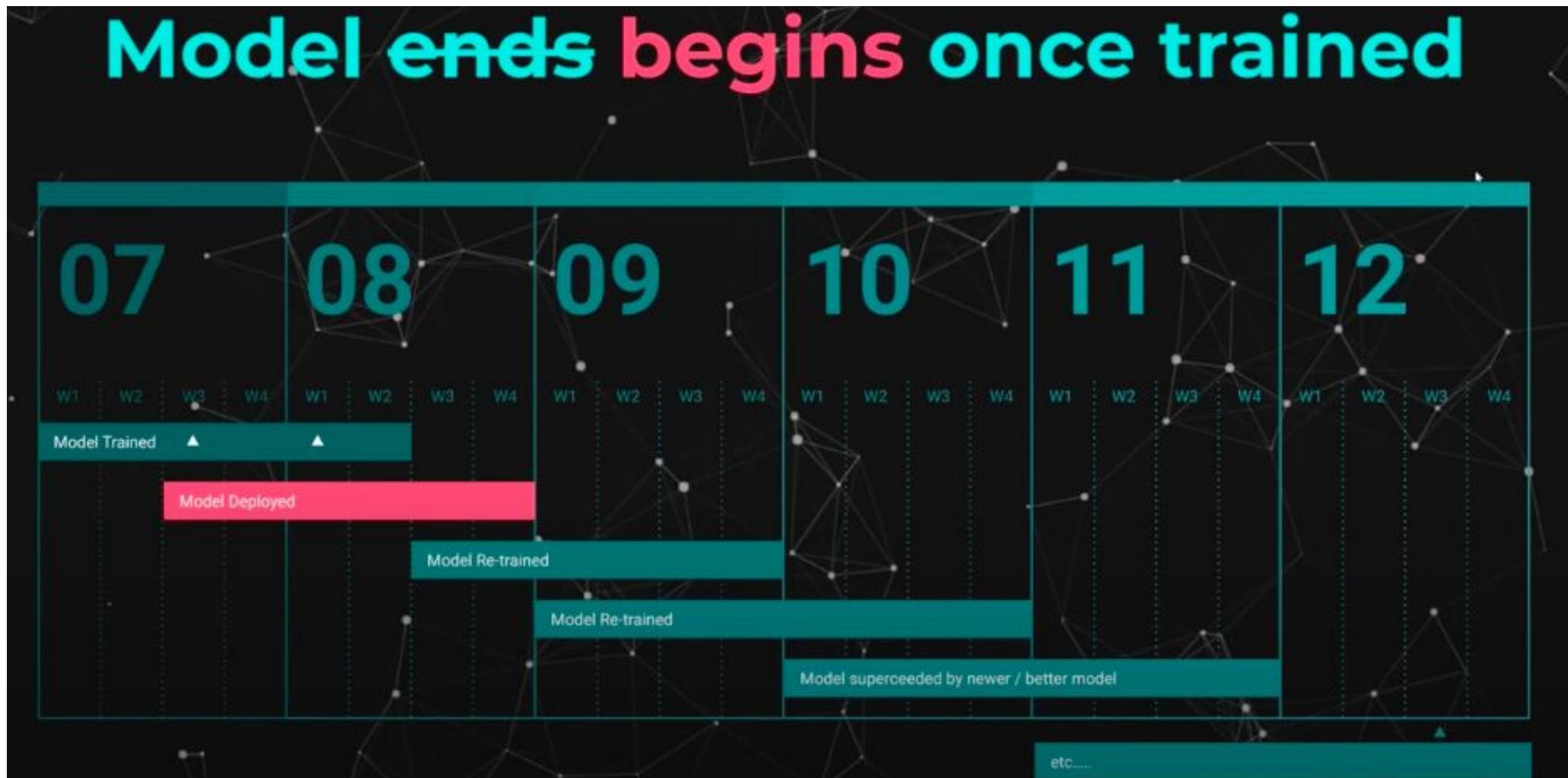


Machine Learning in Production and **AISys** Research

Saeid Ghafouri
ML Systems Course
Sep 14th 2023

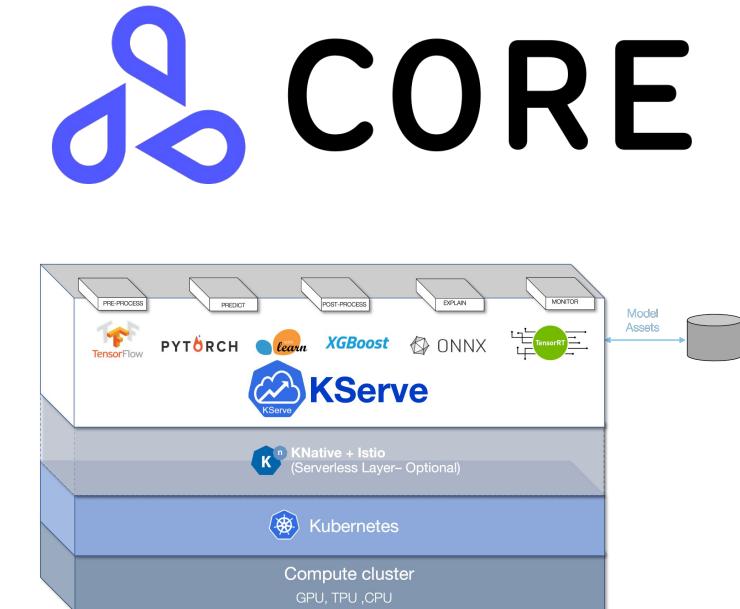
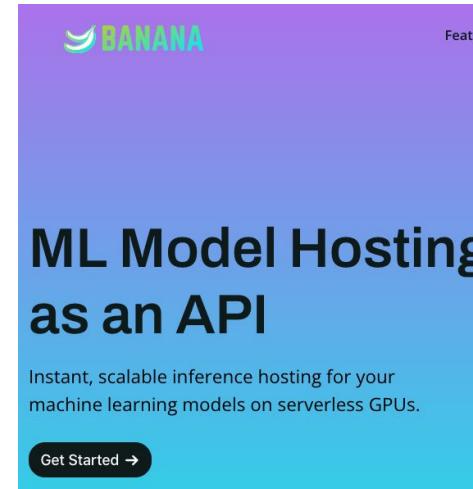
Production Machine learning life-cycle



Source: <https://youtu.be/QcevzK9ZuDg>

ML Serving In production

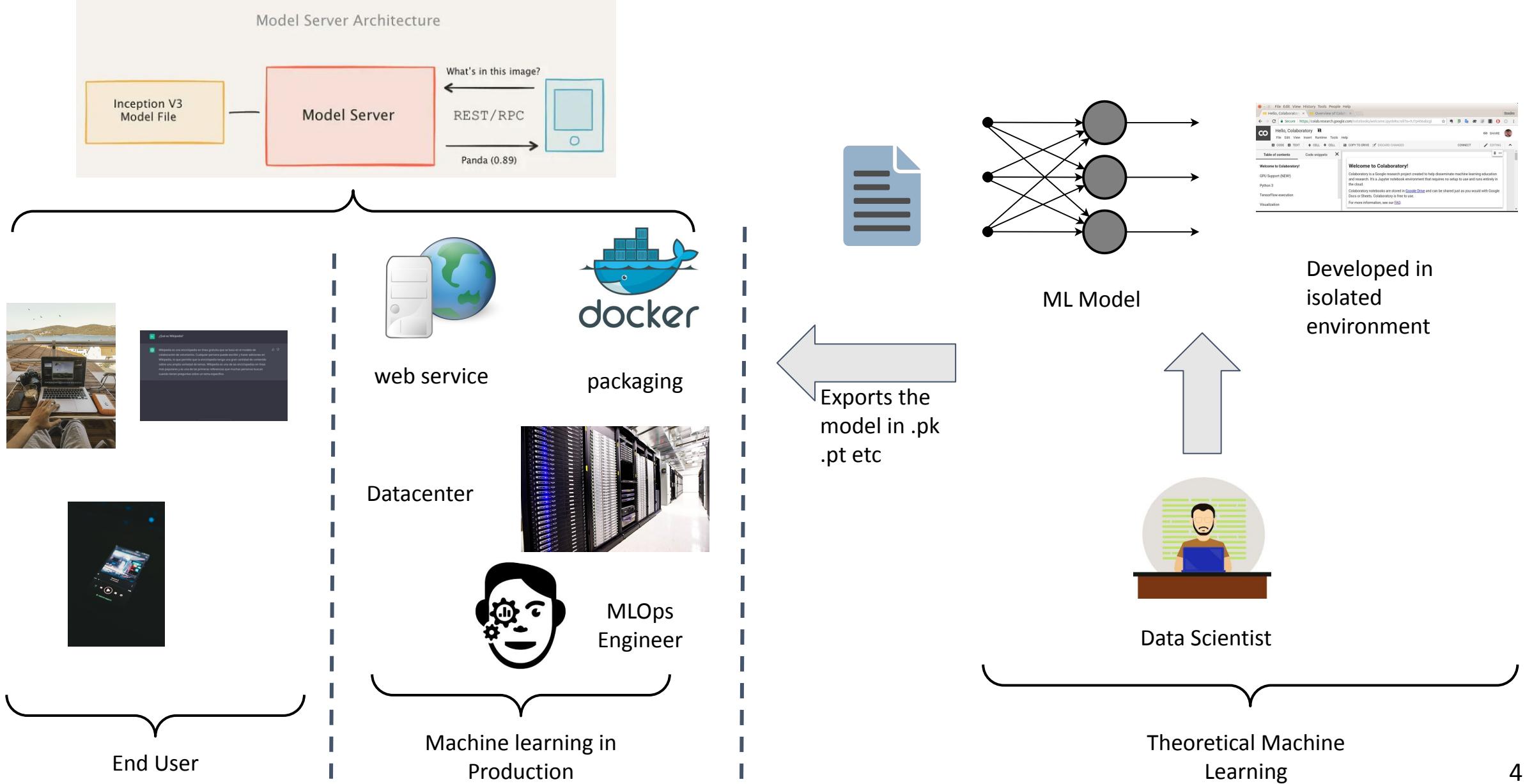
- Optimizing the resources
 - Subject to the variable workload
 - Hardware
 - Budget
- Model drift detection
- Serving Infrastructure
- Model server
- Streaming Data
- Observability of the model
- Automatic retraining
- Explainability
- ...



**The AI community
building the future.**

Build, train and deploy state of the art models powered by
the reference open source in machine learning.

ML Serving In production



Different format of ML Deployment in production

- Full API

Inference Endpoints Starting at \$0.06/hour

Inference Endpoints offers a secure production solution to easily deploy any ML model on dedicated and autoscaling infrastructure, right from the HF Hub.

CPU Instances				GPU Instances			
Provider	Architecture	vCPUs	Memory	Provider	Architecture	GPUs	Memory
aws	Intel Xeon - Icex	1	2GB	aws	NVIDIA T4	1	34GB
aws	Intel Xeon - Icex	2	4GB	aws	NVIDIA A10G	1	24GB
aws	Intel Xeon - Icex	4	8GB	aws	NVIDIA T4	4	56GB
aws	Intel Xeon - Icex	8	16GB	aws	NVIDIA A10G	1	88GB
azurite	Intel Xeon	1	2GB	aws	NVIDIA A10G	2	160GB
azurite	Intel Xeon	2	4GB	aws	NVIDIA A10G	4	320GB
azurite	Intel Xeon	4	8GB	aws	NVIDIA A10G	4	96GB
azurite	Intel Xeon	8	16GB	aws	NVIDIA A10G	8	640GB

Free Plan

Your Current Plan

ChatGPT Plus USD \$20/mo

Upgrade plan

Available when demand is low
Standard response speed
Regular model updates

Available even when demand is high
Faster response speed
Priority access to new features

- Serverless

GPU Pricing

Per Hour Per Second

1x A100 (40GB)

8x A100 (40GB)

\$7.4868 / hr

Contact Sales

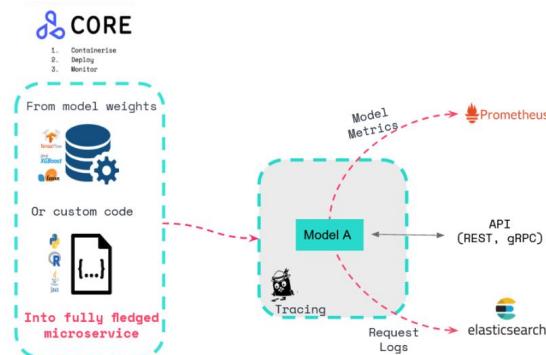
Get Started → Let's Chat →

✓ Autoscaling
✓ Scales to zero
✓ Up to 40% Volume Discounts

```
polassium.py
```

```
1 #!/usr/bin/env python
2 # app.init
3 # def init():
4 #     model = torch.load('gptj.pt')
5 #     context = [
6 #         "model": model
7 #     ]
8 #
9 # @app.handler()
10 # def handler(context, request):
11 #     prompt = request.json.get("prompt")
12 #     model = context.get("model")
13 #     outputs = model(prompt)
14 #     return Response(
15 #         json={"outputs": outputs},
16 #         status=200
17 #     )
18 #
```

- Self managed infrastructure



Main players

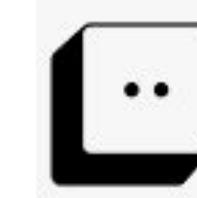
- Full API



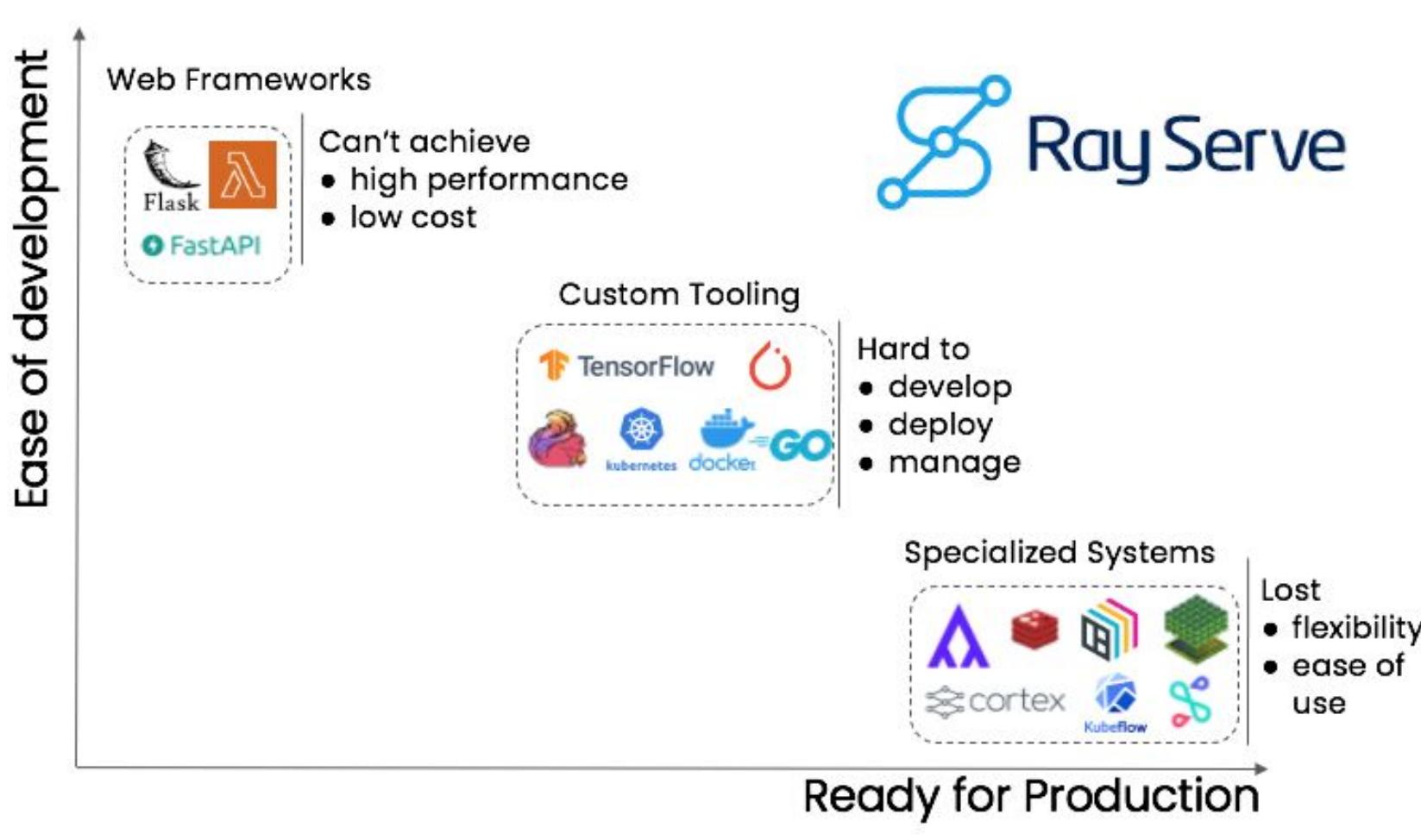
- Serverless



- Self managed Infrastructure

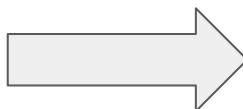
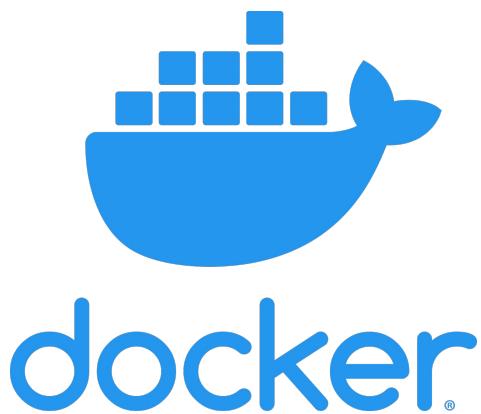


Spectrum of different ways of deployment



<https://www.anyscale.com/blog/serving-ml-models-in-production-common-patterns>

Cloud Orchestration



Borg

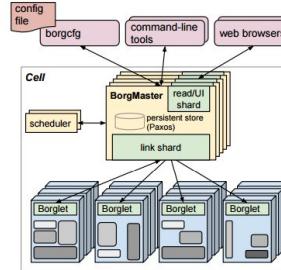
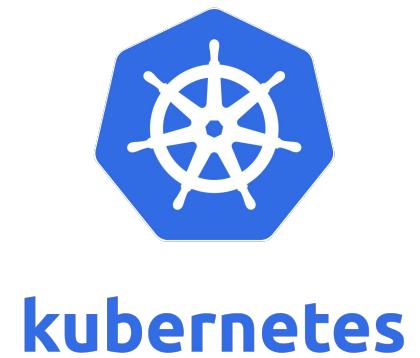
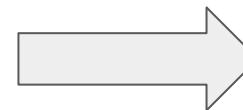
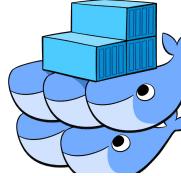


Figure 1: The high-level architecture of Borg. Only a tiny fraction of the thousands of worker nodes are shown.

Swarm

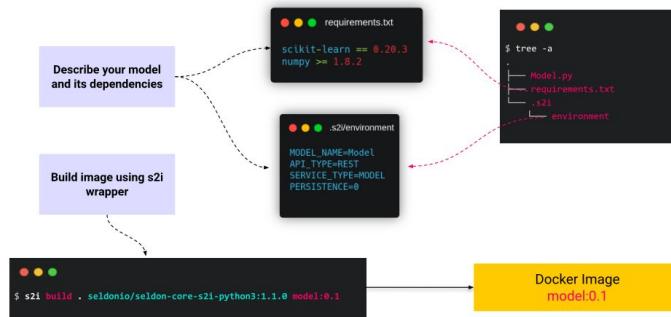


kubernetes

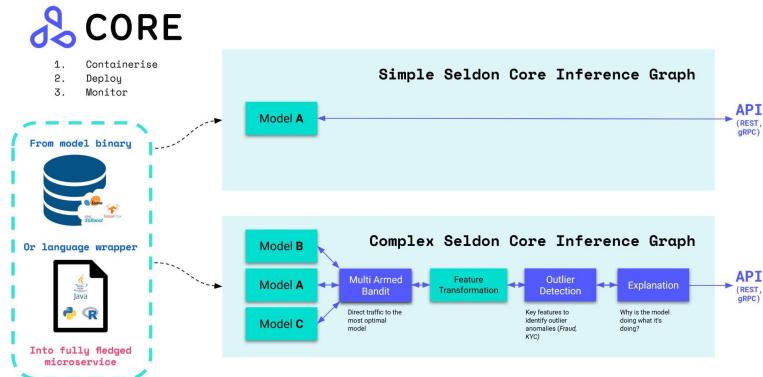


MESOS

Seldon Core and ML Deployment on K8S

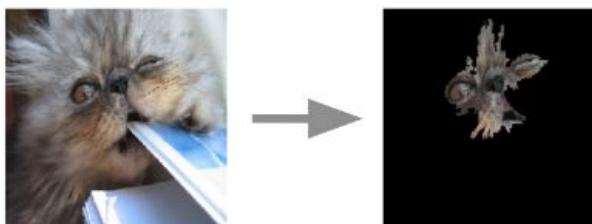


Kubernetes Deployment



Inference pipelines

Anchor explanations for images



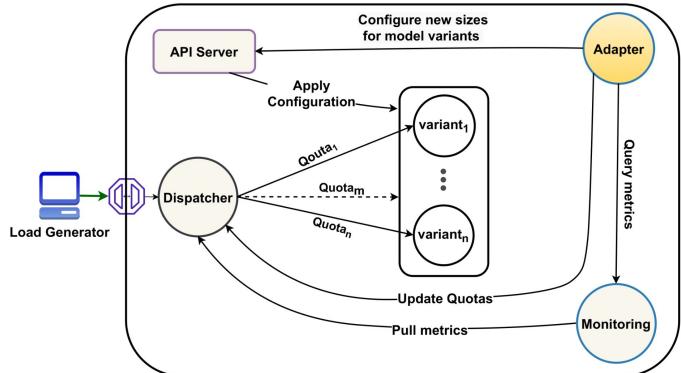
Integrated Gradients for text

a powerful study of loneliness sexual UNK and desperation be patient UNK up the atmosphere and pay attention to the wonderfully written script br br i praise robert altman this is one of his many films that deals with unconventional **loneliness** subject matter this film is disturbing but it's sincere and it's sure to UNK a strong emotional response from the viewer if you want to see an unusual film some might even say bizarre this is worth the time br br unfortunately it's very difficult to find in video stores you may have to buy it off the internet

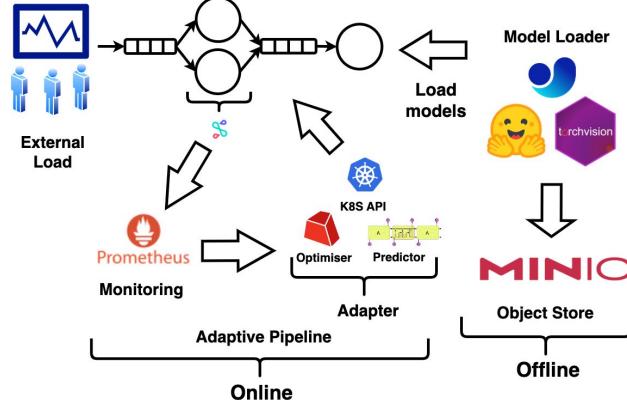
Model Explainability

Recent Projects of AISys

InfAdapter
(EuroMLSys 2023)



IPA
(under review)



Inference_x
(In Progress)



InfAdapter: Reconciling High Accuracy, Cost-Efficiency, and Low Latency of Inference Serving Systems

Mehran Salmani, Saeid Ghafouri, Alireza Sanaee, Kamran Razavi
Joseph Doyle, Max Mühlhäuser, Pooyan Jamshidi, Mohsen Sharifi



Queen Mary
University of London



TECHNISCHE
UNIVERSITÄT
DARMSTADT



UNIVERSITY OF
South Caro

“More than 90% of data center compute for ML workload, is used by inference services”



ML inference services have strict requirements

Highly Responsive!



ML inference services have strict requirements

Highly Responsive!



Cost-Efficient!



ML inference services have strict requirements

Highly Responsive!



Cost-Efficient!



Highly Accurate!



ML inference services have strict & conflicting requirements

Highly Responsive!



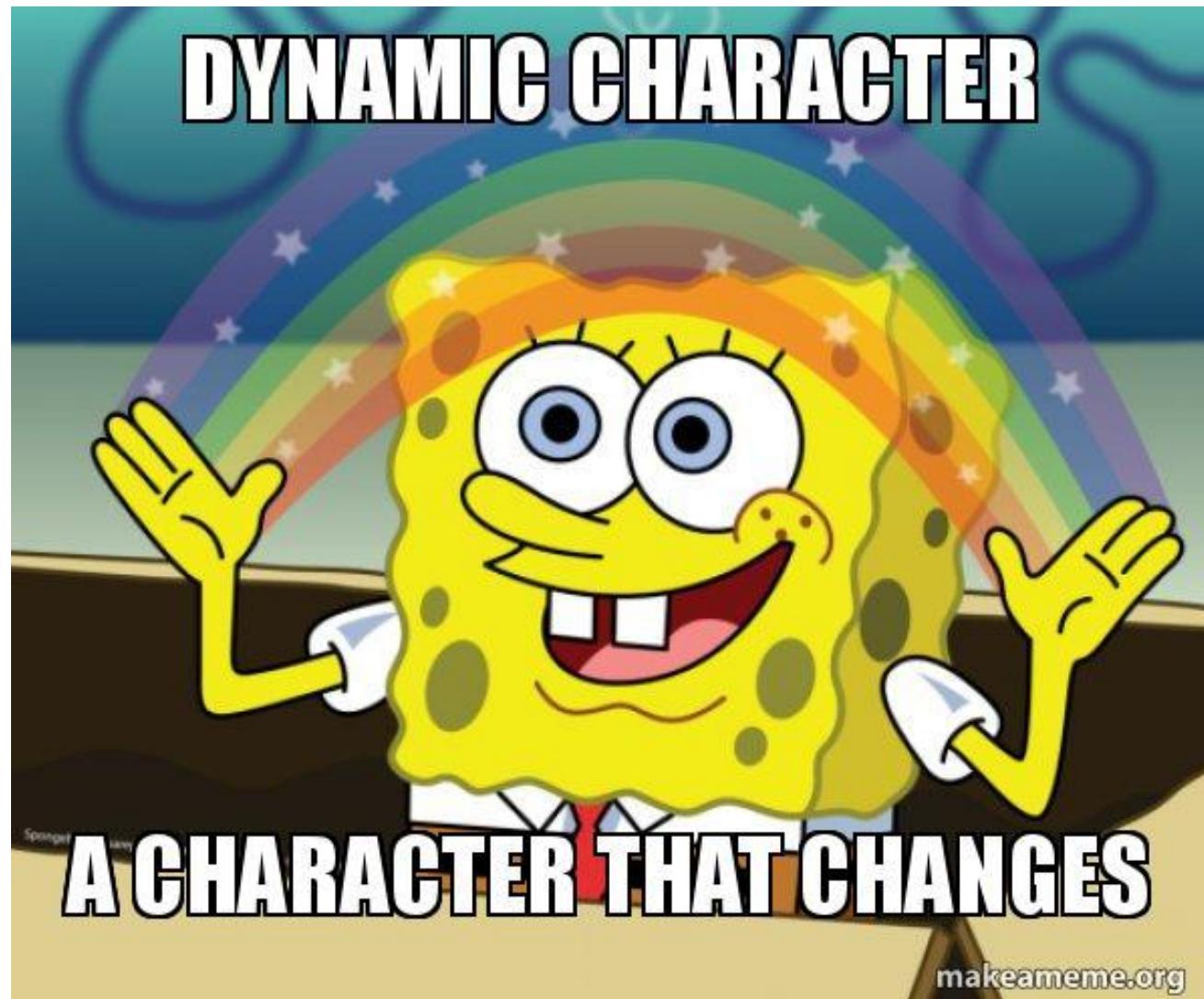
Cost-Efficient!



Highly Accurate!



More challenge: Dynamic workload

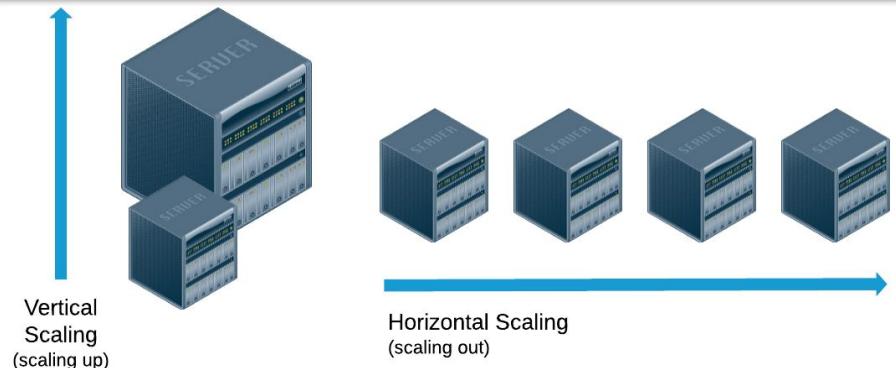


Existing adaptation mechanisms

Resource Scaling

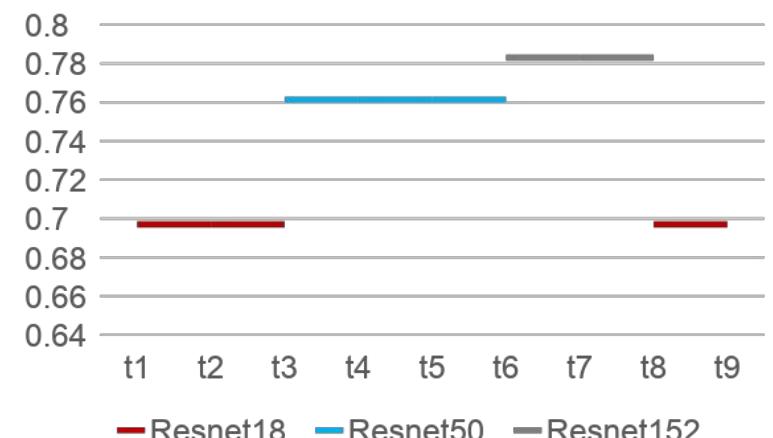
Vertical Scaling (AutoPilot EuroSys'20)

Horizontal Scaling (MArk ATC'19)



Quality Adaptation

Multi Variants (Model-Switching Hotcloud'20)



Resource allocation

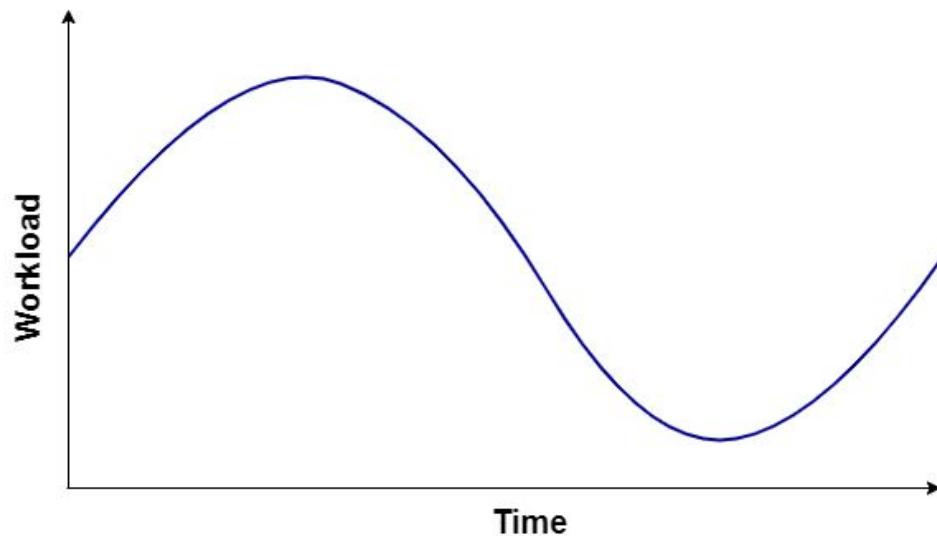
Over
Provisioning



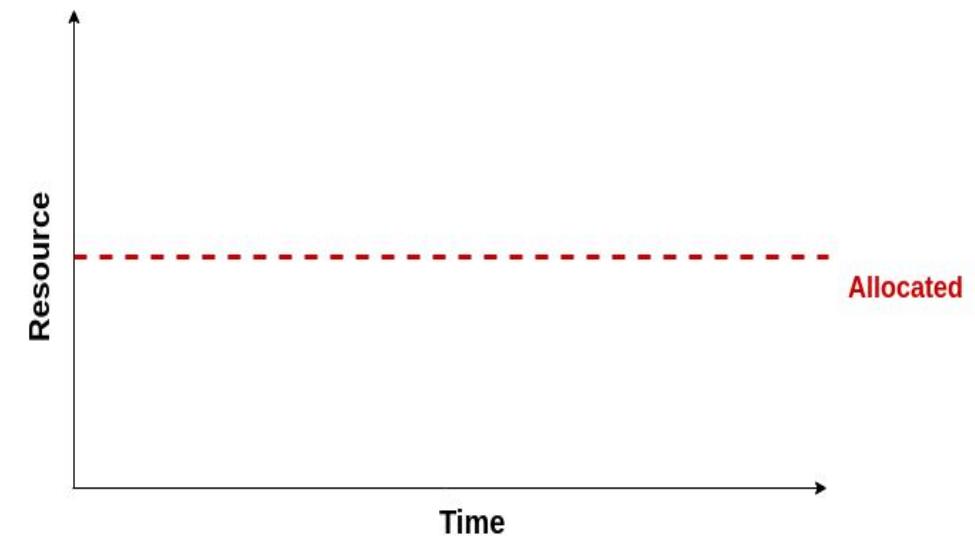
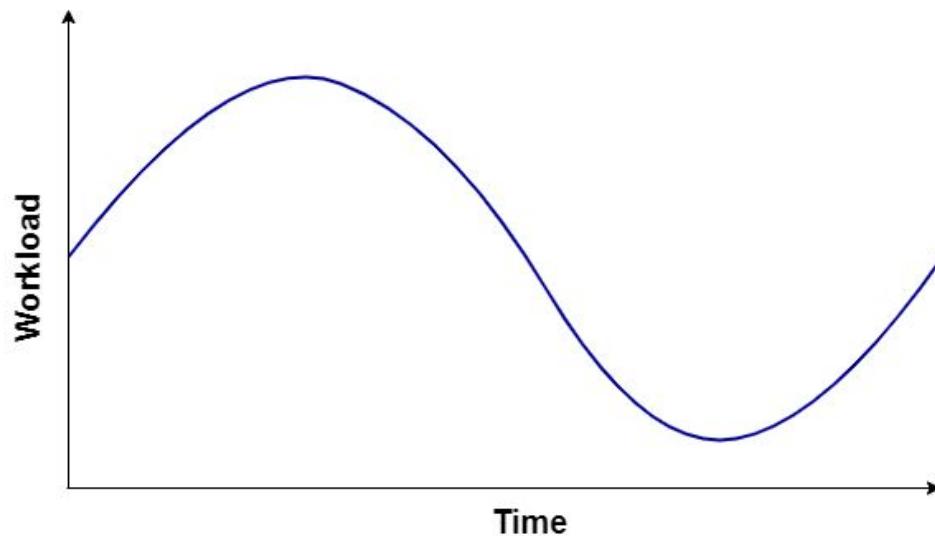
Under
Provisioning



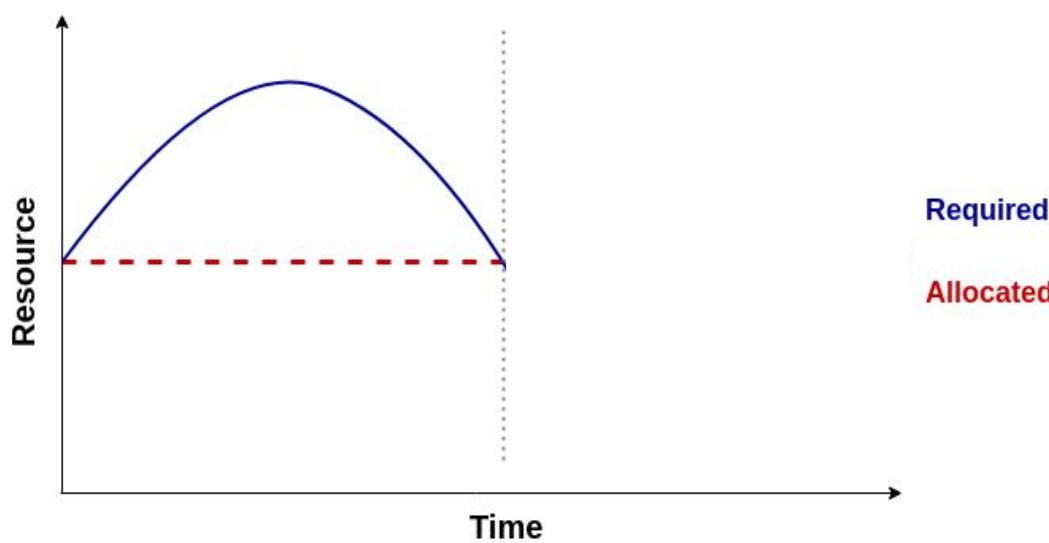
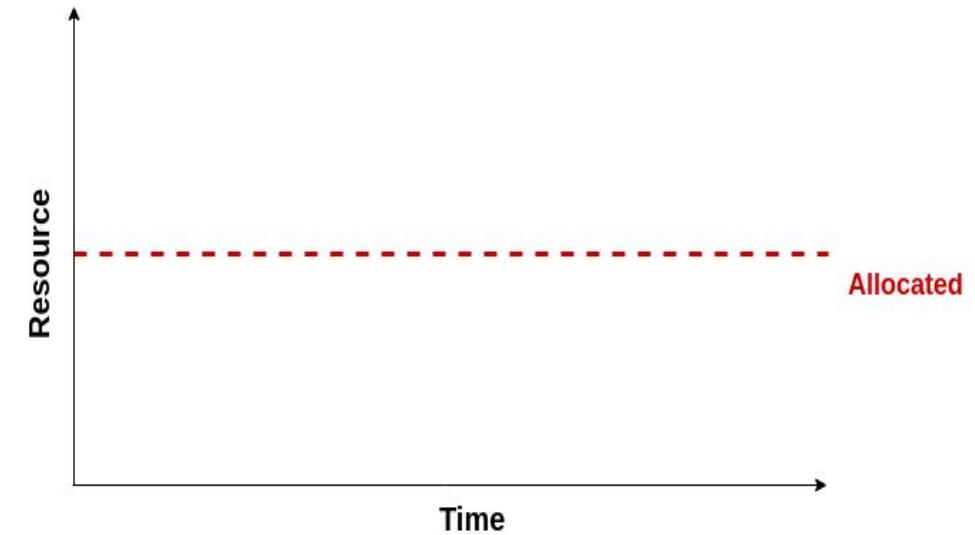
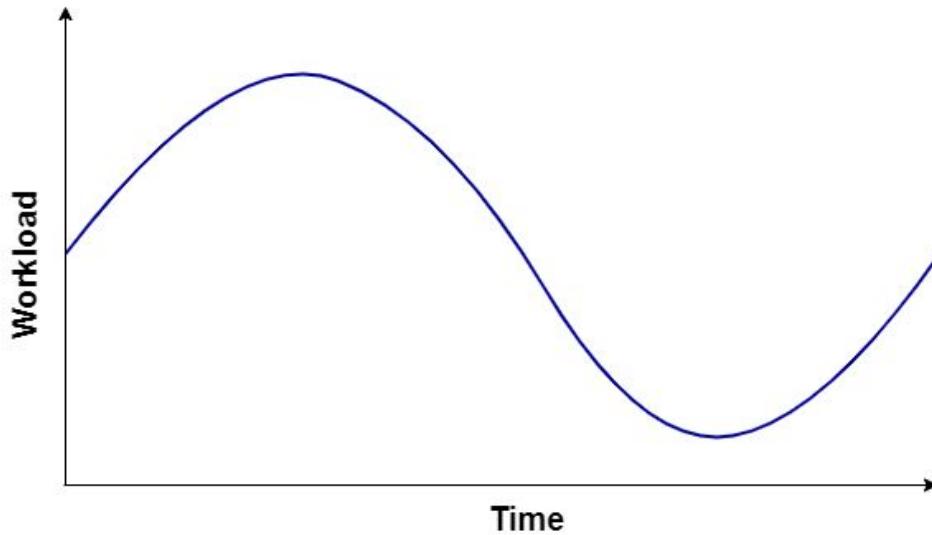
Resource allocation



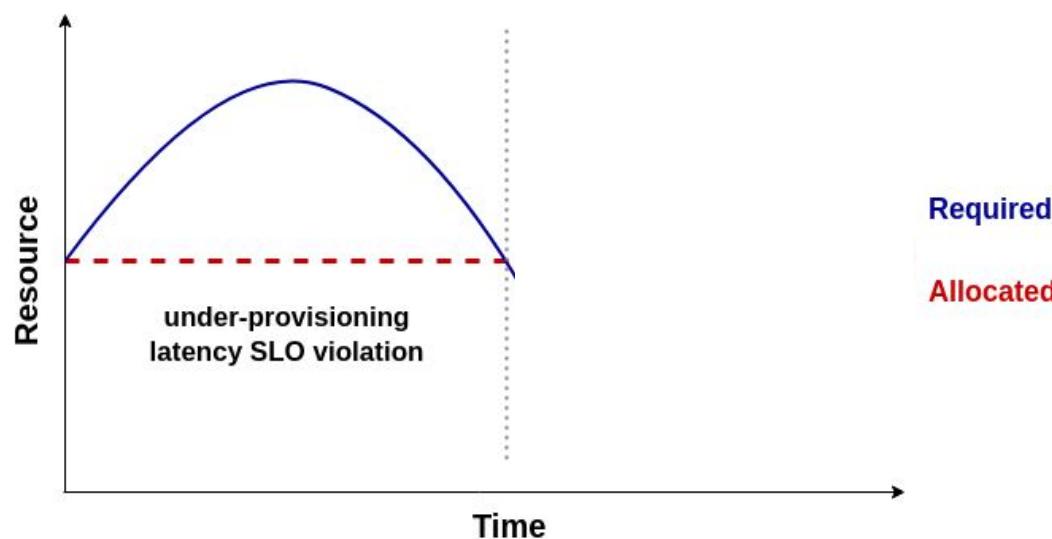
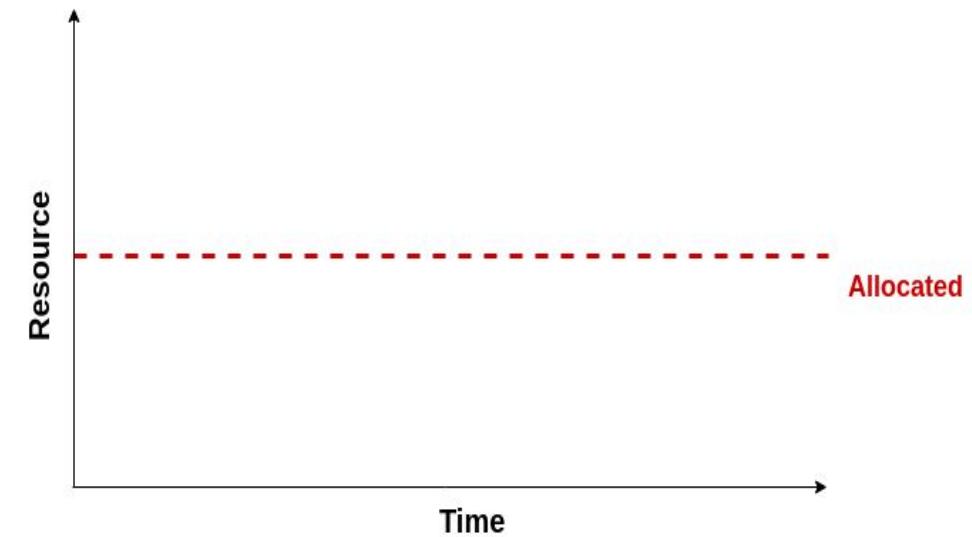
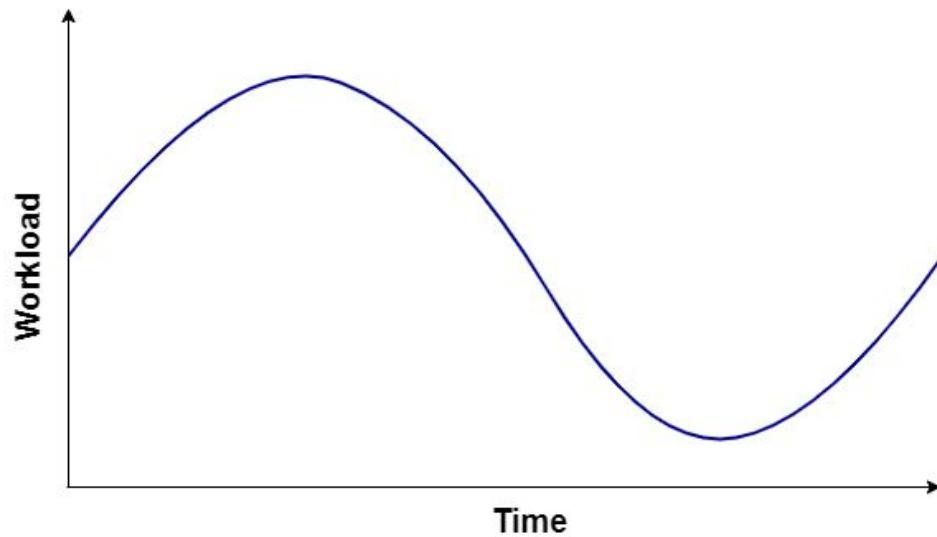
Resource allocation



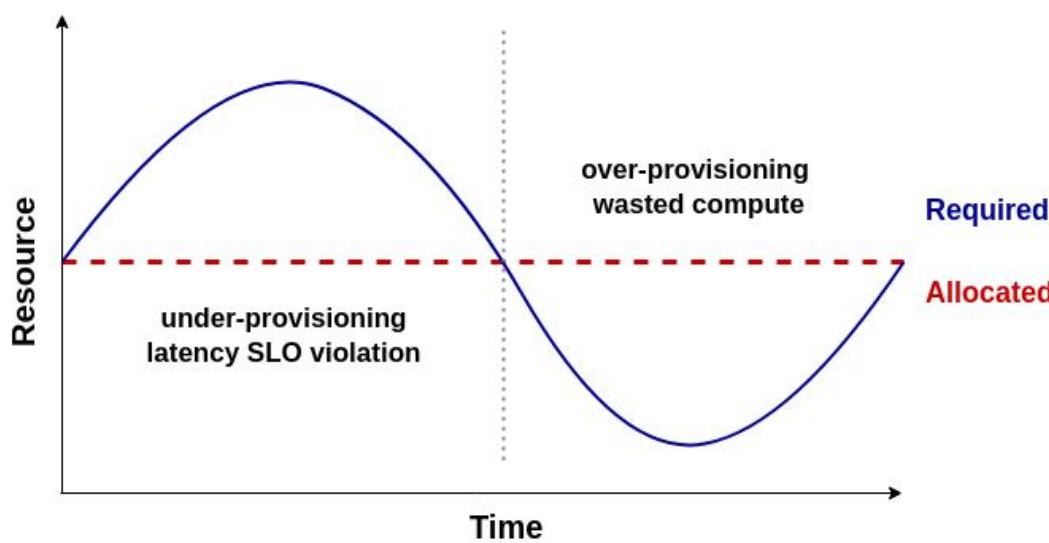
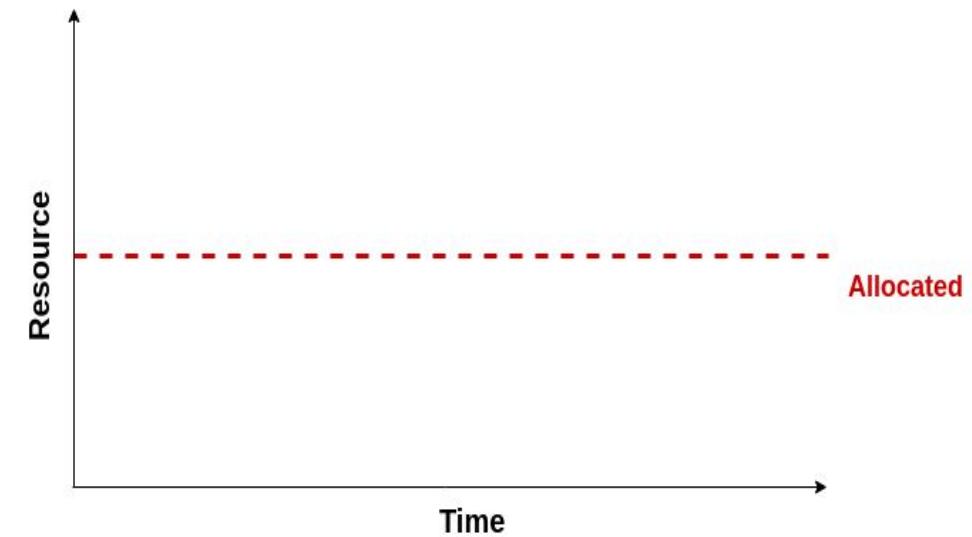
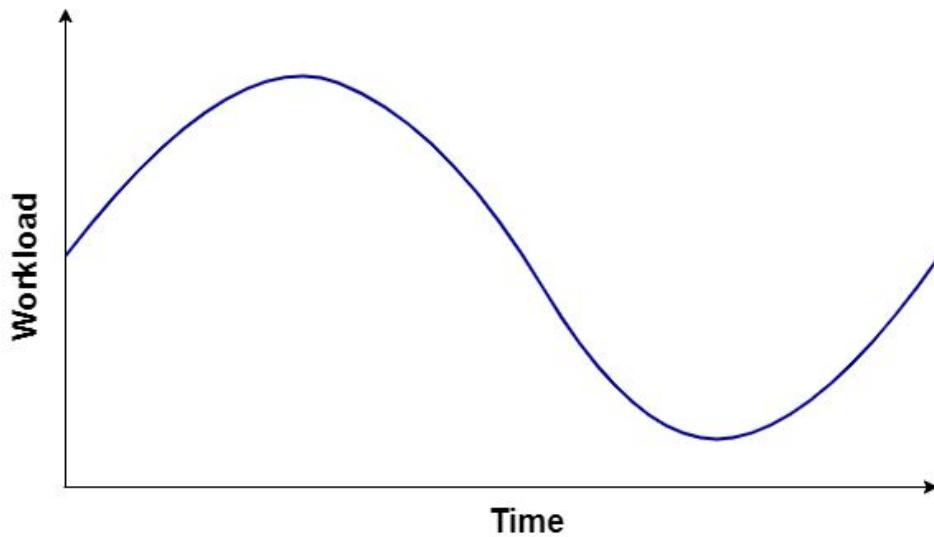
Resource allocation



Resource allocation



Resource allocation



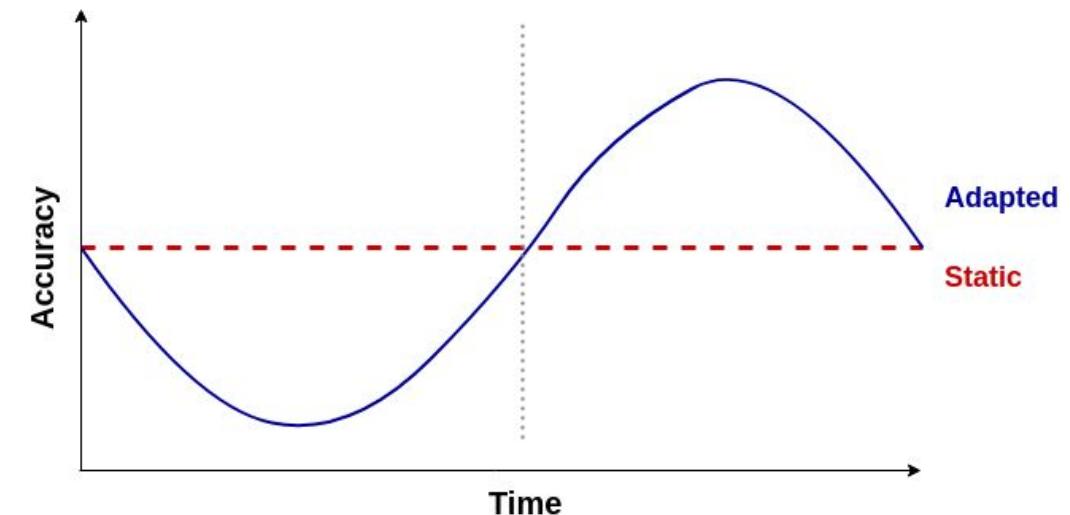
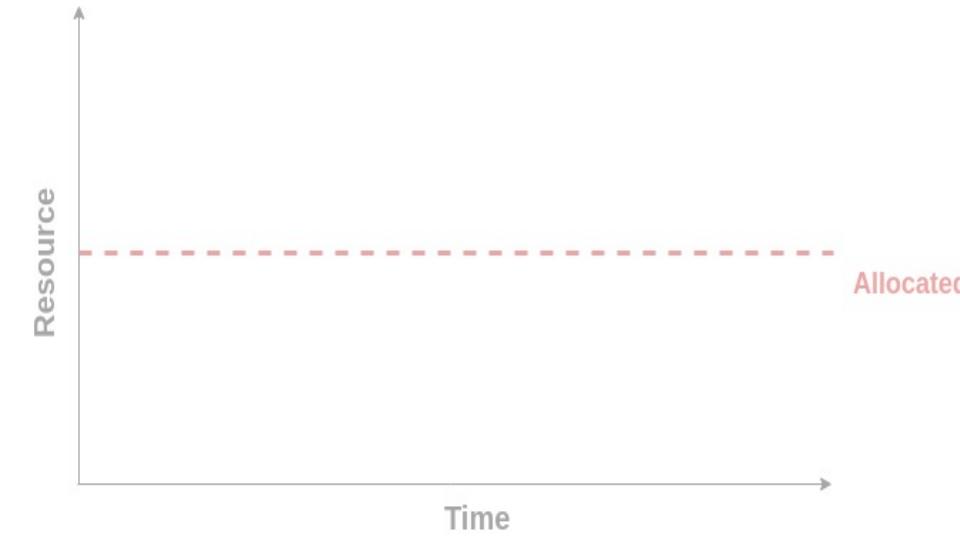
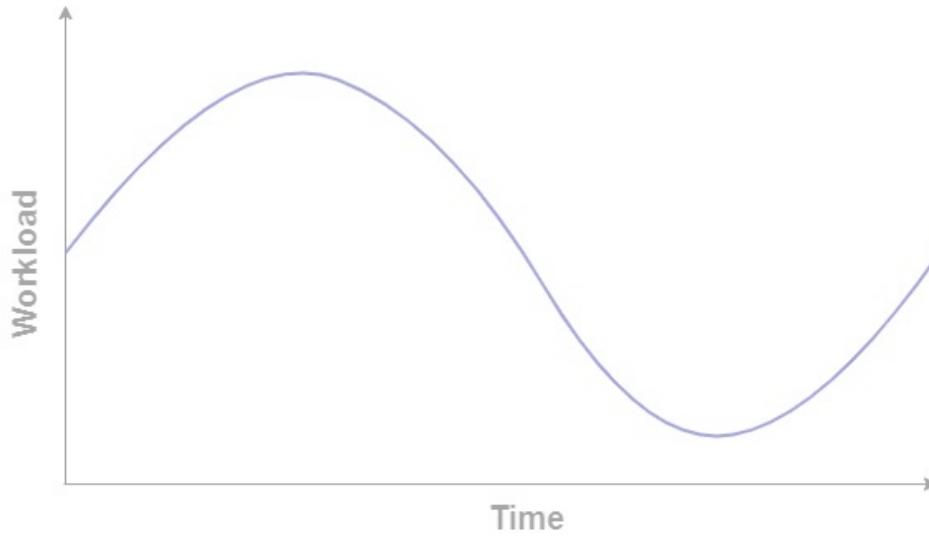
Quality adaptation

ResNet18: Tiger

ResNet152: Dog



Quality adaptation

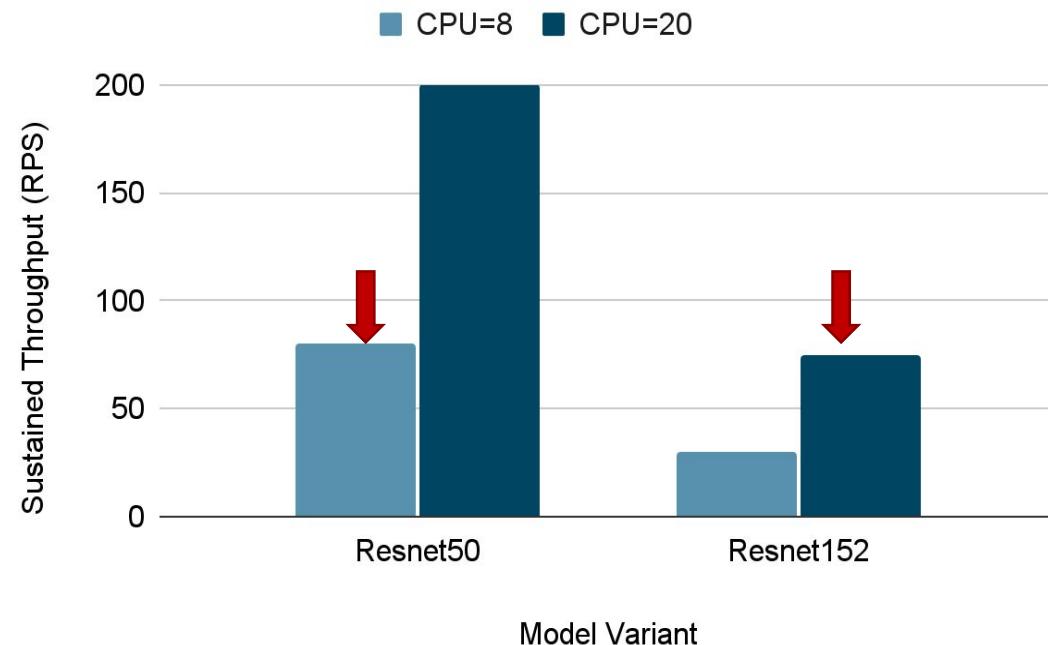


Solution: InfAdapter

InfAdapter is a latency SLO-aware, highly accurate, and cost-efficient inference serving system.

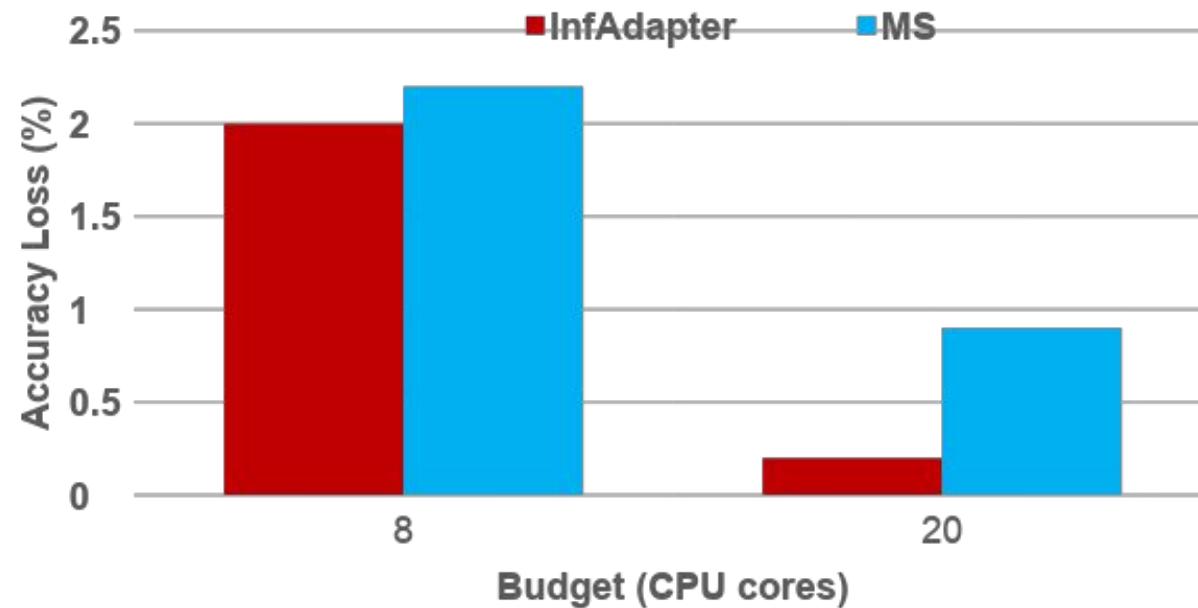
InfAdapter: Why?

Different throughputs with different model variants

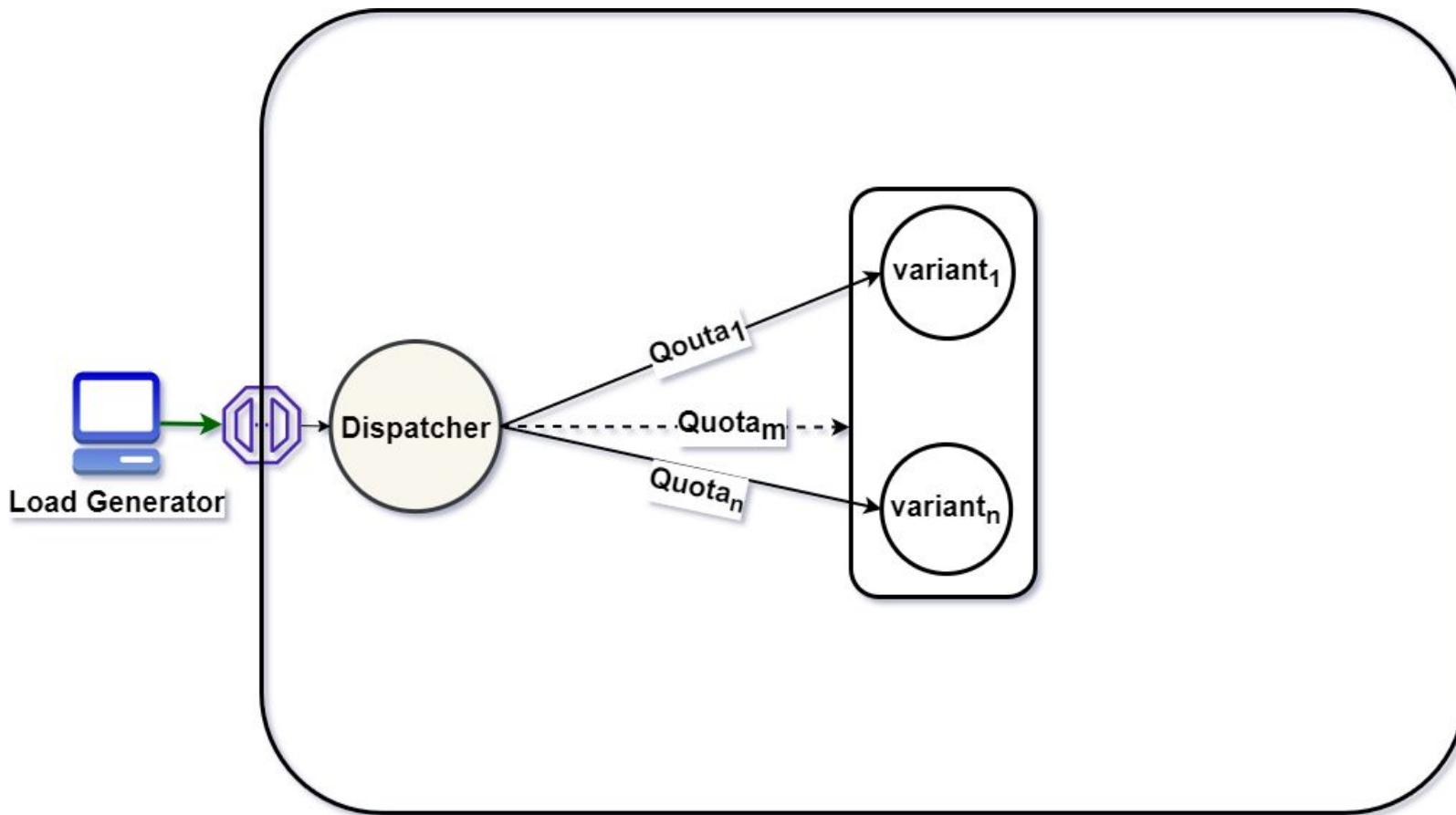


InfAdapter: Why?

Higher average accuracy by using multiple model variants

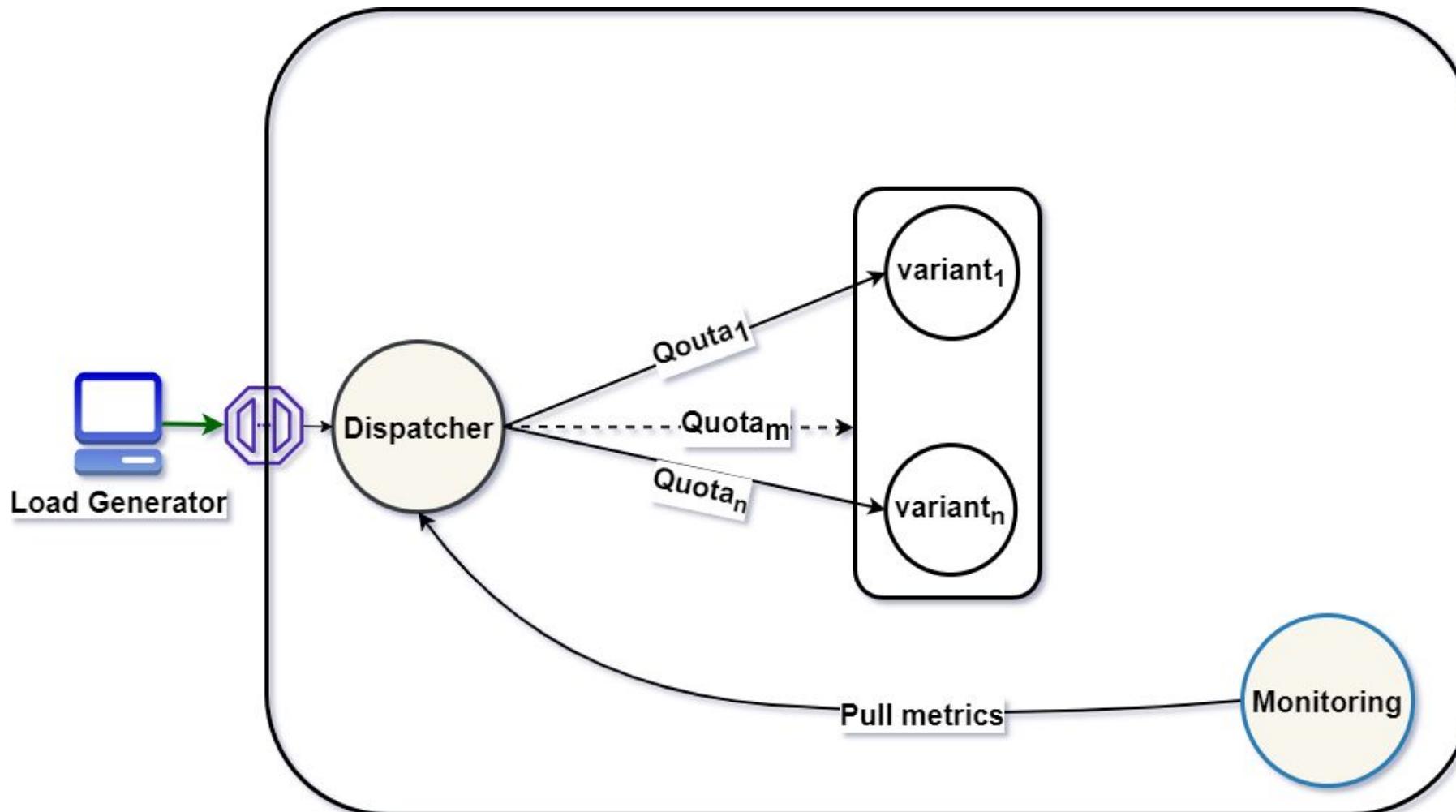


InfAdapter: How?

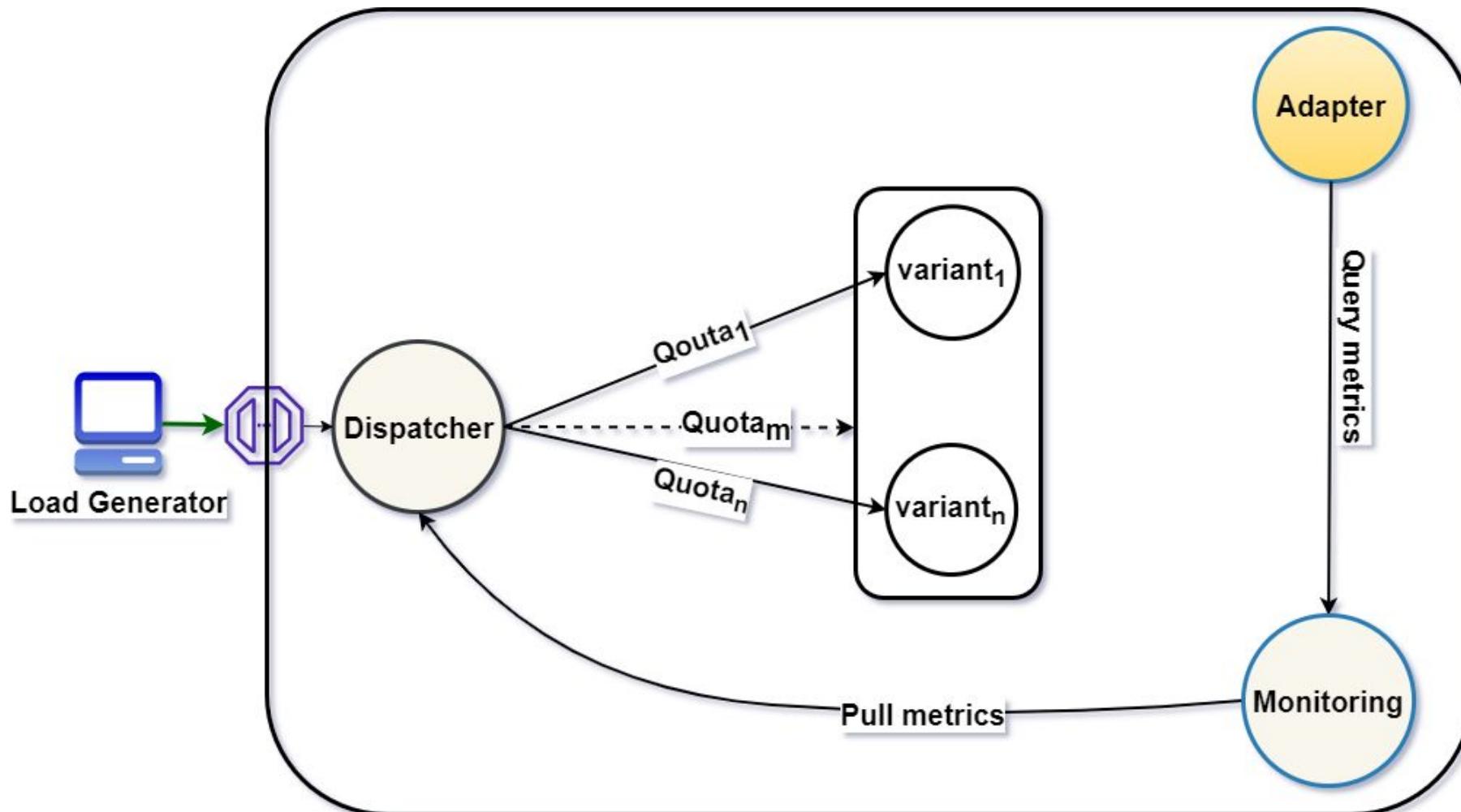


Selecting a subset of model variants, each having its own size
Meeting latency requirement for the predicted workload while maximizing accuracy
and minimizing cost

InfAdapter: Design



InfAdapter: Design



InfAdapter: Formulation

$$\max \quad \alpha \cdot AA - (\beta \cdot RC + \gamma \cdot LC)$$

InfAdapter: Formulation

$$\max \quad \alpha \cdot AA - (\beta \cdot RC + \gamma \cdot LC)$$

Maximizing Average Accuracy

InfAdapter: Formulation

$$\max \quad \alpha \cdot AA - (\beta \cdot RC + \gamma \cdot LC)$$

Maximizing Average Accuracy

Minimizing Resource and Loading Costs

InfAdapter: Formulation

$$\begin{aligned} \max \quad & \alpha \cdot AA - (\beta \cdot RC + \gamma \cdot LC) \\ \text{subject to} \quad & \lambda \leq \sum_{m \in M} th_m(n_m), \\ & \lambda_m \leq th_m(n_m) \\ & p_m(n_m) \leq L, \forall m \in M, \\ & RC \leq B, \\ & n_m \in \mathbb{W}, \forall m \in M. \end{aligned}$$

InfAdapter: Formulation

$$\begin{aligned} \max \quad & \alpha \cdot AA - (\beta \cdot RC + \gamma \cdot LC) \\ \text{subject to} \quad & \lambda \leq \sum_{m \in M} th_m(n_m), \\ & \lambda_m \leq th_m(n_m) \\ & p_m(n_m) \leq L, \forall m \in M, \\ & RC \leq B, \\ & n_m \in \mathbb{W}, \forall m \in M. \end{aligned}$$

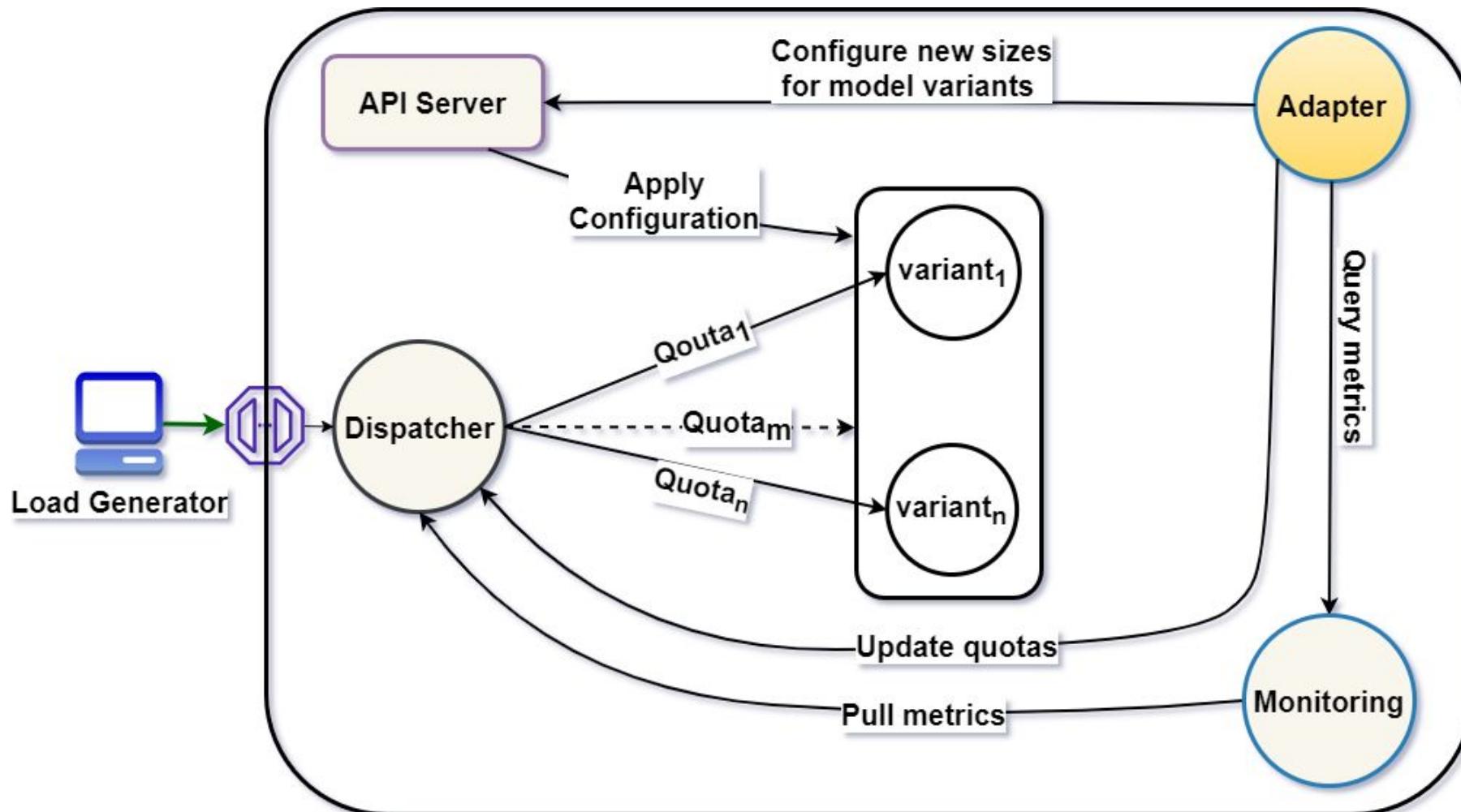
Supporting incoming workload

InfAdapter: Formulation

$$\begin{aligned} \max \quad & \alpha \cdot AA - (\beta \cdot RC + \gamma \cdot LC) \\ \text{subject to} \quad & \lambda \leq \sum_{m \in M} th_m(n_m), \\ & \lambda_m \leq th_m(n_m) \\ \text{Guaranteeing end-to-end latency} \quad & p_m(n_m) \leq L, \forall m \in M, \\ & RC \leq B, \\ & n_m \in \mathbb{W}, \forall m \in M. \end{aligned}$$

Supporting incoming workload

InfAdapter: Design



InfAdapter: Experimental evaluation setup

Twitter-trace sample (2022-08)

Baselines

Kubernetes VPA and adapted Model-Switching

Used models

Resnet18, Resnet34, Resnet50, Resnet101, Resnet152

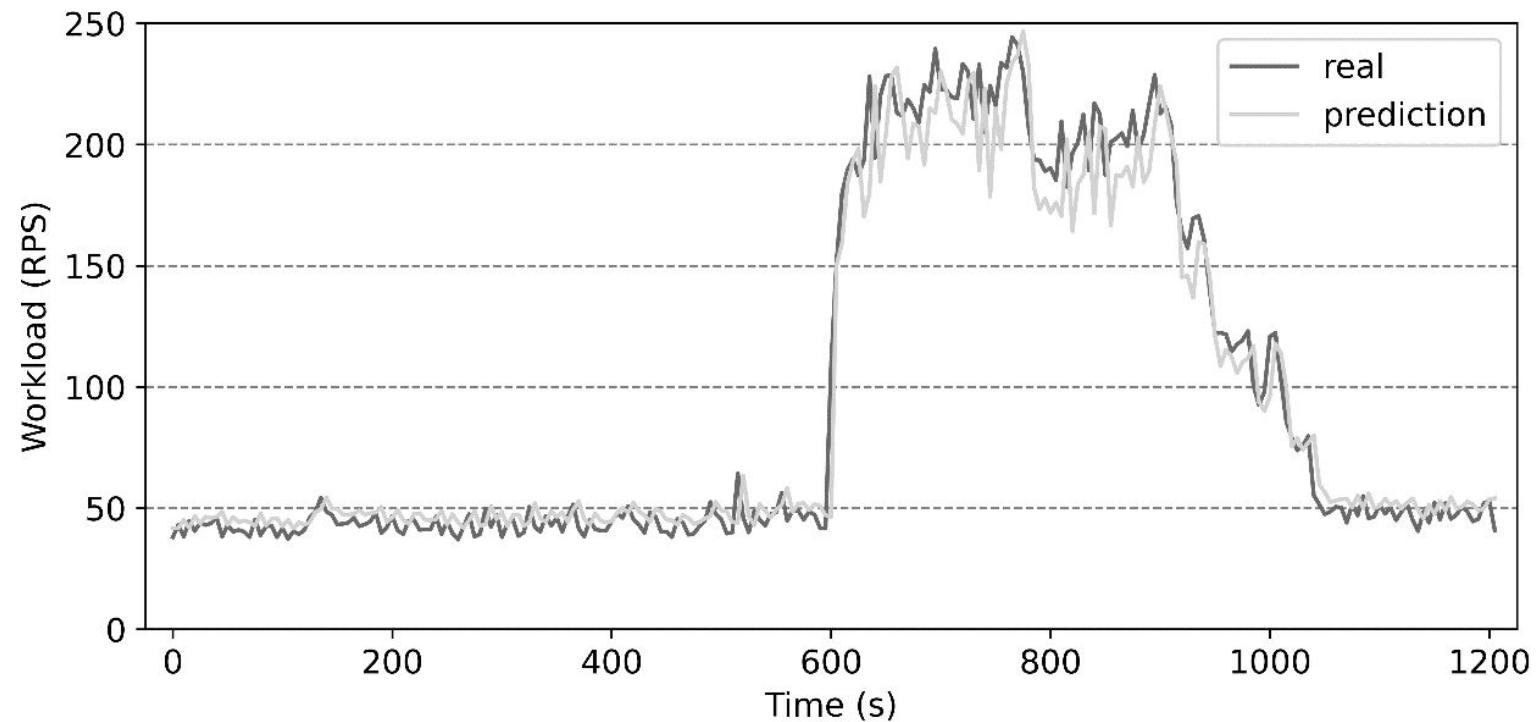
Interval adaptation

30 seconds

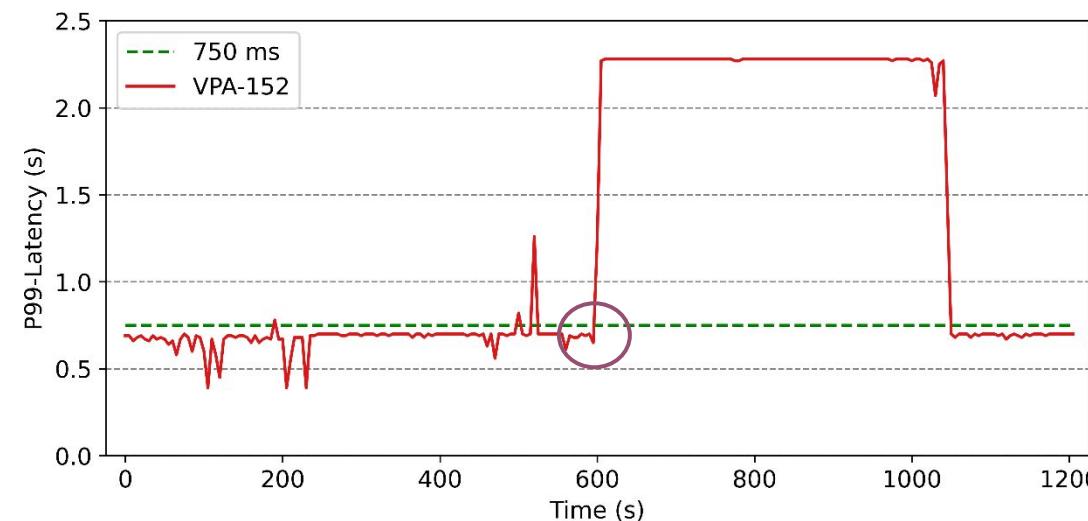
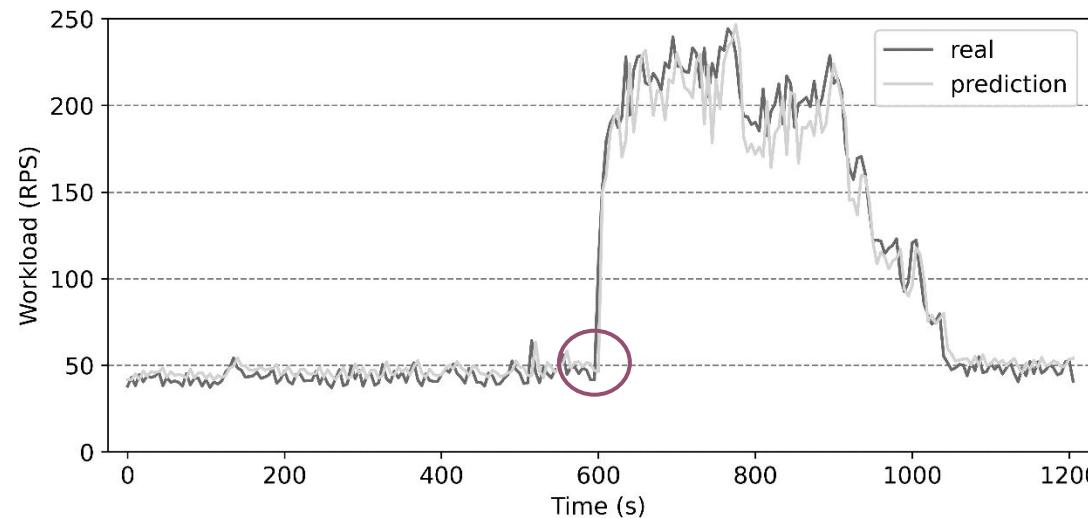
A Kubernetes cluster of 2 computing nodes

48 Cores, 192 GiB RAM

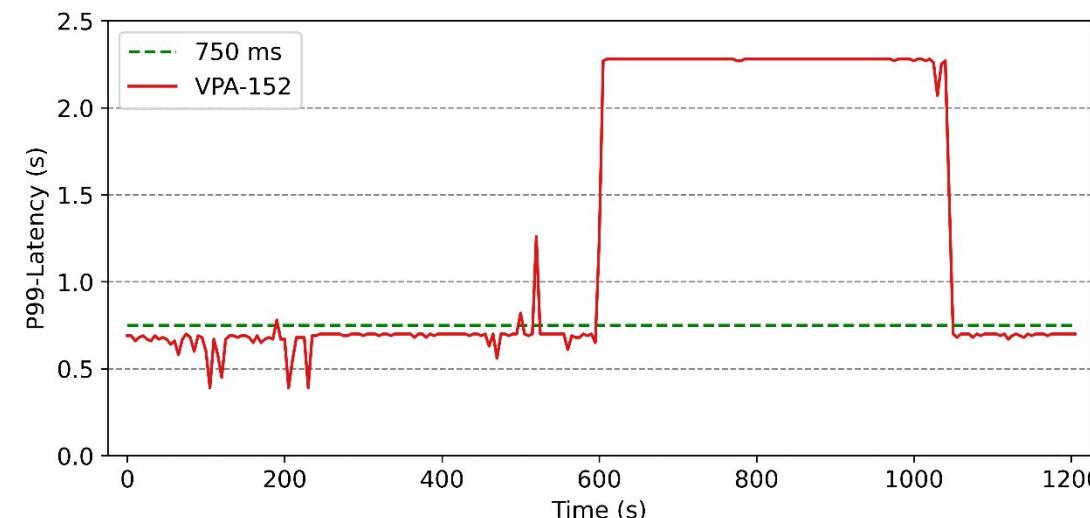
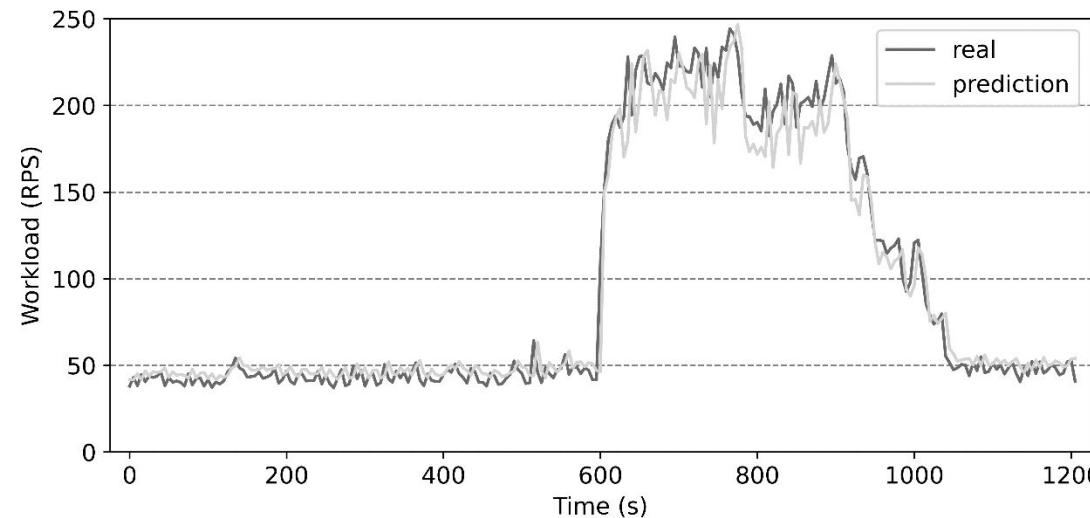
Workload Pattern



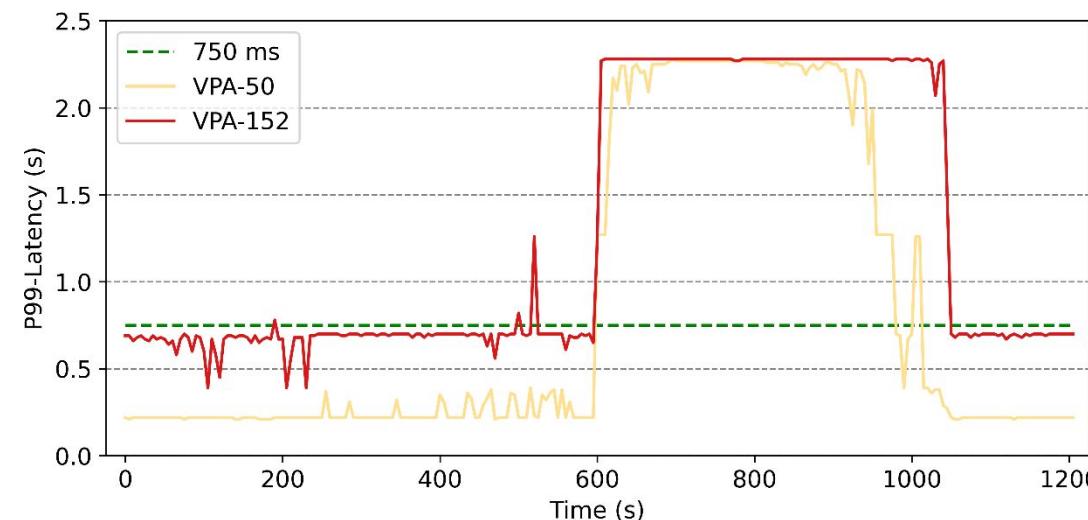
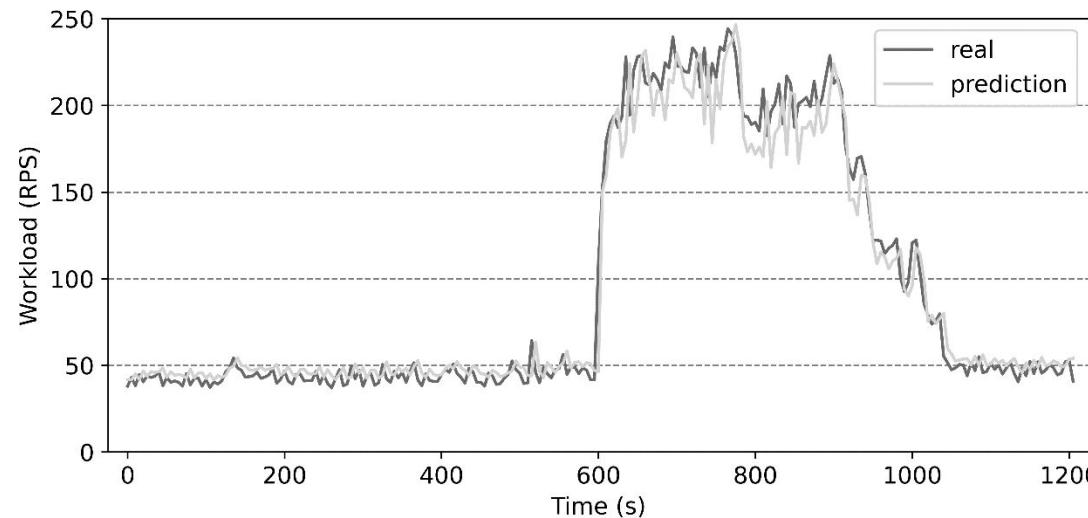
InfAdapter: P99-Latency evaluation



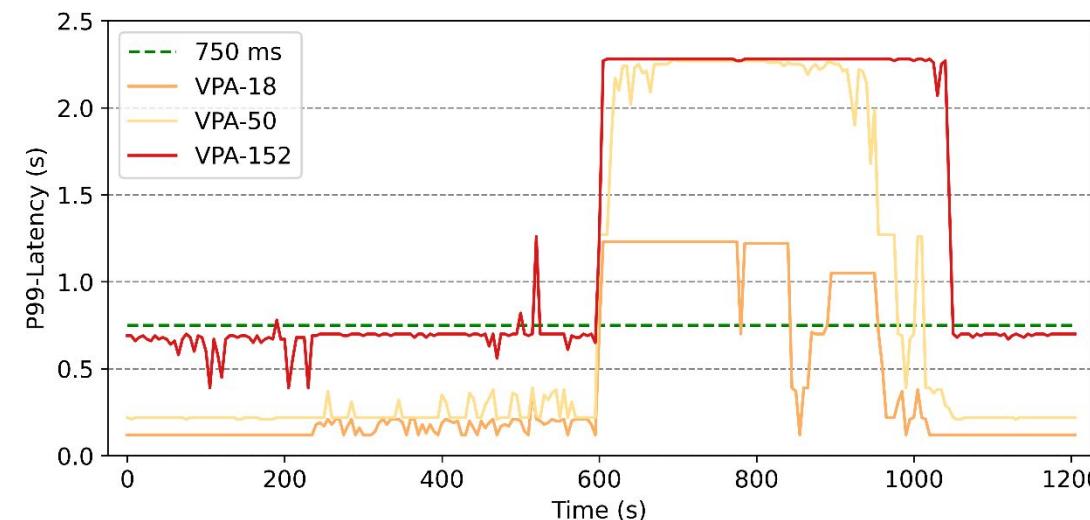
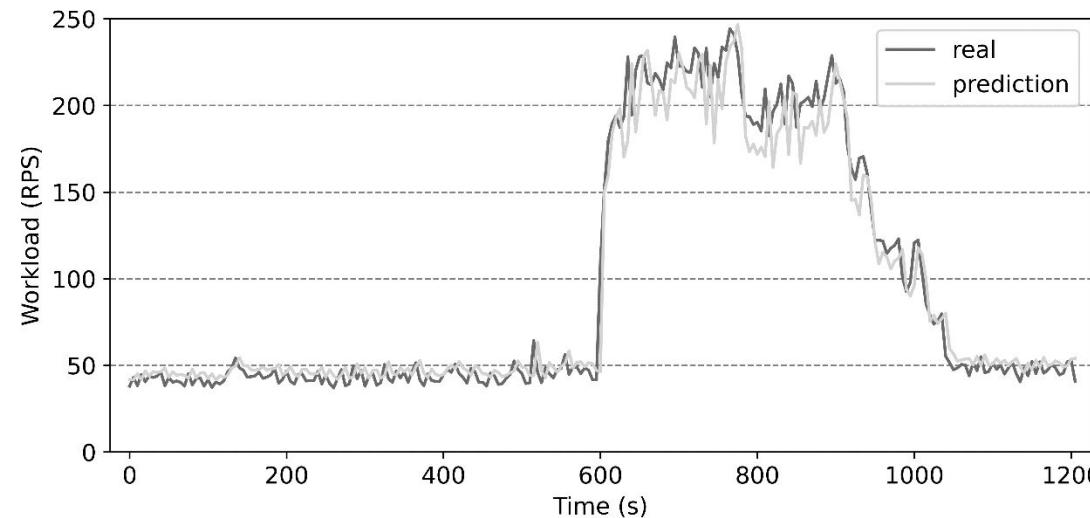
InfAdapter: P99-Latency evaluation



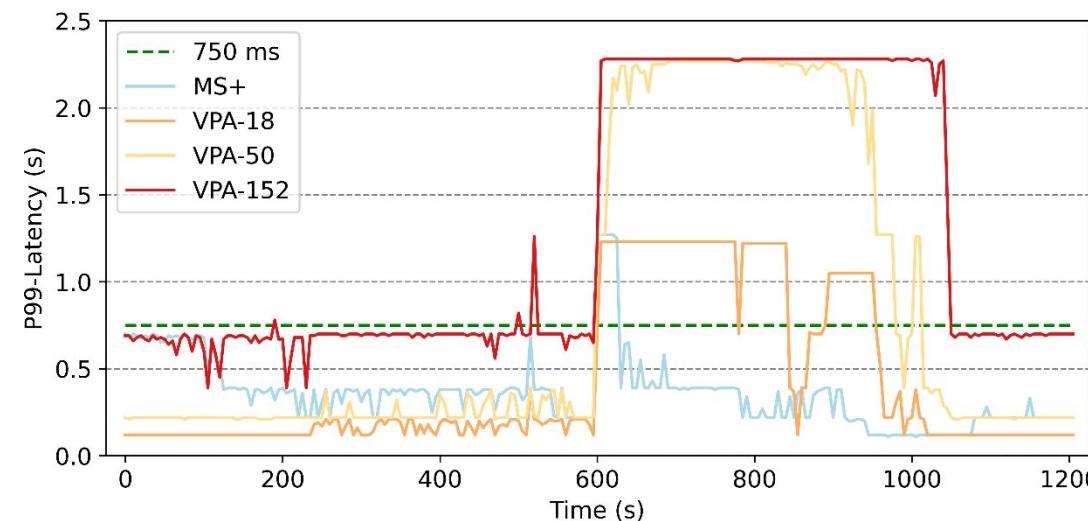
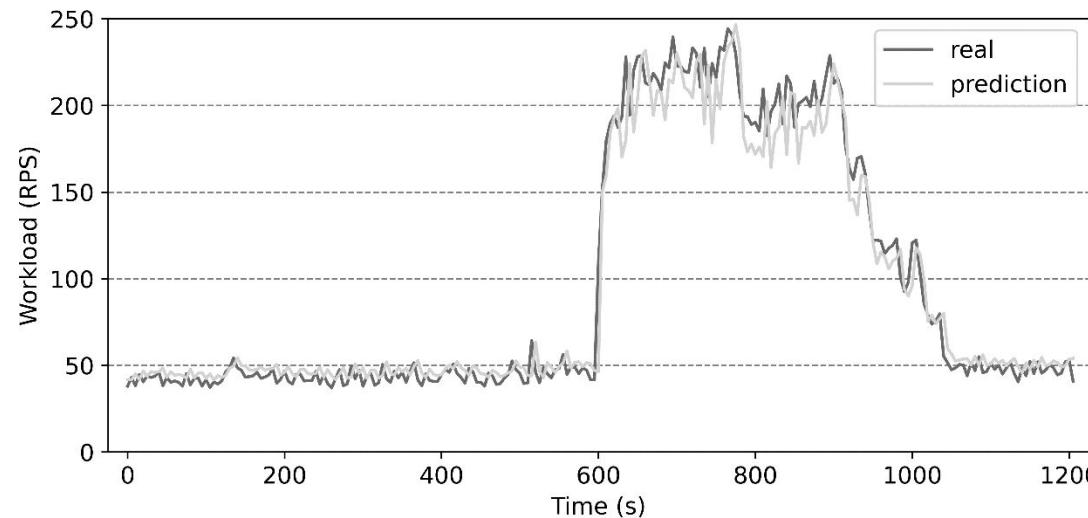
InfAdapter: P99-Latency evaluation



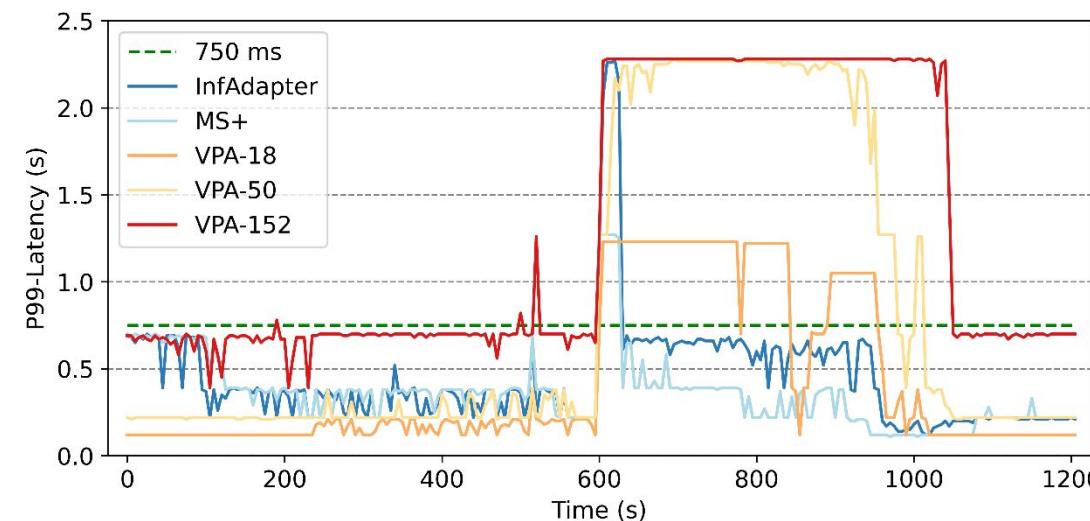
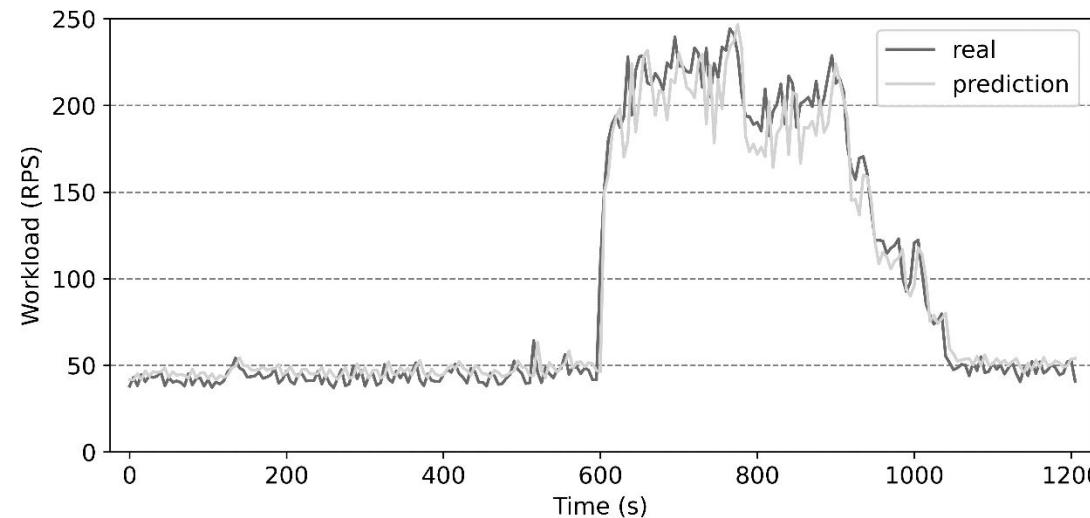
InfAdapter: P99-Latency evaluation



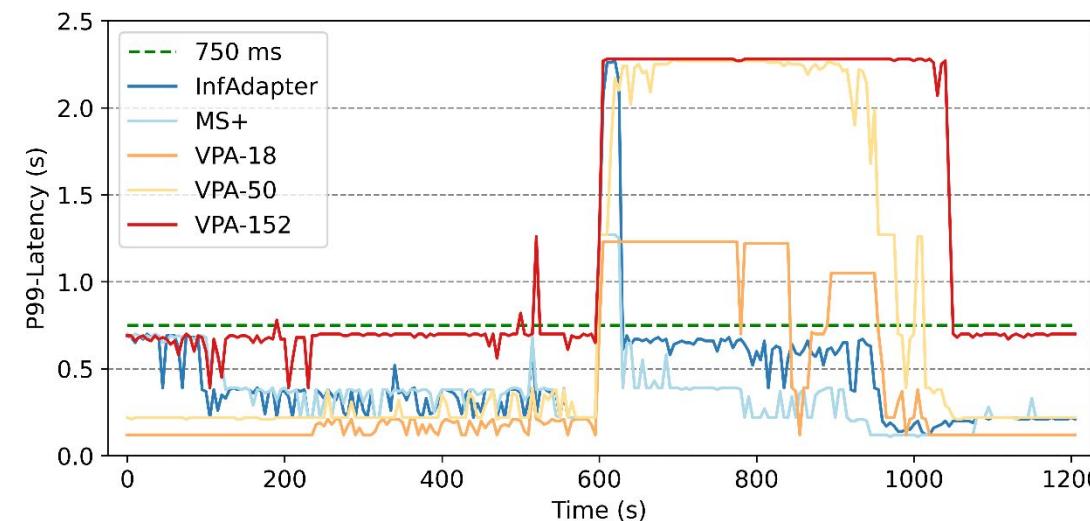
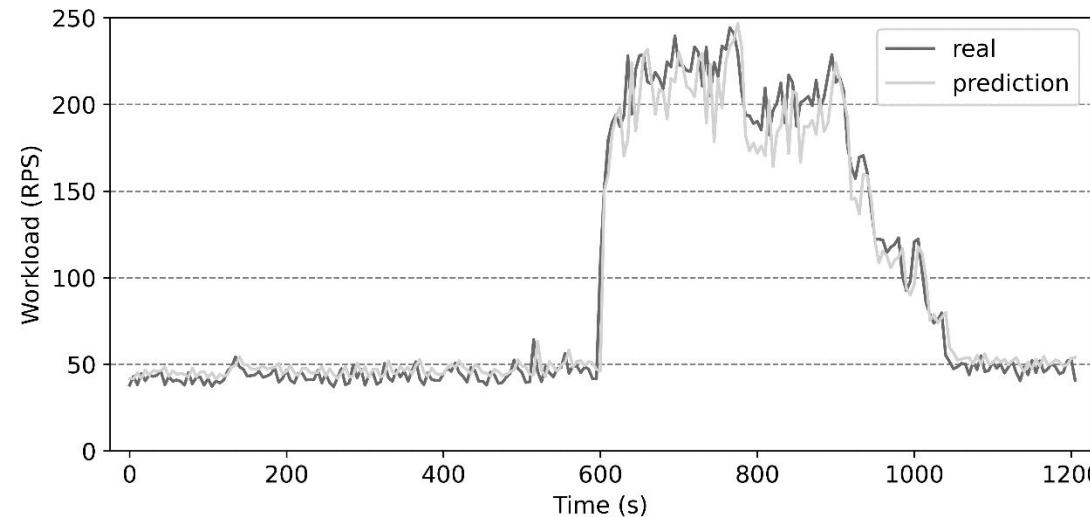
InfAdapter: P99-Latency evaluation



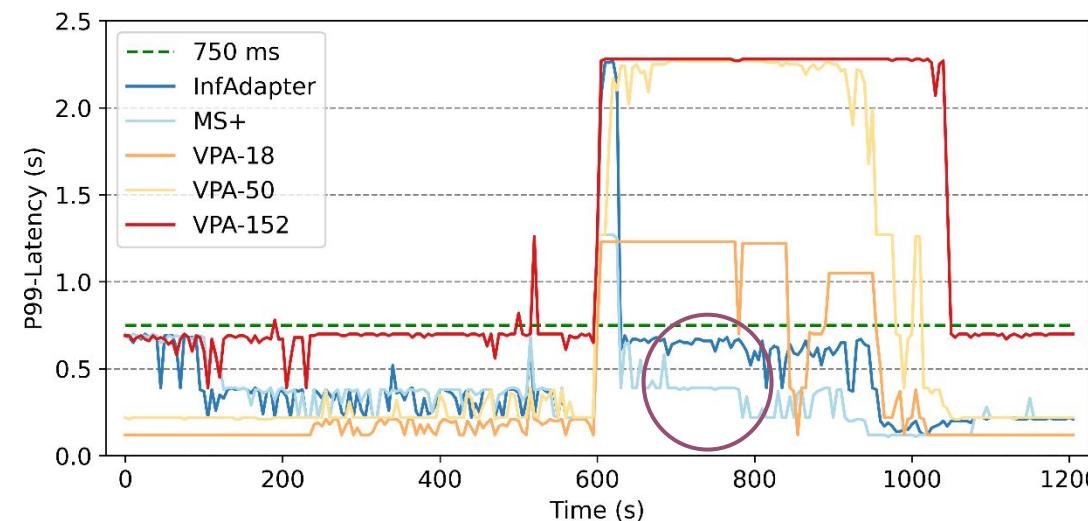
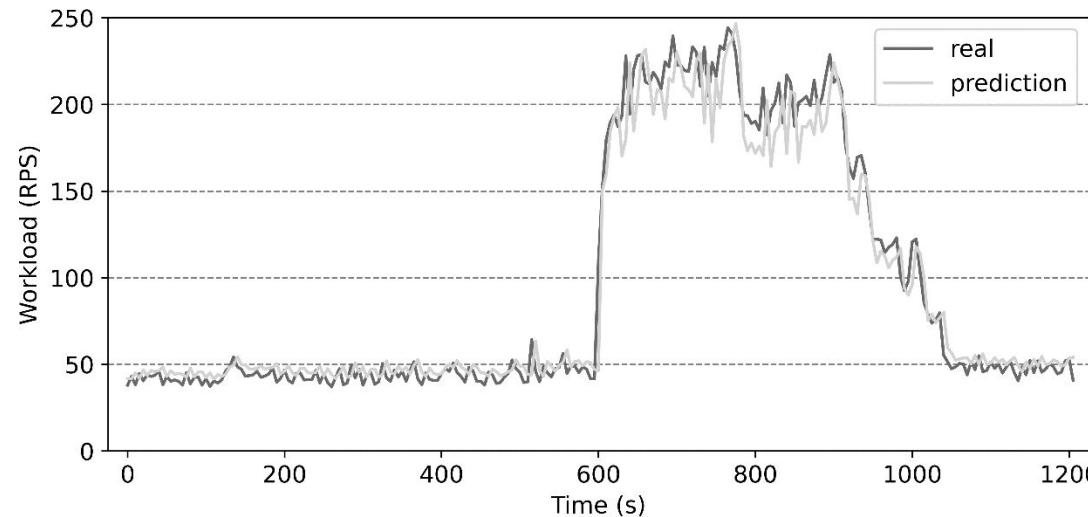
InfAdapter: P99-Latency evaluation



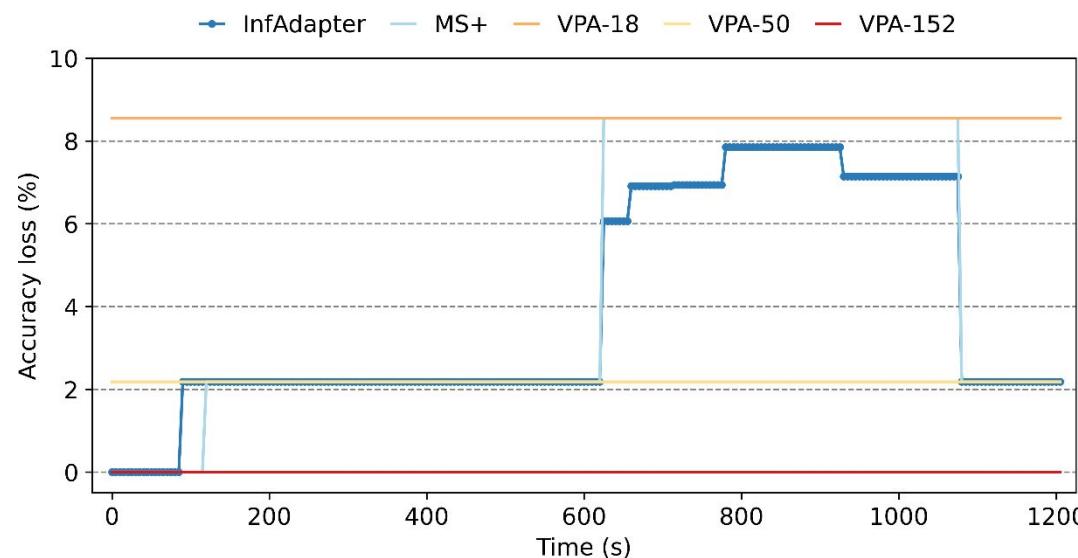
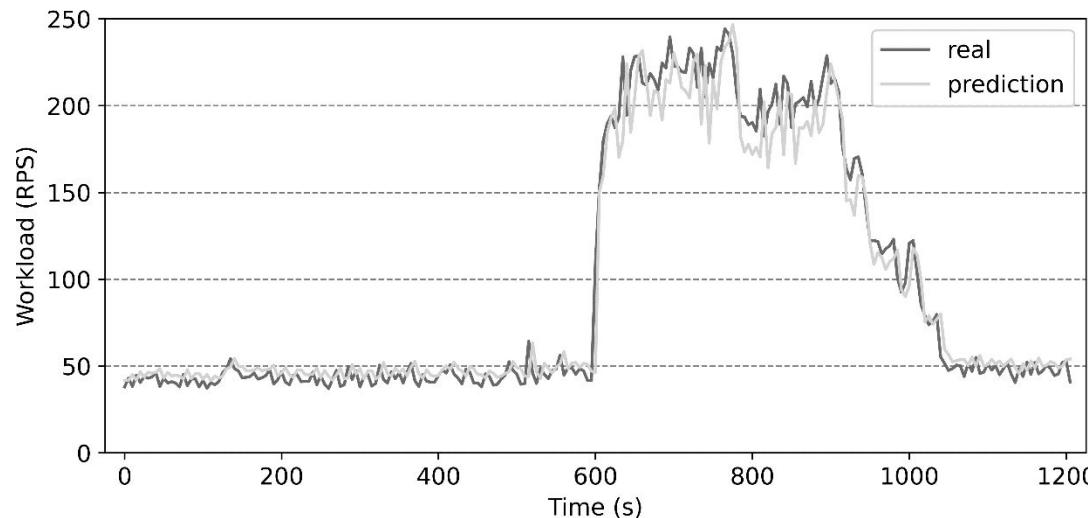
InfAdapter: P99-Latency evaluation



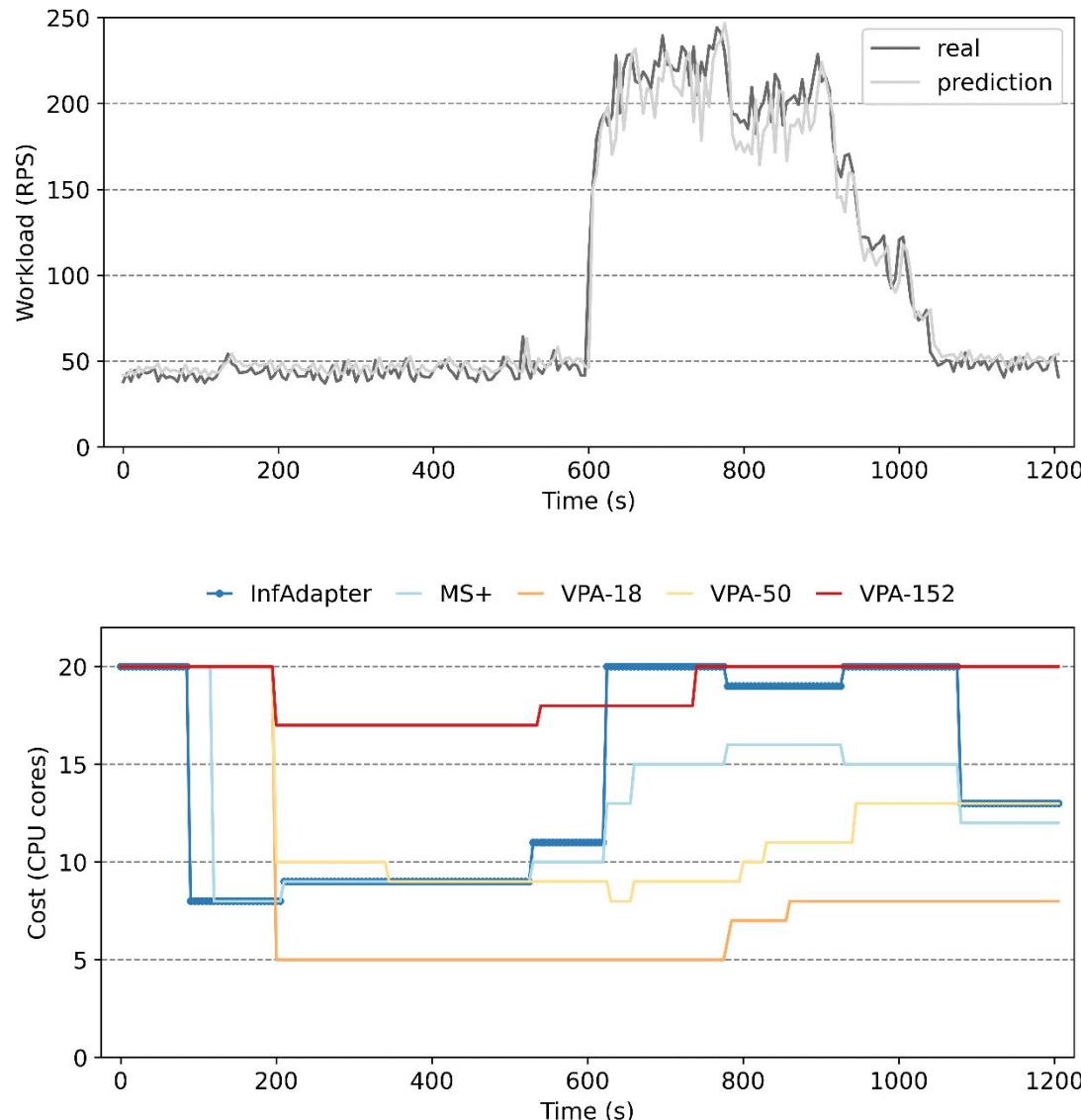
InfAdapter: P99-Latency evaluation



InfAdapter: Accuracy evaluation

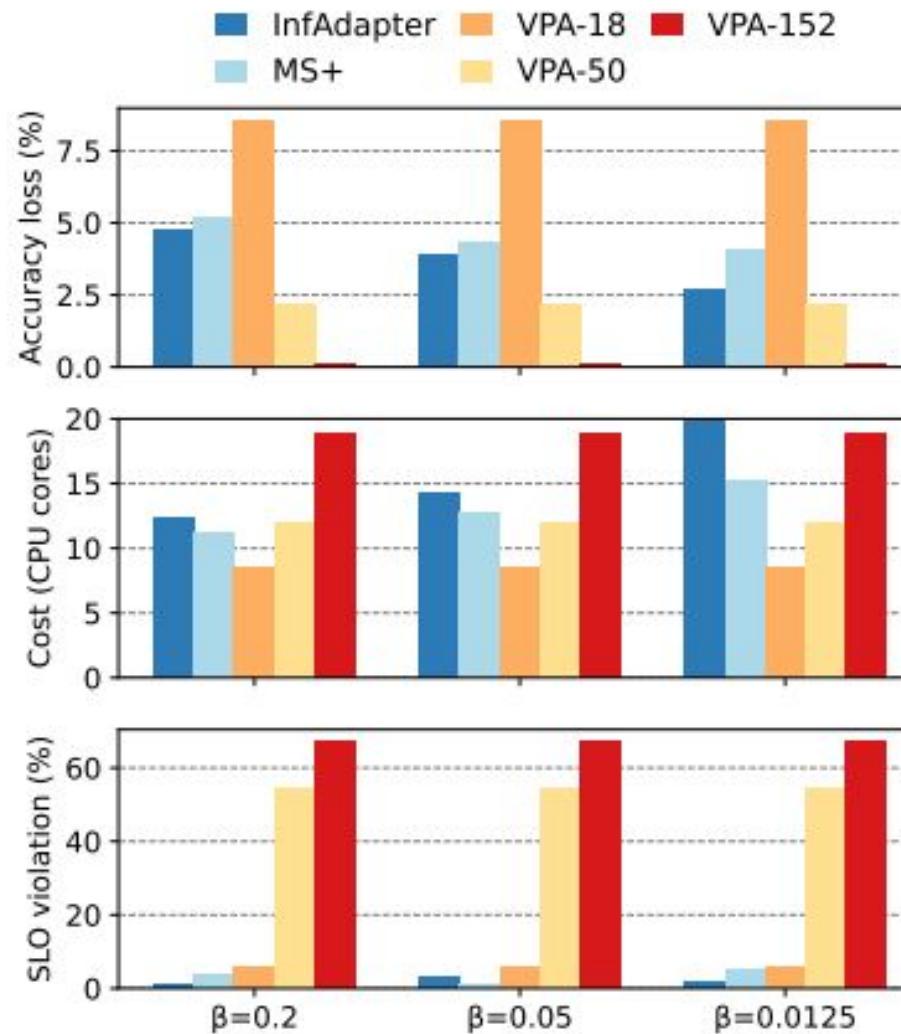


InfAdapter: Cost evaluation



InfAdapter: Experimental evaluation

Compare aggregated metrics of latency SLO violation, accuracy and cost with other works on different β values to see how they perform on different accuracy-cost trade-off



Takeaway



Inference Serving Systems should consider accuracy, latency, and cost at the same time.

Takeaway



Inference Serving Systems should consider accuracy, latency, and cost at the same time.



Model variants provide the opportunity to reduce resource costs while adapting to the dynamic workload.

Using a set of model variants simultaneously provides higher average accuracy compared to having one variant.

Takeaway



Inference Serving Systems should consider accuracy, latency, and cost at the same time.



Model variants provide the opportunity to reduce resource costs while adapting to the dynamic workload.

Using a set of model variants simultaneously provides higher average accuracy compared to having one variant.



InfAdapter!





<https://github.com/reconfigurable-ml-pipeline/InfAdapter>

ML inference services have strict & conflicting requirements

Highly Responsive! Cost-Efficient! Highly Accurate!



6

Takeaway



Inference Serving Systems should consider accuracy, latency, and cost at the same time.



Model variants provide the opportunity to reduce resource costs while adapting to the dynamic workload.

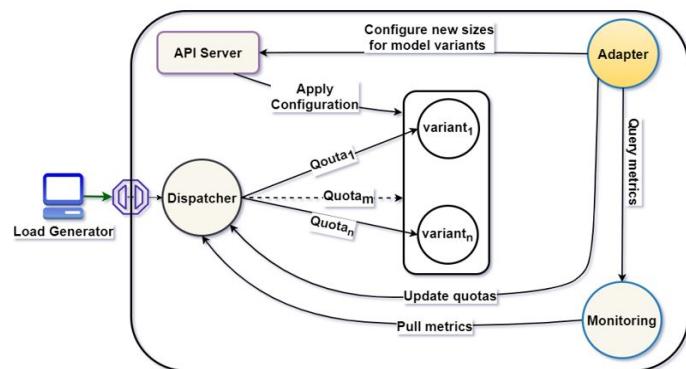
Using a set of model variants simultaneously provides higher average accuracy compared to having one variant.



InfAdapter!

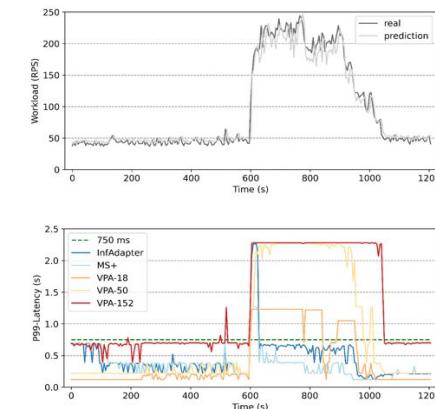
41

InfAdapter: Design



29

InfAdapter: P99-Latency evaluation



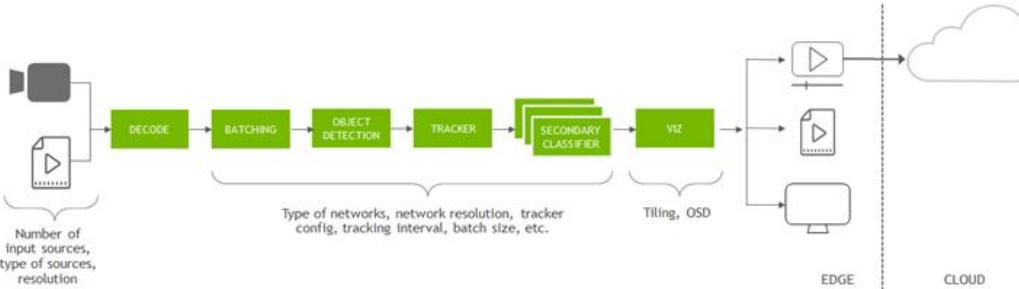
36

IPA: Inference Pipeline Adaptation to Achieve High Accuracy and Cost-Efficiency

Saeid Ghafouri, Kamran Razavi, Mehran Salmani, Alireza Sanaee
Tania Lorida-Botran, Lin Wang, Joseph Doyle, Pooyan Jamshidi

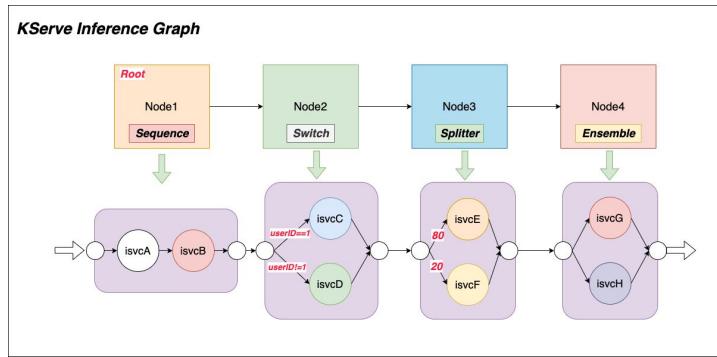


Inference Pipeline

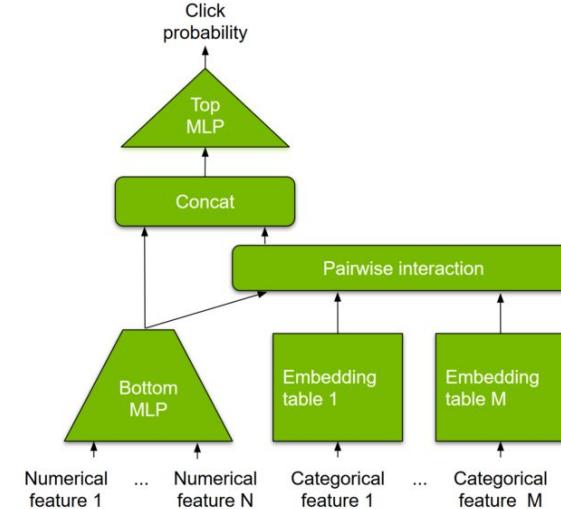
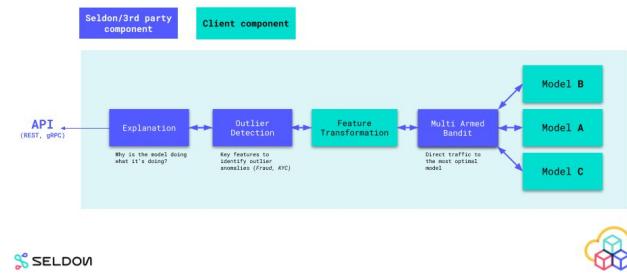


Video Pipelines

Source:
https://docs.nvidia.com/metropolis/deepstream/5.0/dev-guide/index.html#page/DeepStream_Development_Guide/deepstream_overview.html

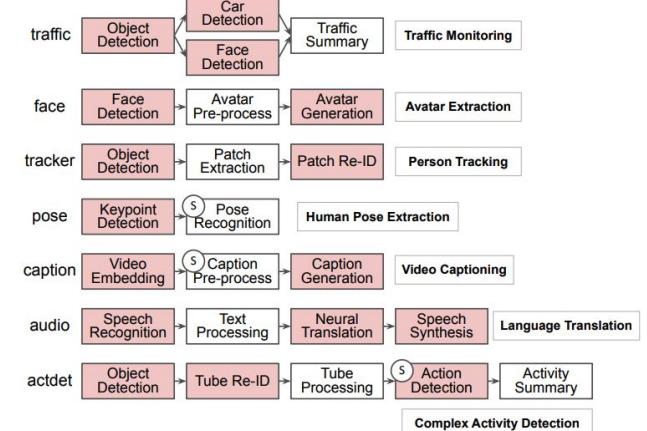


Seldon Core: Define Inference Pipeline

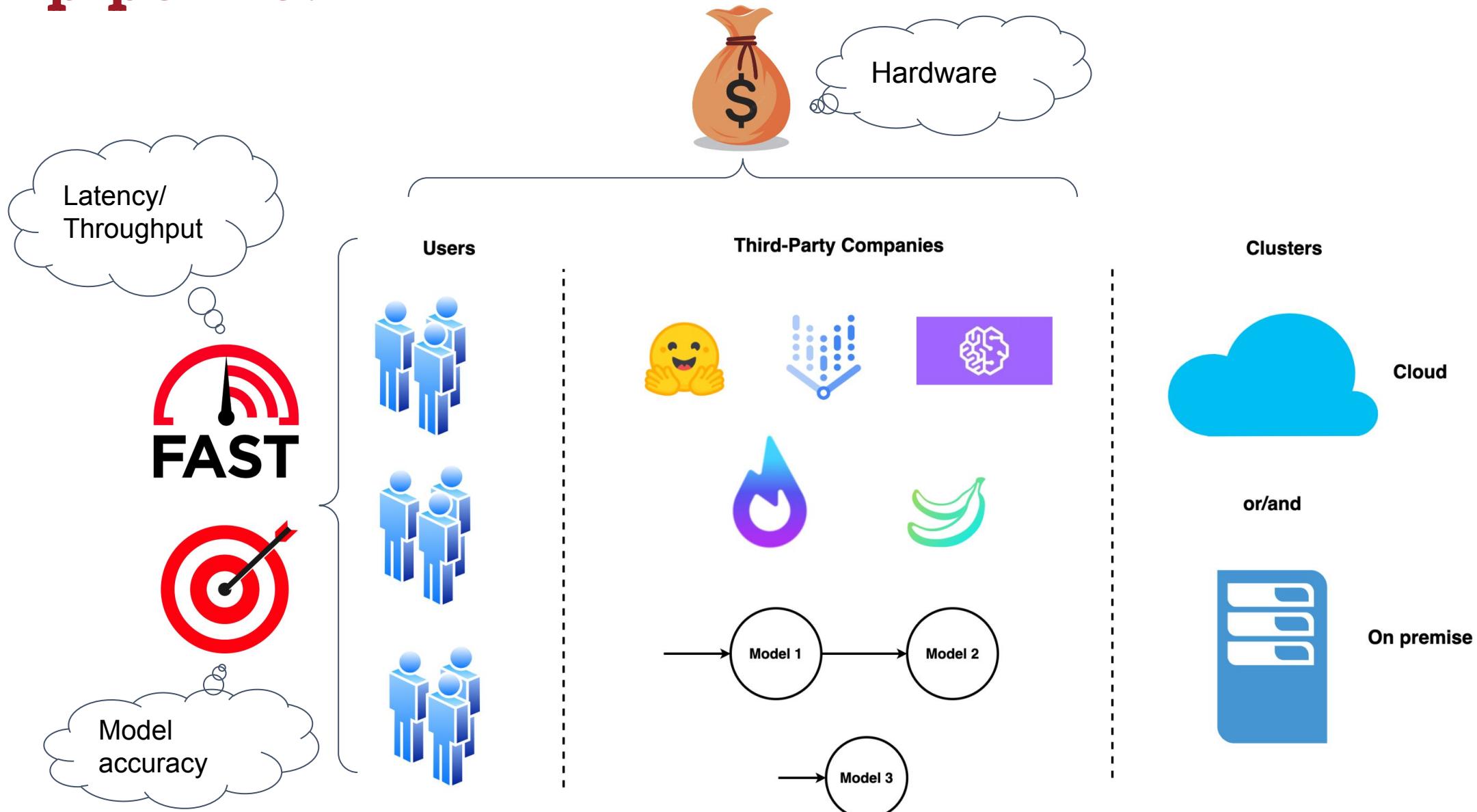


Recommender Systems

Source:
<https://developer.nvidia.com/blog/optimizing-dlrm-on-nvidia-gpus/>



What should be characteristic of an inference pipeline?



Autoscaling

Previous works have used auto scaling for cost optimization of inference pipeline

InferLine: Latency-Aware Provisioning and Scaling for Prediction Serving Pipelines

Daniel Crankshaw
Microsoft Research
dacranks@microsoft.com

Corey Zumar
Databricks
czumar@berkeley.edu

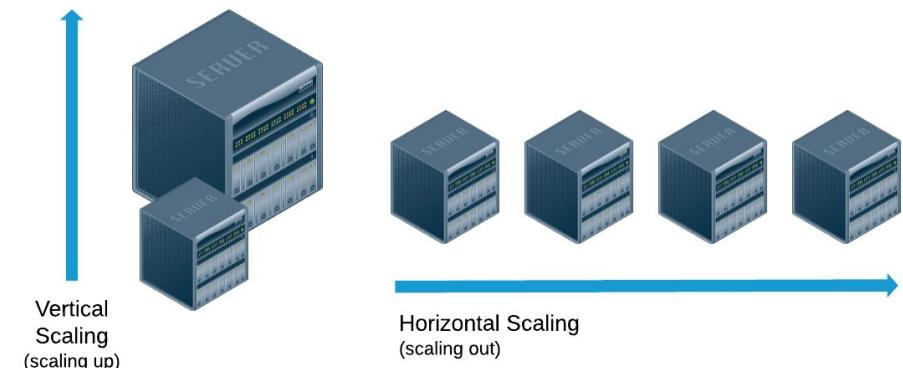
Gur-Eyal Sela
UC Berkeley
ges@berkeley.edu

Ion Stoica
UC Berkeley, Anyscale
istoica@berkeley.edu

Alexey Tumanov
Georgia Tech
atumanov@gatech.edu

Xiangxi Mo
UC Berkeley, Anyscale
xmo@berkeley.edu

Joseph Gonzalez
UC Berkeley
jegonzal@berkeley.edu



FA2: Fast, Accurate Autoscaling for Serving Deep Learning Inference with SLA Guarantees

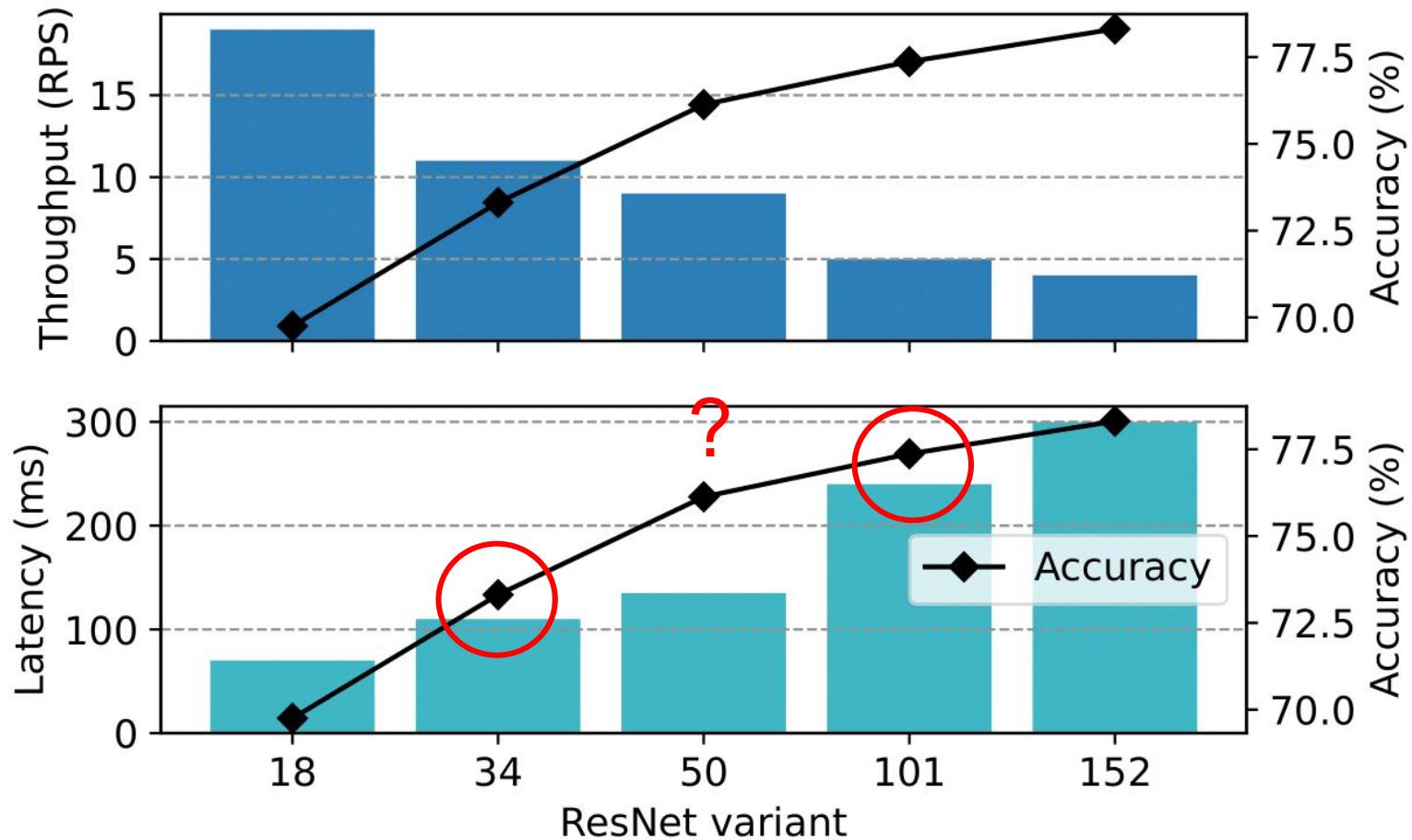
Kamran Razavi[†], Manisha Luthra[†], Boris Koldehofe^{†,‡}, Max Mühlhäuser[†], Lin Wang^{†,§}

[†]Technische Universität Darmstadt

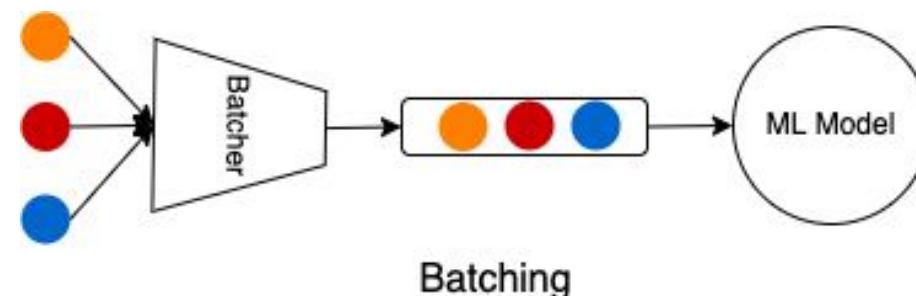
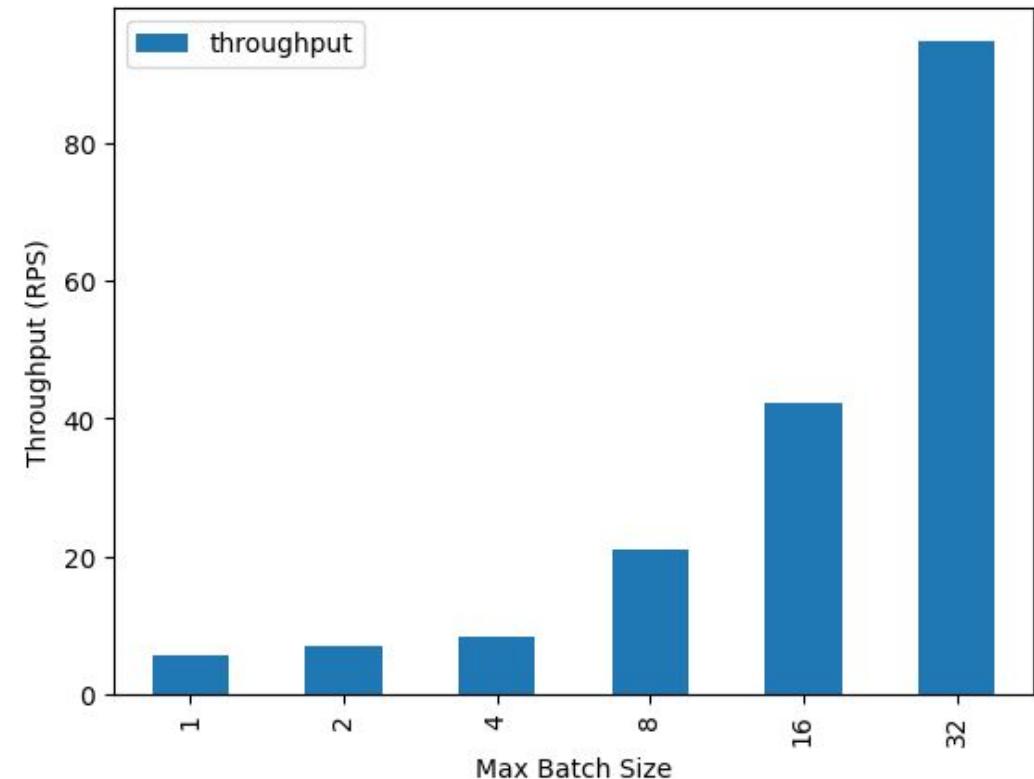
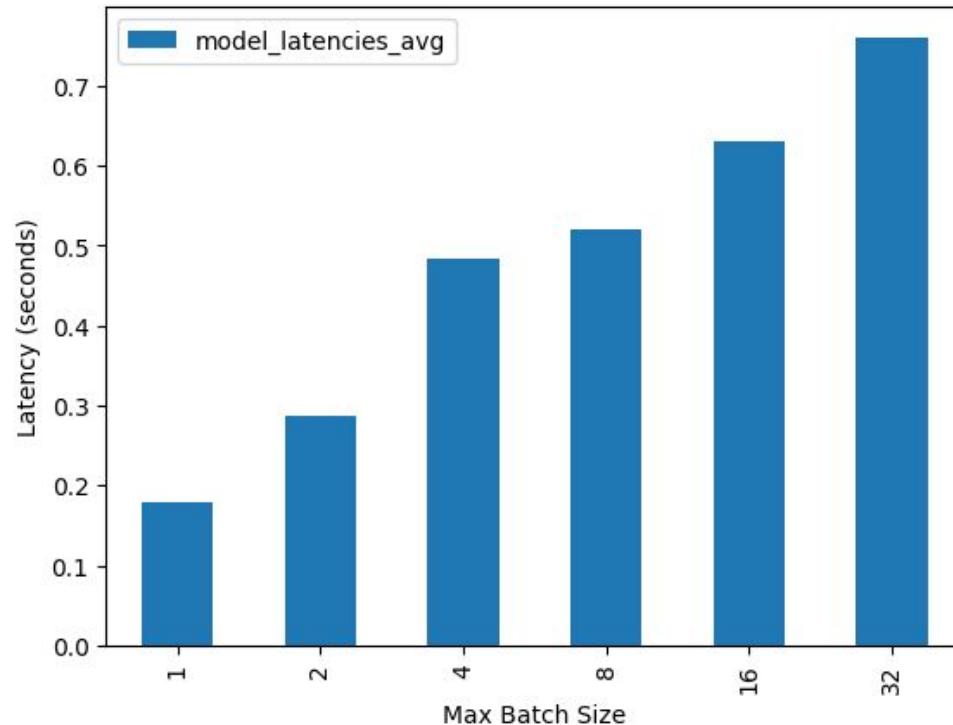
[‡]University of Groningen

[§]Vrije Universiteit Amsterdam

Is only scaling enough?

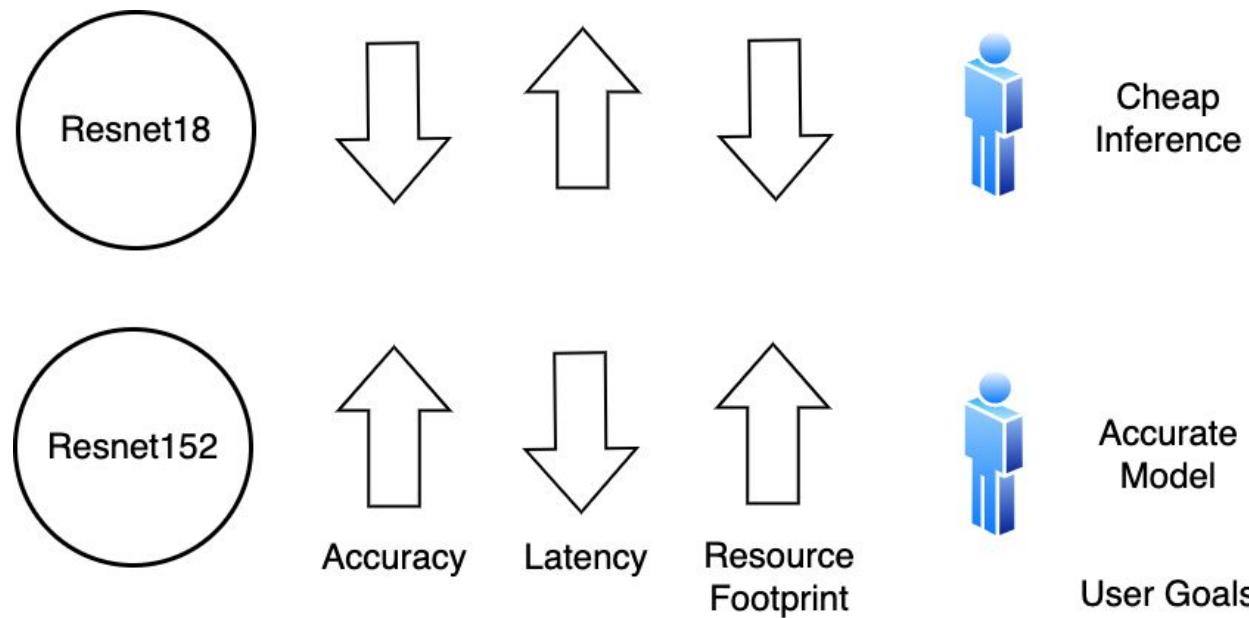


Effect of Batching

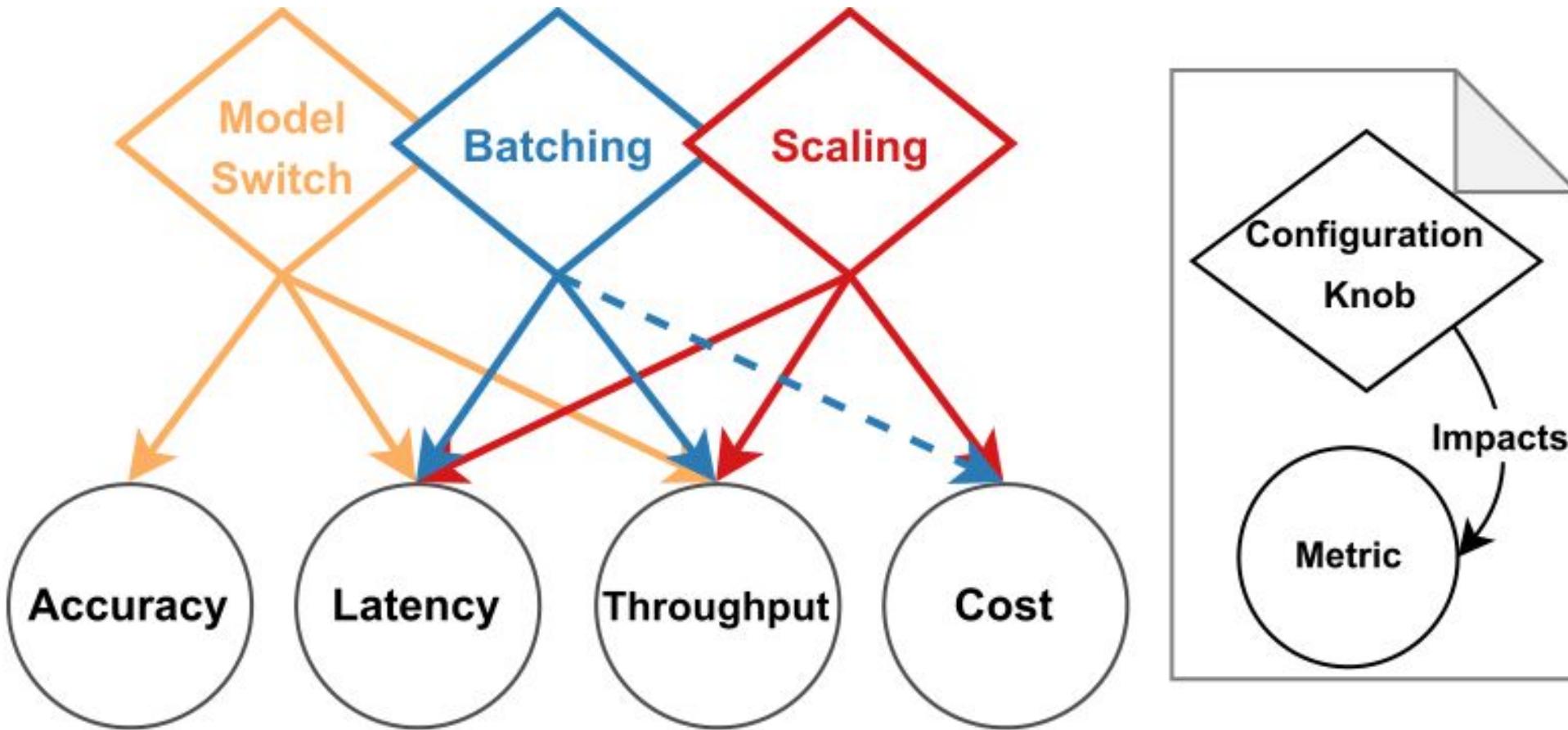


How to navigate Accuracy/latency trade off? Model Variants and Model Switching!

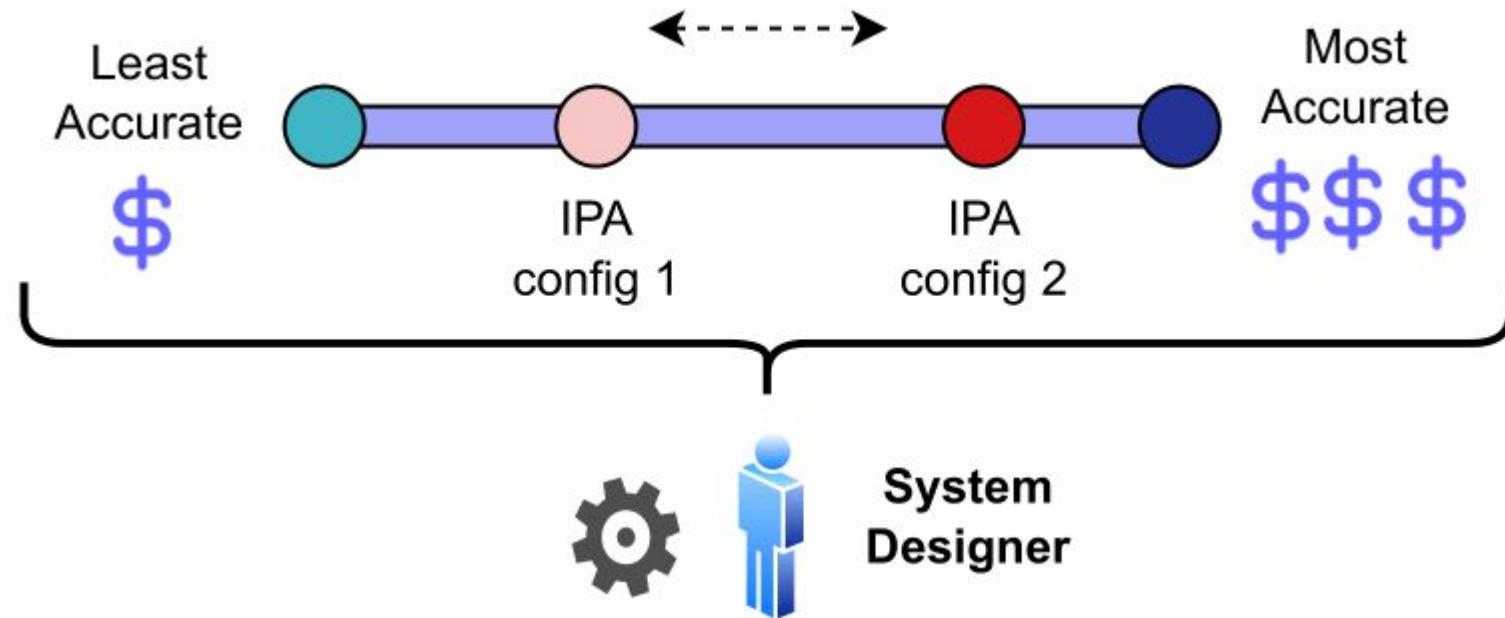
Previous works INFaaS and Model-Switch have proven that there is a big a latency-accuracy-resource footprint tradeoffs of models trained for the same task



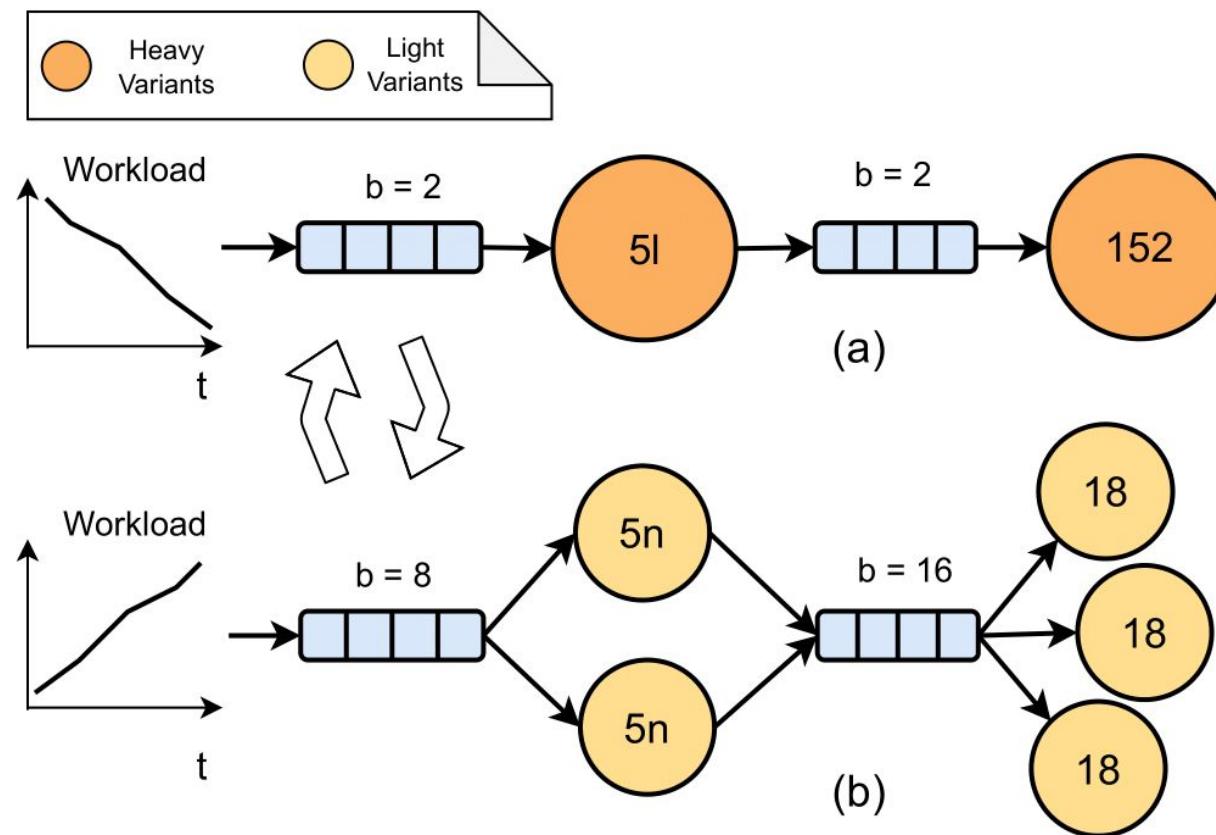
Search Space



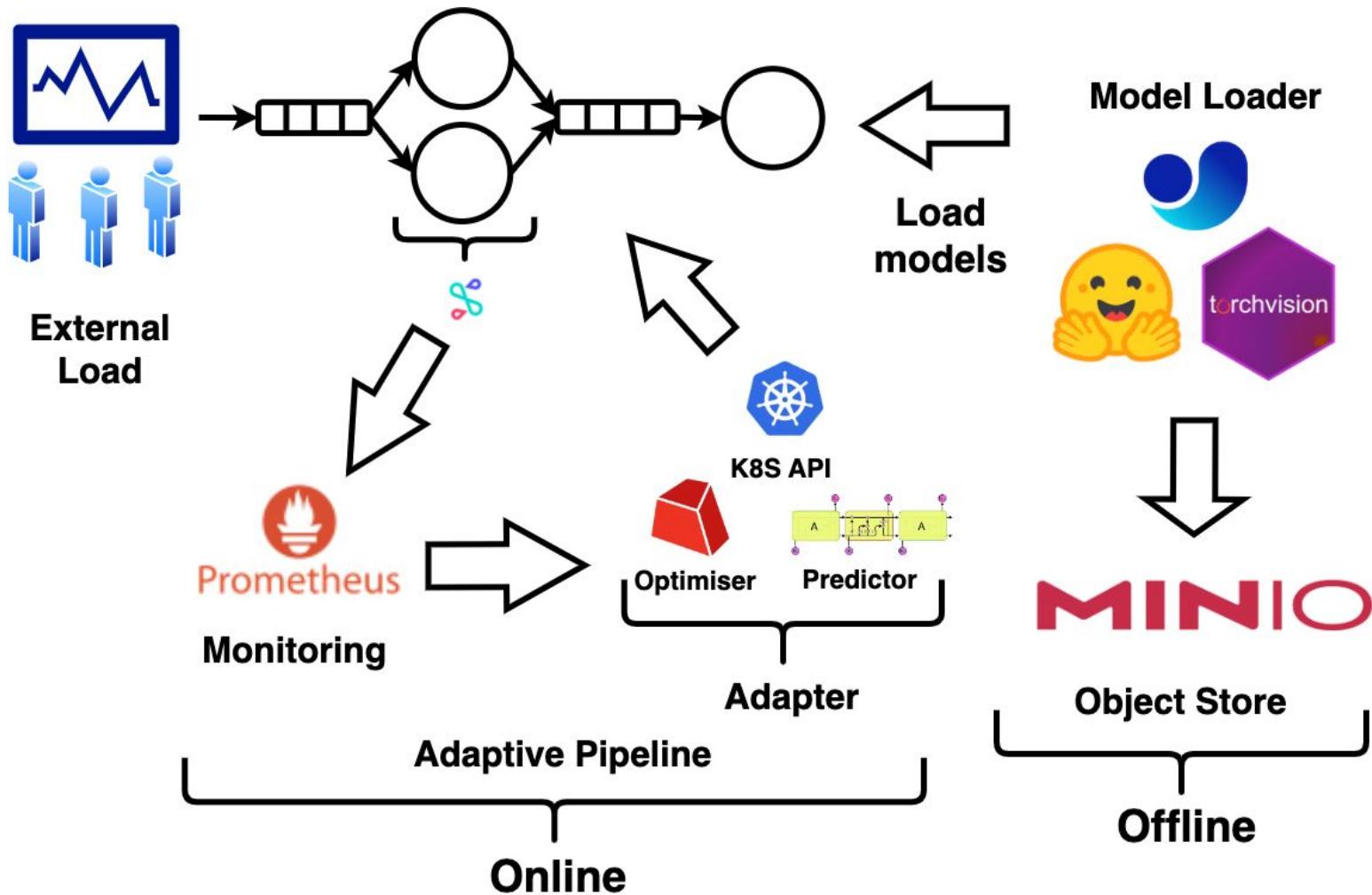
Goal: Providing a flexible inference pipeline



Snapshot of the System



System Design



Problem Formulation

$$f(n, s, I) = \alpha \sum_{s \in P} \left(\sum_{m \in M_s} a_{s,m} \cdot I_{s,m} \right)$$

Accuracy Objective

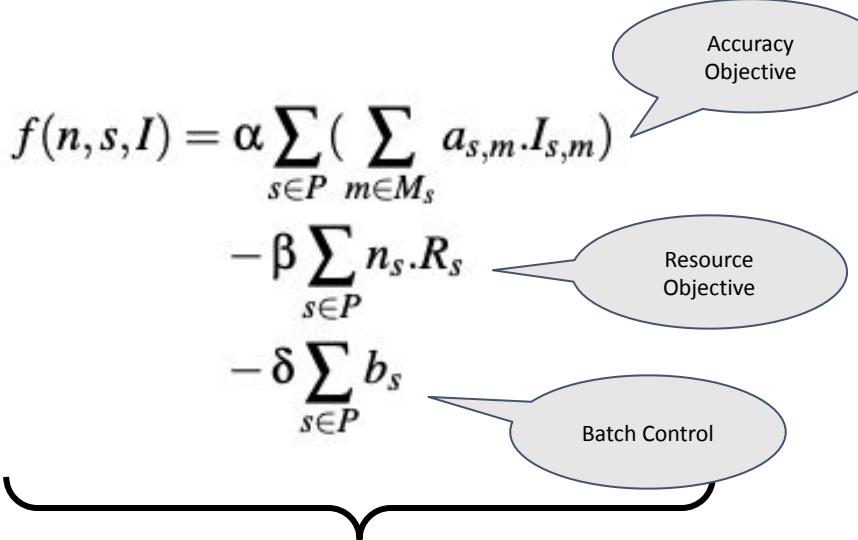
$$- \beta \sum_{s \in P} n_s \cdot R_s$$

Resource Objective

$$- \delta \sum_{s \in P} b_s$$

Batch Control

Objective function



$$\begin{aligned} & \max \quad f(n, s, I) \\ \text{subject to} \quad & \sum_{s \in P} l_s(b_s) + q_s(b_s) \leq SLA_P, \\ & \text{if } I_{s,m} = 1, \text{ then} \\ & n_s \cdot h_s(b_s) \geq \lambda_p, \quad \forall s \in P \\ & \sum_{m \in M_s} I_{s,m} = 1, \quad \forall s \in P \\ & n_s, b_s \in \mathbb{Z}^+, \quad I_{s,m} \in \{0, 1\}, \end{aligned}$$

Latency SLA

Throughput Constraint

One active model per node

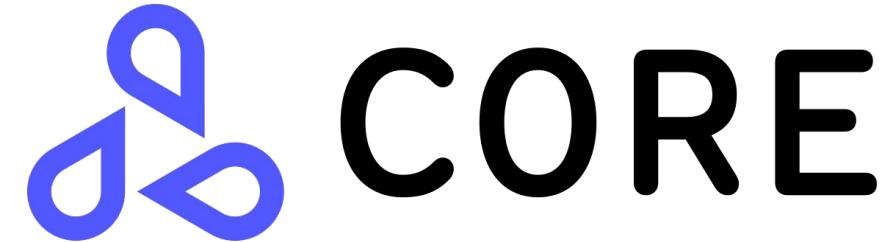
Implementation and Experimental Setup

How to navigate Model Variants



kubernetes

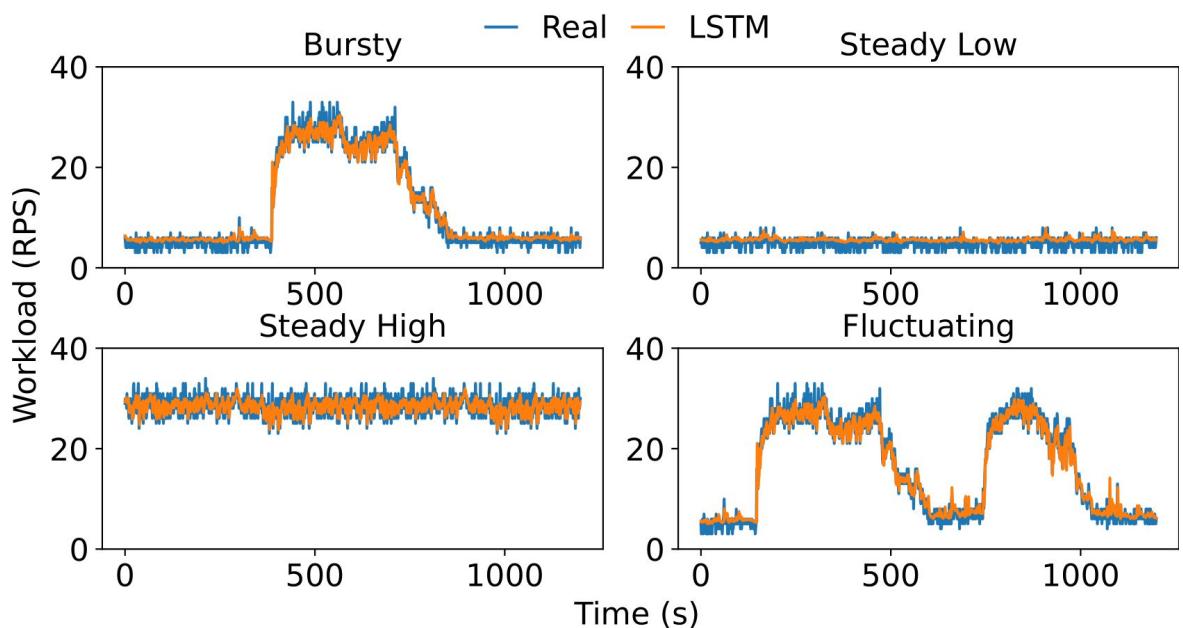
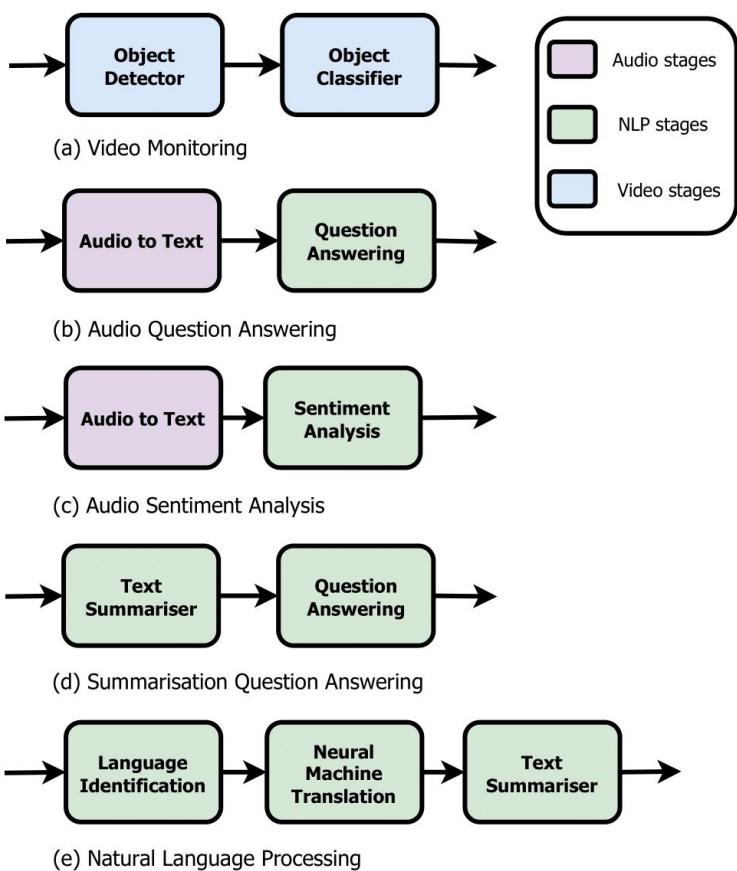
1. Industry standard
2. Used in recent research
3. Complete set of autoscaling, scheduling, observability tools (e.g. CPU usage)
4. APIs for changing the current AutoScaling algorithms



1. Industry standard ML server
2. Have the ability make inference graph
3. Rest and GRPC endpoints
4. Have many of the features we need like monitoring stack out of the box

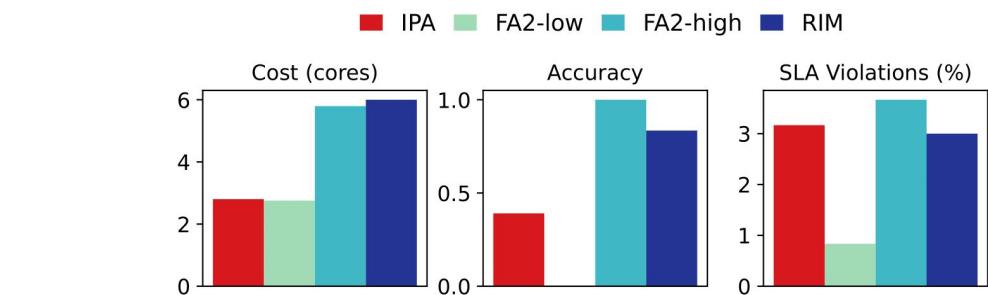
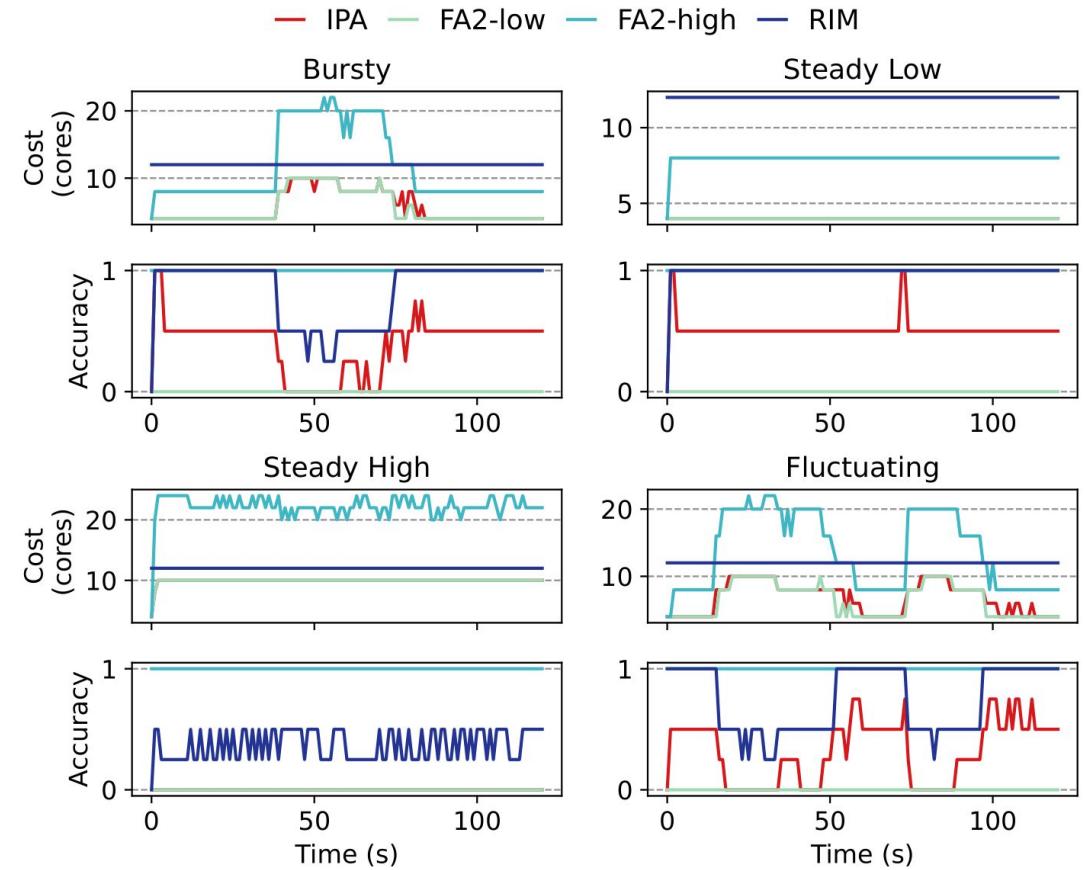
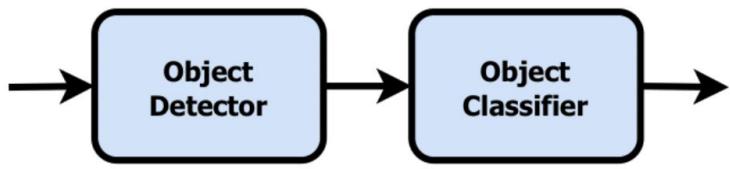
Experimental Setup

- A six node Kubernetes cluster

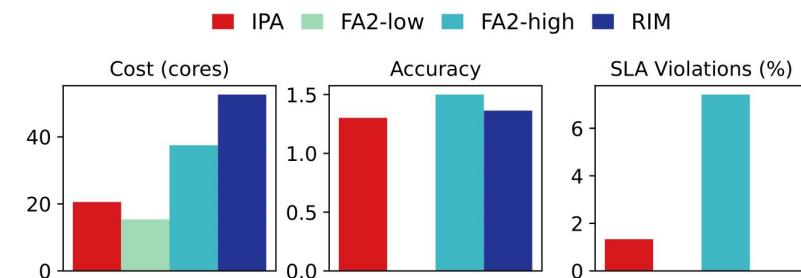
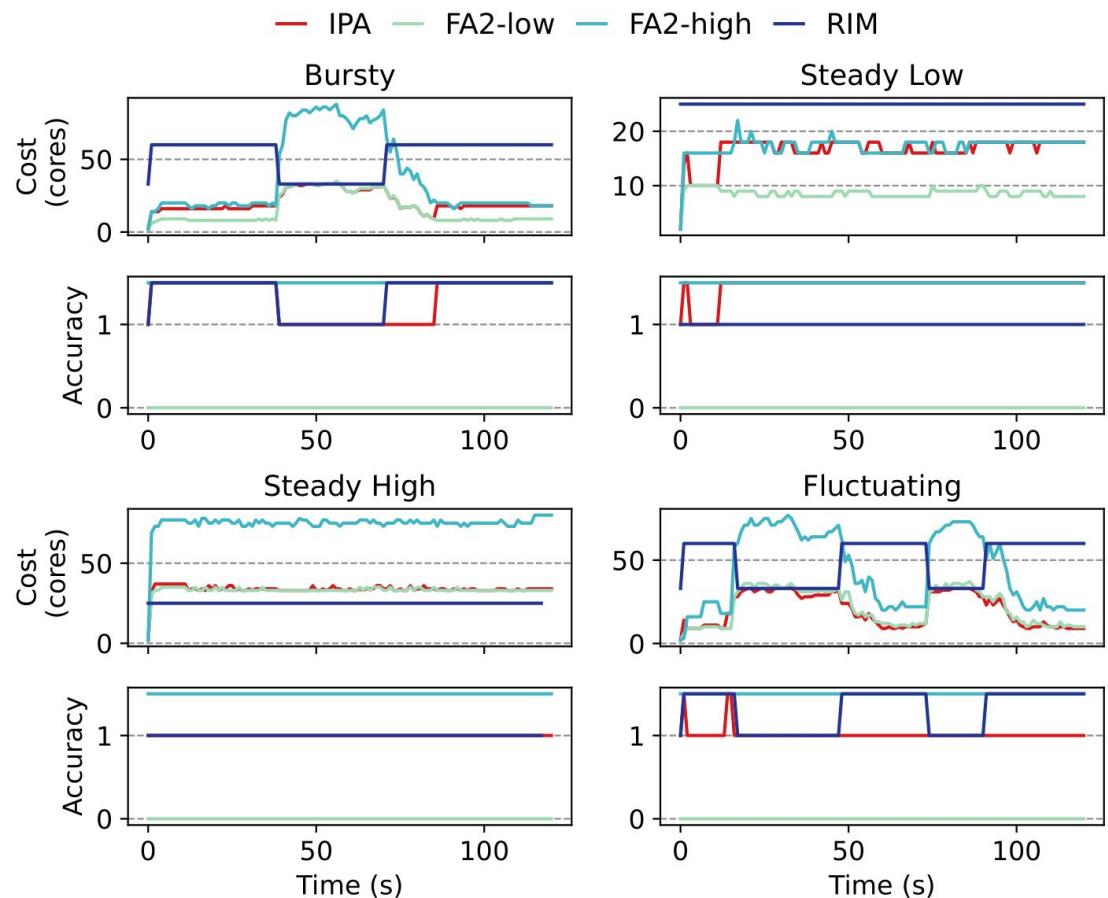
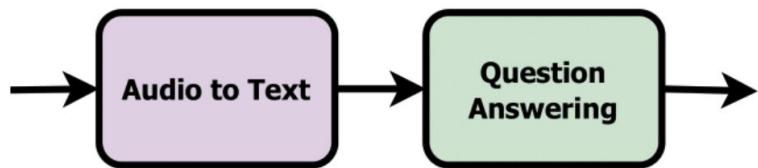


Experimental Results

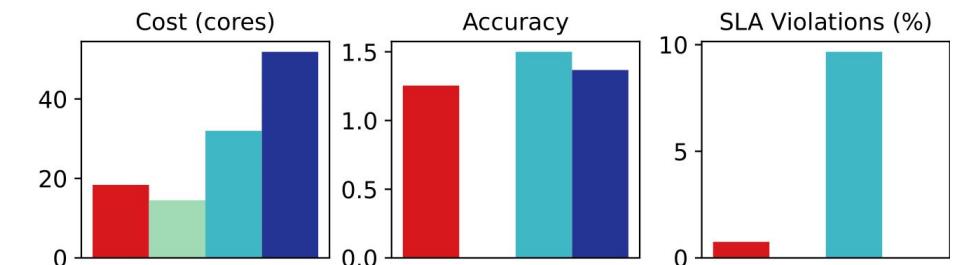
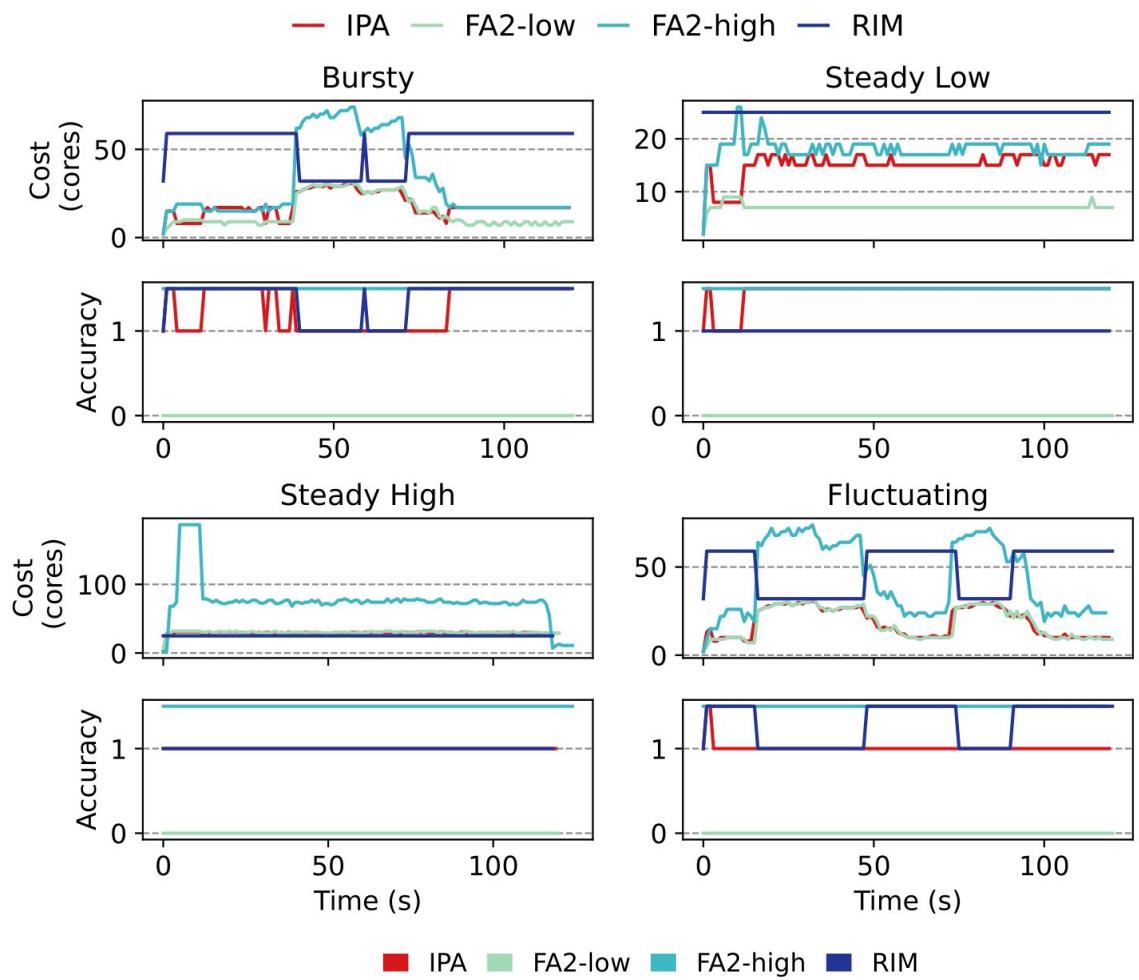
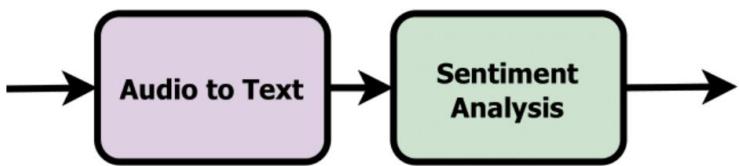
Video Pipeline



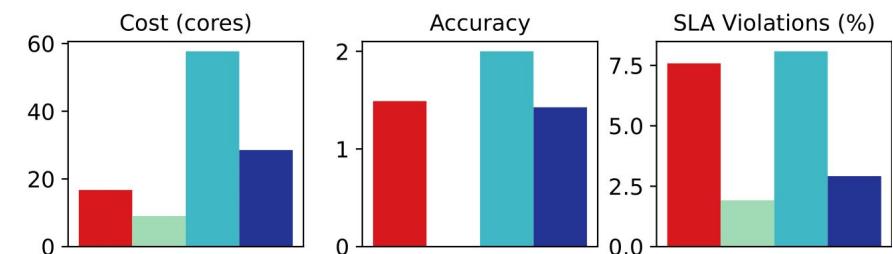
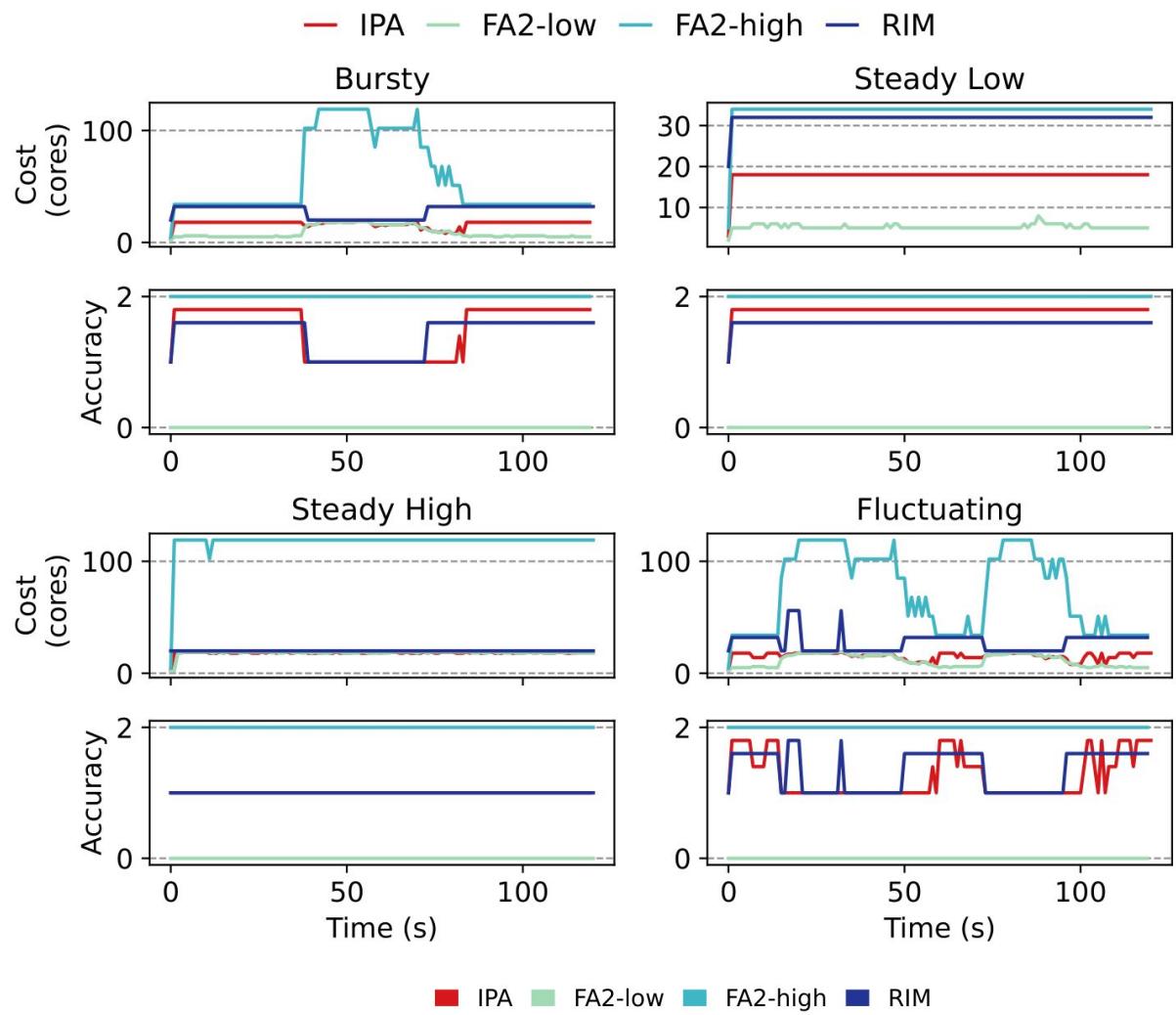
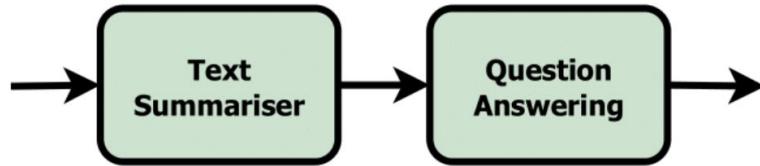
Audio + QA Pipeline



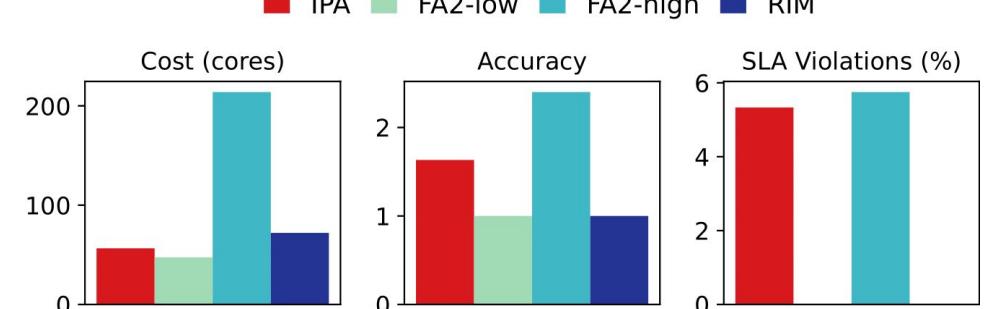
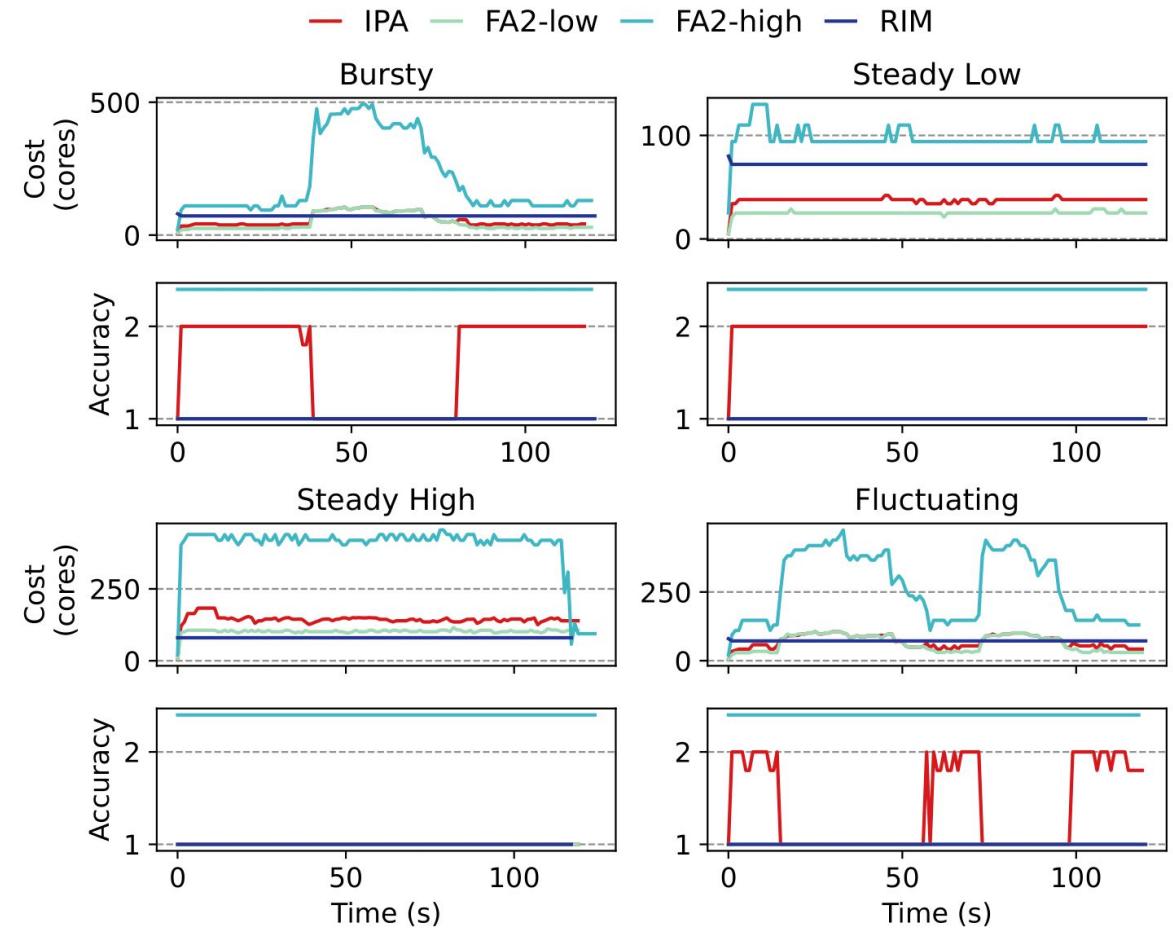
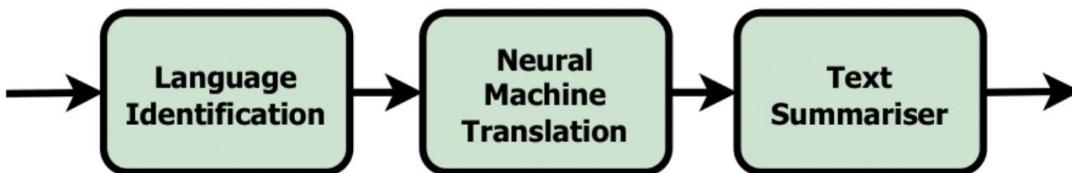
Summarization + QA Pipeline



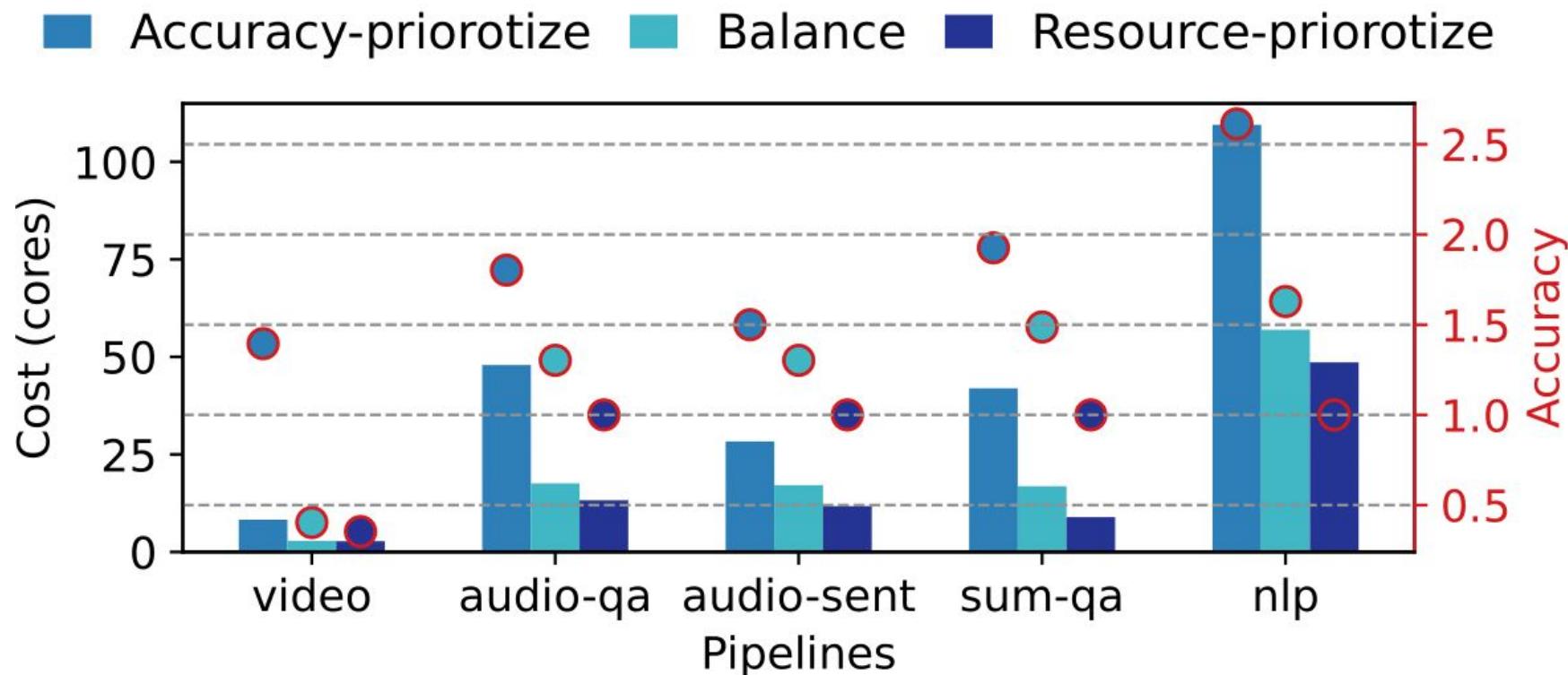
Summarization + QA Pipeline



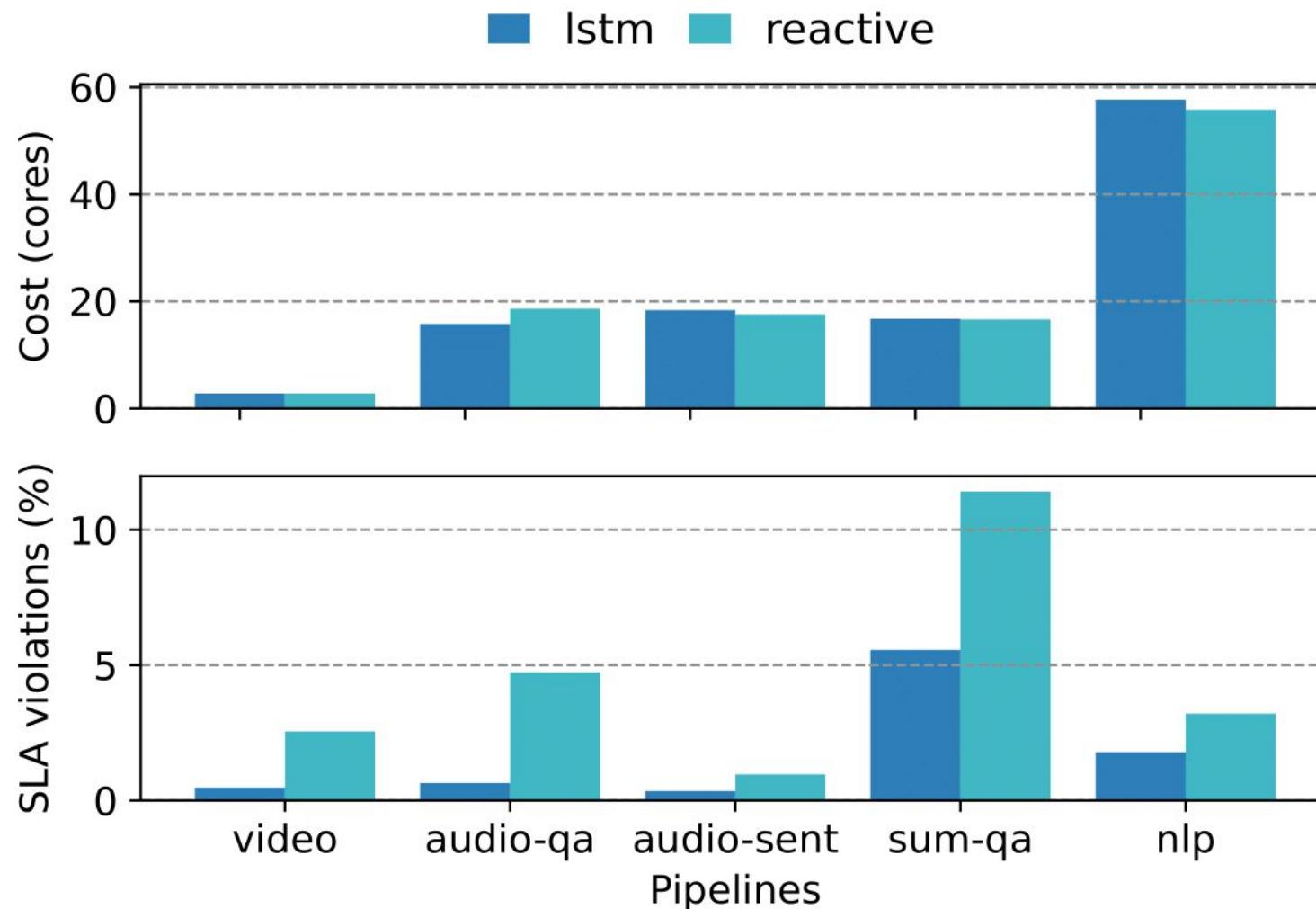
NLP Pipeline



Adaptivity to multiple objectives



Effect of predictor



Gurobi solver scalability

