

Bắt đầu với Pipeline

```
@card
@step
def start(self):
    """Start and prepare the Training pipeline."""
    import mlflow

    mlflow.set_tracking_uri(self.mlflow_tracking_uri)
    logging.info("MLflow tracking server: %s", self.mlflow_tracking_uri)

    self.mode = "production" if current.is_production else "development"
    logging.info("Running flow in %s mode.", self.mode)

    self.data = self.load_dataset()

    try:
        # Let's start a new MLflow run to track the execution of this flow. We want
        # to set the name of the MLflow run to the Metaflow run ID so we can easily
        # recognize how they relate to each other.
        run = mlflow.start_run(run_name=current.run_id)
        self.mlflow_run_id = run.info.run_id
    except Exception as e:
        message = f"Failed to connect to MLflow server {self.mlflow_tracking_uri}."
        raise RuntimeError(message) from e

    # Now that everything is set up, we want to run a cross-validation process
    # to evaluate the model and train a final model on the entire dataset. Since
    # these two steps are independent, we can run them in parallel.
    self.next(self.cross_validation, self.transform)
```

1. Decorators

```
@card
@step
def start(self):
```

- **@step** :
Đánh dấu đây là một bước (step) trong Metaflow pipeline. Mỗi step sẽ được thực thi tuần tự hoặc song song tùy vào luồng thiết kế.
- **@card** :
Tạo một báo cáo trực quan (report) trong giao diện UI của Metaflow, giúp theo dõi thông tin chi tiết của step này (ví dụ: logs, artifacts).

2. Khởi tạo MLflow Tracking

```
import mlflow

mlflow.set_tracking_uri(self.mlflow_tracking_uri)

logging.info("MLflow tracking server: %s", self.mlflow_tracking_uri)
```

- **mlflow.set_tracking_uri()** :
Thiết lập địa chỉ của MLflow Tracking Server (nơi lưu trữ logs, metrics, models).
 - Giá trị được lấy từ tham số `mlflow_tracking_uri` (mặc định là `http://127.0.0.1:5000` hoặc biến môi trường `MLFLOW_TRACKING_URI`).
 - Ví dụ: Nếu dùng MLflow trên AWS, URI có thể là `http://<ip>:5000`.
- **logging.info()** :
Ghi log thông tin để kiểm tra địa chỉ MLflow server đã được thiết lập.

3. Xác định chế độ chạy

```
self.mode = "production" if current.is_production else "development"

logging.info("Running flow in %s mode.", self.mode)
```

- **current.is_production** :
Kiểm tra xem pipeline đang chạy ở chế độ production hay development (dựa trên cách khởi chạy Metaflow).
 - **Production**: Chạy với `--production` flag (ví dụ: `python training.py --production`).
 - **Development**: Chạy mặc định.
- **Ứng dụng**:
Có thể điều chỉnh hành vi pipeline tùy theo chế độ (ví dụ: dùng dataset khác nhau).

4. Tải dữ liệu

```
self.data = self.load_dataset()
```

- **load_dataset()** :
Phương thức kế thừa từ `DatasetMixin`, dùng để tải dữ liệu huấn luyện.
 - Dataset thường là file CSV/JSON hoặc từ database (ví dụ: bảng thông tin chim cánh cụt với các cột như `bill_length`, `flipper_length`, `species`).

```
def load_dataset(self):  
    return pd.read_csv("penguins.csv")
```

5. Thiết lập MLflow Run

```
try:  
    run = mlflow.start_run(run_name=current.run_id)  
    self.mlflow_run_id = run.info.run_id  
except Exception as e:  
    message = f"Failed to connect to MLflow server {self.mlflow_tracking_uri}."  
    raise RuntimeError(message) from e
```

- **mlflow.start_run()** :

Bắt đầu một MLflow Run để theo dõi thí nghiệm.

- **run_name=current.run_id** :

Đặt tên run bằng ID của Metaflow run (ví dụ: `penguins-12345`), giúp liên kết giữa Metaflow và MLflow.

- **run.info.run_id** :

Lưu ID của MLflow Run vào `self.mlflow_run_id` để sử dụng ở các step sau.

- **Xử lý lỗi:**

Nếu kết nối đến MLflow Server thất bại (ví dụ: server chưa khởi động), sẽ raise exception với thông báo rõ ràng.

6. Chia nhánh pipeline

```
self.next(self.cross_validation, self.transform)
```

- **self.next()** :

Chia luồng thành **2 nhánh song song**:

1. **cross_validation** : Đánh giá mô hình bằng K-Fold Cross-Validation.
2. **transform** : Tiền xử lý toàn bộ dataset để huấn luyện mô hình cuối cùng.

- **Lý do song song**:

Hai bước này độc lập, không phụ thuộc nhau → Tối ưu thời gian chạy.

Tóm tắt luồng xử lý

1. Thiết lập MLflow Tracking Server.
2. Xác định chế độ chạy (production/development).
3. Tải dữ liệu từ nguồn (CSV, database, API...).
4. Bắt đầu MLflow Run và liên kết với Metaflow Run.
5. Chia thành 2 nhánh song song: Cross-Validation và Transform.

Phiên bản #1

Được tạo 19 tháng 4 2025 16:22:49 bởi Đỗ Ngọc Tú

Được cập nhật 22 tháng 4 2025 17:43:00 bởi Đỗ Ngọc Tú