ML Continuous Integration System

Haocheng Lin

UCL

Author Note

Supervisor: Dr. Daniel Buchan

ML Continuous Integration System

Aims and Objectives

Aim: Create a platform for taking Machine Learning Code and Data repositories as inputs for running automated training. The platform will execute, user defined, downstream processes if the ML training passes the benchmark tests.

Objectives

1. Review and configure the ML pipelines for training.
2. Integrate GitHub to monitor the code changes.
3. Implement a tracking method on the datasets.
4. Any changes to the code or dataset will run the ML training pipeline.
5. Optimize for the best ML model by comparing new scores with historical ML models' scores.
6. Run automated steps (such as sending email or pushing code to servers) when ML benchmarks pass or fail targets.

Expected Outcomes

Minimum Requirements

Monitoring

Function	Stage
Integrate GitHub to track the code changes.	✅
Run the ML Code whenever new changes are made to the repository and the datasets.	✅
Allow the user to monitor the datasets.	⏳

Configuration

Function	Stage
Configure the application via YAML text file	✅

Execute ML Code

Function	Stage
Can run a ML pipeline on the local computer.	✅

Results Presentation

Function	Stage
Present a visualized exhibition of the ML results and its scores via email.	✅

Benchmarking

Function	Stage
Allow the user to use ML code for calculating a benchmark score to determine whether the training code has passed the tests.	⏳
Enable the user to compare test results with existing benchmark scores to determine whether the current performance fits the passing target.	⏳

Downstream Task

Function	Stage
The user would receive a notification via email about the test progresses and a successful result would push the data and the result onto the server.	⏳

Perfect Solution

Monitoring

Function	Stage
Allow the user to monitor the changes across several repositories.	⏳

Configuration

Function	Stage
Provide a GUI application to help with configuring the code.	⏳

Execute ML Code

Function	Stage
Enables running the pipeline on the different computational platforms.	⏳

Results Presentation

Function	Stage
Present the results on a web interface alongside an email for the user.	⏳

Work Plan

July – October: Planning

1. Selected the topic.
2. Conducted research on the ML Continuous Integration System.

October – November: Write-up a preliminary specification outlining my progresses.

November: Design a small-scale prototype with basic functionalities.

November – January: Extend upon the prototype into the full system.

January – February:

1. Design Tests for evaluating each task.
2. Interim report for checking the progress.

February – March: Record a video preview outlining the project.

March: Finalize the Project Report for submission.