INNOMATICS®
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

Experiment Tracking and Model Management

(Sentiment Analysis Project)

Submitted by:

Pandey Abhishek Nath Roy

Intern ID : 1240134

About me

- Hi there! I am **Pandey Abhishek Nath Roy (IN1240134),** a fresher data enthusiast, currently learning various things to crack an opportunity to go further.
- Apart from this, I possess the problem solving ability and I am good at learning new things that makes me an ideal candidate to follow my dreams.

- I have previously worked as a data science intern at **"Innomatics Research Labs"** and right now, I am doing the internship to refine my skills at their highest level.

- Feel free to reach out! You can do so by following the below links:

  Github @vjabhi000985

  Linkedin

INNOMATICS RESEARCH LABS

OBJECTIVE OF THE PROJECT

- The objective of this task is to introduce you to MLflow for experiment tracking, model management, and reproducibility in machine learning projects for the **"Sentiment Analysis Project".**

INNOMATICS RESEARCH LABS

MLFLOW

- "MLflow" is an open source platform for managing the machine learning lifecycle.
- **What it does :-**
  - ❑ ***Tracks experiments:*** Logs parameters, metrics, and artifacts (models, code) during training runs.
  - ❑ ***Enables reproducibility:*** Compares experiments and facilitates replicating successful runs.
  - ❑ ***Model registry:*** Stores, versions, manages, and deploys trained models.
- **Benefits :-**
  - ❑ Improves Collaboration.
  - ❑ Streamline deployments and Automates workflows.

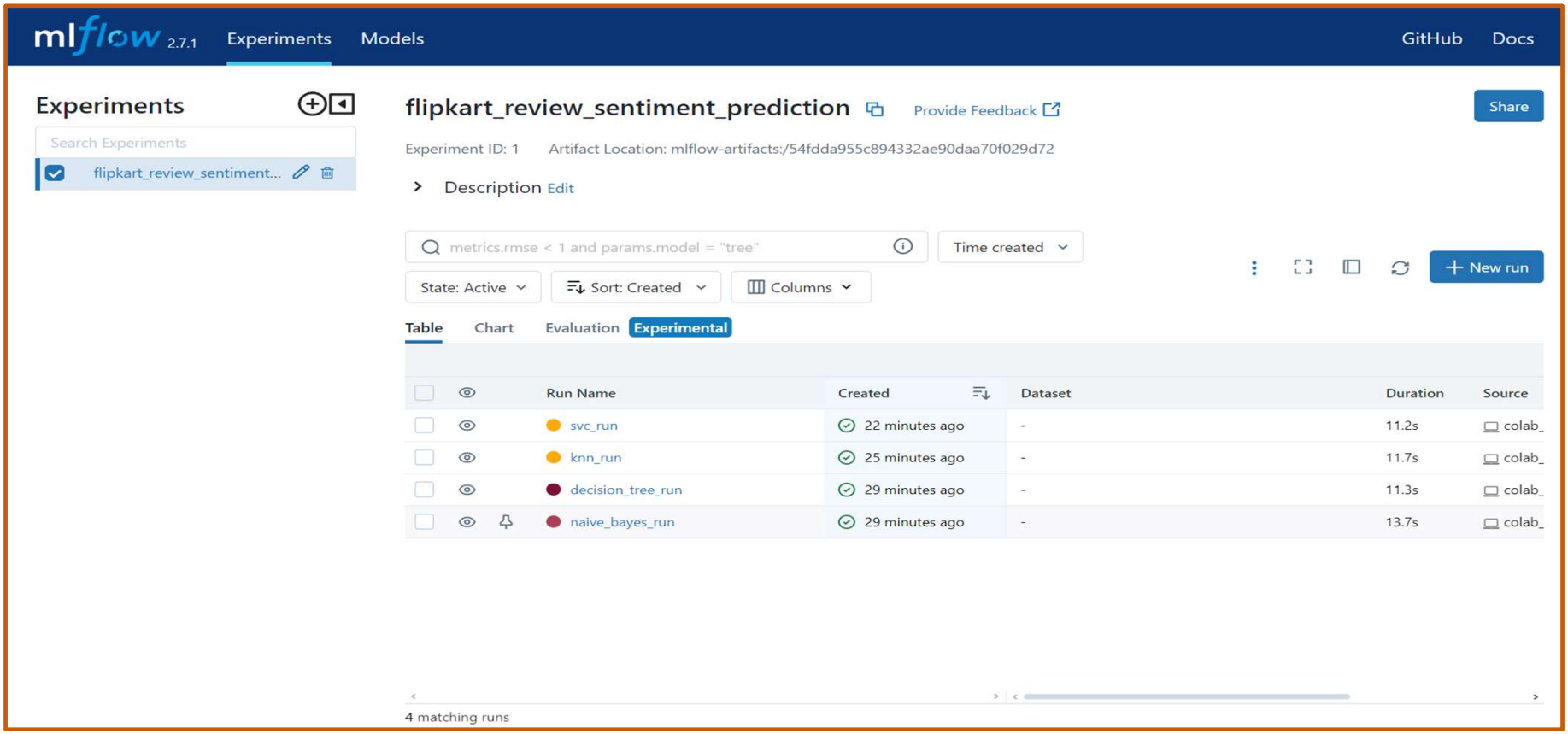
INNOMATICS RESEARCH LABS

Integration of MLflow into Projects

```
 # Install the required packages
$ pip3 install mlflow dagshub --quiet


import mlflow

mlflow.set_experiment("flipkart_review_sentiment_prediction")

# Start the experiment run
   with mlflow.start_run() as run:
         %time grid_search.fit(X_train, y_train)
```

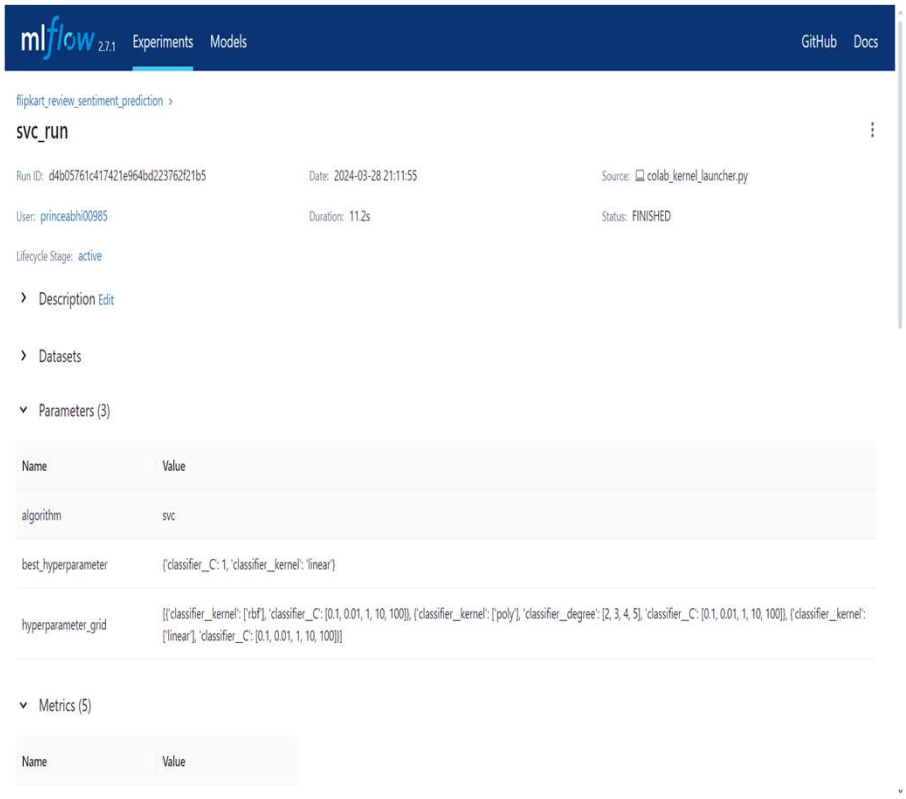
INNOMATICS RESEARCH LABS

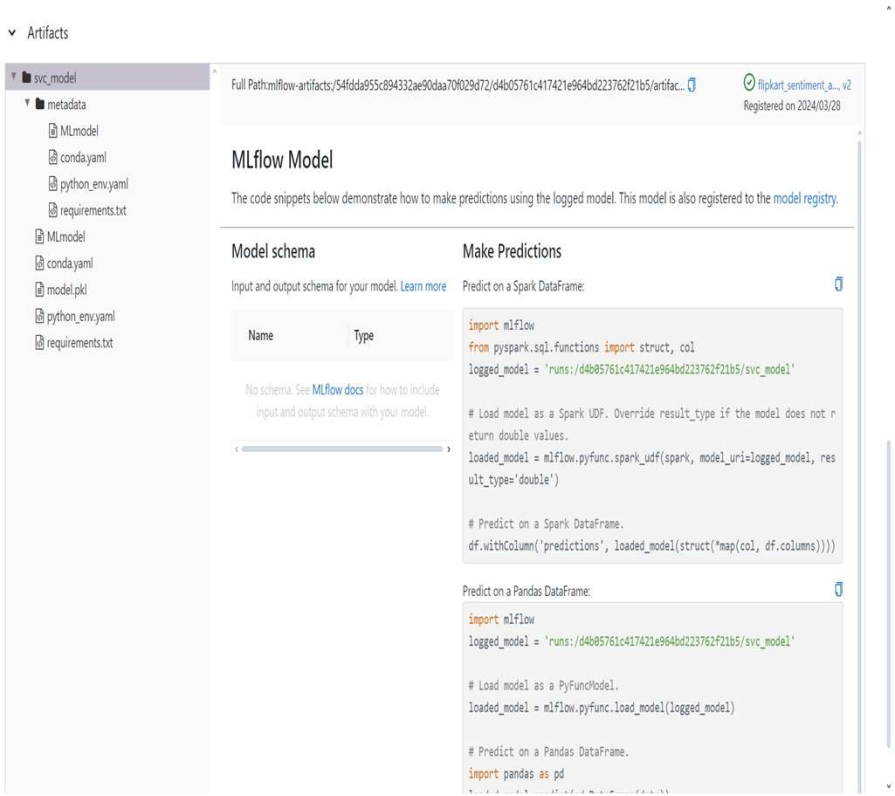
MLflow Dashboard



INNOMATICS RESEARCH LABS

Demonstration of Parameters, Metrics and Artifacts





INNOMATICS RESEARCH LABS

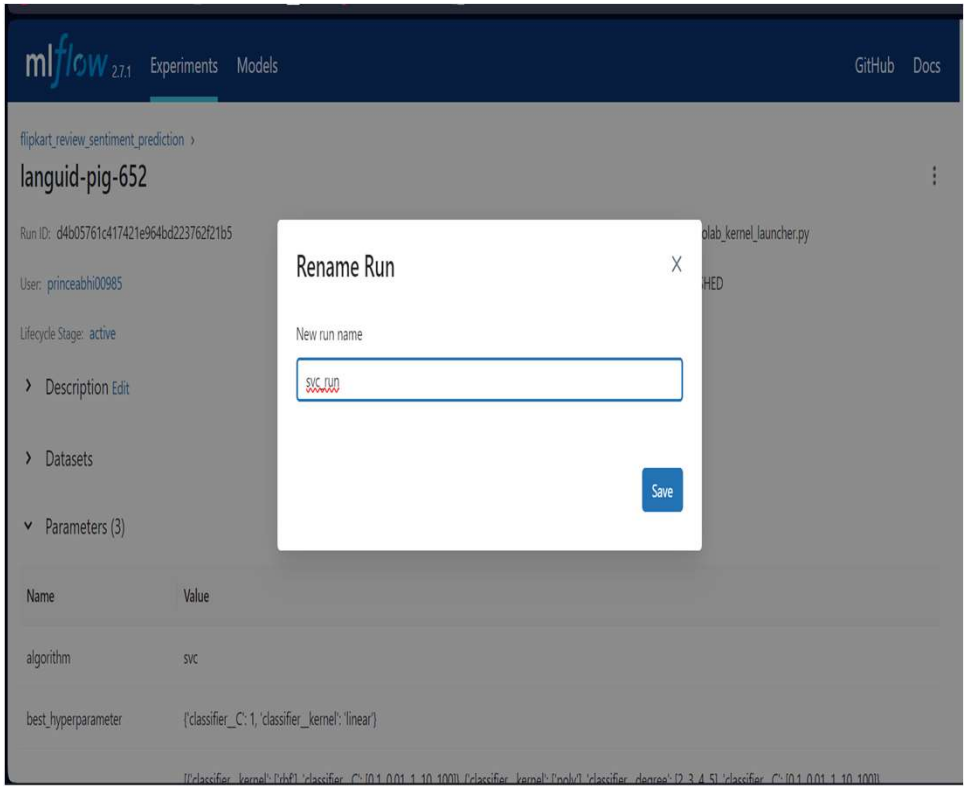
Demonstration of Parameters, Metrics and Artifacts using MLflow tracking API

Click on "Experiment Name" -> Run Name -> (scroll down for) -> Parameters, Metrics and Artifacts .

INNOMATICS RESEARCH LABS

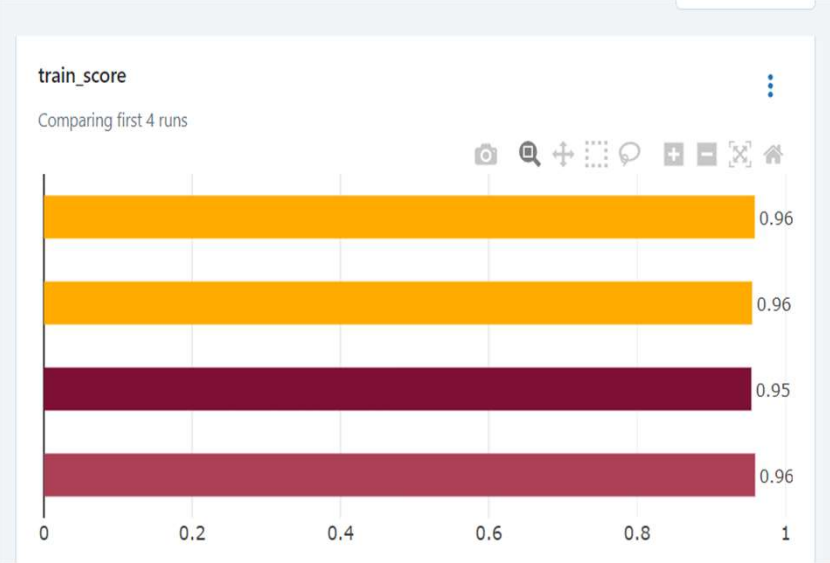
Customizing Mlflow with Run Names

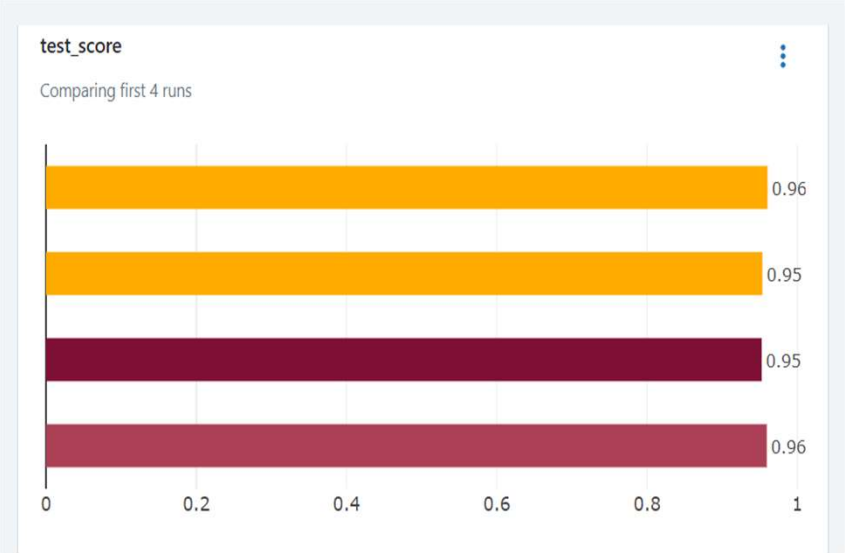
Click on the run name -> Right corner : 3 dots -> Rename -> Give the name of run -> Save



INNOMATICS RESEARCH LABS

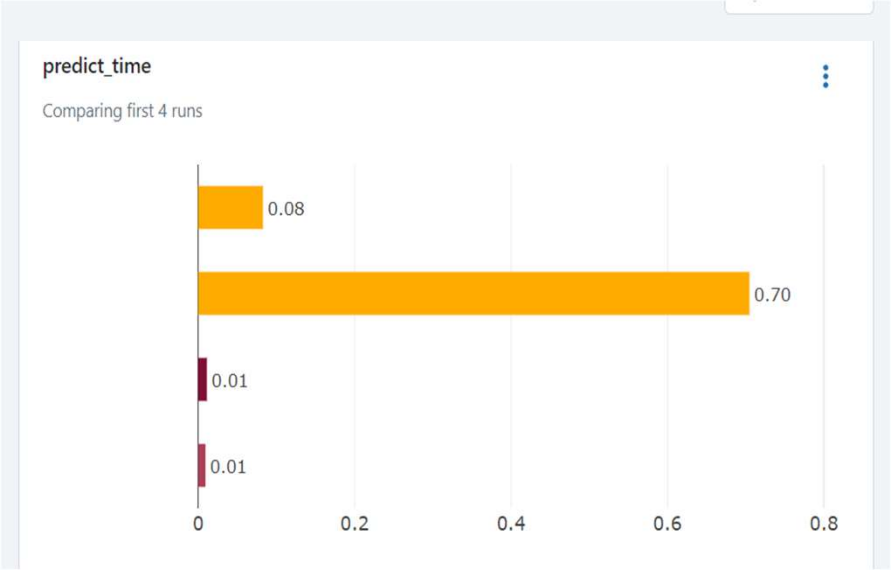
Metric Plots :





INNOMATICS RESEARCH LABS

Metric Plots :



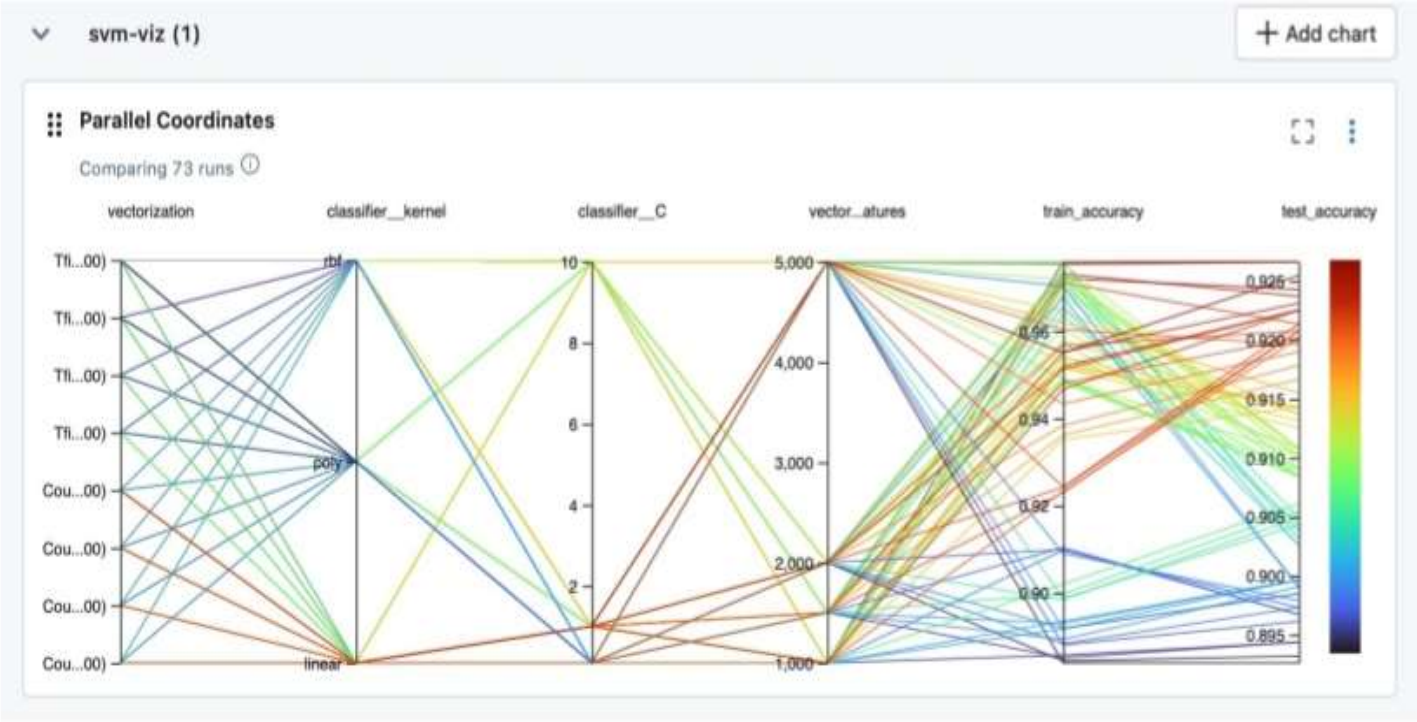
INNOMATICS RESEARCH LABS

Hyperparameter Plots Creation

Click on charts -> Add Section -> Name hyperparameter with model ->
Add Parallel coordinates chart -> give parameters and metrics of particular run ->
Click on the run-name's "+" : This will pop interactive chart.

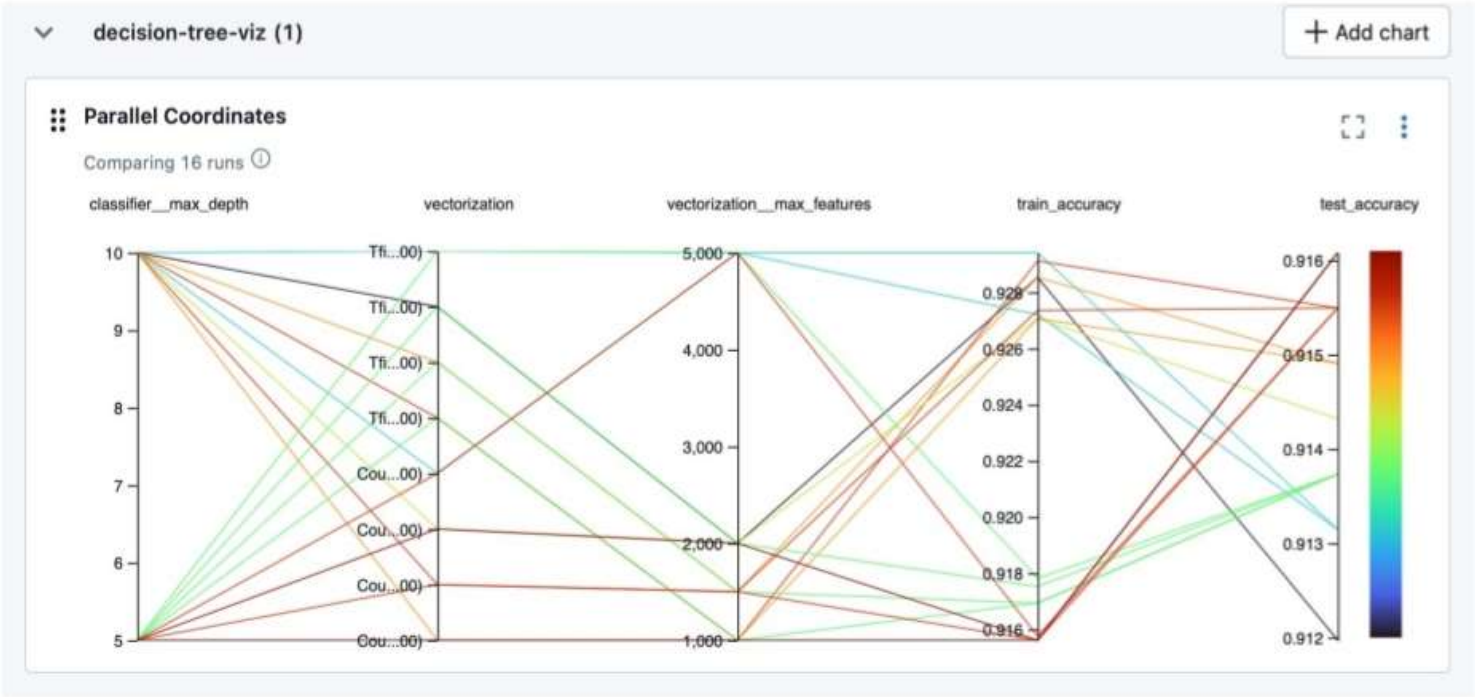
INNOMATICS RESEARCH LABS

Hyperparameter Plots



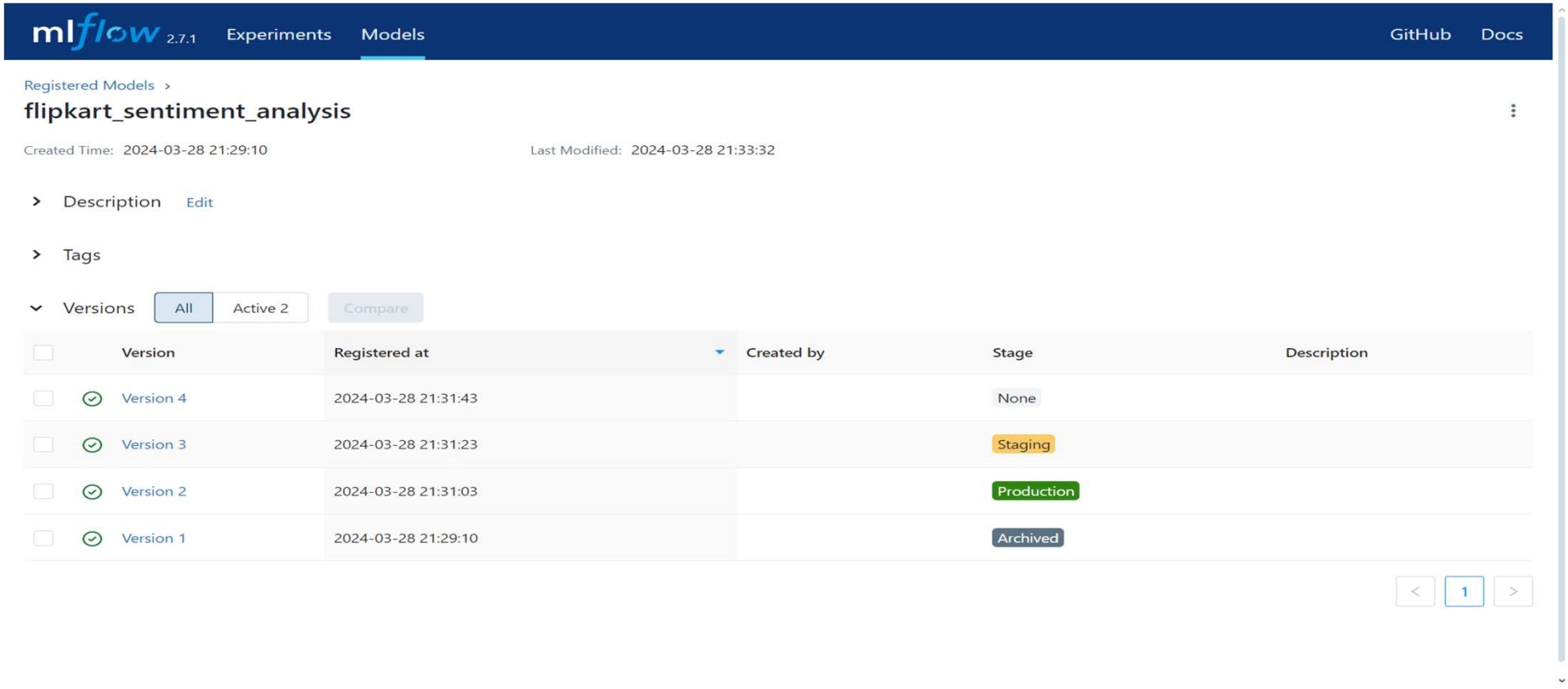
INNOMATICS RESEARCH LABS

Hyperparameter Plots



INNOMATICS RESEARCH LABS

Registering the models



INNOMATICS RESEARCH LABS

Prefect

Prefect is a Python library that helps you build organized and efficient data pipelines, especially for machine learning. It allows you to:

- **Break down complex tasks:** Prefect lets you divide your data processing or machine learning workflow into smaller, reusable steps called tasks.
- **Manage task order:** It cleverly manages the order in which these tasks run, ensuring everything happens at the right time.
- **Keep an eye on progress:** Prefect monitors your workflows as they run, catching any errors and offering visualizations to help you troubleshoot.

INNOMATICS RESEARCH LABS

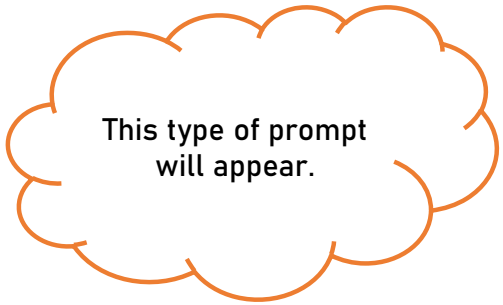
Prefect : Benefits

- **Less code, more reuse:** By using modular tasks, you can write code once and use it in multiple workflows, saving time and effort.
- **Easier maintenance:** Clear organization with tasks makes your workflows easier to understand, modify, and debug later on.
- **Scales as you grow:** Prefect can handle large and complex workflows by allowing tasks to be run on multiple machines.

Prefect : Installation and First run

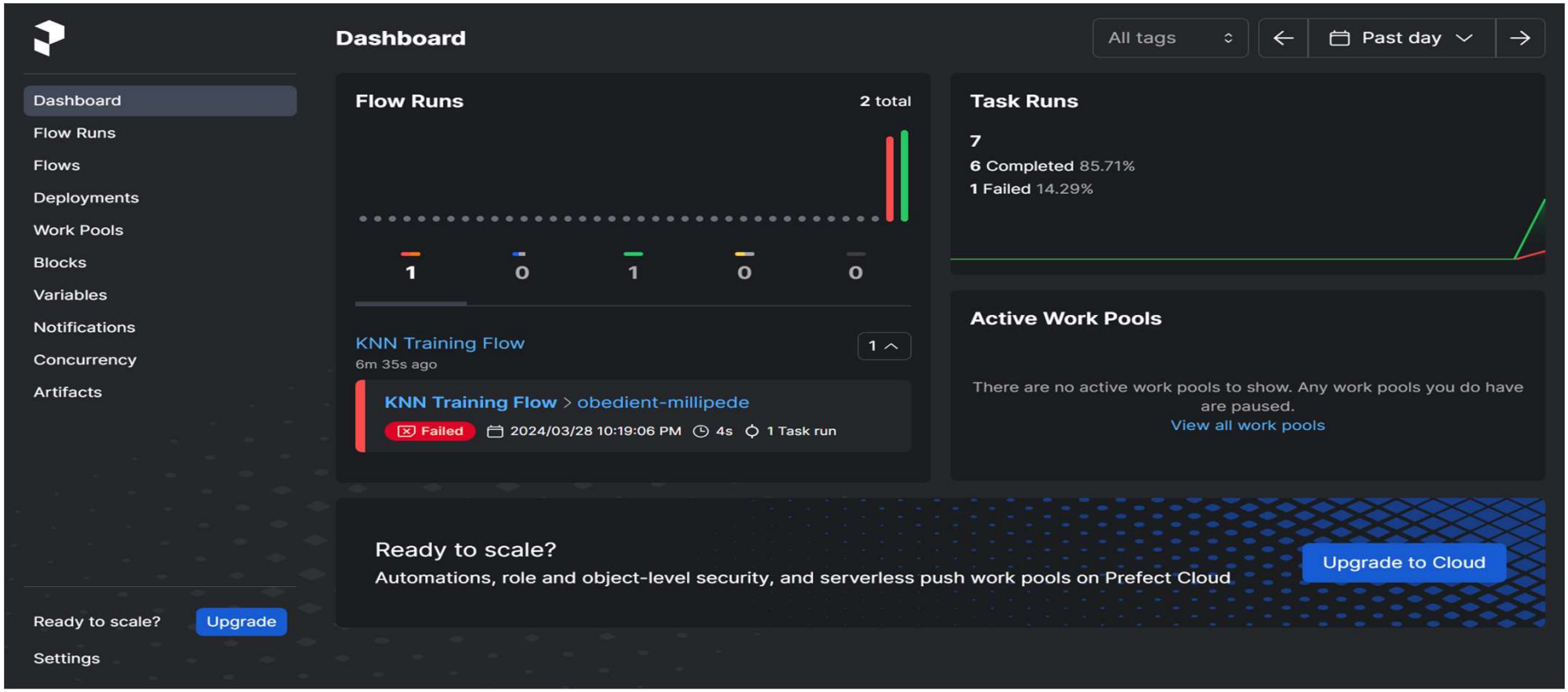
- **Run the following commands to install the prefect and to run it.**
  - ❑ **$ pip install prefect**
  - ❑ **$ prefect server start**

```
PS C:\Users\vjavi\Desktop\my_flask_projects\MLflow_Prefect_Orchestration> prefect server start


 ___ ___ ___ ___ ___ ___ _____
| _ \ _ \ __| __| __/ __|_   _|
|  _/   / _|| _|| _| (__  | |
|_| |_|_\___|_| |___\___| |_|

Configure Prefect to communicate with the server with:

    prefect config set PREFECT_API_URL=http://127.0.0.1:4200/api
```



INNOMATICS RESEARCH LABS

Prefect Dashboard



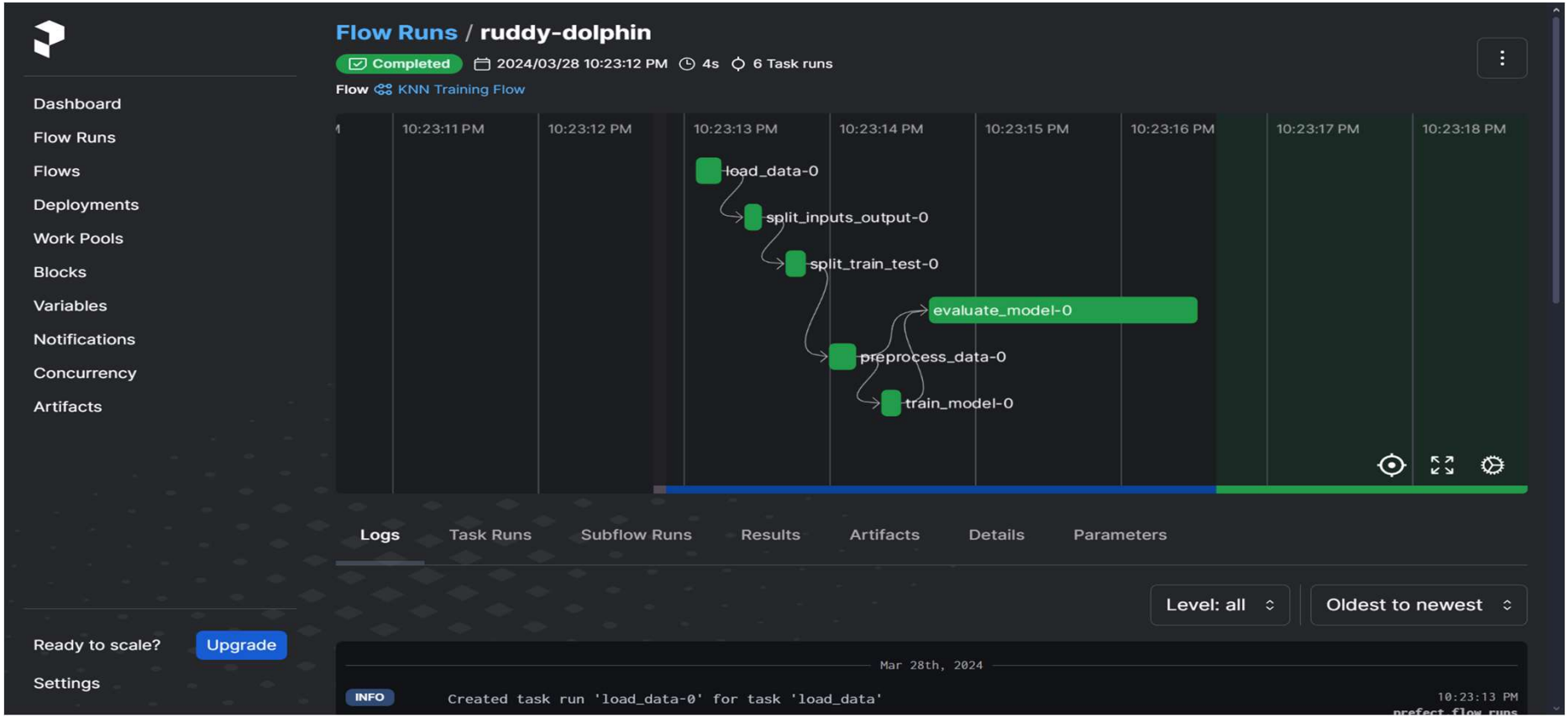
INNOMATICS RESEARCH LABS

Building a Prefect workflow

- ✓ **Import Prefect Models *(step 1).***
- ✓ **Define Prefect Tasks *(step 2).***
- ✓ **Define Prefect Flow *(step 3).***
- ✓ **Run Prefect Workflow *(step 4).***

INNOMATICS RESEARCH LABS

Prefect workflow



INNOMATICS RESEARCH LABS

THANK YOU

INNOMATICS
RESEARCH LABS