INNOMATICS®
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

Sentimental Analysis Experimentation using MLFlow Integration

Objective of the project

The key aims of this project include:

1. Integrate MLflow into machine learning project.
2. Customizing MLflow UI with run names.
3. Demonstration of log parameters, metrics, and artifacts using MLflow tracking APIs.
4. Demonstration of metric plots and hyperparameter plots.
5. Demonstration on registering models and managing them by tagging.

Introduction

MLflow is an open-source platform for managing the end-to-end machine learning lifecycle. It provides a suite of tools and components designed to streamline the development, experimentation, productionization, and collaboration aspects of machine learning projects. MLflow is widely used by data scientists, machine learning engineers, and researchers to track experiments, package and share code, and deploy models at scale.

Key Features:

1. Experiment Tracking
2. Model Registry

Integrating MLFlow

Run below commands to install mlflow on your system:

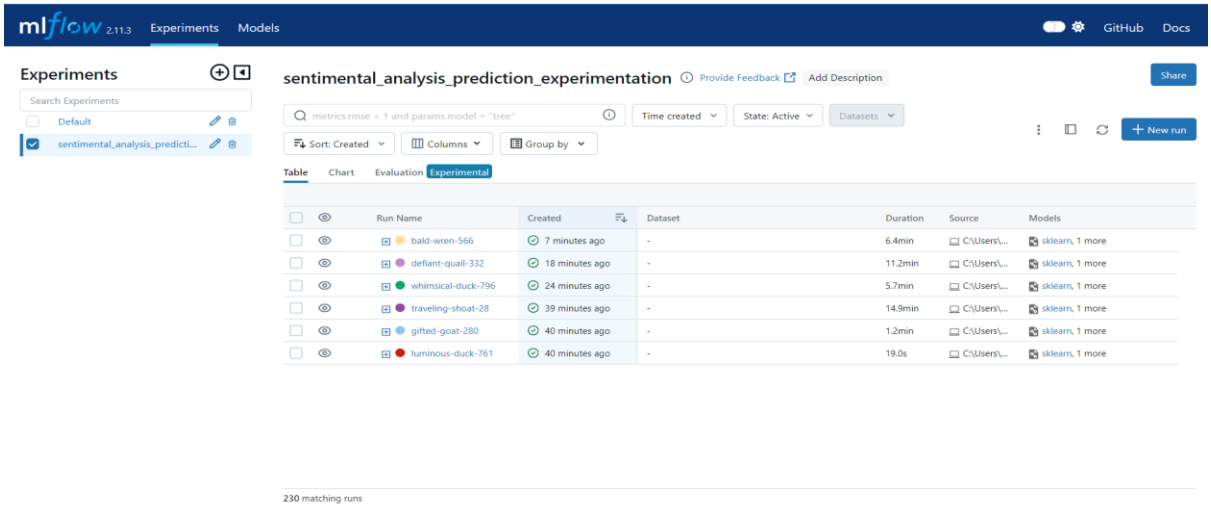
1. Create Virtual Environment
   python -m venv .env

2. Activate Virtual Environment
   .env\Scripts\activate

3. Install MLFlow
   pip install mlflow

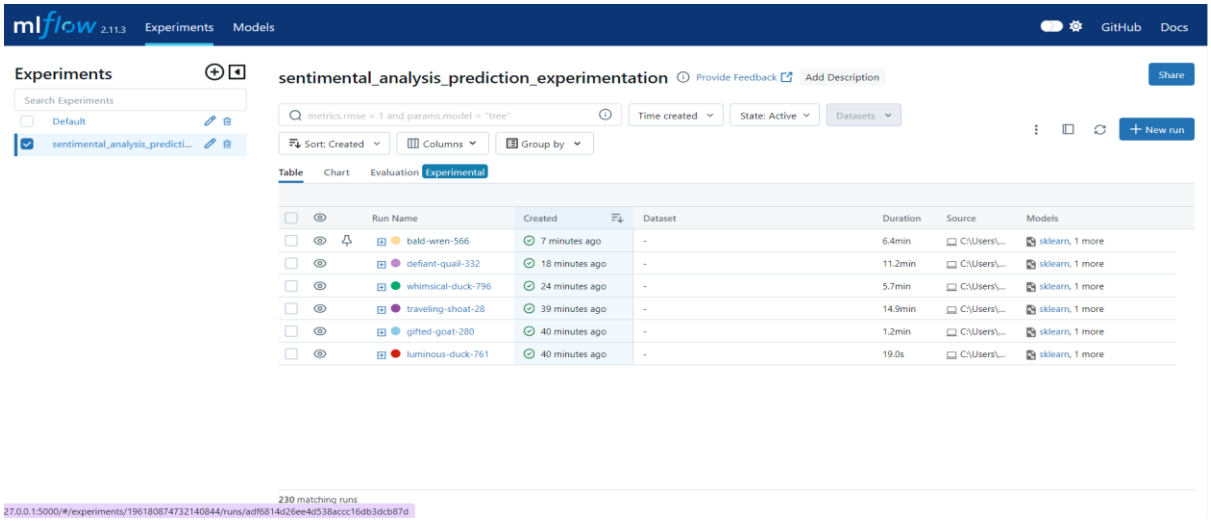
4. Install Setuptools (Optional)
   pip install setuptools

5. To open MLFlow Dashboard
   mlflow ui

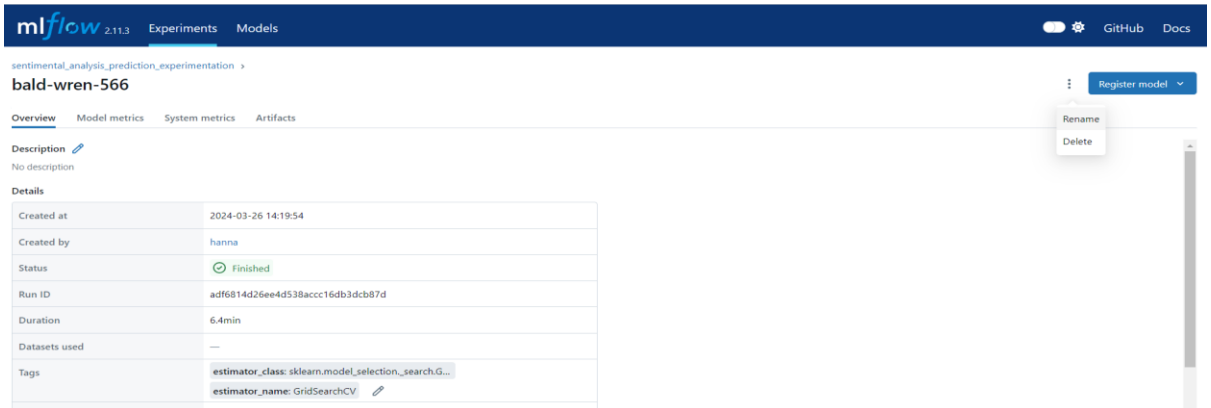


Customizing MLFlow

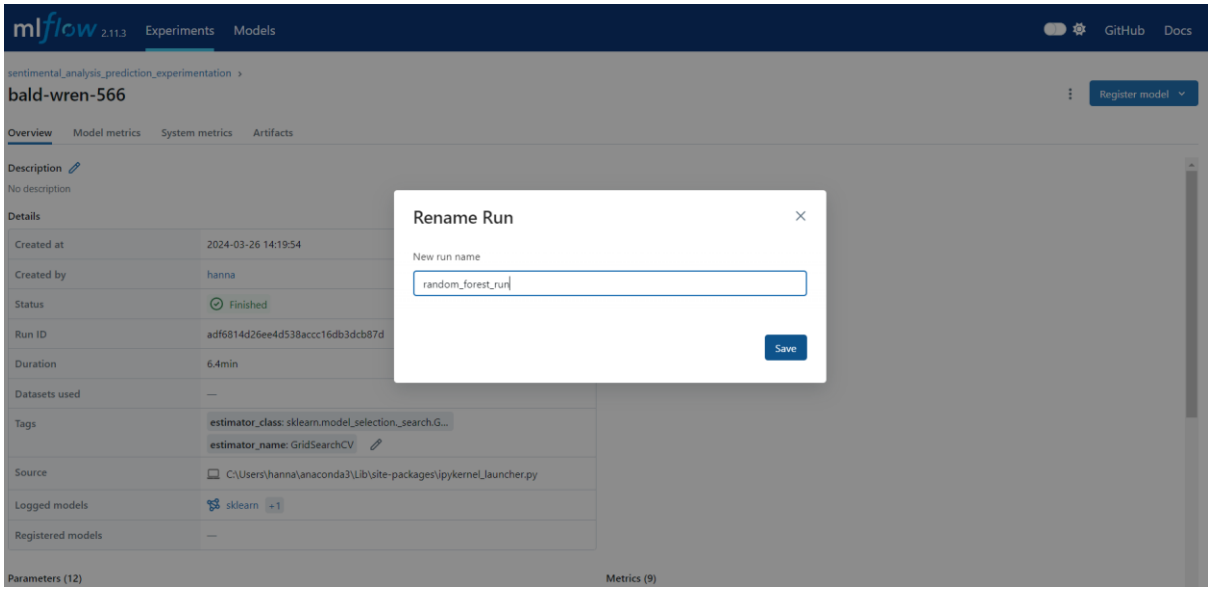
1. Click on Run Name “**bald-wren-566**”



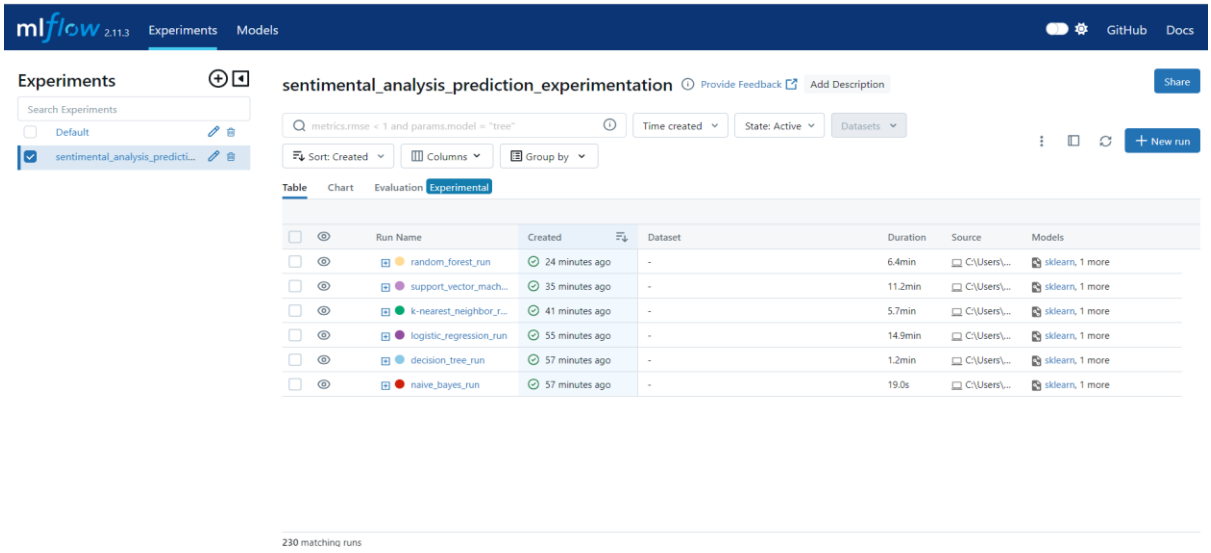
2. Click on 3 dots → Select Rename



3. Enter new run name → click Save

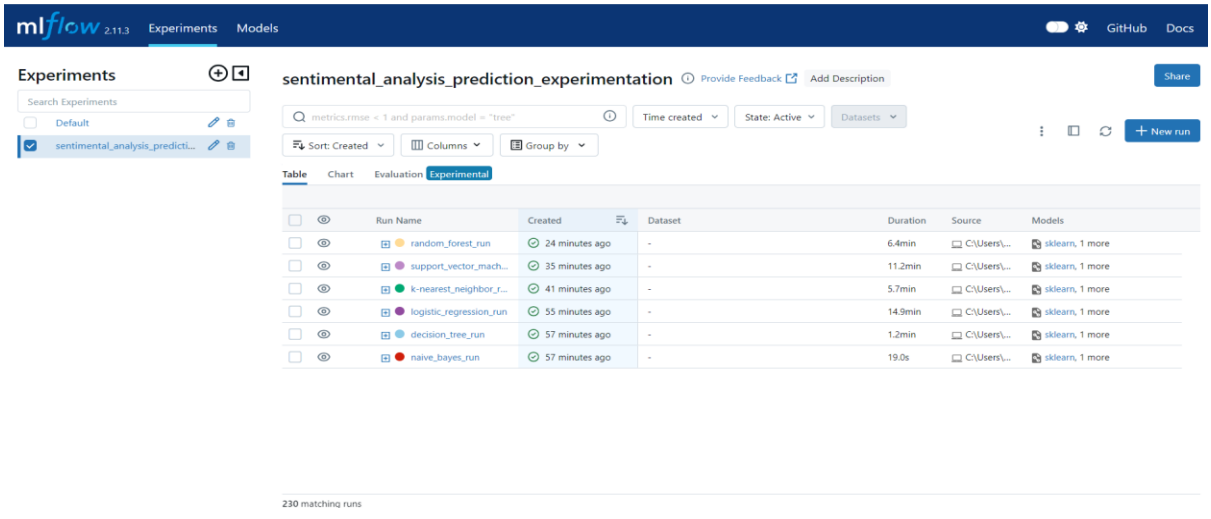


4. Repeat step 1 to 3 for remaining runs

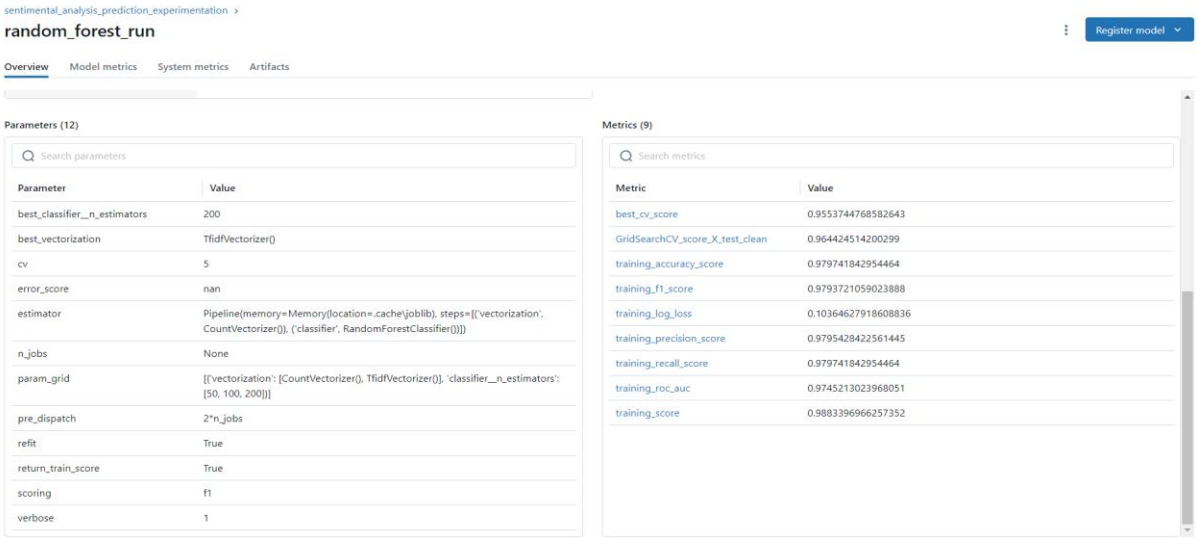


Log Parameters, Metrics and Artifacts

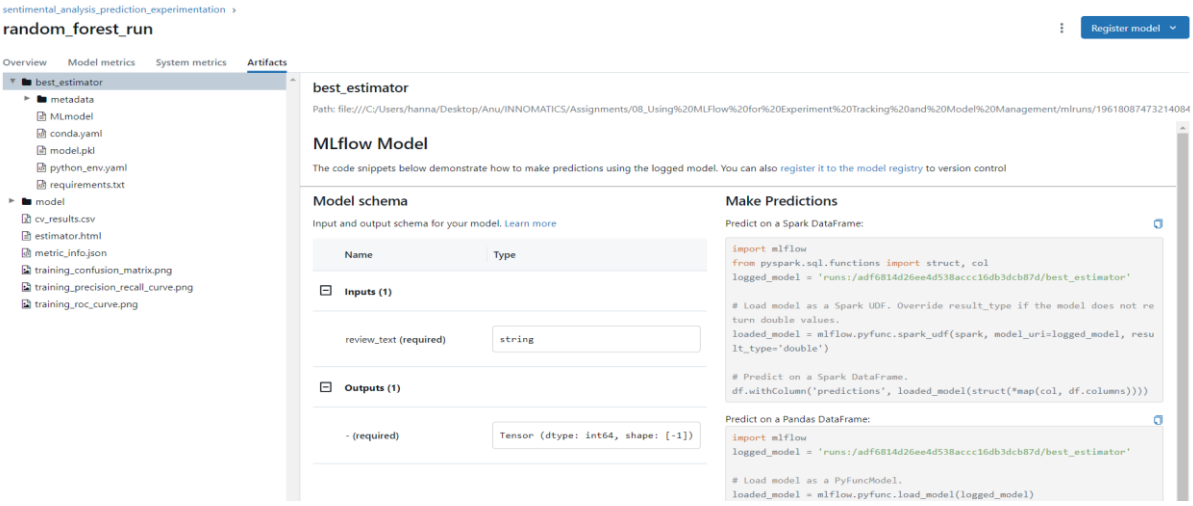
1. Click on Run name



2. Click on **Overview** Tab and Scroll down to View Log **Parameters** and **Metrics** of that run

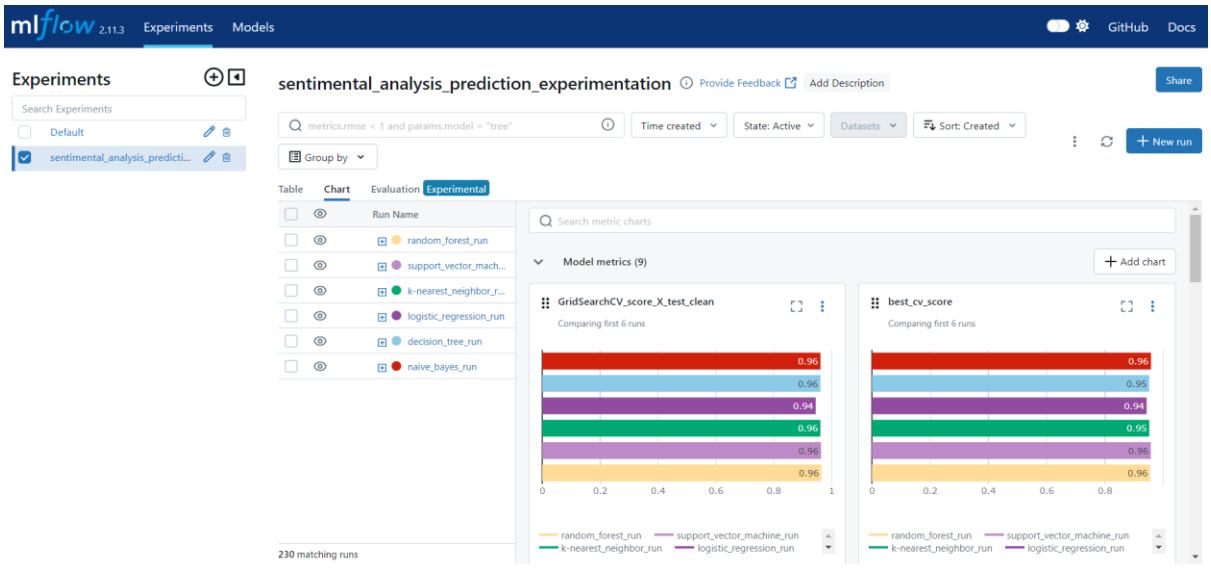


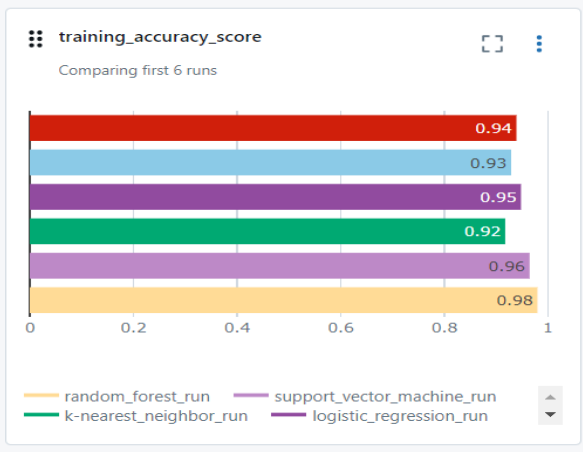
3. Click on **Artifacts** Tab to view the artifacts which includes best_estimator, MLflow Model etc.

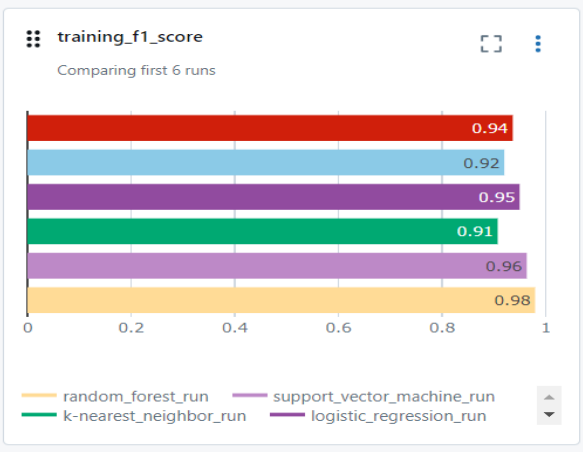


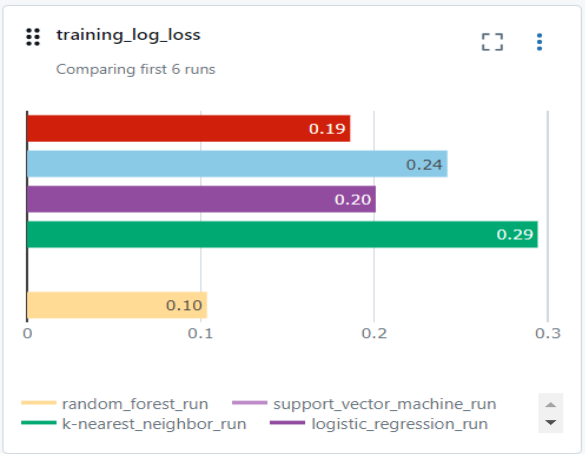
Metric Plots

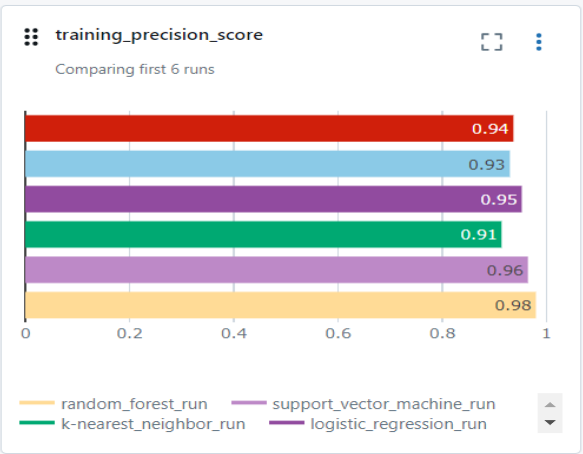
1. Click on **Chart** Tab to view Model **Metric** Plots

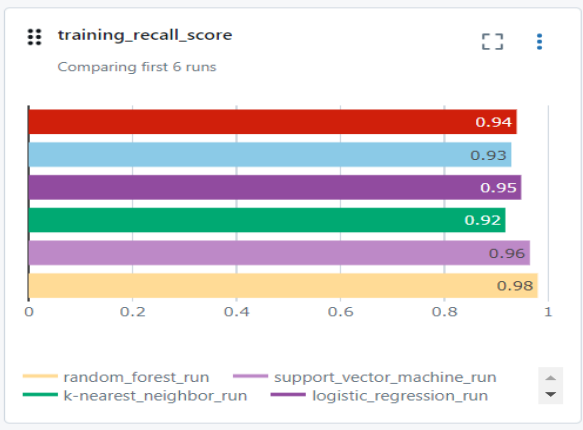


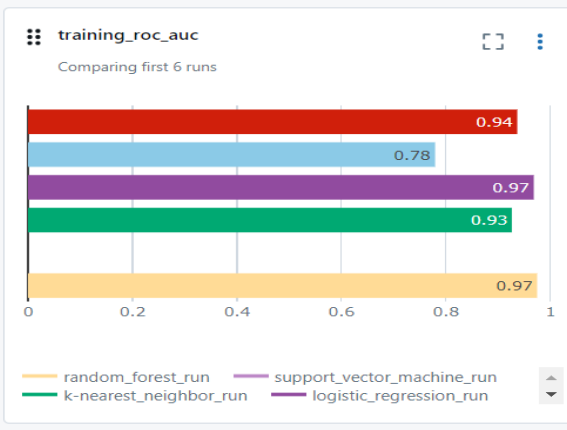

training_accuracy_score
Comparing first 6 runs
0.94
0.93
0.95
0.92
0.96
0.98
0
0.2
0.4
0.6
0.8
1
random_forest_run
support_vector_machine_run
k-nearest_neighbor_run
logistic_regression_run

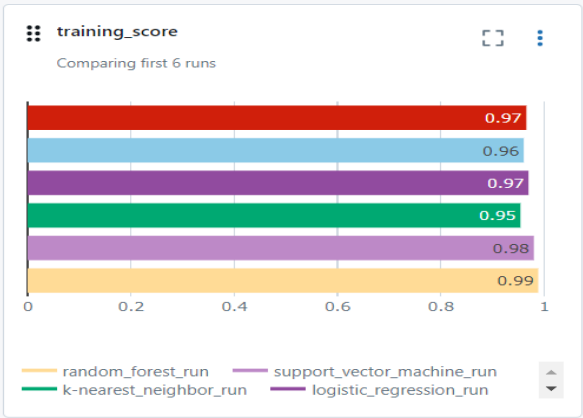

training_f1_score
Comparing first 6 runs
0.94
0.92
0.95
0.91
0.96
0.98
0
0.2
0.4
0.6
0.8
1
random_forest_run
support_vector_machine_run
k-nearest_neighbor_run
logistic_regression_run


training_log_loss
Comparing first 6 runs
0.19
0.24
0.20
0.29
0.10
0
0.1
0.2
0.3
random_forest_run
support_vector_machine_run
k-nearest_neighbor_run
logistic_regression_run


training_precision_score
Comparing first 6 runs
0.94
0.93
0.95
0.91
0.96
0.98
0
0.2
0.4
0.6
0.8
1
random_forest_run
support_vector_machine_run
k-nearest_neighbor_run
logistic_regression_run

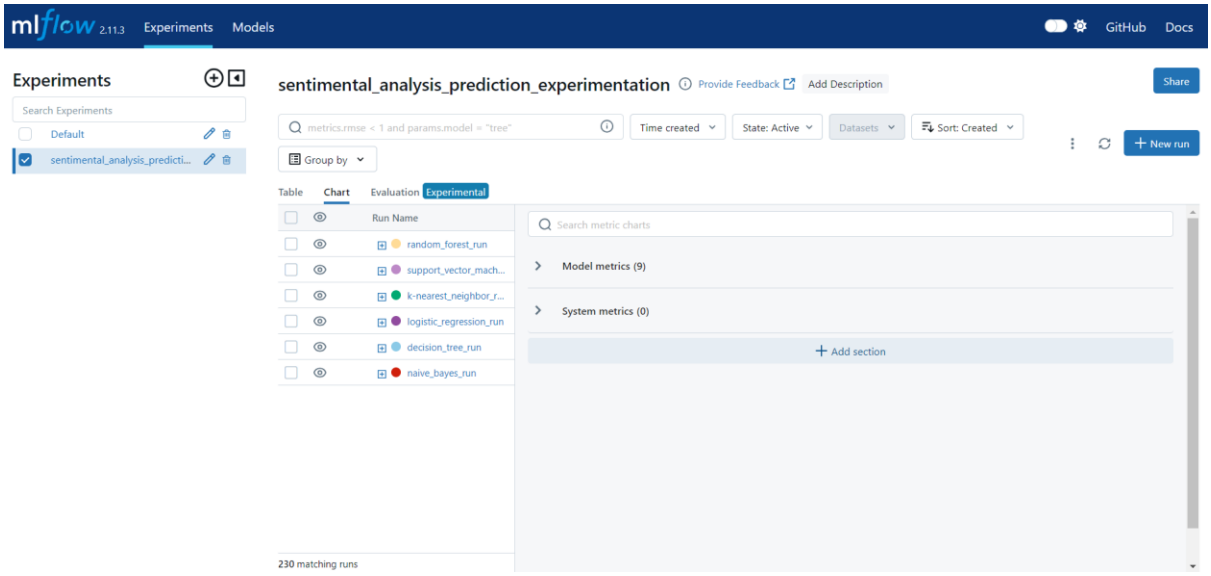

training_recall_score
Comparing first 6 runs
0.94
0.93
0.95
0.92
0.96
0.98
0
0.2
0.4
0.6
0.8
1
random_forest_run
support_vector_machine_run
k-nearest_neighbor_run
logistic_regression_run


training_roc_auc
Comparing first 6 runs
0.94
0.78
0.97
0.93
0.97
0
0.2
0.4
0.6
0.8
1
random_forest_run
support_vector_machine_run
k-nearest_neighbor_run
logistic_regression_run

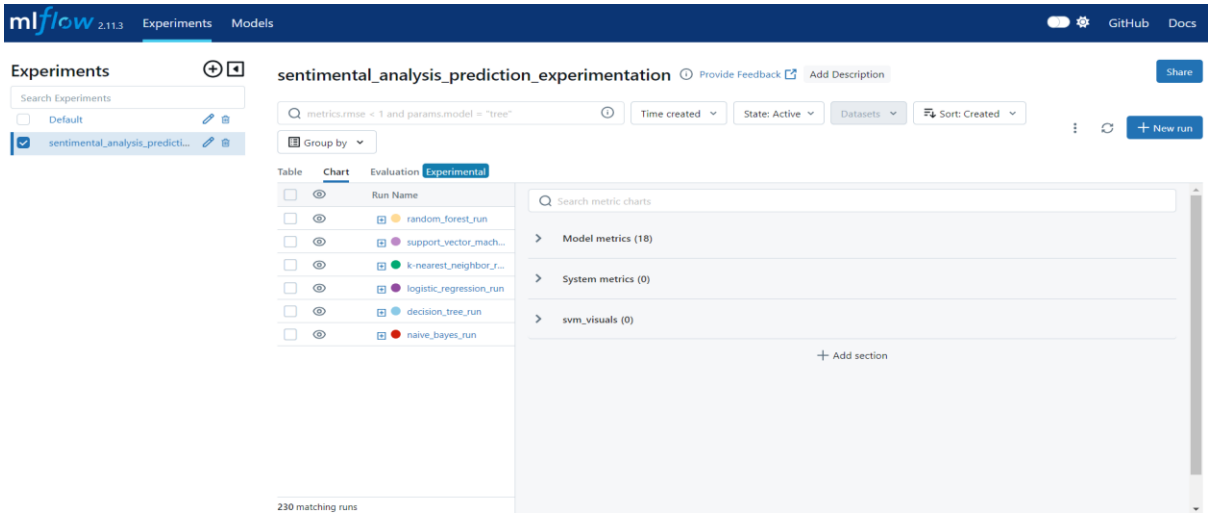

training_score
Comparing first 6 runs
0.97
0.96
0.97
0.95
0.98
0.99
0
0.2
0.4
0.6
0.8
1
random_forest_run
support_vector_machine_run
k-nearest_neighbor_run
logistic_regression_run

Hyperparameter Plots

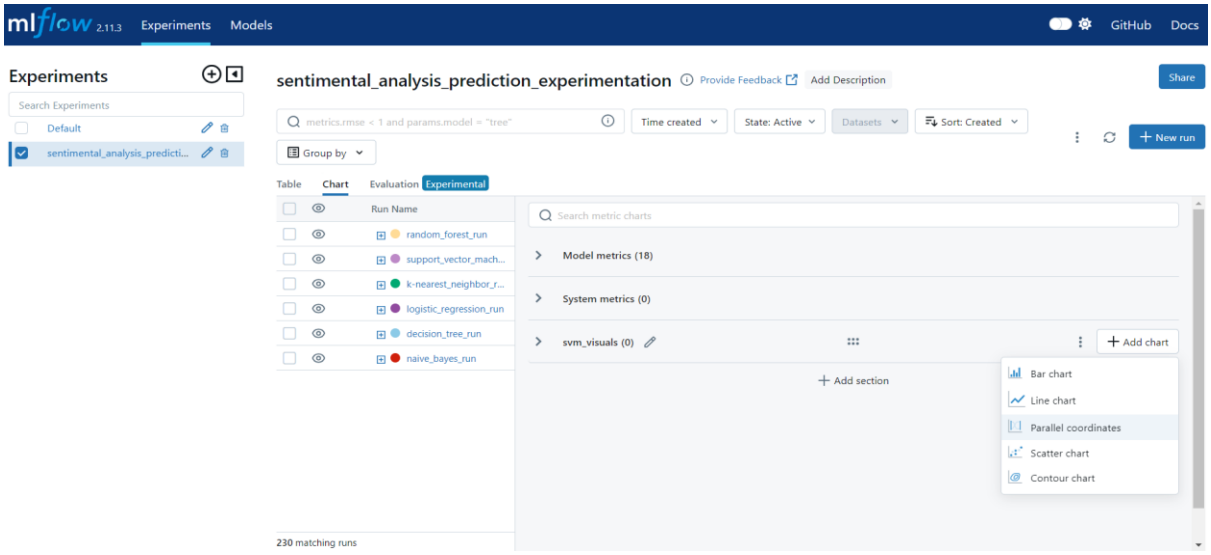
1. Click on **Add Section**



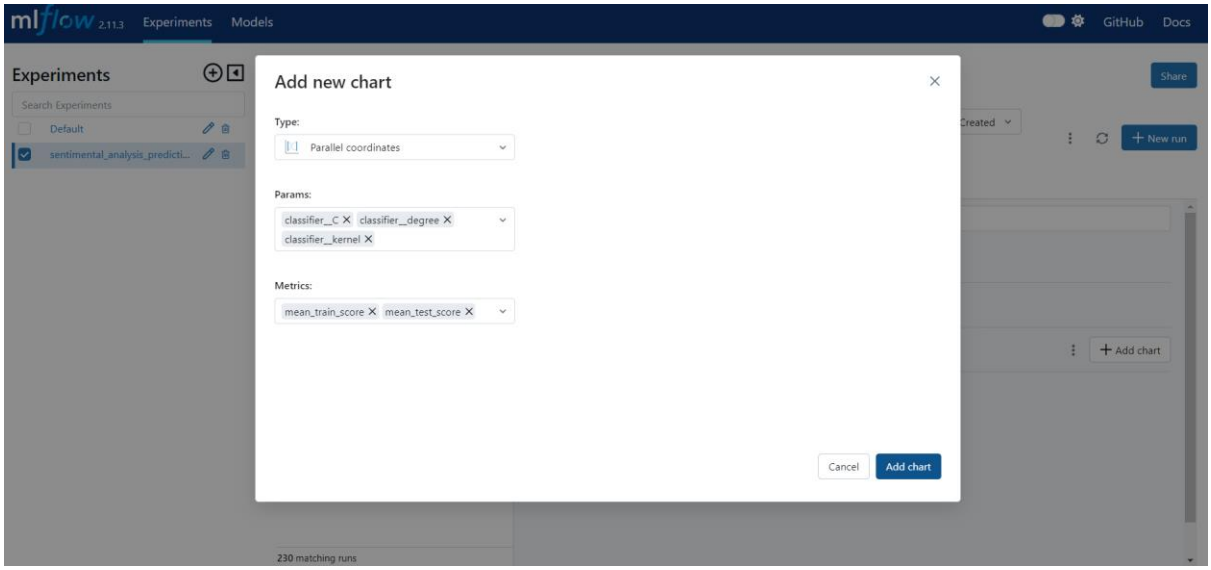
2. Enter the **Name** to the section



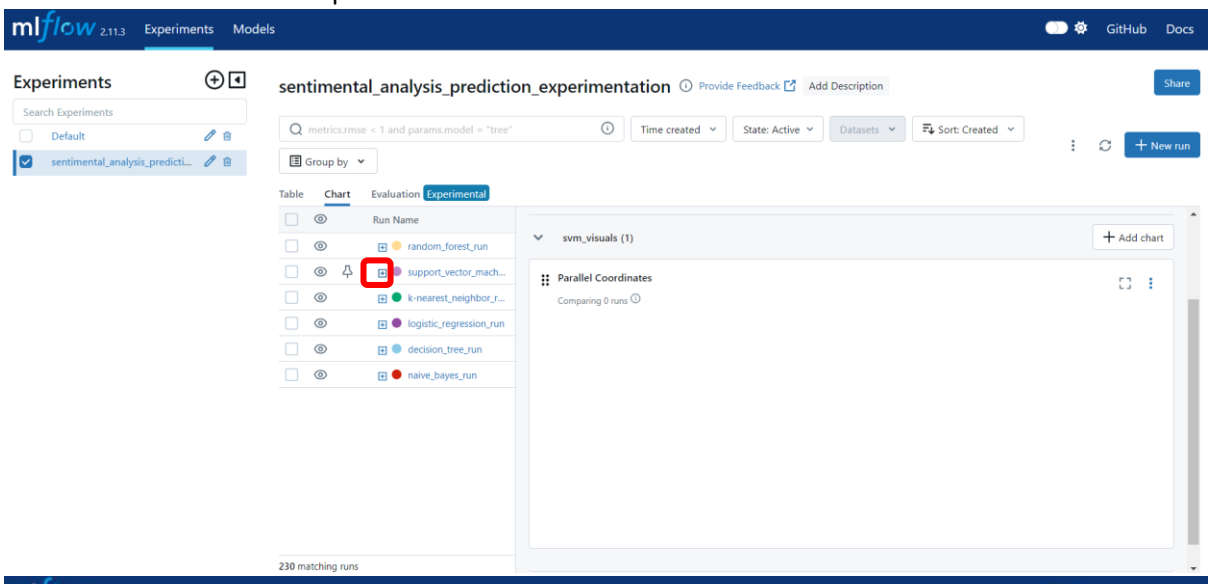
3. Click on **Add chart** → Select **Parallel coordinates**

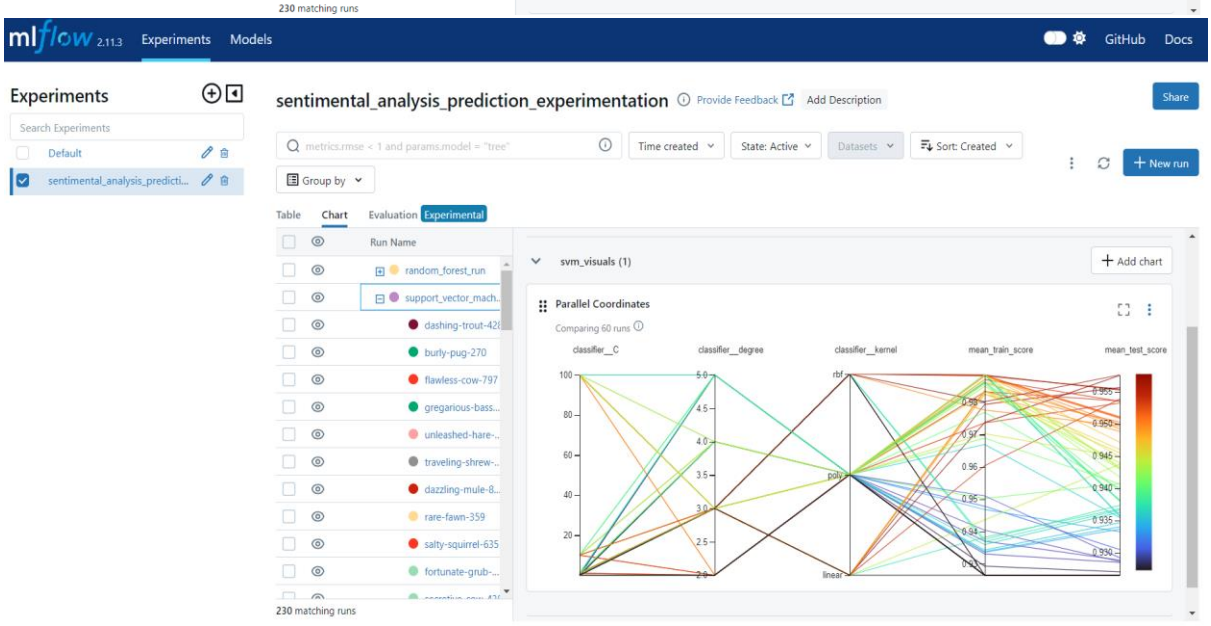


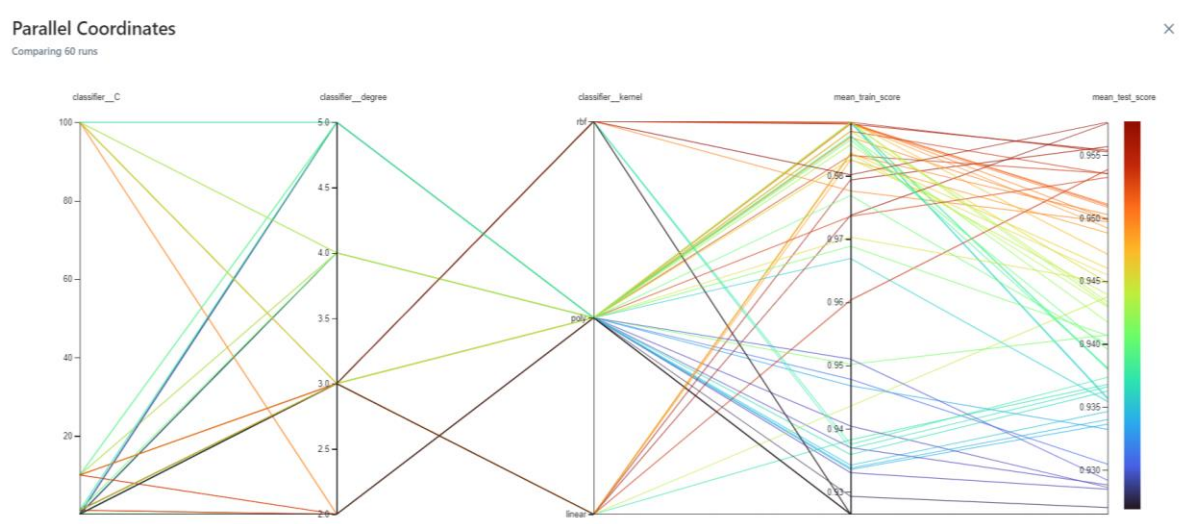
4. Select the **Params** and **Metrics** from drop-down → click on **Add chart**

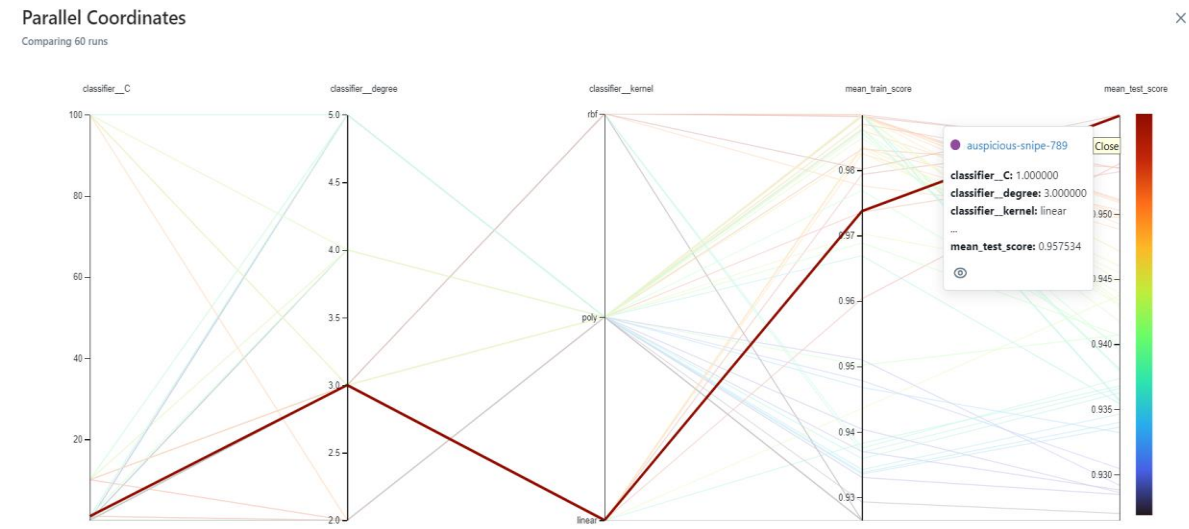


5. Click on '+' beside specified run name under **Chart** Tab



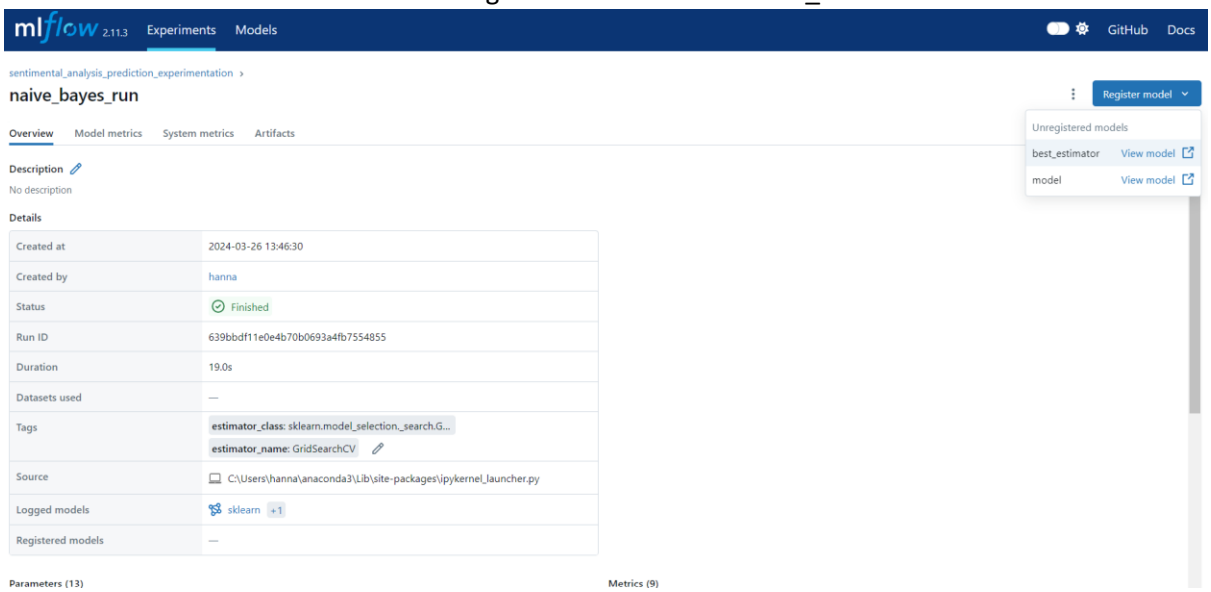




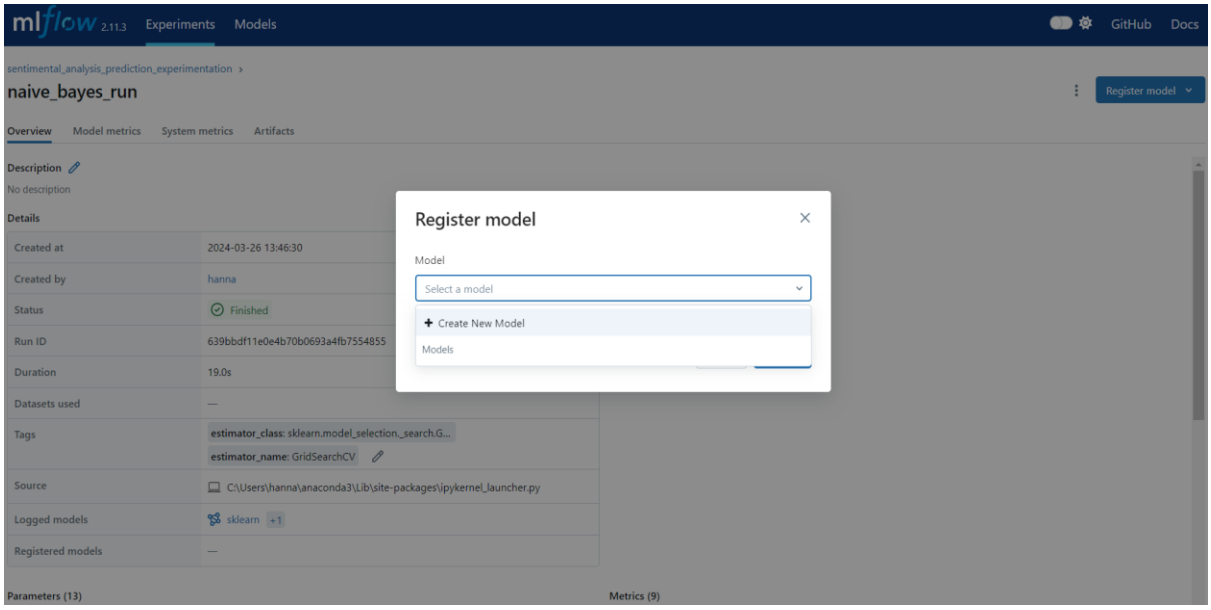


Registering Models

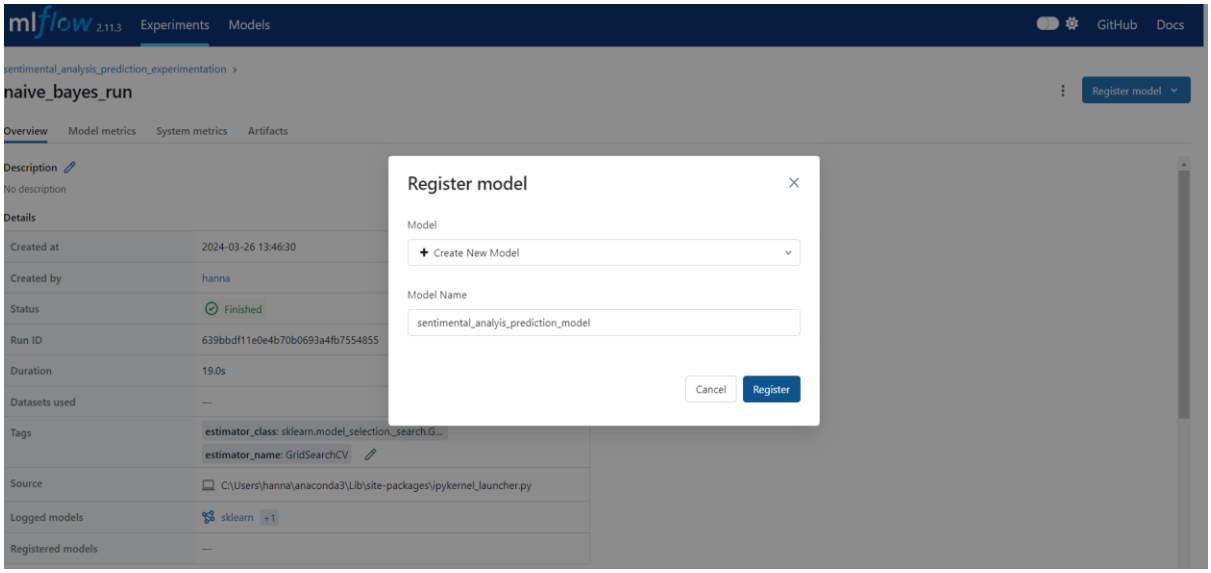
1. Click on run name → Click on Register model → Select best_estimator



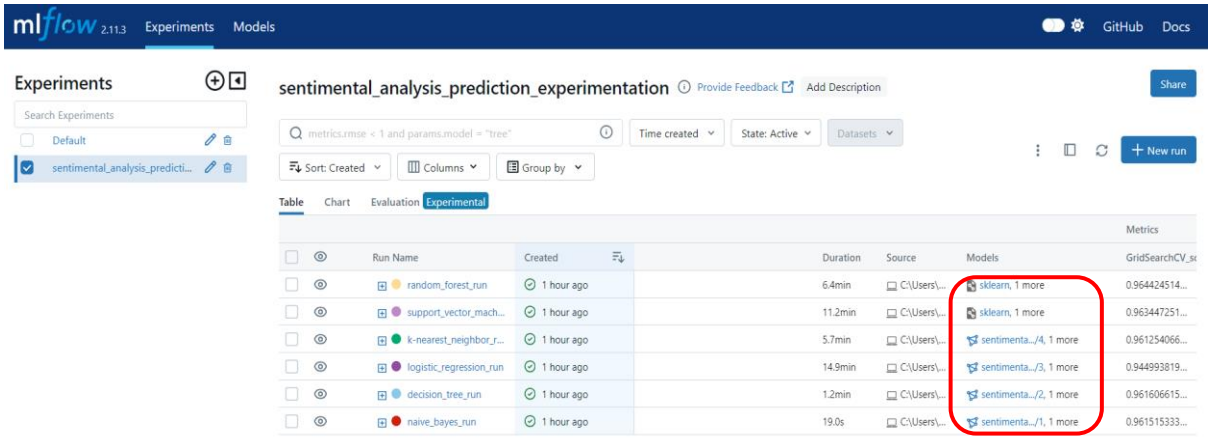
2. Click on **+ Create New Model** (only for first time as I don't have any Registered Model)



3. Enter Model Name → click Register

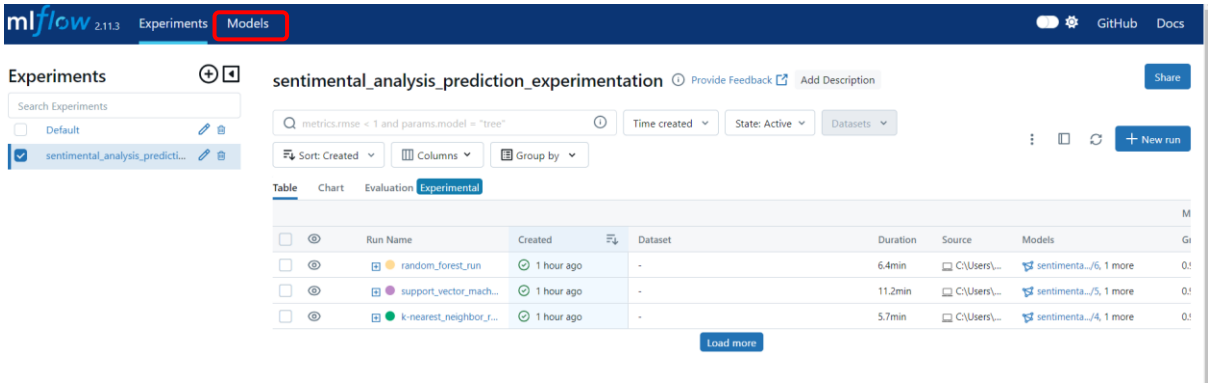


We can view under Models column which models are registered and its version.

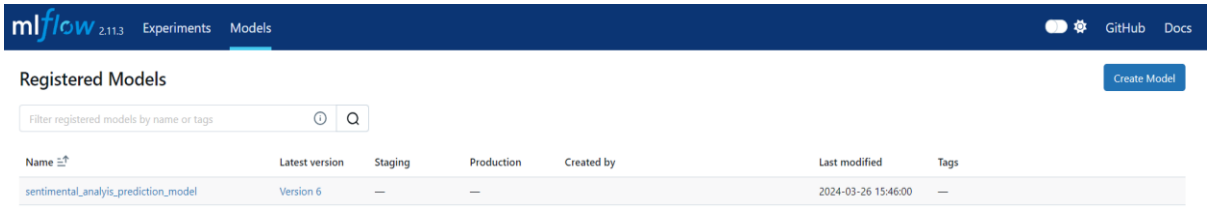


Managing and Tagging the Models

1. Click on ‘Models’ Tab on the top of the dashboard.

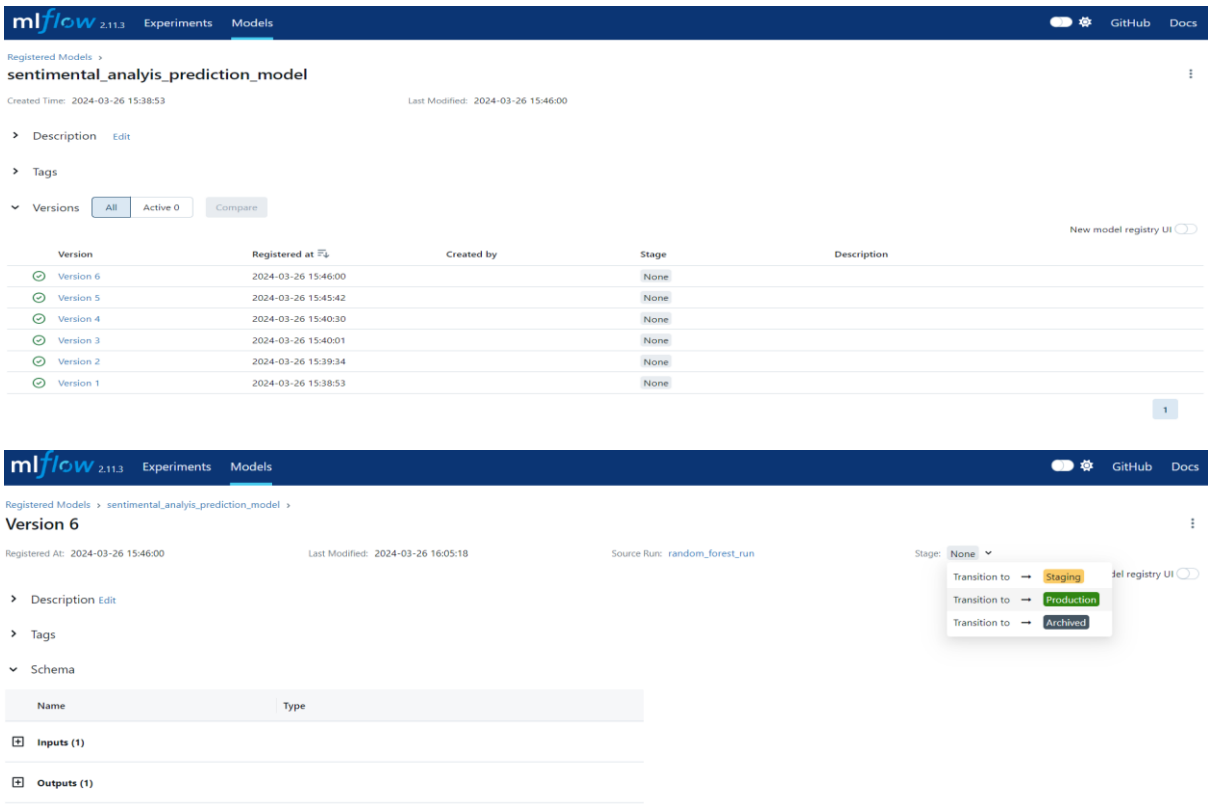


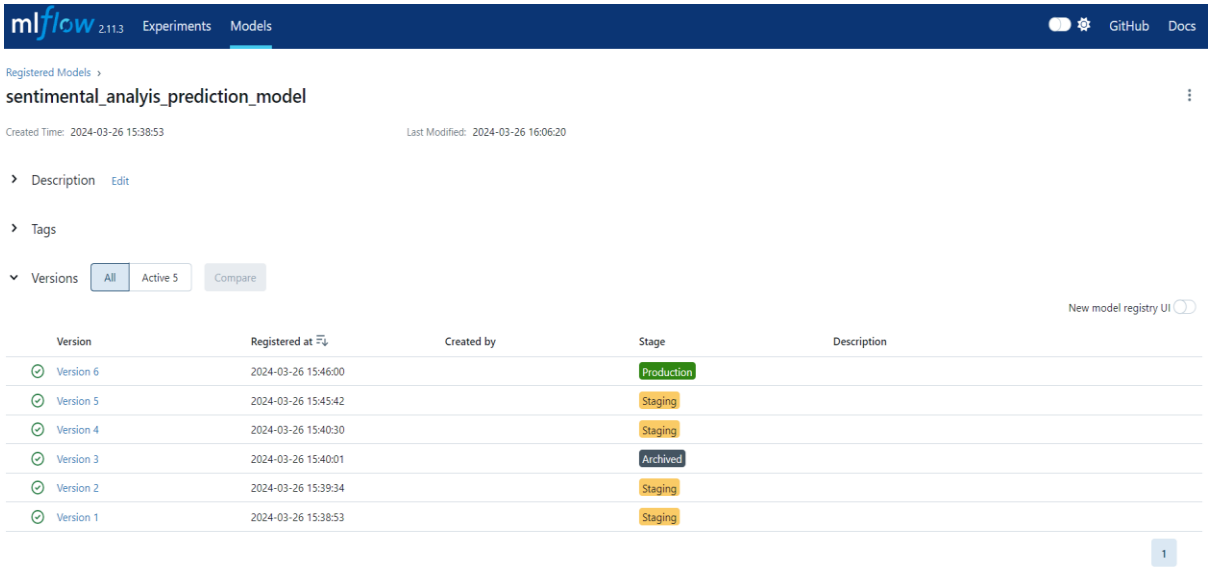
2. Click on the Registered model name i.e., sentimental_analysis_prediction_model.



3. Choose the version → click on Stage menu, Select the appropriate option as ‘Staging’, ‘Production’ and ‘Archived’.

Archived: These versions are no longer in active use.
Staged: These versions are ready for deployment pending final validation.
Production: These versions are actively serving users in live environments.




mlflow 2.11.3
Experiments
Models
GitHub
Docs
Registered Models ›
sentimental_analyis_prediction_model
Created Time: 2024-03-26 15:38:53
Last Modified: 2024-03-26 16:06:20
Description
Edit
Tags
Versions
All
Active 5
Compare
New model registry UI
Version
Registered at
Created by
Stage
Description
Version 6
2024-03-26 15:46:00
Production
Version 5
2024-03-26 15:45:42
Staging
Version 4
2024-03-26 15:40:30
Staging
Version 3
2024-03-26 15:40:01
Archived
Version 2
2024-03-26 15:39:34
Staging
Version 1
2024-03-26 15:38:53
Staging
1