Serving ML Inference Pipelines in Containers

Mingyu Guan, Alind Khare, Alexey Tumanov
School of Computer Science, College of Computing, Georgia Institute of Technology

Problem

Production development of a ML pipeline is full of challenges:

- Heterogeneous environments across models.
- Heterogeneous library requirements across models.
- Heterogeneous resource requirements across models.
- Scaling at model level.

Model

SRTML

Input

Ray Actor

RPC

Ray Actor

RPC

Data Flow

Ray Actor

RPC

Data Flow

Ray Actor

RPC

Output

Figure 1: Main Flow of Container-based Development.

Design:

- Supporting pipeline and batched queries functionality.
- Serving models in Containers across ML frameworks.

Data Flow:

- Models are connected to a pipeline by Ray Actors.
- Each model is running in a separate Docker Container.
- A Ray Actor talks to its container via RPC.

Latency Comparison

Figure 2: No-operation Model Latency.

Figure 3: Single Squeezenet Model Latency.

Figure 4: Transform-Squeezenet Pipeline Latency

Start-up Time Comparison

Figure 5: Model Start-up Time.

Contribution

- Serving models in containers across frameworks.
- Providing process and resource isolation.
- Isolating each model’s variability with SRTML.

Related Work


References

