ETH zürich

Systems@ETH Zürich

Towards A Platform and Benchmark Suite for Model Training on Dynamic Datasets

Maximilian Böther, Foteini Strati, Viktor Gsteiger, Ana Klimovic

Problem Statement

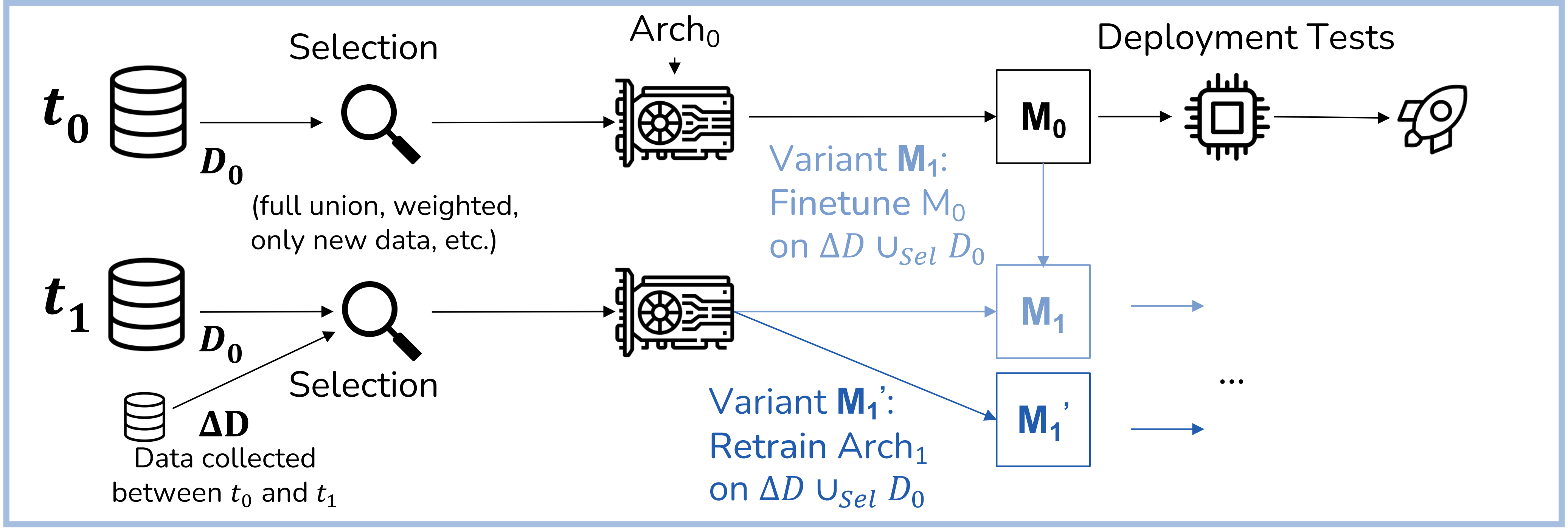
In practice, the training dataset is frequently updated. Hence, models are regularly (re)trained, which is very expensive.

Research Question

How can we lower the cost of updating production models on dynamic datasets?

To investigate this question, we build a system for model training on dynamic datasets that enables research on (a) training policies and (b) data selection policies.

Model Update Strategies


Selection
Arch_0
Deployment Tests
t_0
D_0
(full union, weighted, only new data, etc.)
M_0
Variant $\mathbf{M_1}$: Finetune M_0 on $\Delta D \cup_{Sel} D_0$
t_1
D_0
M_1
...
Selection
ΔD
Data collected between t_0 and t_1
Variant $\mathbf{M_1}$': Retrain Arch_1 on $\Delta D \cup_{Sel} D_0$
M_1'

Training Policy Design Space

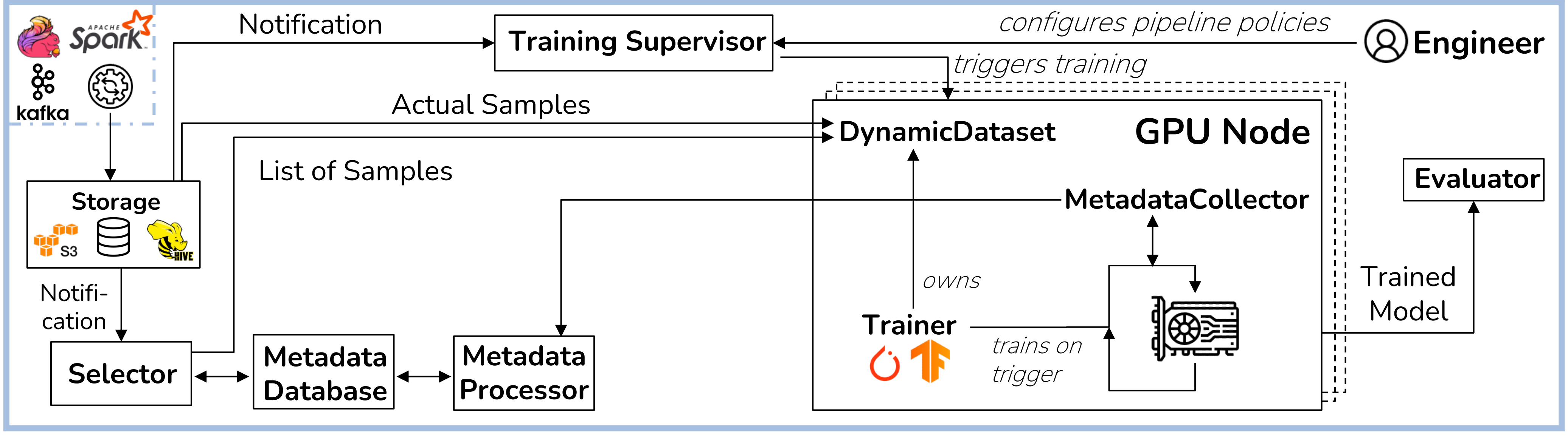
When to update the model?

1. Do we train with a fixed schedule, when a certain number of new data points has arrived or on data shifts?
2. How do we detect data distribution shifts?

How to update the model?

1. Do we retrain from scratch, finetune the existing model, or switch between both?
2. On which old and new data points do we train?
   a. Which metrics do we need for this decision?
   b. How do we efficiently collect and store these metrics?
3. What do we do when old data is deleted?

Modyn System Architecture


Apache Spark
kafka
Notification
Training Supervisor
configures pipeline policies
Engineer
triggers training
Actual Samples
DynamicDataset
GPU Node
List of Samples
Storage
S3
HIVE
MetadataCollector
Evaluator
Notifi-cation
owns
Trained Model
Trainer
trains on trigger
Selector
Metadata Database
Metadata Processor