Report

Setup

I'm leveraging makefile to run commands. In order to run the normal flow, use `run-df-flow-filtered-salary-range` command. and to run the version which I intentionally introduced a fault in training step of the flow, use `run-df-flow-filtered-salary-range-with-fault-tolerance`.

There are other commands like `test`. It will run the test using pytest. To run those tests, users should either have pytest in the shell environment or use the provided dockerfile in the test directory.

Flow

Detailed by a flowchart in the appendix.

Decisions Argumentation

For evaluation in my flow, I have introduced 2 custom parameters, name of a metric and value for it. I'm logging it to the model registry, it would be independent of model type, architecture and so on. By default I'm using r^2, which in my base decision tree model is a metric for how much information from the data is being incorporated. The best model result I had with my basic feature is 0.35 (pretty poor).

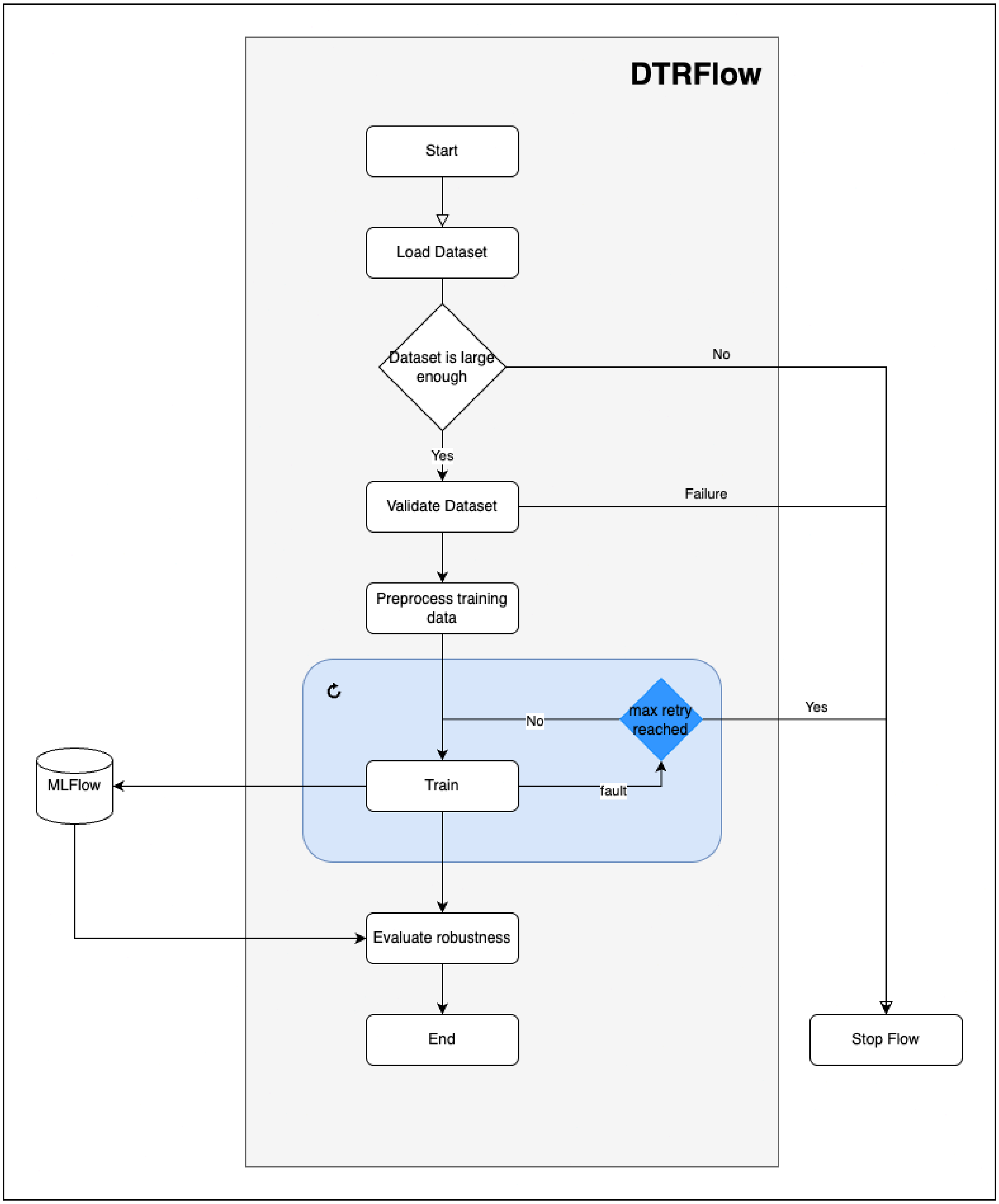
In the final evaluation step, I load the last model and compare it with the model tagged "Champion", the model with this label is the one which passed the baseline. If the current model has better r^2, I label it as Champion and use it as the baseline for the current model.

Fault Tolerance

To handle the intentional fault in the training step, I have benefited from the ***retry*** mechanism of metaflow. It's a decorator which retries the failed step for upmost 4 times. Intentional fault is at line 324.

Reference: https://docs.metaflow.org/scaling/failures#retrying-tasks-with-the-retry-decorator

Appendix


DTRFlow
Start
Load Dataset
Dataset is large enough
No
Yes
Validate Dataset
Failure
Preprocess training data
No
max retry reached
Yes
MLFlow
Train
fault
Evaluate robustness
End
Stop Flow