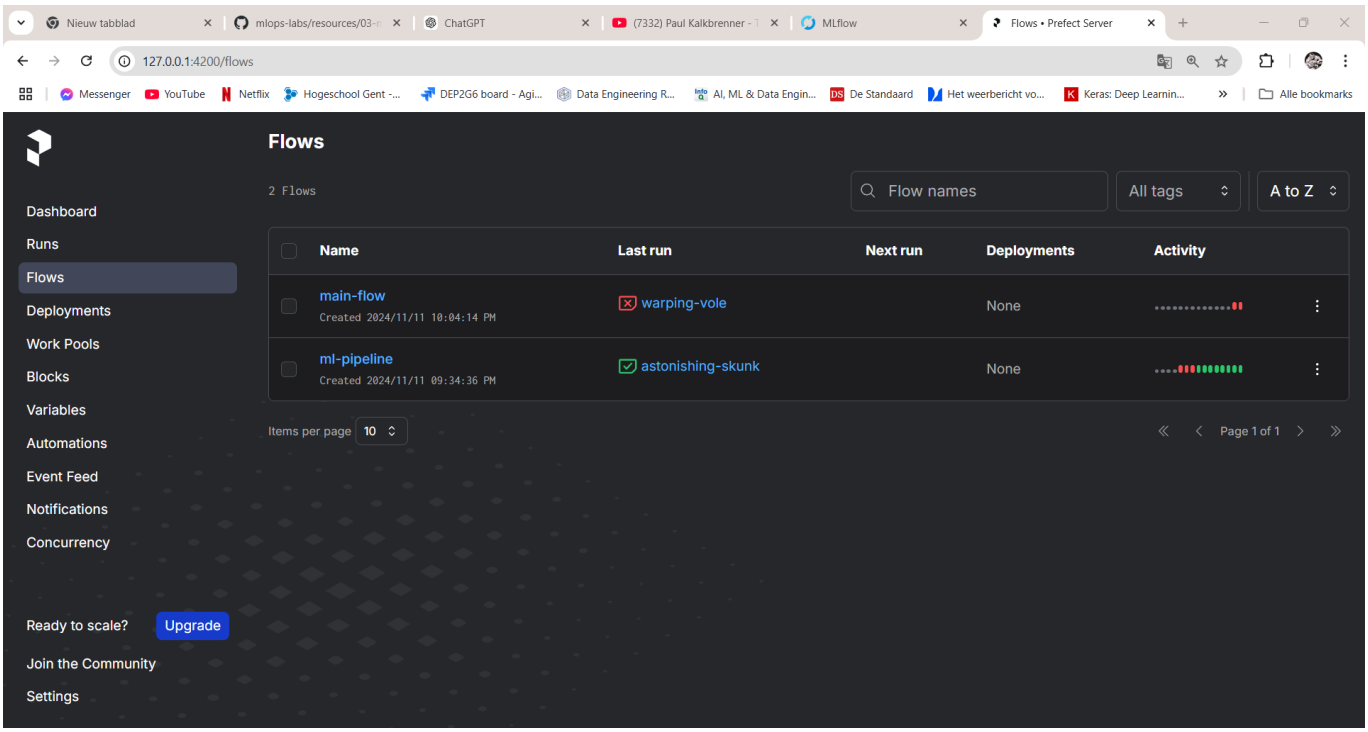
Model Deployen en Predictie Maken: Ik deployde het model naar een Azure Managed Endpoint en maakte een voorspelling met het gedeployede model.

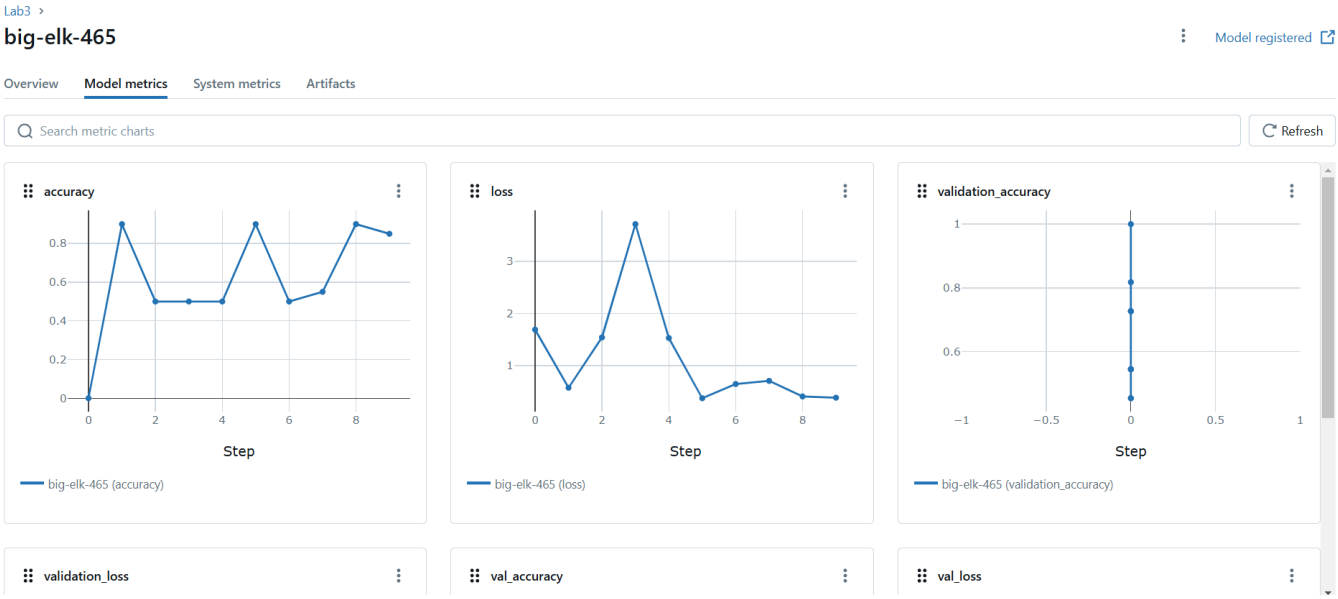
Verwachte Output: Een volledig uitgevoerde notebook in de repository. Screenshots van de Prefect en MLFlow dashboards. Een labverslag met antwoorden op vragen en screenshots van de resultaten. Een werkende deployment van het model en succesvolle voorspellingen

## Proof of work done

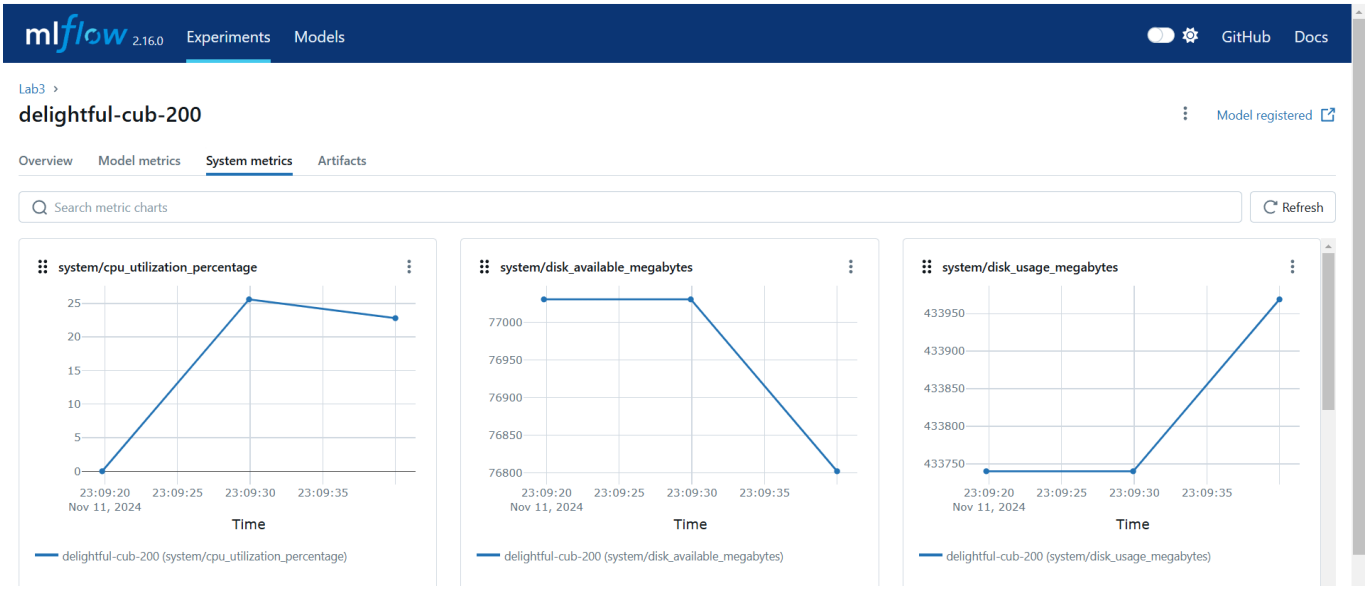

powershell
Python 03-...
Python 03-...
prefect 03-...
C:\Users\Xander\mlops-2425-xvanderlinden\.venv\lib\site-packages\keras\src\trainers\data_adapters\py_dataset_adapter.py:122: UserWarning: Your `PyDataset` class should call `super().__init__(**kwargs)` in its constructor. `**kwargs` can include `workers`, `use_multiprocessing`, `max_queue_size`. Do not pass these arguments to `fit()`, as they will be ignored.
self._warn_if_super_not_called()
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step - accuracy: 1.0000 - loss: 0.1829
22:15:46.561 | INFO | Task run 'evaluate_model-a1a' - Finished in state Completed()
22:15:46.701 | INFO | Flow run 'ludicrous-albatross' - Finished in state Completed()
PS C:\Users\Xander\mlops-2425-xvanderlinden\resources\03-ml-workflow> & c:/Users/Xander/mlops-2425-xvanderlinden/.venv/Scripts/python.exe c:/Users/Xander/mlops-2425-xvanderlinden/resources/03-ml-workflow/ml_workflow.py
22:25:24.535 | INFO | prefect.engine - Created flow run 'adventurous-chimpanzee' for flow 'ml-pipeline'
22:25:24.537 | INFO | prefect.engine - View at http://127.0.0.1:4200/runs/flow-run/a2210bba-d4ad-4de8-81a4-0a646d61e40e
22:25:24.634 | INFO | Task run 'organize_data-218' - Created task run 'organize_data-218' for task 'organize_data'
22:25:24.730 | INFO | Task run 'organize_data-218' - Finished in state Completed()
22:25:24.742 | INFO | Task run 'preprocess_data-dc4' - Created task run 'preprocess_data-dc4' for task 'preprocess_data'
Found 20 images belonging to 5 classes.
Found 11 images belonging to 5 classes.
22:25:24.801 | INFO | Task run 'preprocess_data-dc4' - Finished in state Completed()
22:25:24.815 | INFO | Task run 'build_model-835' - Created task run 'build_model-835' for task 'build_model'
C:\Users\Xander\mlops-2425-xvanderlinden\.venv\lib\site-packages\keras\src\layers\convolutional\base_conv.py:107: UserWarning: Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
super().__init__(activity_regularizer=activity_regularizer, **kwargs)
2024-11-11 22:25:24.842238: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instructions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
22:25:24.995 | INFO | Task run 'build_model-835' - Finished in state Completed()
22:25:25.010 | INFO | Task run 'train_model-bcc' - Created task run 'train_model-bcc' for task 'train_model'
Epoch 1/10
C:\Users\Xander\mlops-2425-xvanderlinden\.venv\lib\site-packages\keras\src\trainers\data_adapters\py_dataset_adapter.py:122: UserWarning: Your `PyDataset` class should call `super().__init__(**kwargs)` in its constructor. `**kwargs` can include `workers`, `use_multiprocessing`, `max_queue_size`. Do not pass these arguments to `fit()`, as they will be ignored.
self._warn_if_super_not_called()
1/1 ━━━━━━━━━━━━━━━━━━━━ 3s 3s/step - accuracy: 0.0000e+00 - loss: 1.5794 - val_accuracy: 0.4545 - val_loss: 4.0469
Epoch 2/10
1/1 ━━━━━━━━━━━━━━━━━━━━ 1s 1s/step - accuracy: 0.5000 - loss: 3.4749 - val_accuracy: 0.5455 - val_loss: 5.6114
Epoch 3/10
1/1 ━━━━━━━━━━━━━━━━━━━━ 2s 2s/step - accuracy: 0.5000 - loss: 5.6618 - val_accuracy: 0.5455 - val_loss: 3.3672


Nieuw tabblad
mlops-labs/resources/03-ml-w
ChatGPT
(7332) Paul Kalkbrenner - Train
MLflow
127.0.0.1:5000/#/experiments/551252170475217538/runs/122620af185a4c568bd712becddc98b7
Messenger
YouTube
Netflix
Hogeschool Gent -...
DEP2G6 board - Agi...
Data Engineering R...
AI, ML & Data Engin...
De Standaard
Het weerbericht vo...
Keras: Deep Learnin...
Alle bookmarks
mlflow 2.16.0
Experiments
Models
GitHub
Docs
Lab3 ›
traveling-lynx-546
Overview
Model metrics
System metrics
Artifacts
Description
No description
Details
Created at 2024-11-11 22:35:17
Created by Xander
Experiment ID 551252170475217538
Status Finished
Run ID 122620af185a4c568bd712becddc98b7
Duration 1.2s
Datasets used —
Tags Add
Source ml_workflow.py 0df7182


Nieuw tabblad
mlops-labs/resources/03-
ChatGPT
(7332) Paul Kalkbrenner -
MLflow
Flows • Prefect Server
127.0.0.1:4200/flows
Messenger
YouTube
Netflix
Hogeschool Gent -...
DEP2G6 board - Agi...
Data Engineering R...
AI, ML & Data Engin...
De Standaard
Het weerbericht vo...
Keras: Deep Learnin...
Alle bookmarks
Flows
Dashboard
Runs
Flows
Deployments
Work Pools
Blocks
Variables
Automations
Event Feed
Notifications
Concurrency
Ready to scale?
Upgrade
Join the Community
Settings
2 Flows
Flow names
All tags
A to Z
Name
Last run
Next run
Deployments
Activity
main-flow
Created 2024/11/11 10:04:14 PM
warping-vole
None
ml-pipeline
Created 2024/11/11 09:34:36 PM
astonishing-skunk
None
Items per page
10
Page 1 of 1


Lab3
big-elk-465
Model registered
Overview
Model metrics
System metrics
Artifacts
Search metric charts
Refresh
accuracy
Step
big-elk-465 (accuracy)
loss
Step
big-elk-465 (loss)
validation_accuracy
Step
big-elk-465 (validation_accuracy)
validation_loss
val_accuracy
val_loss

```
(venv) PS C:\Users\Xander\mlops-2425-xvanderlinden\resources\03-ml-workflow> & c:/Users/Xander/mlops-2425-xvanderlinden/.venv/Scripts/python.exe
c:/Users/Xander/mlops-2425-xvanderlinden/resources/03-ml-workflow/predict_image.py
2024-11-11 23:02:38.632592: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to use available CPU instruc
tions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
WARNING:absl:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until you train or e
valuate the model.
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 94ms/step
Predicted Label: class_1
Class Probabilities: [0.798334   0.1769082  0.00929314 0.00718282 0.00828189]
```




## Evaluation criteria

- ☑ Show that you've executed the notebook and pushed it to the repository ☑ Show that your Jupyter notebook contains all cells' output
- [x]Show that you created a virtual environment for the project
- [x]Show the Prefect and MLFlow dashboards
- [x]Show that your ML pipeline is working
- [x]Show the logs and metrics in the MLFlow dashboard
- [x]Show that you pushed a model to MLFlow
- [x]Show that you wrote an elaborate lab report in Markdown and pushed it to the repository
    - [x]Show that it contains the answers to the questions in the lab assignment
    - [x]Show that it contains the screenshots of the MLFlow dashboard
- [x]Show that you are able to make a prediction with the deployed model

## Issues

Ik had problemen met het opzetten van de virtuele enviroment. Er waren problemen met Versies van Keras pip en Python. Dit heb ik opgelost door een nieuwe Python versie te installeren, maar toen had ik problemen omdat sommige afhankelijkheden niet werkte met de nieuwste versie van python en python niet juist in het Path van windows stond.Ik had problemen met het MLflow script te runnen. Aangezien er enviromental variabelen waren die niet correct waren ingesteld.Ook waren er problemen met de github Url en de directories

## Reflection

Ik leerde hoe ik een Machine learning pipeline opzetten en hergebruikte via MLflow. Nu ik weet hoe dit moet is het zeker een nuttige skill voor in de praktijk te gebruiken. Het model van nu is niet zo bruikbaar aangezien

de dataset redelijk klein was. De Voorspellingen zijn dus niet mega accuraat.

## Antwoorden op de vragen

### Wat doet het commando `python3 -m venv venv`?

Dit commando maakt een nieuwe virtuele omgeving voor Python aan. Het is een geïsoleerde ruimte waar je de benodigde afhankelijkheden (zoals bibliotheken) kunt installeren voor je project, zonder dat dit invloed heeft op andere projecten of het systeem zelf.

### Wat betekent het eerste argument `venv`?

Het eerste `venv` is de naam van de virtuele omgeving die je wilt maken. Dit is de naam van de map waarin de virtuele omgeving wordt aangemaakt.

### Wat betekent het tweede `venv`?

Het tweede `venv` is de naam van de directory waarin de virtuele omgeving wordt aangemaakt. Dit kan je aanpassen naar een andere naam als je wilt, bijvoorbeeld `mijn_omgeving`.

### Welke van de twee kun je naar eigen voorkeur wijzigen?

Je kunt het tweede `venv` (de naam van de map) aanpassen. De eerste `venv` is de naam van de module die door Python wordt gebruikt om de virtuele omgeving aan te maken, dus die kun je niet wijzigen.

### Hoe zorg je ervoor dat je virtuele omgeving niet door Git wordt gevolgd?

Om ervoor te zorgen dat je virtuele omgeving niet door Git wordt gevolgd, voeg je de naam van de virtuele omgeving (bijvoorbeeld `venv/` of de naam van jouw virtuele omgeving) toe aan het `.gitignore`-bestand. Dit bestand zorgt ervoor dat Git de virtuele omgeving en andere tijdelijke bestanden negeert. Voeg de volgende regel toe aan het `.gitignore`

venv/

### Waar worden de afhankelijkheden geïnstalleerd?

De afhankelijkheden worden geïnstalleerd in de virtuele omgeving zelf, meestal in de map venv in je projectdirectory. Dit zorgt ervoor dat de geïnstalleerde pakketten enkel beschikbaar zijn voor dit specifieke project, en niet voor andere projecten of systemen. De geïnstalleerde pakketten staan dus niet in de globale Python-installatie.

### Waarom moeten we de omgeving variabele PREFECT_HOME instellen?

De omgeving variabele PREFECT_HOME wordt gebruikt om de opslaglocatie van de Prefect configuratie en runtime data aan te geven. Door deze variabele in te stellen, geef je aan waar Prefect zijn configuratiebestanden en logs moet opslaan. Dit is belangrijk omdat het Prefect systeem afhankelijk is van deze configuratie om de werkstromen goed te beheren.

### Wat is het nut van het starten van de Prefect server?

Het starten van de Prefect server is nodig om toegang te krijgen tot het Prefect dashboard. Dit dashboard biedt een visuele interface voor het beheren en monitoren van je werkstromen (pipelines). Het stelt je in staat om de voortgang van taken te volgen, logs te bekijken en fouten te debuggen.

### Hoe maak je een Prefect pipeline?

In de pipeline moet je verschillende taken (tasks) definiëren, zoals het downloaden van data, het voorverwerken van data, het trainen van het model en het evalueren van het model. Deze taken worden uitgevoerd in een bepaalde volgorde, wat je definieert met een flow.

Voorbeeldtructuur pipeline:

Download data: Download de afbeeldingen van de GitHub repository. Preprocess data: Verwerk de afbeeldingen voor en splits ze in trainings-, validatie- en testsets. Train het model: Train een model op de trainingsset. Evalueer het model: Evalueer het model op de testset. Elke stap moet worden gedefinieerd als een Prefect taak en aan elkaar worden gekoppeld in de juiste volgorde met behulp van een Prefect flow.

### Hoe log je experimenten en modelinformatie in MLFlow?

In MLFlow kun je experimenten en modelinformatie loggen door het instellen van de tracking URI en de experimentnaam in je script. Door gebruik te maken van MLFlow's autologging, kun je automatisch belangrijke informatie loggen, zoals modelparameters, metrics en zelfs het model zelf.

Autologging inschakelen: Voor het trainen en evalueren van het model kun je autologging inschakelen om automatisch de hyperparameters, metrics, modelgewichten en andere belangrijke informatie op te slaan. Je kunt bijvoorbeeld mlflow.keras.autolog() gebruiken om Keras modeltraining automatisch te loggen.

System metrics loggen: Door de omgevingsvariabele MLFLOW_ENABLE_SYSTEM_METRICS_LOGGING in te stellen op true, kun je systeemmetrics zoals CPU-gebruik, geheugenverbruik en andere statistieken loggen tijdens de uitvoering van je pipeline.

## Resources

https://docs.prefect.io/3.0/develop/write-flows https://docs.prefect.io/3.0/develop/write-tasks