Project on

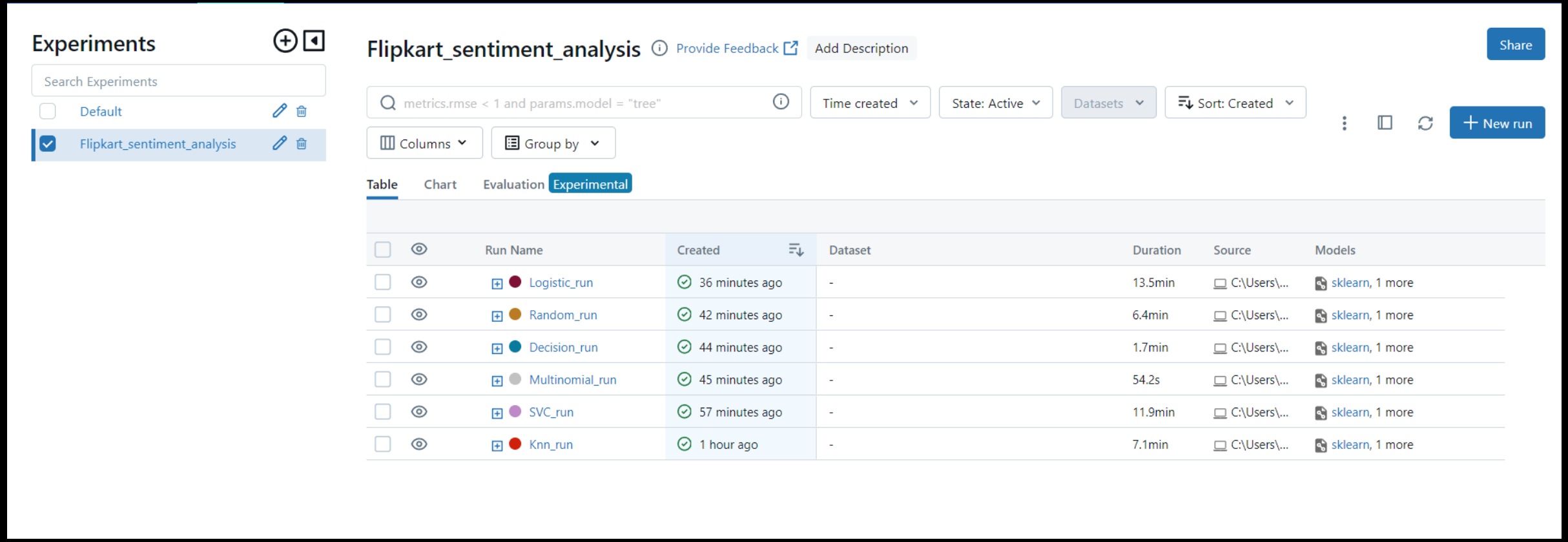
Sentiment Analysis of Real-time Flipkart Product Reviews Using MLflow for Experiment Tracking and Model Management

T. Mahesh Reddy

Intern ID: IN1240834

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.



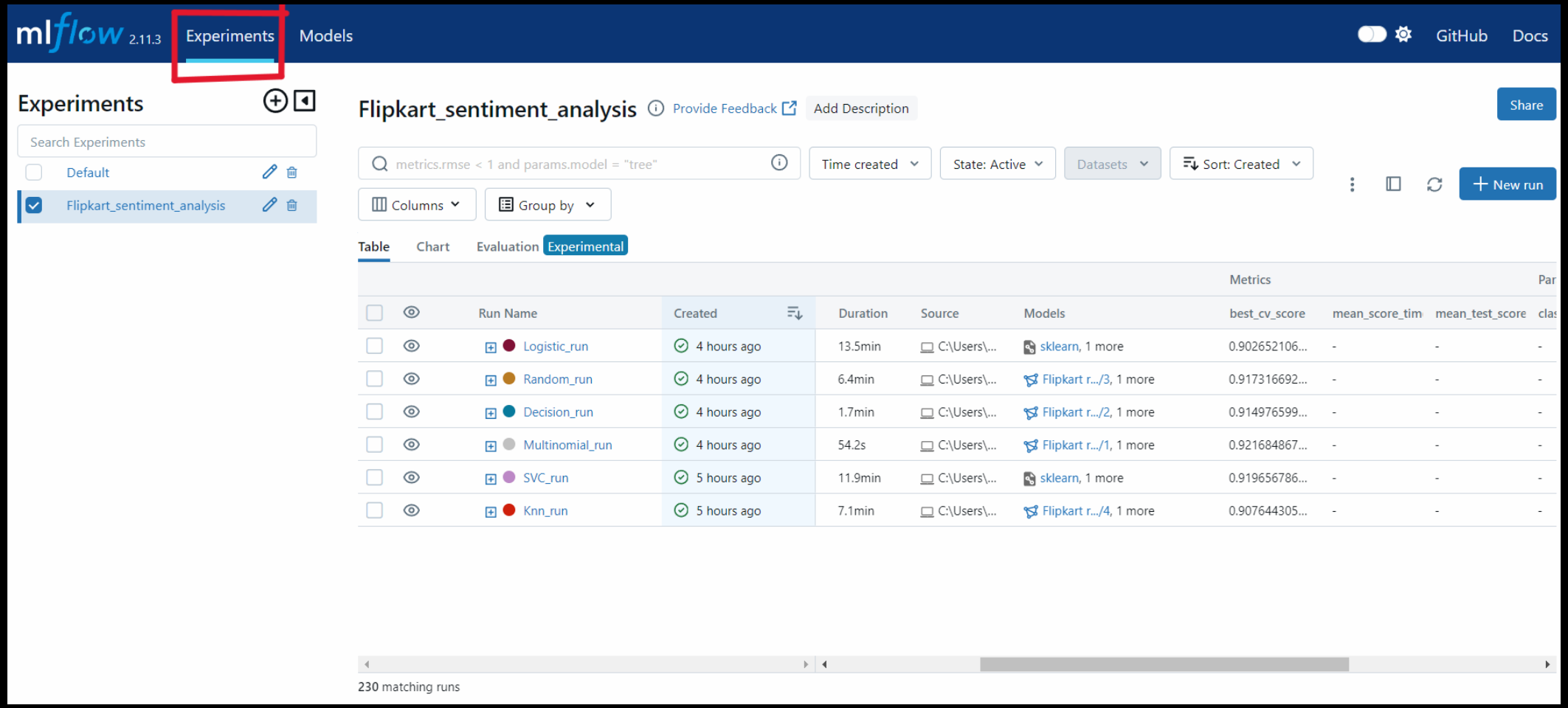
MLFlow

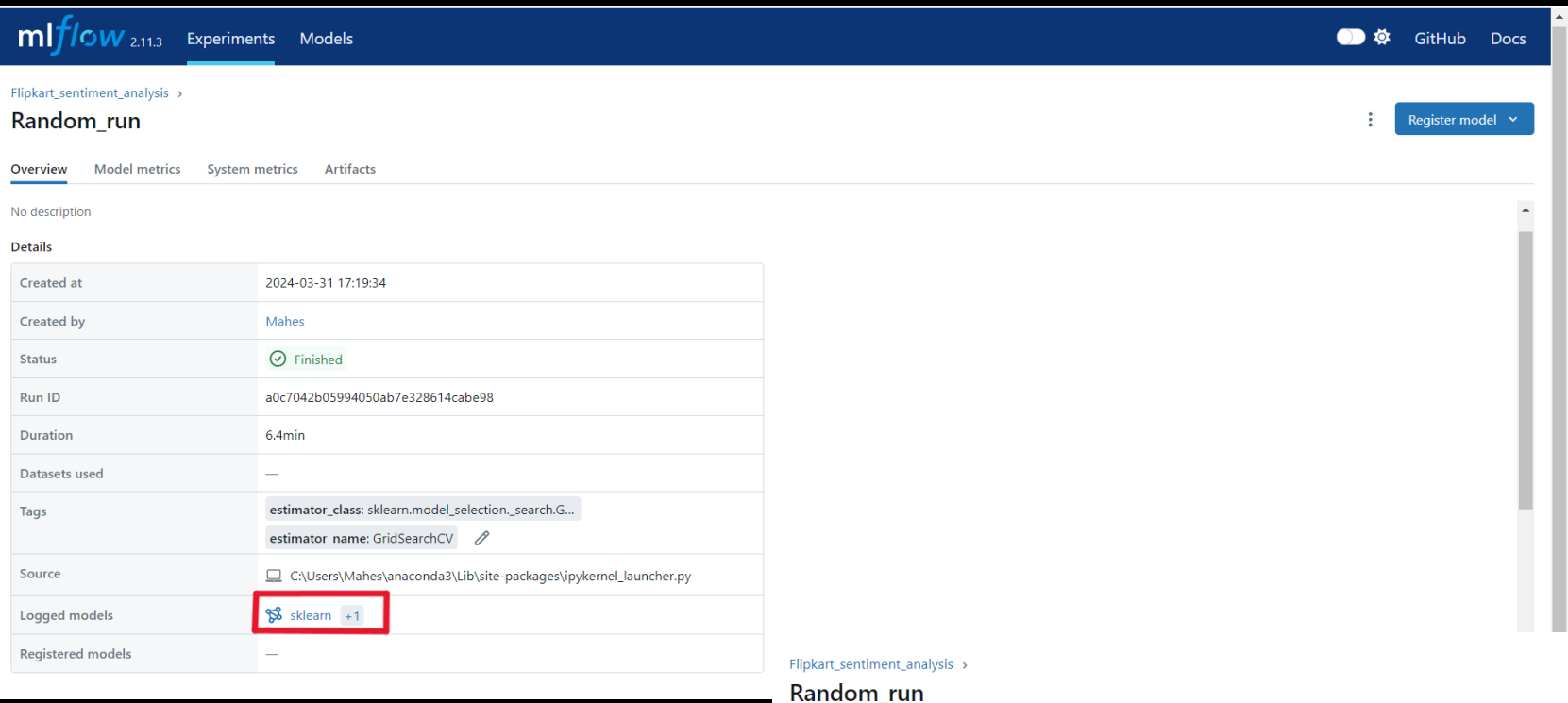
MLflow is an open source platform for managing machine learning workflows. It is used by MLOPs teams and data scientists. MLflow has main components.

- **The experiment tracking** allows you to record machine model training sessions (called runs) and run queries using Python.
- **The model management** provides a standard unit for packaging and reusing machine learning models.
- **The deployment** is a standard format that packages a machine learning model with its metadata, such as dependencies and inference schema.

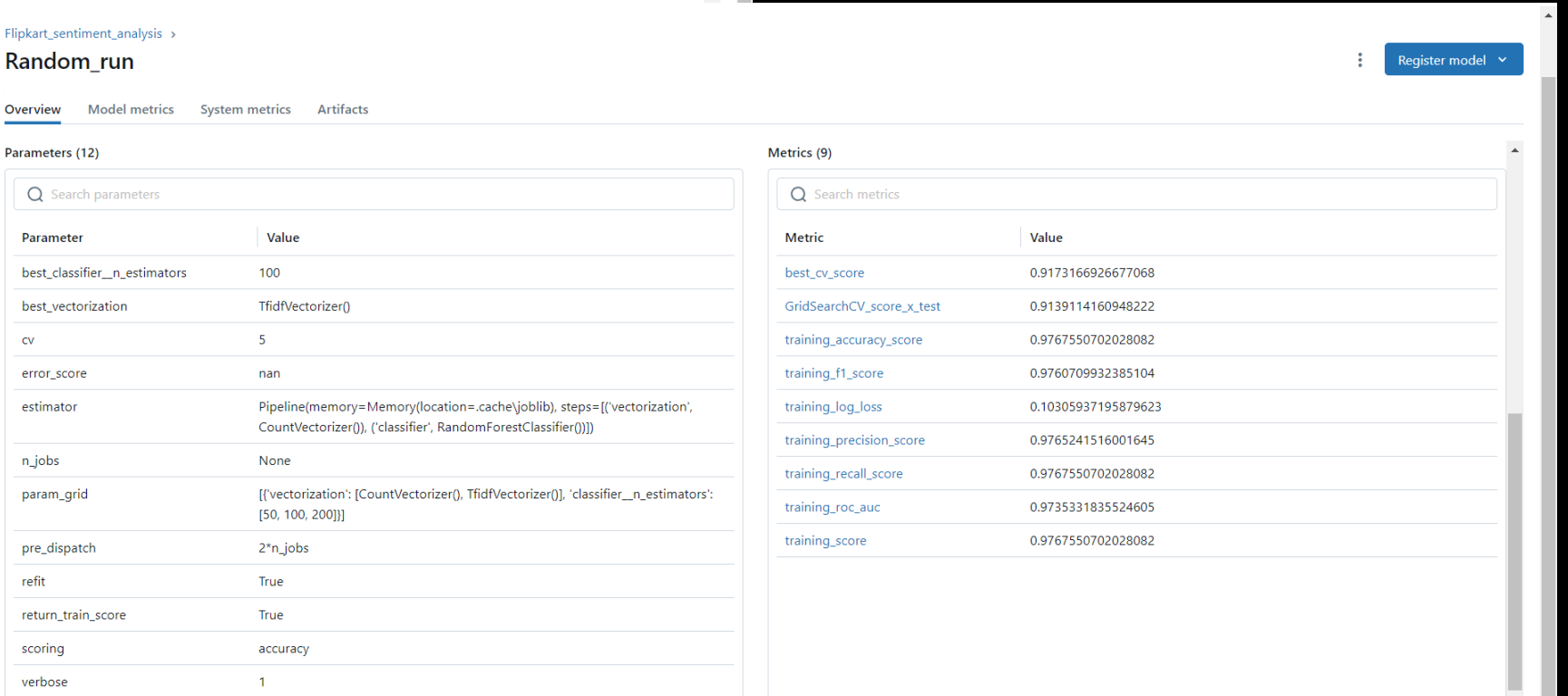
MLFlow Dashboard

- The runs were executed using different models, such as Logistic Regression, Random forest, Decision Tree, Multinomial Naive Bayes, Support Vector Classification (SVC), and K-Nearest Neighbors (Knn).



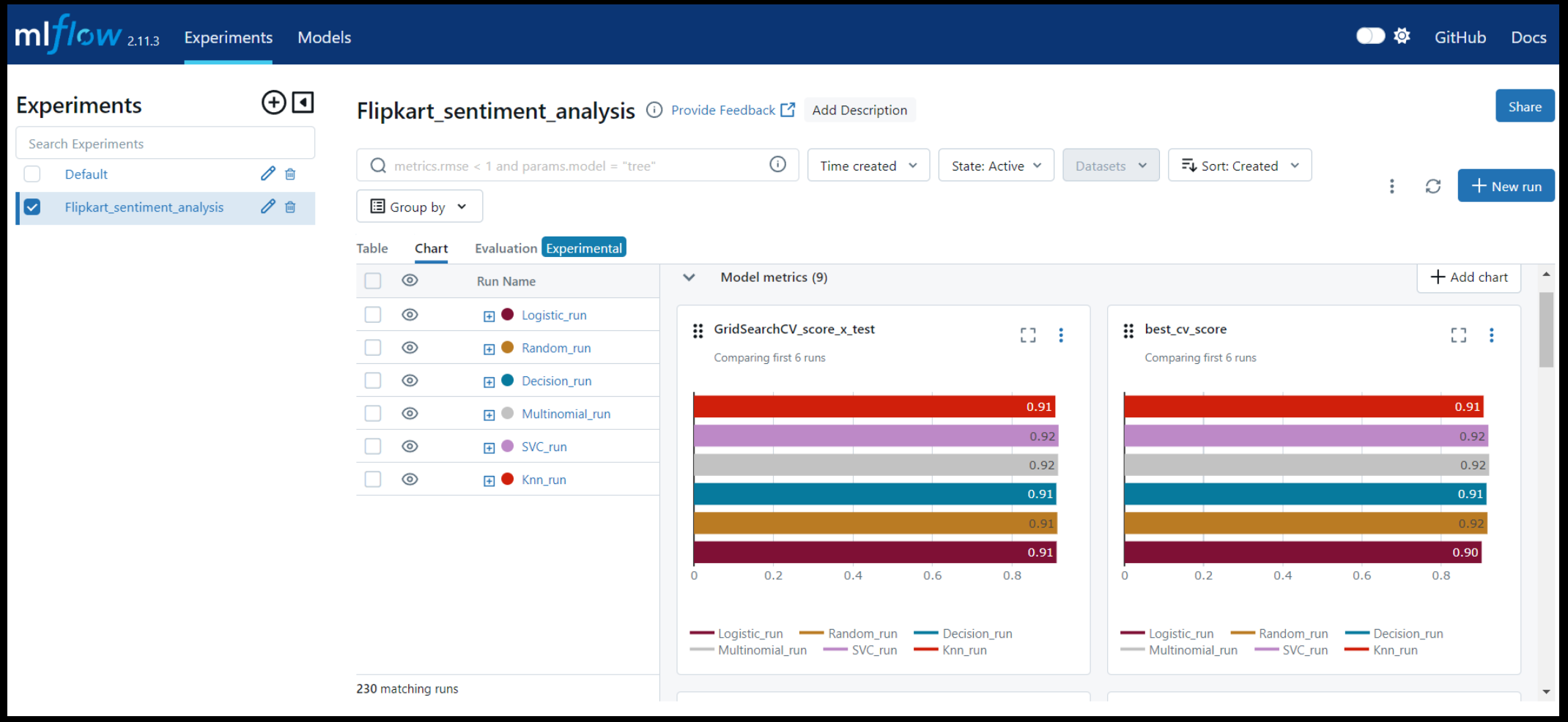


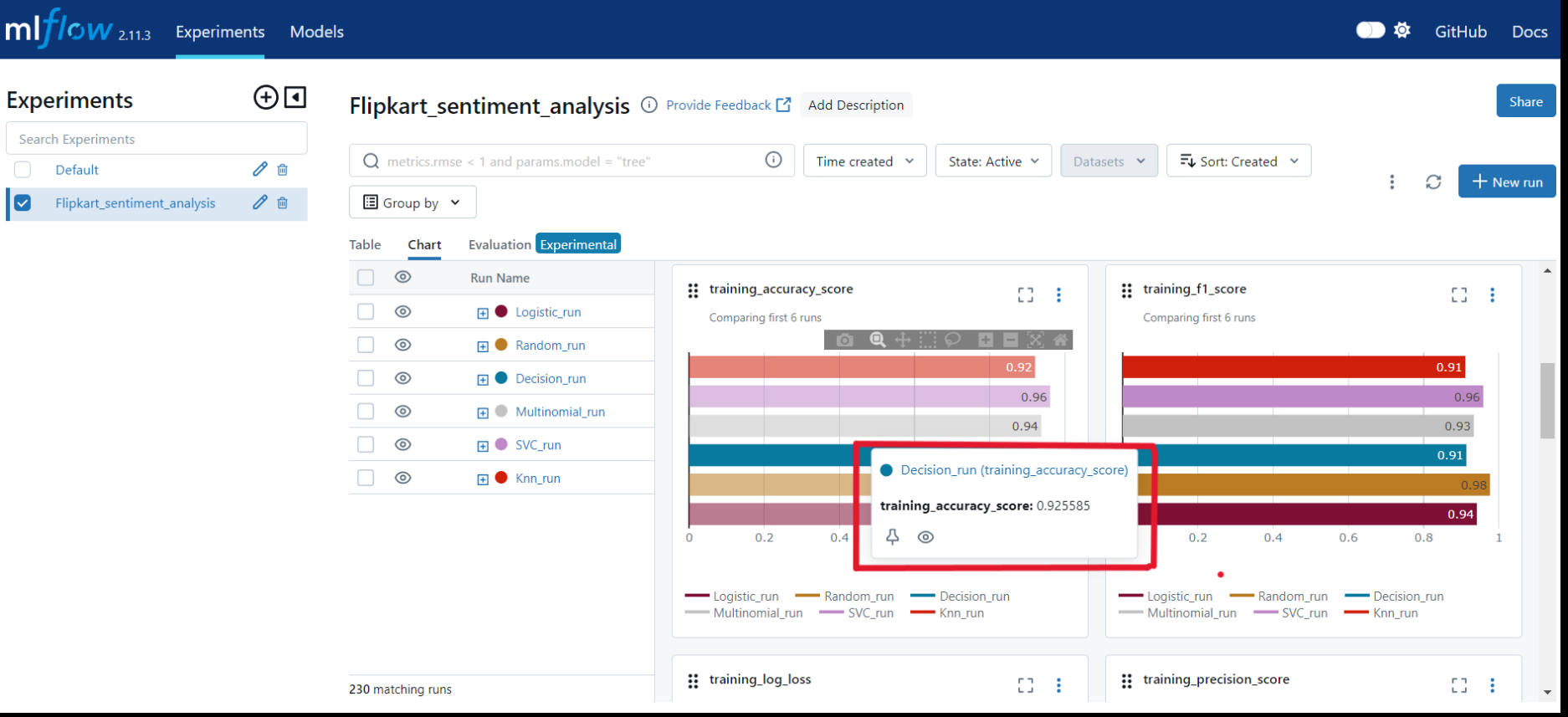
The "Random_run" has no description, and there are no logged or registered models associated with it. However, the run has a "sklearn" tag with a value of "+1", indicating that it used the scikit-learn library for machine learning



Model Metrics

- The table view shows the run name, model metrics, and a chart view. The user can filter the runs based on various parameters and add charts to visualize the performance of the models.
- Overall, MLflow provides a user-friendly interface to manage and track machine learning experiments, making it easy to compare and evaluate the performance of various models.

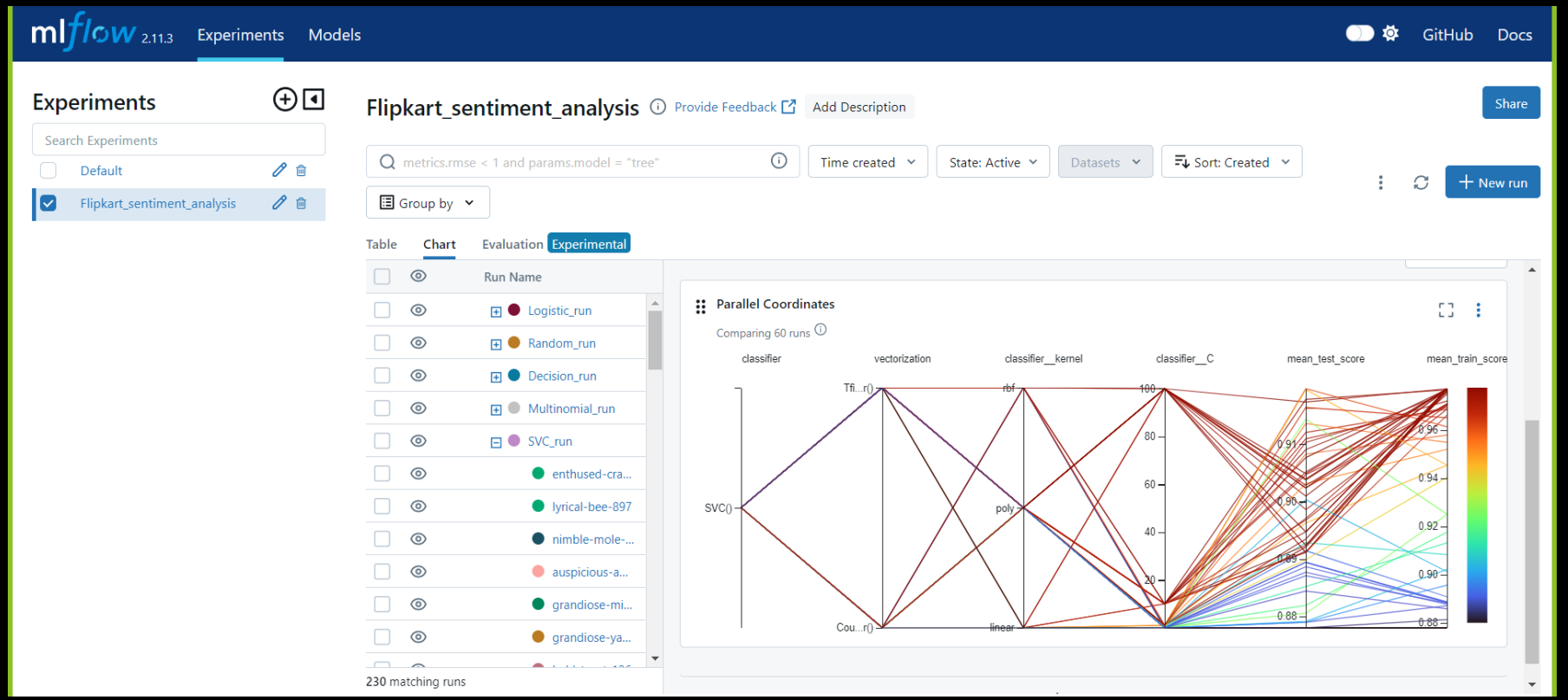


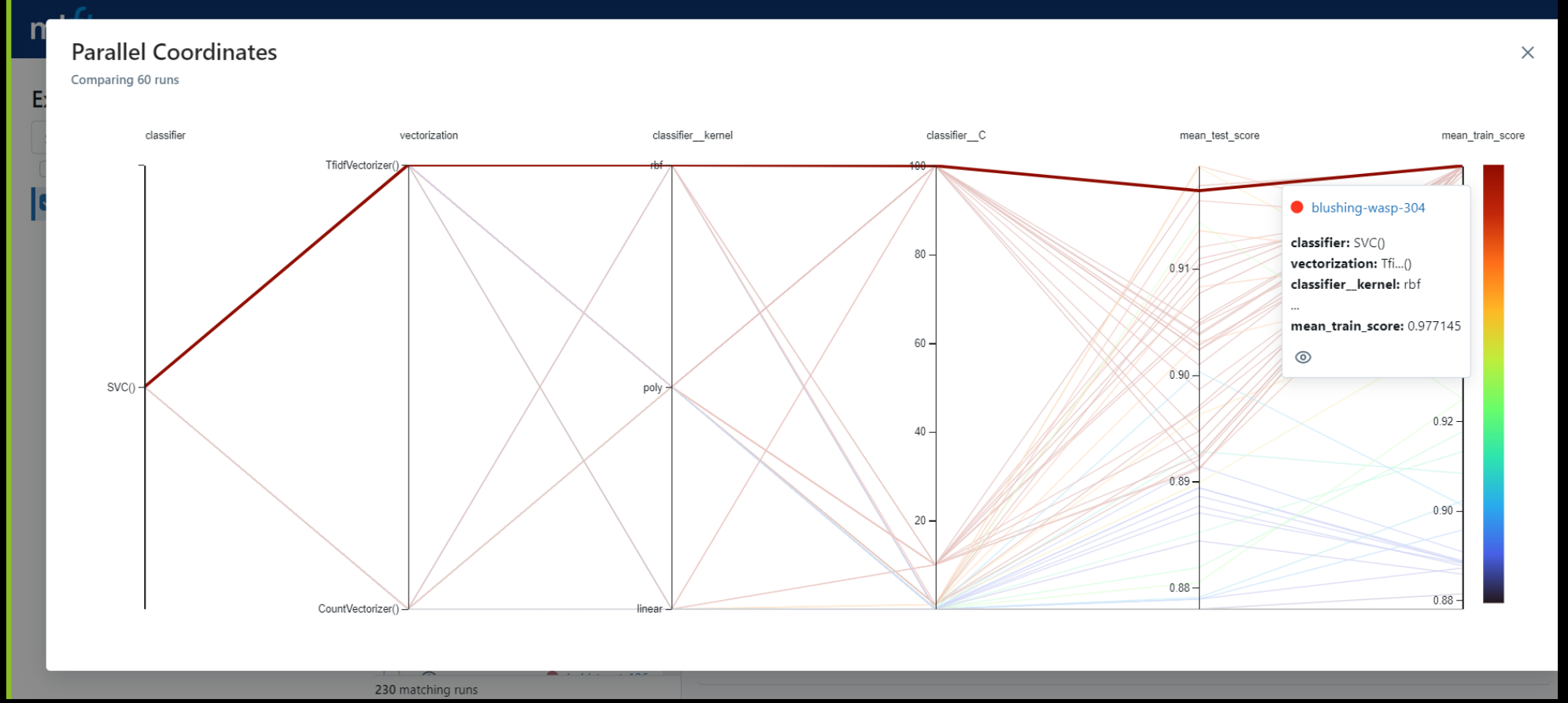


- The table shows the model metrics such as training_accuracy_score, training_f1_score. The user can view the values of these metrics for each run by clicking on the run name.
- For example, clicking on the run name "Logistic_run" shows the metrics values for the Logistic Regression model, such as training_accuracy_score: 0.925585.

Hyperparameters Plots

- The table can be grouped by various parameters, such as classifier, vectorization, and classifier kernel. The user can also sort the table based on various metrics, such as mean_train_score and mean_test_score.
- It shows the run name, time created, state, and datasets used. The user can click on the run name to view more details about the run.



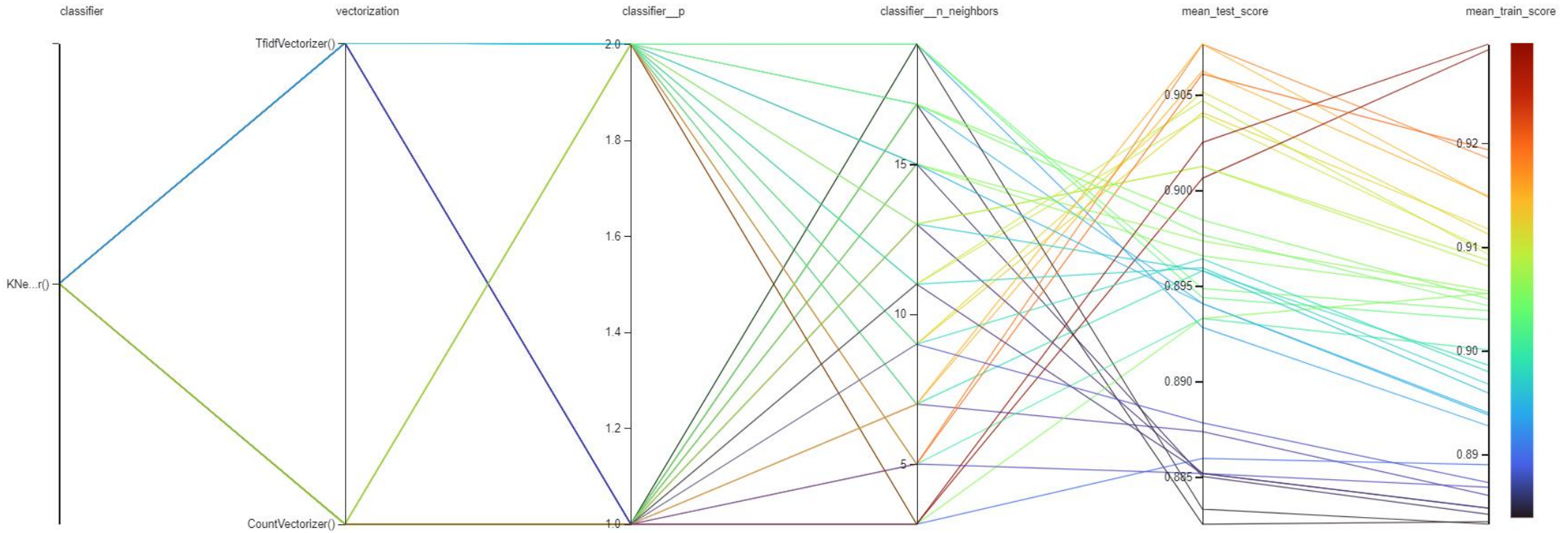


Each horizontal line represents a single run, and the intersection with the vertical axis shows the value of the parameter or metric for that run.

The top horizontal line shows that the classifier used in that run is SVC(), the vectorization is TfidfVectorizer(), and the kernel is rbf. The mean train score is 0.977145, and the mean test score is 0.91.

Parallel Coordinates

Comparing 36 runs

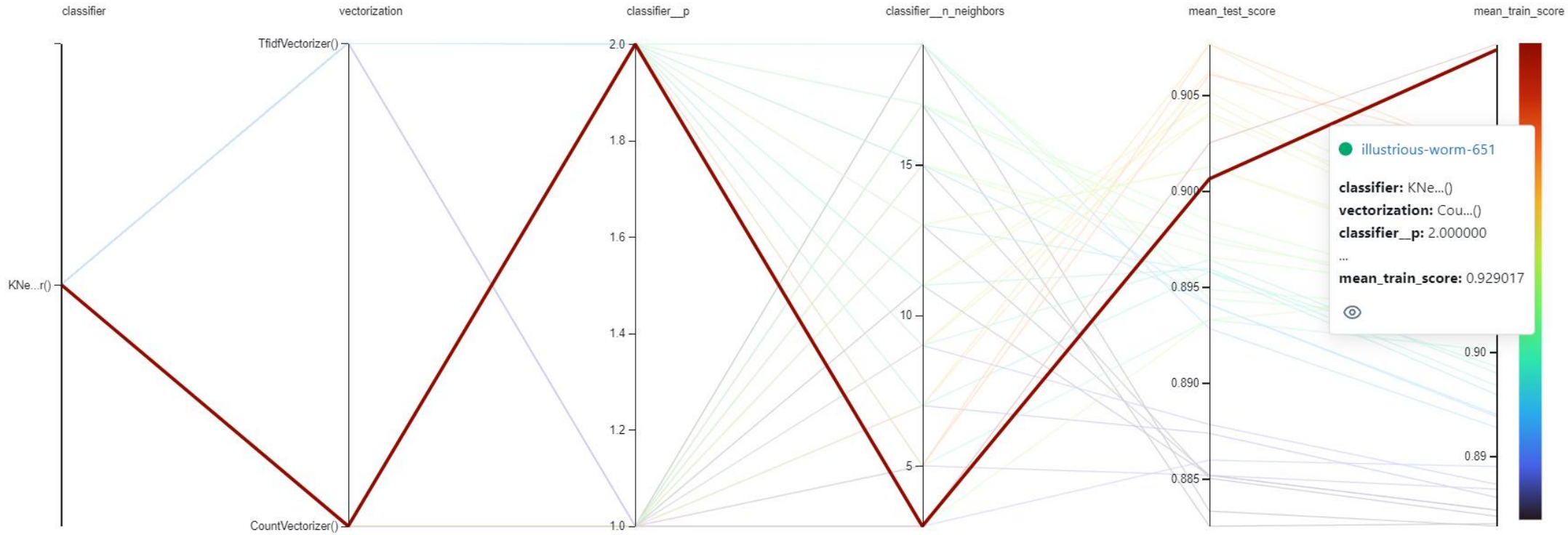


230 matching runs

Add section

Parallel Coordinates

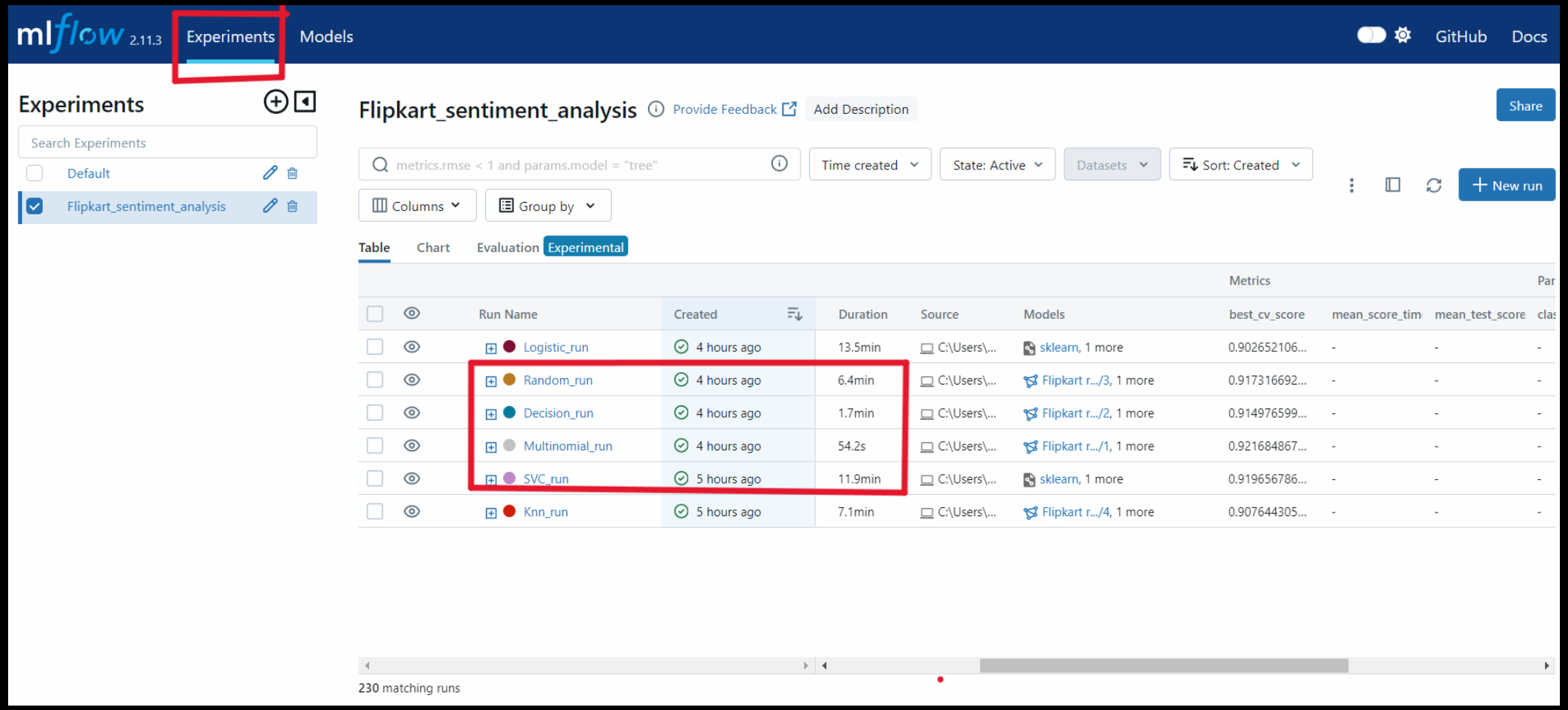
Comparing 36 runs



230 matching runs

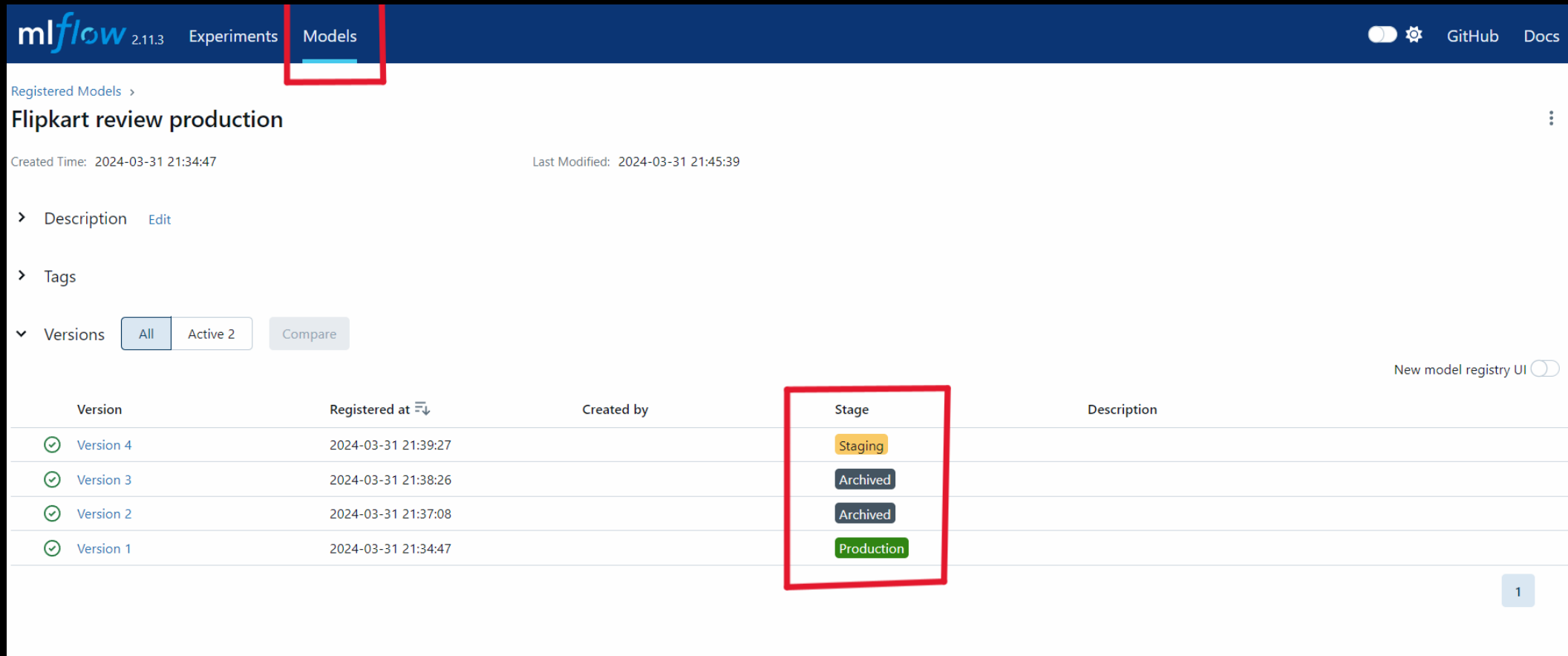
Add section

Registering Models



The table below the Versions section provides information about each version of the model, including the version number, the date and time it was registered, the user who registered it, and its current stage.

Overall, the context shows the status and version history of the "Flipkart review production" experiment in the MLflow Models UI, including the current active version and the previous staging and archived versions.



Prefect

The Prefect is also available as an open source, locally hosted orchestration engine, API server, and UI, giving you insight into the flows running with any local Prefect server instance.

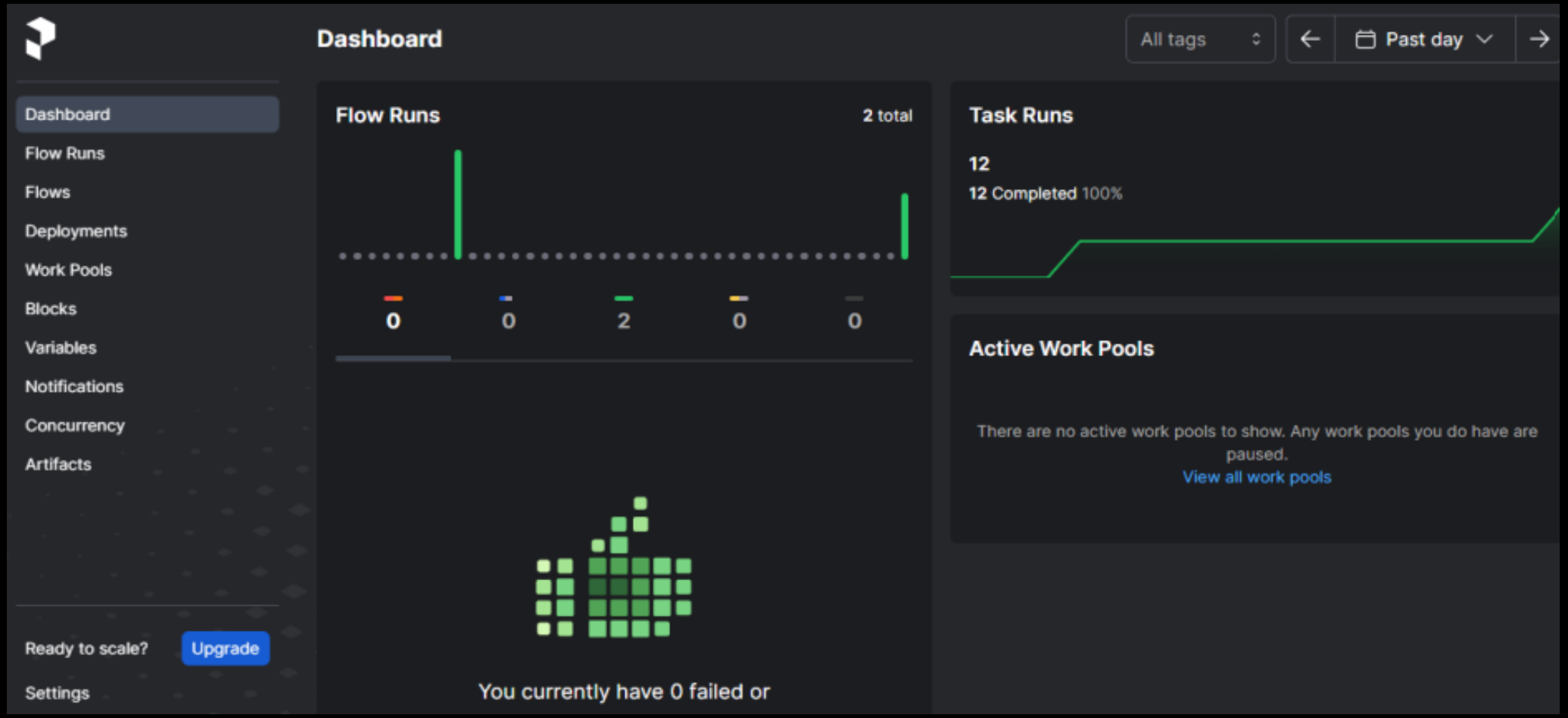
Prefect installation

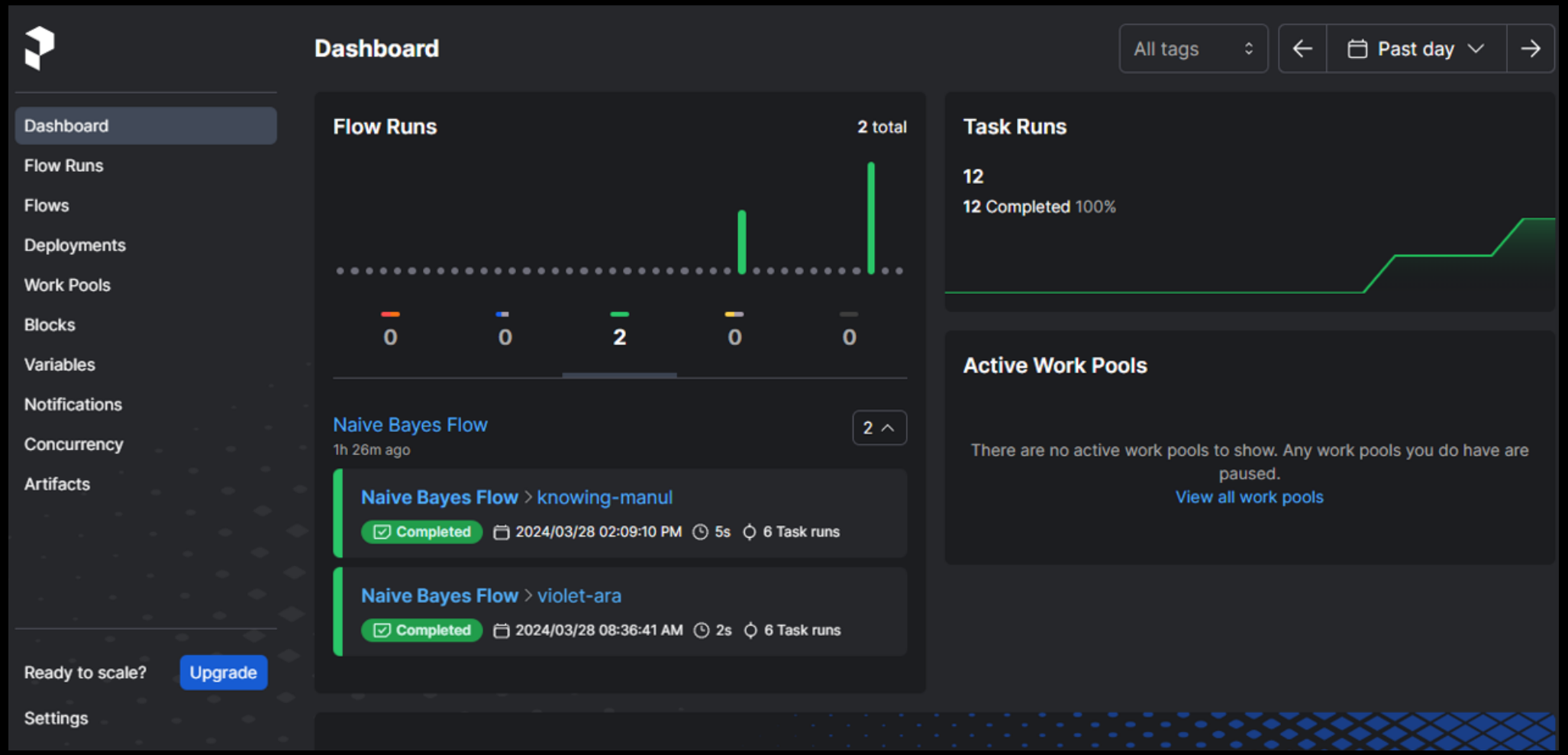
pip install prefect

prefect server start

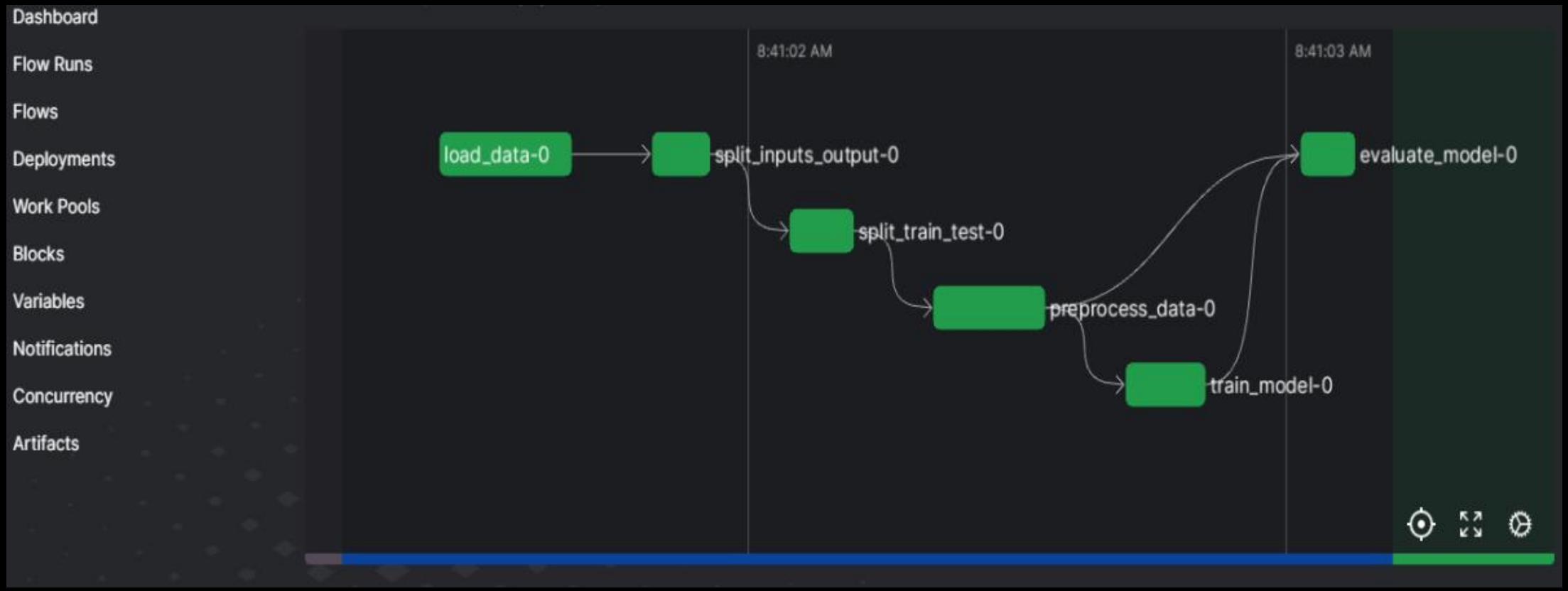
```
 ___ ___ ___ ___ ___ ___ _____
| _ \ _ \ __| __| __/ __|_   _|
|  _/   / _|| _|| _| (__  | |
|_| |_|_\___|_| |___\___| |_|

Configure Prefect to communicate with the server with:

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

View the API reference documentation at http://127.0.0.1:4200/docs

Check out the dashboard at http://127.0.0.1:4200
```

Prefect dashboard




Dashboard
All tags
Past day
Dashboard
Flow Runs
Flows
Deployments
Work Pools
Blocks
Variables
Notifications
Concurrency
Artifacts
Ready to scale?
Upgrade
Settings
Flow Runs
2 total
0
0
2
0
0
Naive Bayes Flow
1h 26m ago
2
Naive Bayes Flow > knowing-manul
Completed
2024/03/28 02:09:10 PM
5s
6 Task runs
Naive Bayes Flow > violet-ara
Completed
2024/03/28 08:36:41 AM
2s
6 Task runs
Task Runs
12
12 Completed 100%
Active Work Pools
There are no active work pools to show. Any work pools you do have are paused.
View all work pools

Prefect Workflow



THANK YOU