

End-to-end ML System Benchmarking

(Bachelor project, Winter 2020) - Data Engineering Systems Group

Motivation

The current development of end-to-end machine learning systems enables users to execute whole workflows in just one system. Moreover, the state-of-the-art machine learning systems cover most of the algorithms, as well as allow users to try and implement novel algorithms and approaches. This points to the growing need of measuring the performance of these systems under load as well as in processing heterogeneous data workflows.

Creating a modular suite for benchmarking machine learning systems will allow researchers and engineers to dissect the performance and resource consumption in the different stages of the end-to-end workflows, as well as spotting potential data processing bottlenecks and points for optimization of the system. For this purpose, systems need to be tested with heterogeneous data requiring different preparation steps and used to train various machine learning models. Interesting case studies that can be used for benchmarking machine learning systems are:

- CS 1: Earth Observation - Large set of image data (~4 PB) used for climate zone classification and monitoring
- CS2: Semiconductor Manufacturing - Simulation and prediction of chip defects and equipment stability
- CS3: Material Degradation - Modelling and simulation of material degradation in semiconductor devices
- CS4: Automotive Vehicle Development - Modelling of the vehicle development process, improvement of decision making in the vehicle design process

These four case studies can be modeled as complex workflows consisting of different stages such as data preparation and integration, model training, evaluation, and inference. To evaluate the systems that run such workflows, we need to take into consideration all of their components and the data manipulation necessary for their execution. In this project, we propose a prototype of a benchmarking platform capable of measuring the performance of an end-to-end machine learning system.

Project description

The main objective of this project is to build a platform for benchmarking and analysis of SystemDS and other ML systems. The students' main contribution is the design and implementation of a platform that will benchmark the data preparation, data cleaning, and machine learning capabilities of an end-to-end system, like SystemDS, and libraries, such as Pandas, NumPy, and Scikit-learn.

Machine learning workflows are commonly composed of several different stages, such as data preparation, data cleaning, model training, and inference. The proposed benchmarking platform needs to measure the performance of each of them.

Data preparation: The participants will select a working dataset from the presented case studies and create a data preparation pipeline, which will transform the data for the cleaning and training phases. The focus of this stage is the benchmarking of each of the data transformation steps.

Data cleaning: The focus will be on measuring the performance of the procedures for handling missing, corrupt or inaccurate data points.

Model training & evaluation: The platform needs to track the execution of the training algorithms and gather metrics such as execution time, number of training steps, and resource consumption. The focus of the evaluation stage is the gathering of model performance metrics such as accuracy, precision, and recall.

Model inference: In order to assess the behaviour of the model in a production environment, we feed it with unseen data points provided by the case studies. The platform will need to measure the throughput and latency of the predictions.

The participants need to design the benchmarking platform as an independent system, which can run machine learning workflows as previously defined. It receives a workflow specification as input and executes it by calling system APIs, services, or methods from other machine learning systems, such as SystemDS, Scikit-learn, Pandas, or NumPy. Finally, in order to ease the analysis of the produced benchmarks, the platform should have a visualization component, which would assist the readability and presentation of the results.

Project partners

During this project, the participants will collaborate with researchers from TU Graz and the KNOW Center (core developers of SystemDS) and potentially other industry partners.

Skills

The participants need to have experience in software engineering and at least one programming language (C++, Java, Python). Basic experience in machine learning is beneficial. Participants should be comfortable with documenting their work and visualizing their results.

Recommended reading

1. Mathias Boehm et al. SystemDS: A Declarative Machine Learning System for the End-to-End Data Science Lifecycle [[Paper](#)] [[Video](#)]
2. Petter Mattson et al. MLPerf Training Benchmark [[Paper](#)]
3. Tilmann Rabl et al. AdaBench - Towards an Industry Standard Benchmark for Advanced Analytics [[Paper](#)]
4. Marco Vogt et al. Chronos: The Swiss Army Knife for Database Evaluations [[Paper](#)]
5. Joaquin Vanschoren et al. OpenML: networked science in machine learning [[Paper](#)]
6. Christoph Boden et al. PEEL: A Framework for Benchmarking Distributed Systems and Algorithms. [[Paper](#)][[website](#)]