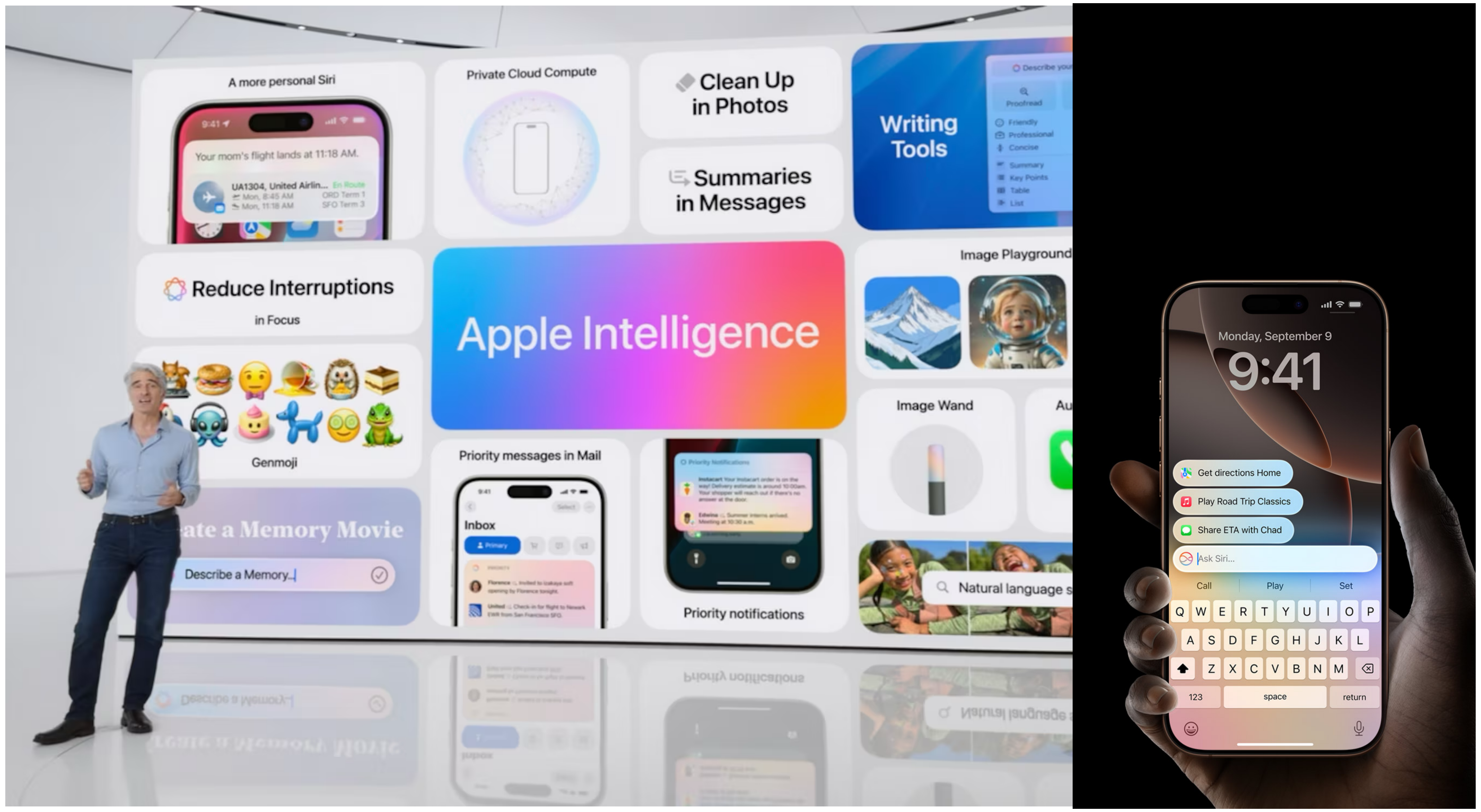
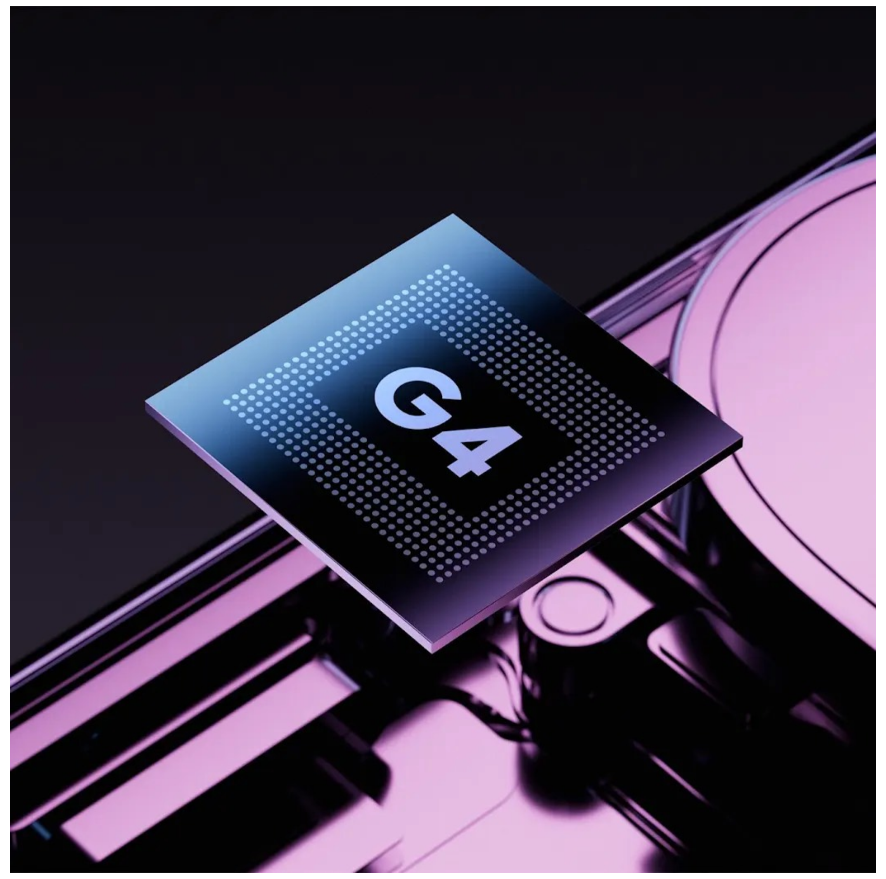
# On-Device AI is becoming the Thing!!




Apple launches iPhone 16 with Apple Intelligence




Google Pixel9 loaded with AI

Snapdragon X Elite

Microsoft Surface Pro powered by Snapdragon

Image sources: apple.com, google.com, qualcomm.com

# Self-adaptation Meets EdgeAI: Model Balancer in Action

**Karthik Vaidhyanathan**

TechFusion 2024: Towards Future Innovations

**September 11, 2024**


Middlesex University

SERC
Software Engineering Research Centre

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

UKIERI
UK-India Education and Research Initiative

SPARC
Scheme for Promotion of Academic and Research Collaboration

# ABOUT ME

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

IIIT Hyderabad

Logic takes you from A to B, Immagination takes you elsewhere -- Albert Einstein


**Karthik Vaidhyanathan**

Assistant Professor

Software Engineering Research Center and

Leadership Member, Smart City Research Center

IIIT Hyderabad, India


## Research Interests

**ML4SA**

1. ML for continuous self-adaptation
2. Gen AI for Architectural Knowlege

**SA4ML**

1. Sustainable ML-enabled systems
2. Autonomous ML-ops

## Education

Double Master Degree - Software Architecture and Machine Learning

PhD from GSSI, Italy

Postdoc, University of L'Aquila, Italy

## Fun Facts!

1. Cricket fanatic!
2. Movie buff!!
3. From God's own Country!!

SERC
Software Engineering Research Centre


https://karthikvaidhyanathan.com

/in/karthikv1392/

karthi_ishere

karthik.vaidhyanathan@iiit.ac.in

# We have come a long way!

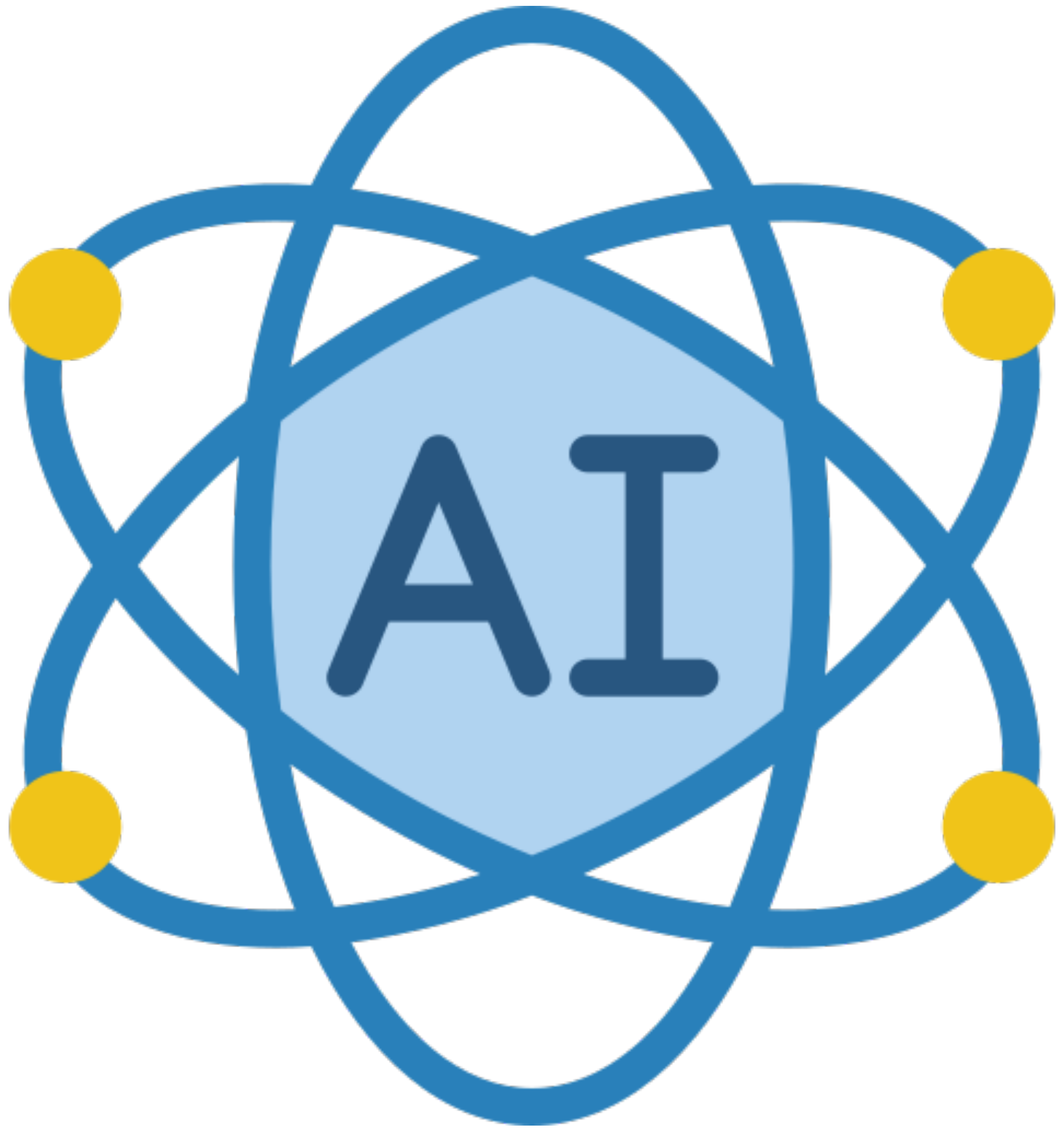
*In fact what I would like to see is thousands of computer scientists let loose to do whatever they want. That's what really advances the field* - Donald Knuth

**Time** →

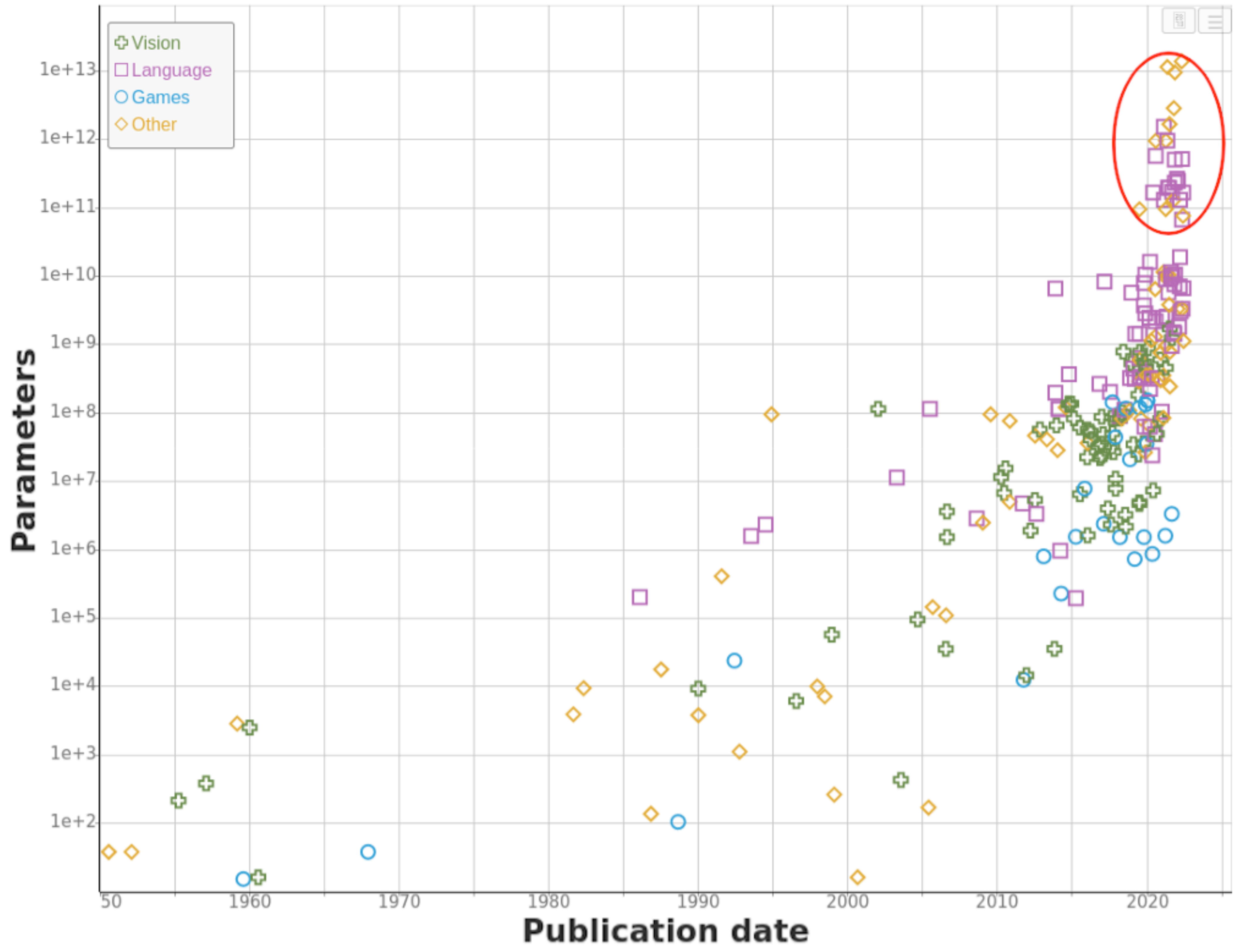
Now we are talking about spatial computing, quantum (will take time)

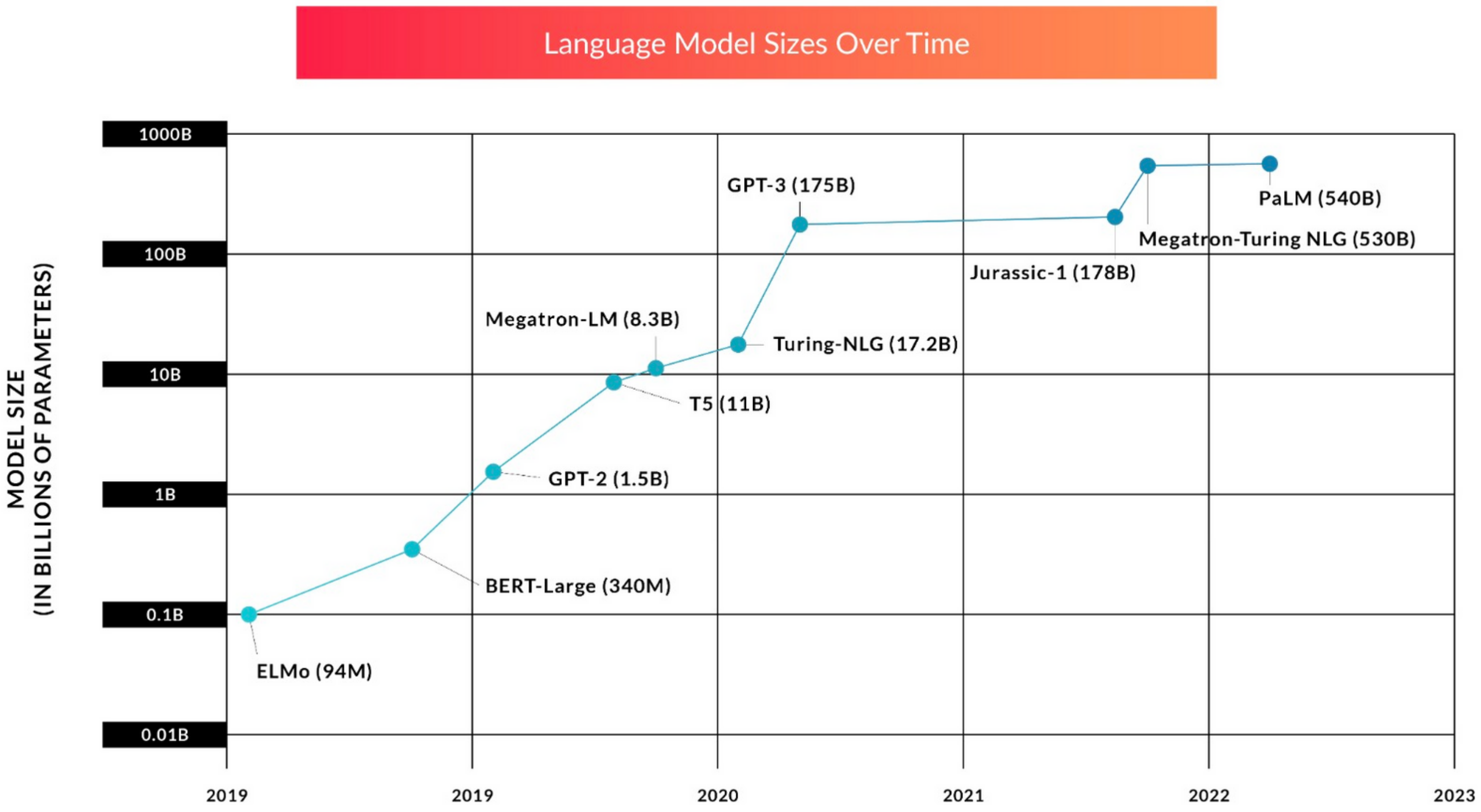
# AI: Over the Years

- **1943** - Warren McCulloh and Walter Pitts, MCP Model
- **1950** - Turing Test, Alan Turing
- **1952** - Computer learns checkers game, Arthur Samuel
- **1958** - Perceptron, Frank Rossenbalt
- **1959** - ADELINE, Bernard Widrow and Marcian Hoff
- **1997** - IBM DeepBlue
- **2000** - Now - Google Brain, DeepFace, AlphaGo, GPT…



# Models are getting bigger and better..But..




INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY



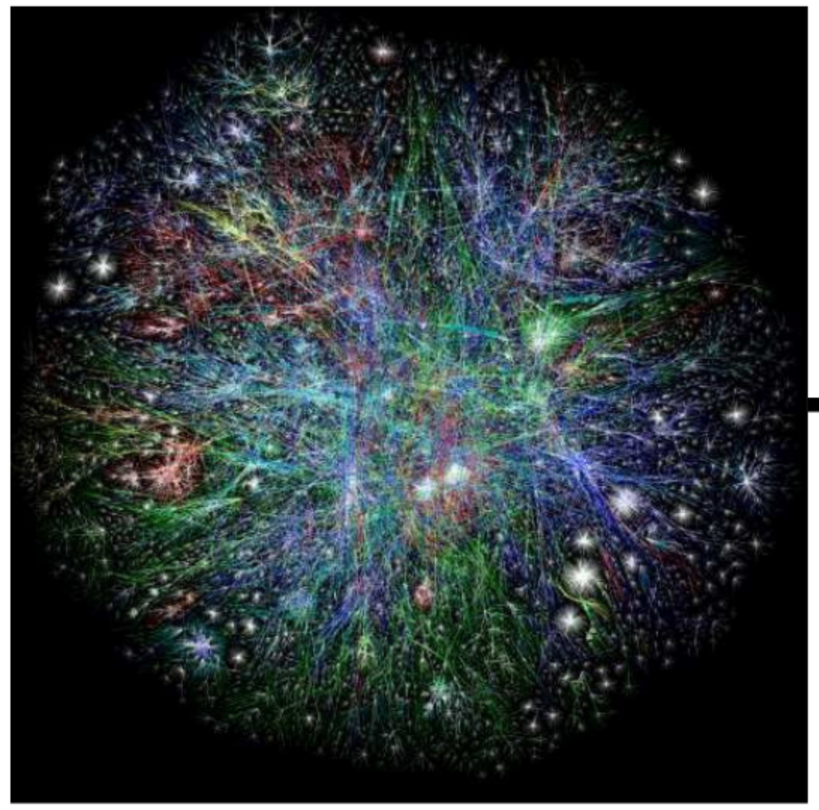
**Source:** Villalobos et al, Machine Learning Model Sizes and the Parameter Gap, https://arxiv.org/pdf/2207.02852

**Source:** assemblyai.com

# “Large” Language Models (LLM)

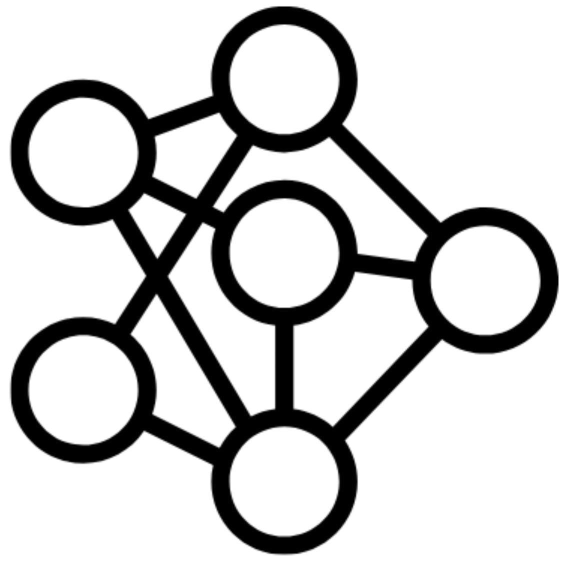
## Do you have a ton of text and compute power?

**Internet**



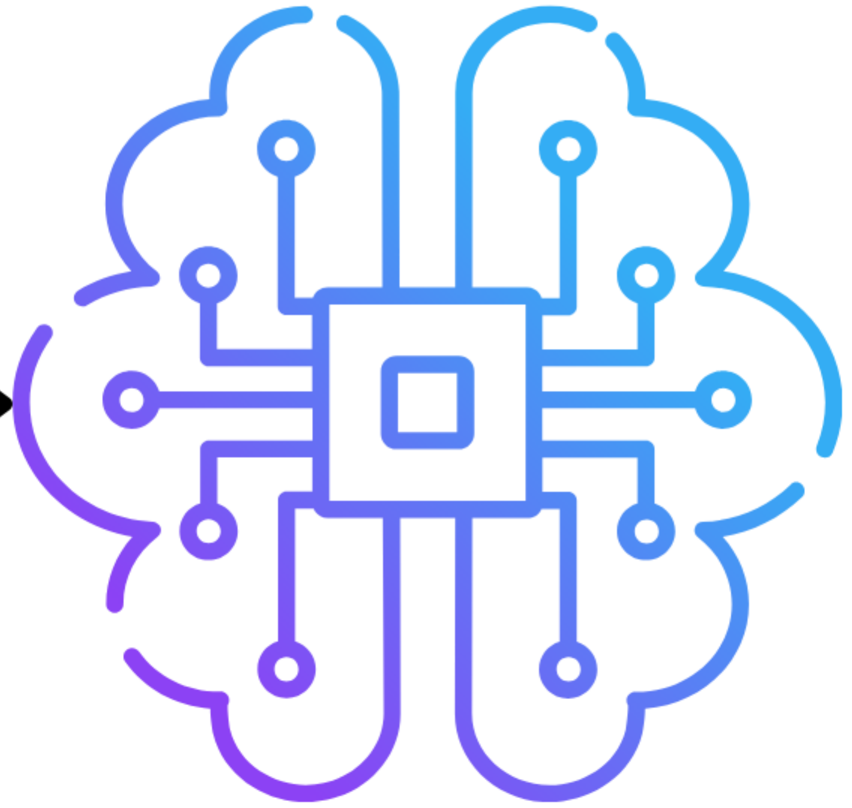
**~10B of text from internet**

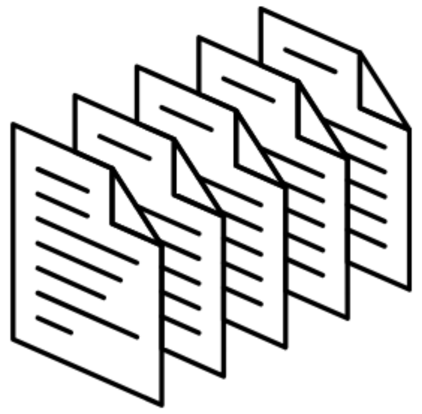
**Compute**



**Transformer NN**
**trained on 1000s of GPUs**
**for days**

**Foundation model/**
**Base model**





**Model parameters**
**~some billions**
**100s of GB**



Inspiration from slides of Andrej Karpathy, Introduction to large language models

# Privacy, Security, Latency…

## The BLERP issue

- **Bandwidth:** Need for high bandwidth connections, intensive in terms of energy!
- **Latency**: Round trip time for communicating with cloud, NASA Rover
- **Economy:** Cost of server side infrastructure, high speed network,..Think about chatGPT cost
- **Reliability:** Large models often require huge amount of compute and GPU (even for inference!)
- **Privacy**: Data going to third-party cloud or servers

SERC Software Engineering Research Centre



INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD

# Taking AI to the Edge

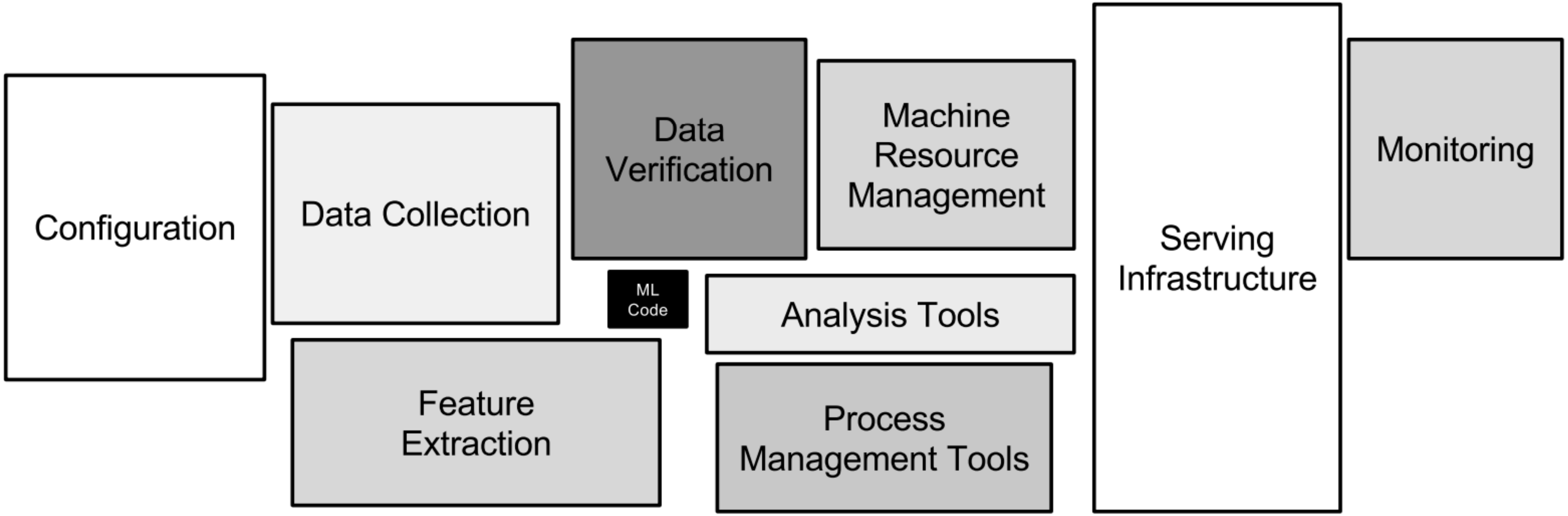
- **EdgeAI:** Practice of doing AI computations near the users at the network's edge instead of central cloud
  - Process data closer to where it is gathered
  - Challenges related to privacy, latency and bandwidth can be better handled
- Hardware has improved, communication standards have grown, AI models are becoming smaller (SLMs)
- EdgeAI market valued at USD 14,787.5 million in 2022


Image source: Raghubir Singh et al, EdgeAI: A Survey, https://doi.org/10.1016/j.iotcps.2023.02.004

# Challenges in EdgeAI

- **Resource Constraints:** Limited compute capabilities, network, battery,..
- **Security:** Edge can become an easy target
- **Scalability and Maintenance:** Models needs to be updated, needs to work in scale
- **Model Compression and Accuracy:** Large models should be able to support compression without a big trade-off on accuracy
- **Communication:** Reliability in communication is always a question

# AI Systems in General



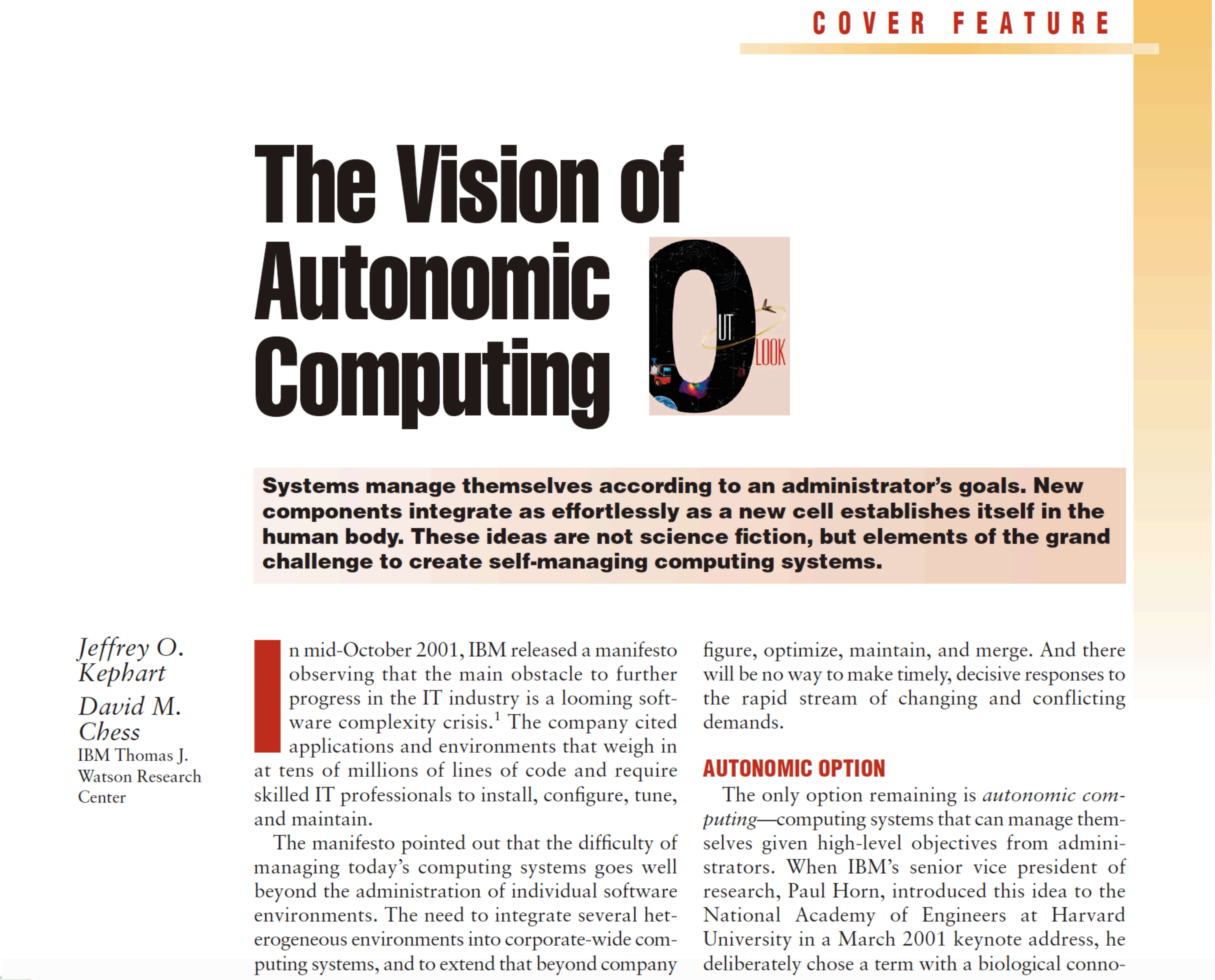

Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.F. and Dennison, D., 2015. **Hidden technical debt in machine learning systems**. *Advances in neural information processing systems*, *28.*

# AI System and Uncertainties

## More than 50% of ML systems remain as prototypes - Gartner

# Self-adaptation: A Potential Solution

## What if Software Systems could adapt like human cells?



COVER FEATURE

# The Vision of Autonomic Computing

OUT LOOK

**Systems manage themselves according to an administrator's goals. New components integrate as effortlessly as a new cell establishes itself in the human body. These ideas are not science fiction, but elements of the grand challenge to create self-managing computing systems.**

*Jeffrey O. Kephart*
*David M. Chess*
IBM Thomas J. Watson Research Center

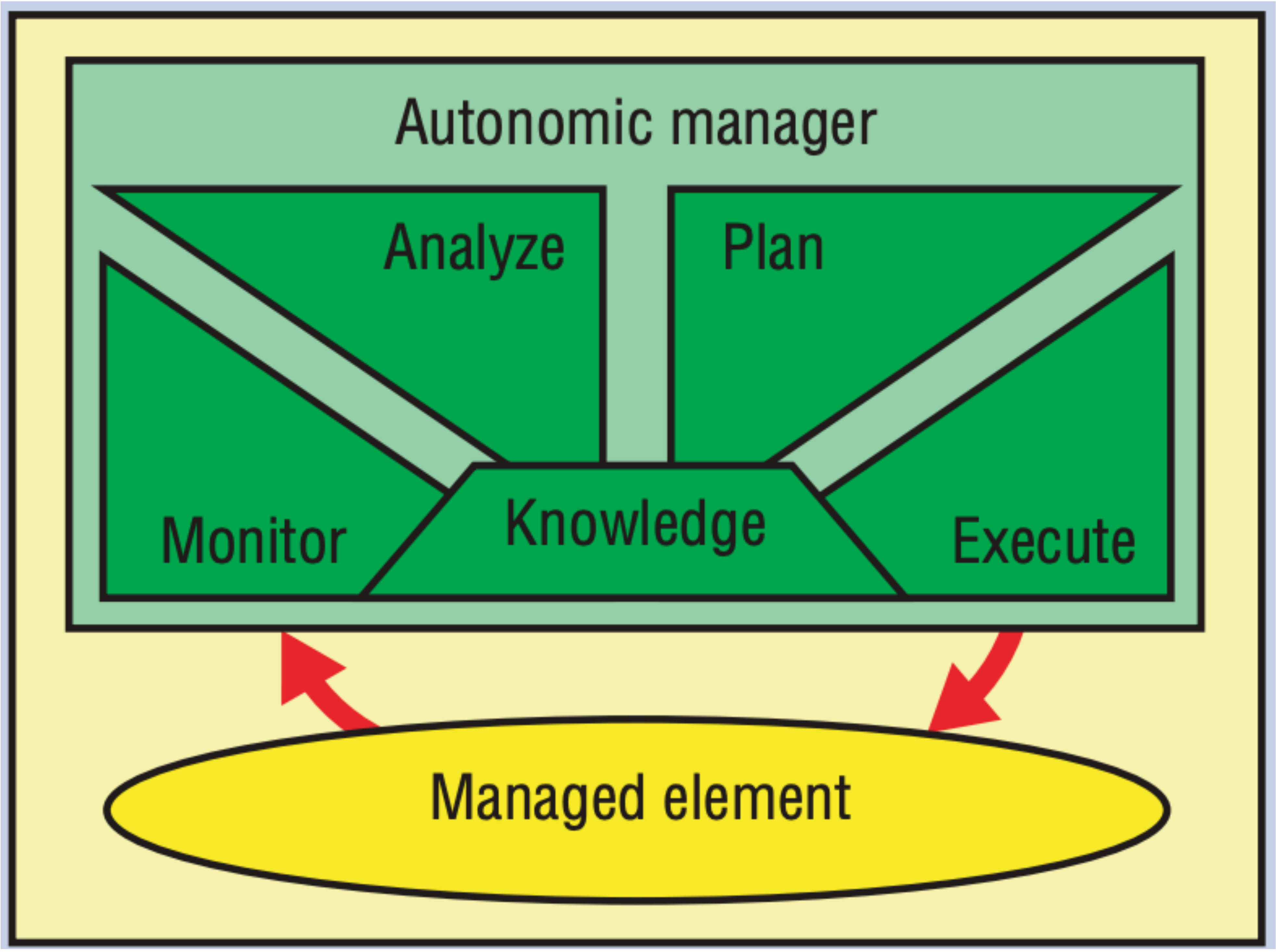
In mid-October 2001, IBM released a manifesto observing that the main obstacle to further progress in the IT industry is a looming software complexity crisis.[1] The company cited applications and environments that weigh in at tens of millions of lines of code and require skilled IT professionals to install, configure, tune, and maintain.

The manifesto pointed out that the difficulty of managing today's computing systems goes well beyond the administration of individual software environments. The need to integrate several heterogeneous environments into corporate-wide computing systems, and to extend that beyond company

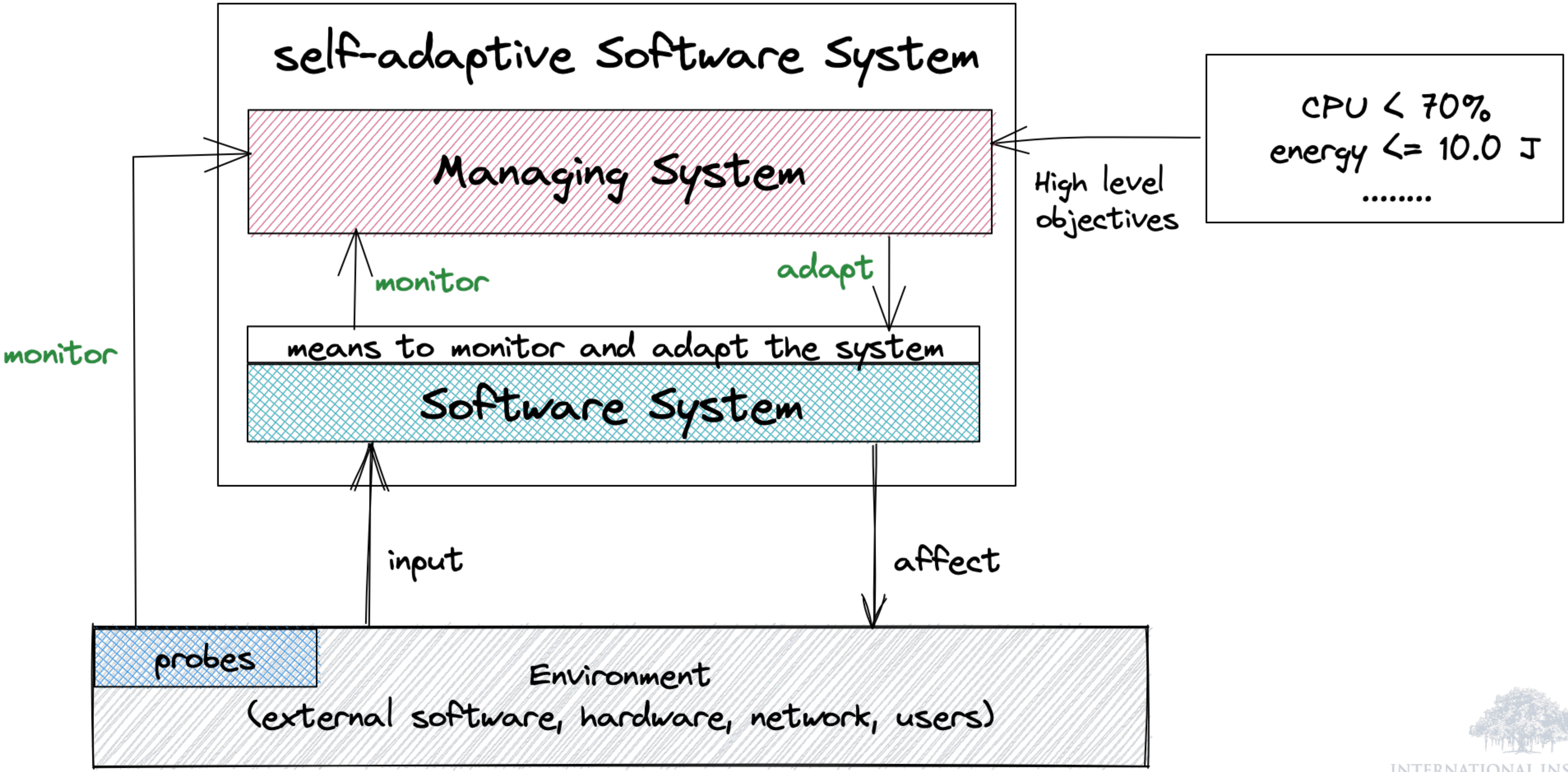
figure, optimize, maintain, and merge. And there will be no way to make timely, decisive responses to the rapid stream of changing and conflicting demands.

### AUTONOMIC OPTION

The only option remaining is *autonomic computing*—computing systems that can manage themselves given high-level objectives from administrators. When IBM's senior vice president of research, Paul Horn, introduced this idea to the National Academy of Engineers at Harvard University in a March 2001 keynote address, he deliberately chose a term with a biological conno-



# Self-adaptive System: Conceptual Model



# ML for Self-adaptation Exists..But

## Can we build self-adaptive ML-enabled System?

### Applying Machine Learning in Self-Adaptive Systems: A Systematic Literature Review

OMID GHEIBI, Katholieke Universiteit Leuven
DANNY WEYNS, Katholieke Universiteit Leuven, Linnaeus University
FEDERICO QUIN, Katholieke Universiteit Leuven

Recently, we witness a rapid increase in the use of machine learning techniques in self-adaptive systems. Machine learning has been used for a variety of reasons, ranging from learning a model of the environment of a system during operation to filtering large sets of possible configurations before analysing them. While a body of work on the use of machine learning in self-adaptive systems exists, there is currently no systematic overview of this area. Such overview is important for researchers to understand the state of the art and direct future research efforts. This paper reports the results of a systematic literature review that aims at providing such an overview. We focus on self-adaptive systems that are based on a traditional MAPE-based feedback loop (Monitor-Analyze-Plan-Execute). The research questions are centered on the problems that motivate the use of machine learning in self-adaptive systems, the key engineering aspects of learning in self-adaptation, and open challenges in this area. The search resulted in 6709 papers, of which 109 were retained for data collection. Analysis of the collected data shows that machine learning is mostly used for updating adaptation rules and policies to improve system qualities, and managing resources to better balance qualities and resources. These problems are primarily solved using supervised and interactive learning with classification, regression and reinforcement learning as the dominant methods. Surprisingly, unsupervised learning that naturally fits automation is only applied in a small number of studies. Key open challenges in this area include the performance of learning, managing the effects of learning, and dealing with more complex types of goals. From the insights derived from this systematic literature review we outline an initial design process for applying machine learning in self-adaptive systems that are based on MAPE feedback loops.

CCS Concepts: • **Software and its engineering**; • **Computing methodologies** → **Machine learning**; **Machine learning**; • **General and reference** → **Surveys and overviews**; **Surveys and overviews**;

ML has been applied to enable self-adaptation in non-ML systems

What if the managed system is an ML-enabled system?

What kind of adaptations can be performed?

SERC Software Engineering Research Centre

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

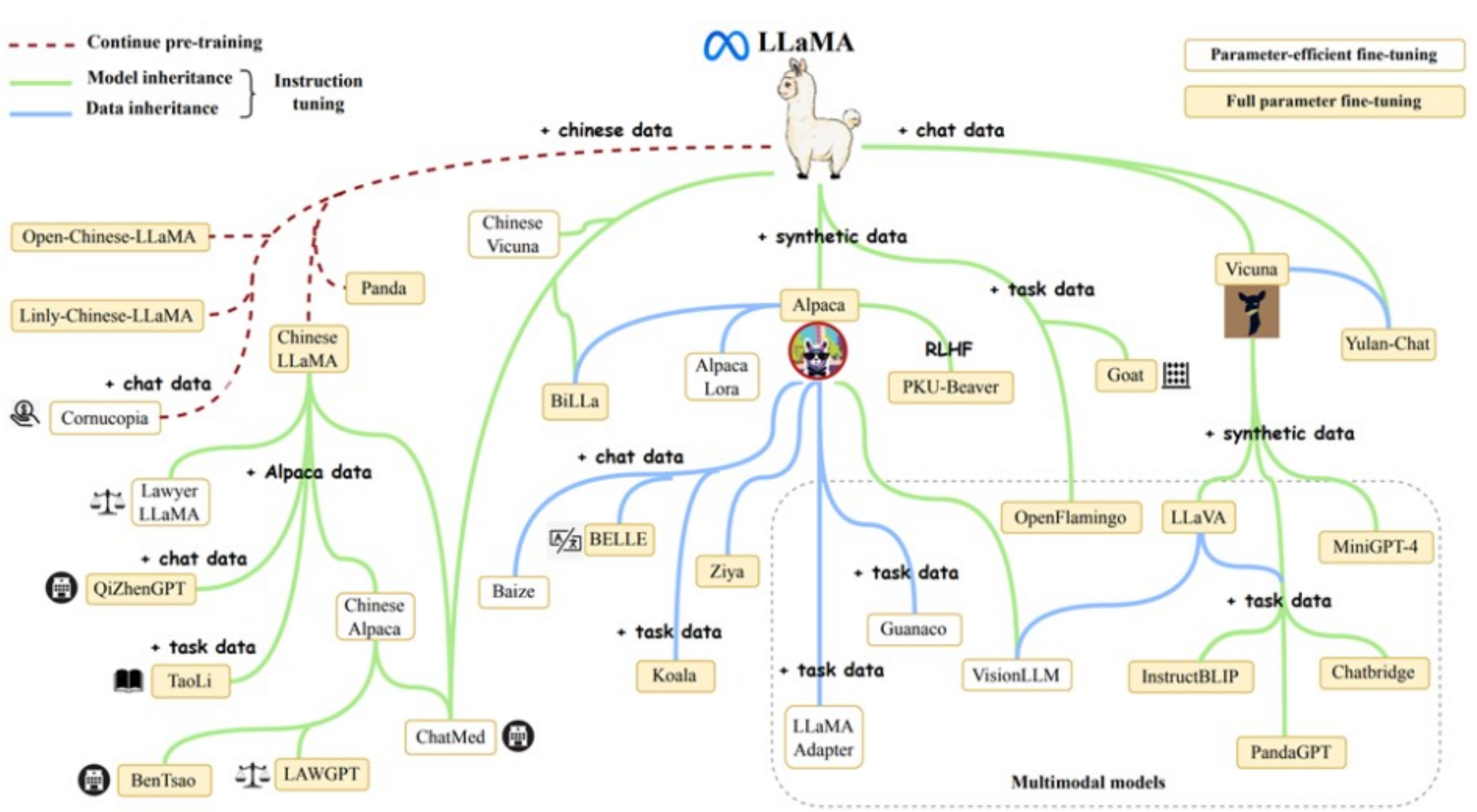
# Same Task can have multiple Models

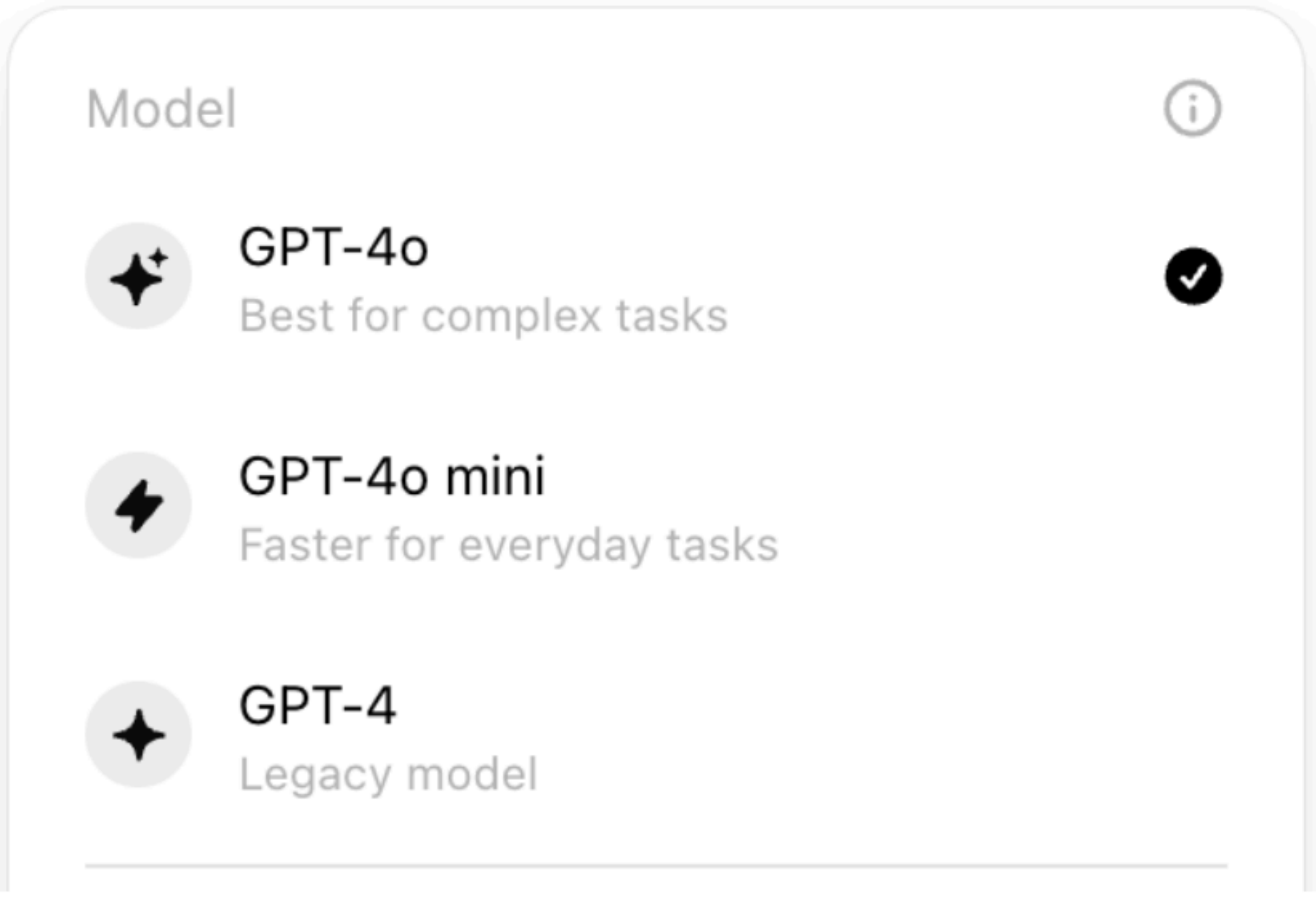
## Glimpse from LLM domain

### Proprietary

Bard

### Open Source




**Which model should I use?**

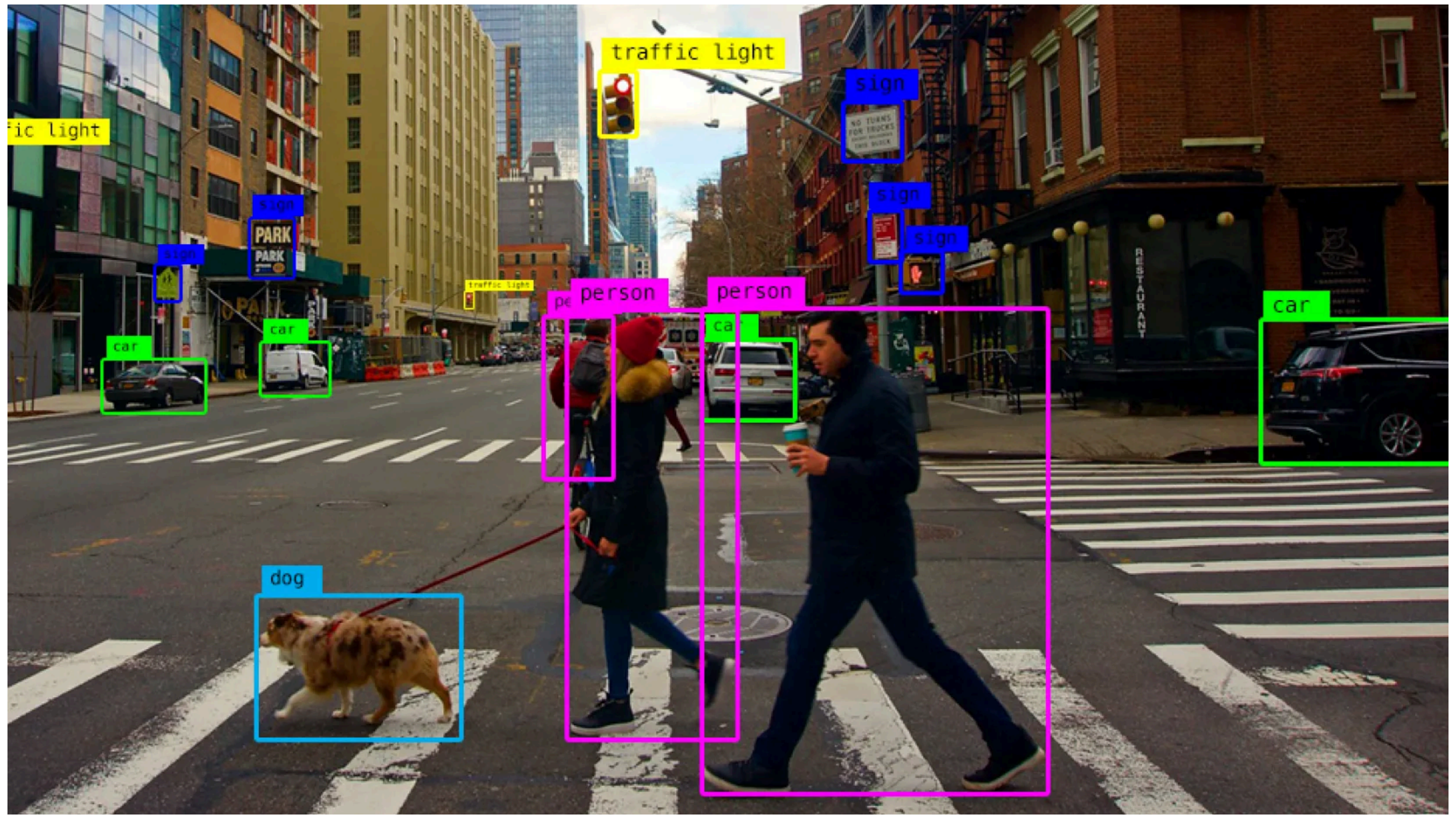
It will depend on your use case. Here are the most common reasons to use each model:

- **GPT-4o**
  - Our latest, fastest, highest intelligence model.
  - 128k context length (i.e. an average to longer novel).
  - Text and image input / text and image output.*
  - Audio input / output.**
- **GPT-4o mini**
  - Our lightest-weight intelligence model.
  - 128k context length (i.e. an average to longer novel).
  - Text and image input / text and image output.*
  - Audio input / output.**
  - `Limitation`: This model does not have access to the advanced tools that GPT-4o has.
- **GPT-4**
  - Our previous high intelligence model.
  - 128k context length (i.e. an average to longer novel).
  - Text and image input / text and image output.*
  - Audio input / output.**
- **GPT-3.5 (API only)**
  - Fast model for the simplest routine tasks.
  - 16k context length (i.e. 1-2 dozen articles or a short story / novella).
  - Text input / text output.
  - Audio input / output.**

Source: help.openai.com

# Same Task can have multiple Models

## Same holds true for Image Domain



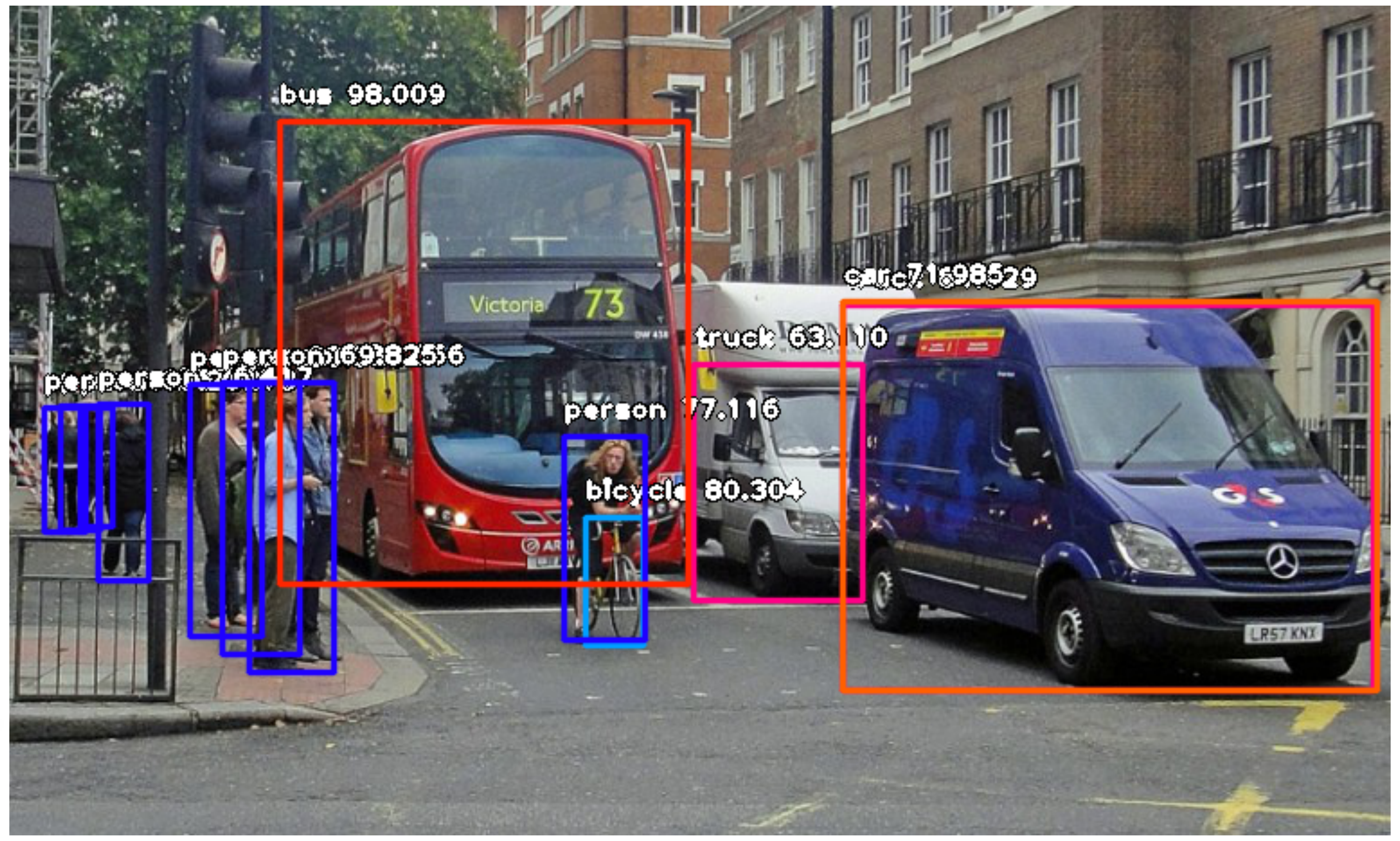



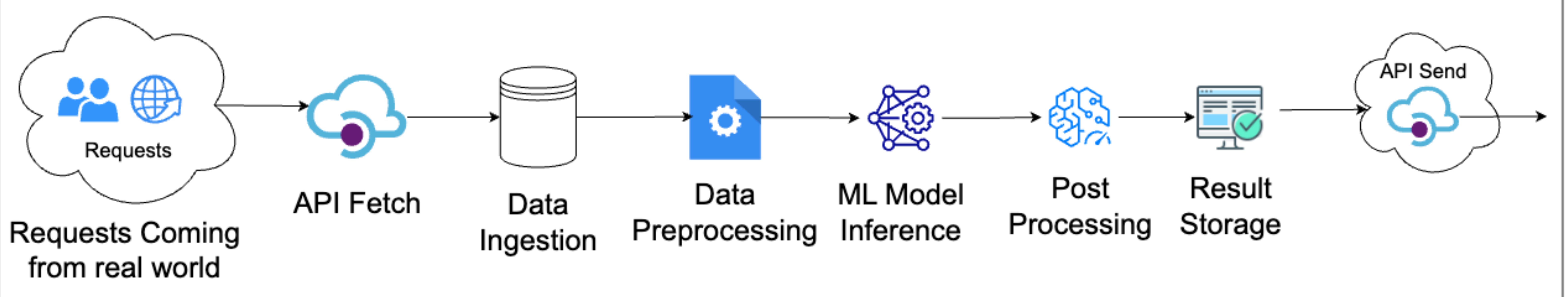

Image source: augmentedstartups.com

| Model | size (pixels) | $mAP^{val}$ 50-95 | $mAP^{val}$ 50 | Speed CPU b1 (ms) | Speed V100 b1 (ms) | Speed V100 b32 (ms) | params (M) | FLOPs @640 (B) |
|---|---|---|---|---|---|---|---|---|
| YOLOv5n | 640 | 28.0 | 45.7 | **45** | **6.3** | **0.6** | **1.9** | **4.5** |
| YOLOv5s | 640 | 37.4 | 56.8 | 98 | 6.4 | 0.9 | 7.2 | 16.5 |
| YOLOv5m | 640 | 45.4 | 64.1 | 224 | 8.2 | 1.7 | 21.2 | 49.0 |
| YOLOv5l | 640 | 49.0 | 67.3 | 430 | 10.1 | 2.7 | 46.5 | 109.1 |
| YOLOv5x | 640 | 50.7 | 68.9 | 766 | 12.1 | 4.8 | 86.7 | 205.7 |

Detectron2

MobileNet v2

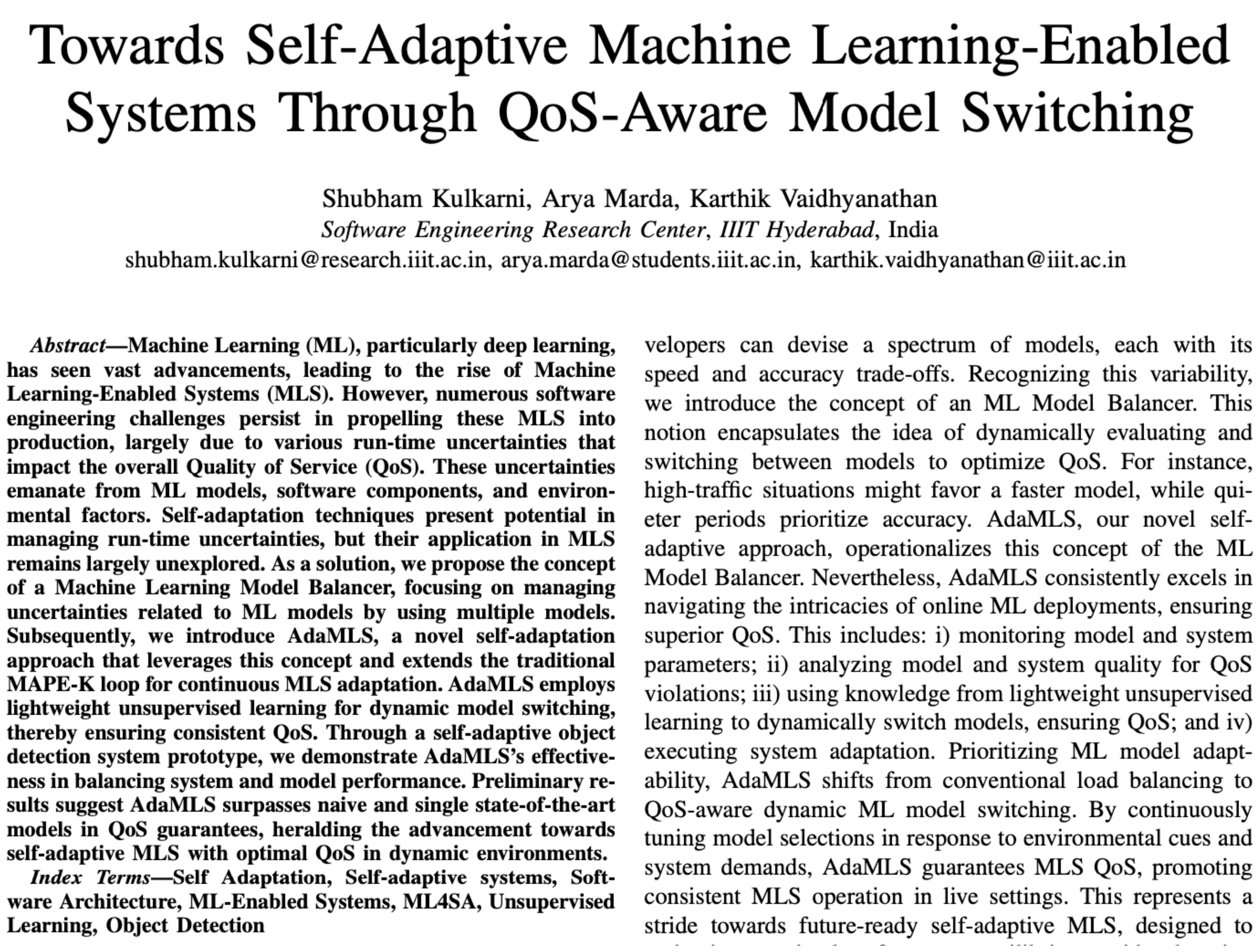
# Can we self-adapt during Inference?




Overview of Workflow in Machine Learning Enabled System

- For a given task, one can use different ML models
- Each model offers different latency, different confidence, energy efficiency, etc.
- **What if we could switch among the ML models - ML Model Balancer!**

# AdaMLS: Balancing Between ML Models



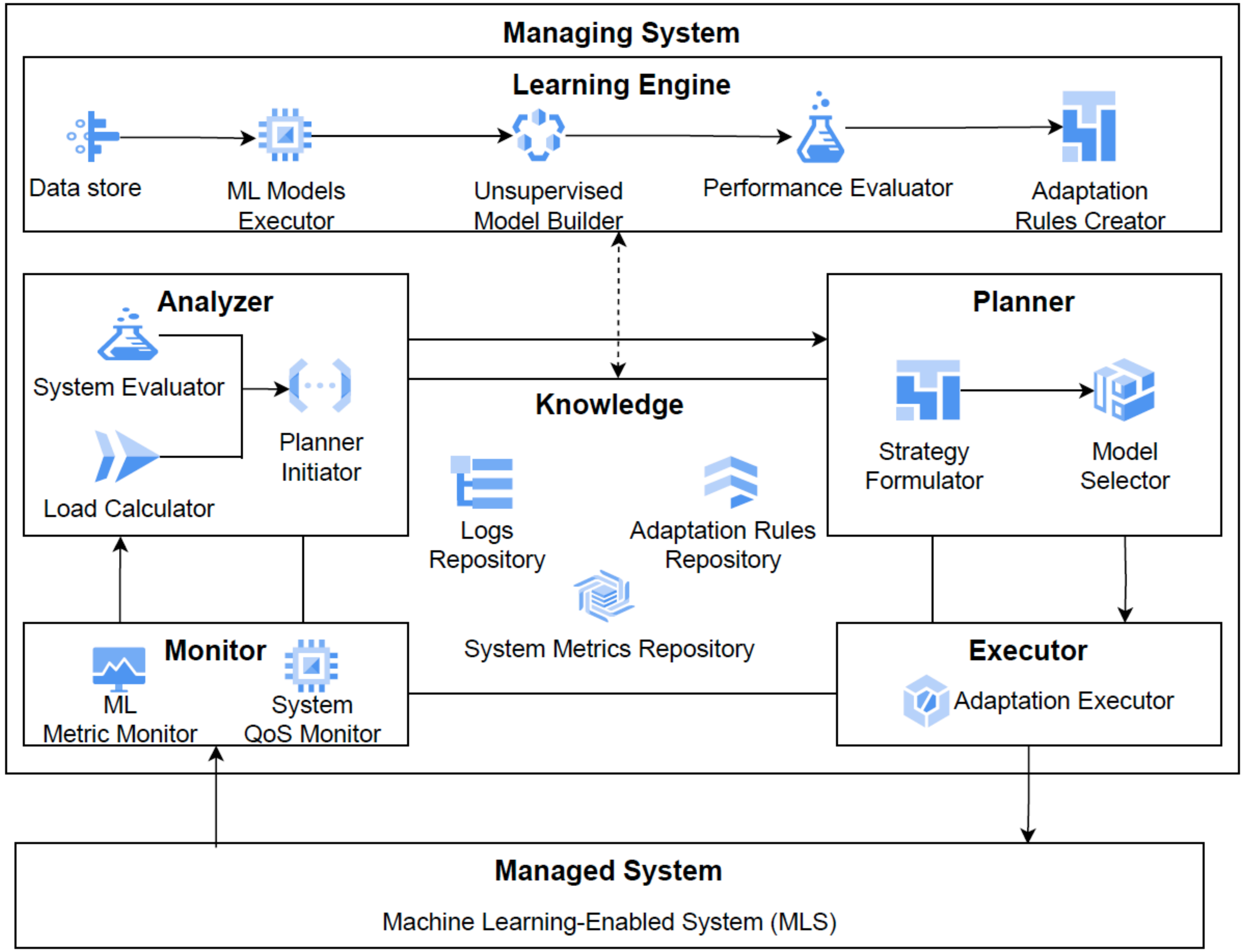
Towards Self-Adaptive Machine Learning-Enabled Systems Through QoS-Aware Model Switching

Shubham Kulkarni, Arya Marda, Karthik Vaidhyanathan
*Software Engineering Research Center, IIIT Hyderabad*, India
shubham.kulkarni@research.iiit.ac.in, arya.marda@students.iiit.ac.in, karthik.vaidhyanathan@iiit.ac.in

***Abstract*—Machine Learning (ML), particularly deep learning, has seen vast advancements, leading to the rise of Machine Learning-Enabled Systems (MLS). However, numerous software engineering challenges persist in propelling these MLS into production, largely due to various run-time uncertainties that impact the overall Quality of Service (QoS). These uncertainties emanate from ML models, software components, and environmental factors. Self-adaptation techniques present potential in managing run-time uncertainties, but their application in MLS remains largely unexplored. As a solution, we propose the concept of a Machine Learning Model Balancer, focusing on managing uncertainties related to ML models by using multiple models. Subsequently, we introduce AdaMLS, a novel self-adaptation approach that leverages this concept and extends the traditional MAPE-K loop for continuous MLS adaptation. AdaMLS employs lightweight unsupervised learning for dynamic model switching, thereby ensuring consistent QoS. Through a self-adaptive object detection system prototype, we demonstrate AdaMLS's effectiveness in balancing system and model performance. Preliminary results suggest AdaMLS surpasses naive and single state-of-the-art models in QoS guarantees, heralding the advancement towards self-adaptive MLS with optimal QoS in dynamic environments.**

***Index Terms*—Self Adaptation, Self-adaptive systems, Software Architecture, ML-Enabled Systems, ML4SA, Unsupervised Learning, Object Detection**

velopers can devise a spectrum of models, each with its speed and accuracy trade-offs. Recognizing this variability, we introduce the concept of an ML Model Balancer. This notion encapsulates the idea of dynamically evaluating and switching between models to optimize QoS. For instance, high-traffic situations might favor a faster model, while quieter periods prioritize accuracy. AdaMLS, our novel self-adaptive approach, operationalizes this concept of the ML Model Balancer. Nevertheless, AdaMLS consistently excels in navigating the intricacies of online ML deployments, ensuring superior QoS. This includes: i) monitoring model and system parameters; ii) analyzing model and system quality for QoS violations; iii) using knowledge from lightweight unsupervised learning to dynamically switch models, ensuring QoS; and iv) executing system adaptation. Prioritizing ML model adaptability, AdaMLS shifts from conventional load balancing to QoS-aware dynamic ML model switching. By continuously tuning model selections in response to environmental cues and system demands, AdaMLS guarantees MLS QoS, promoting consistent MLS operation in live settings. This represents a stride towards future-ready self-adaptive MLS, designed to

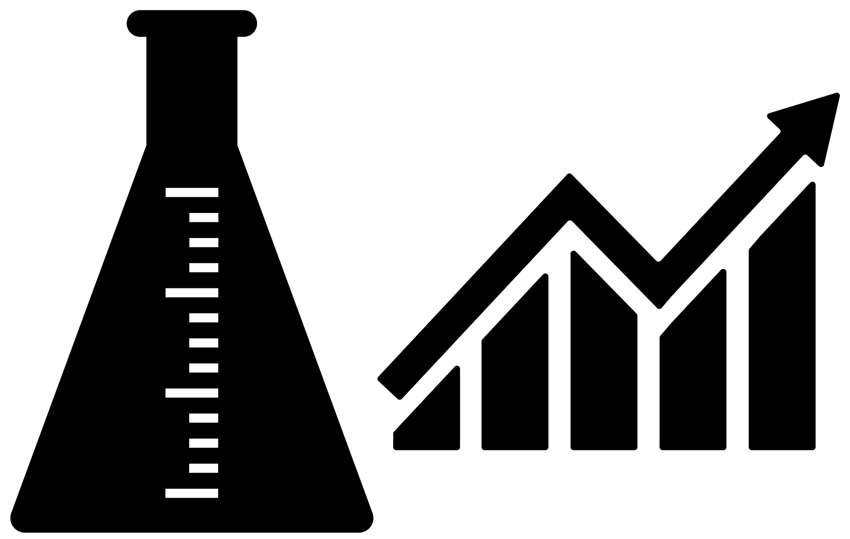



ASE, NIER 2023

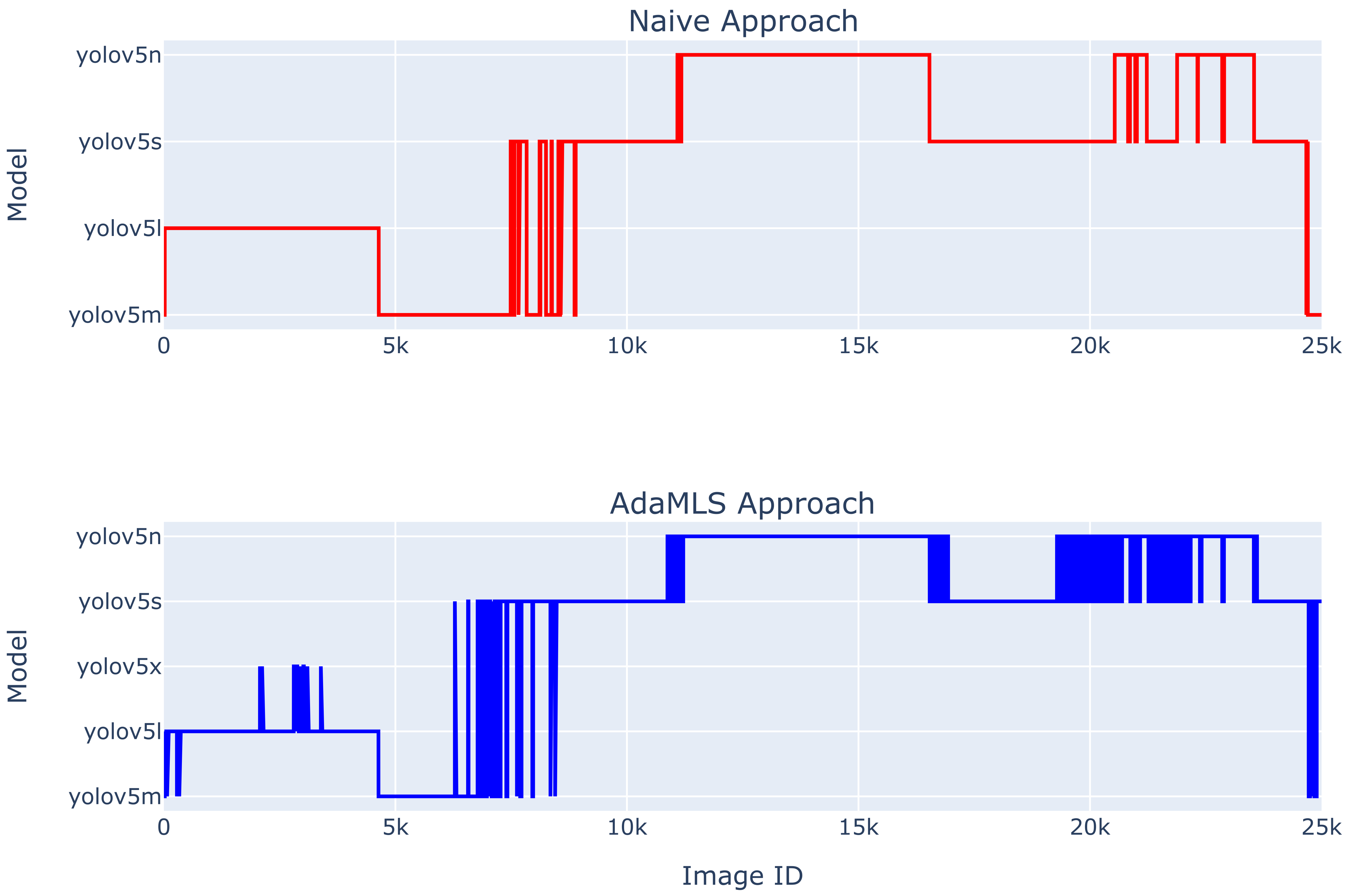
Best student poster@ISEC 2024!

# Application to Object Detection

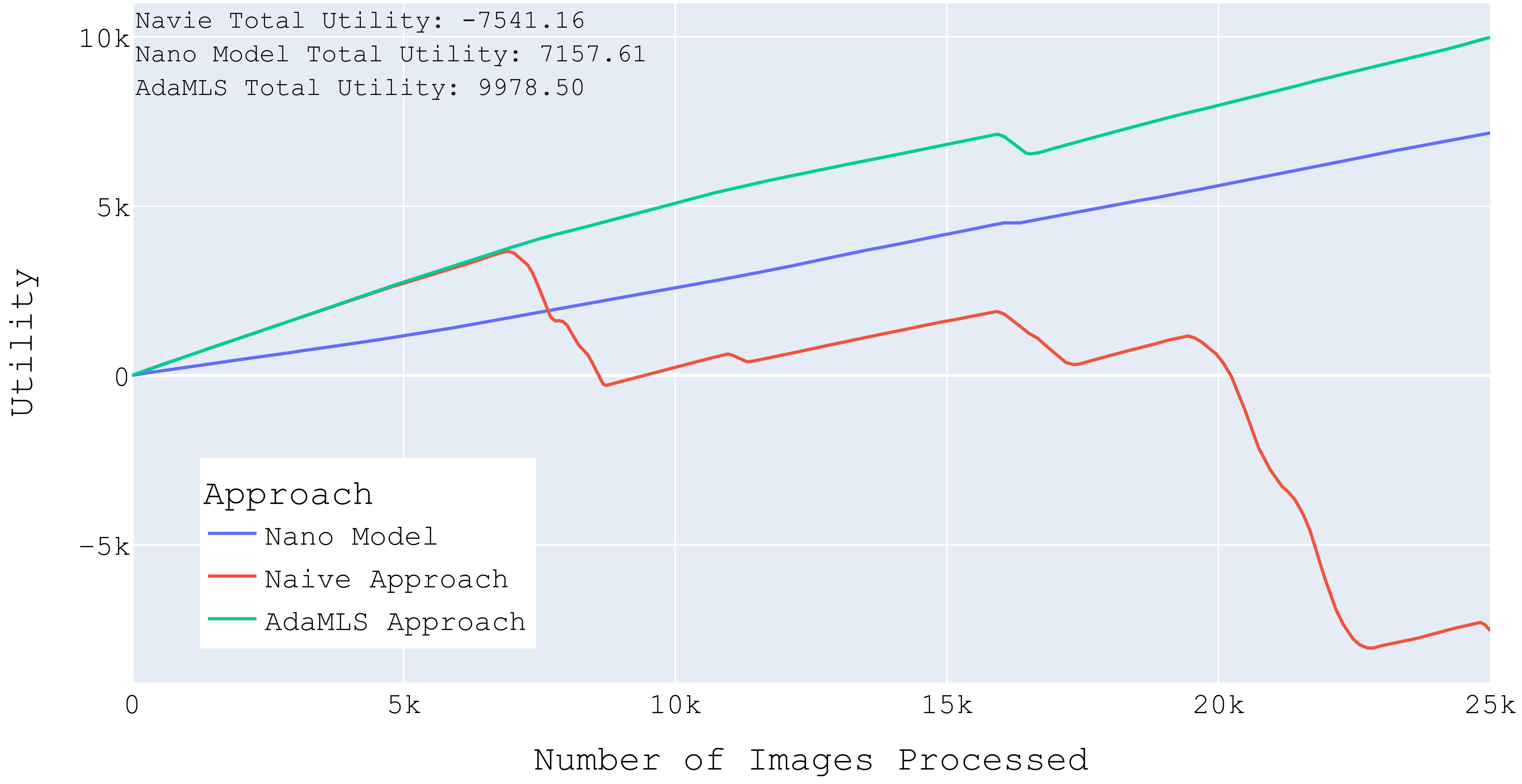
- Implemented AdaMLS on an Object detection system (APIs that serve the models)
- Models used: different variants of YoloV5 (Except large)
- Simulated workload to the system using FIFA98 benchmark trace
- COCO 2017 dataset was used for the evaluation
    - Utility score was defined to compare effectiveness
    - Naive approach uses thresholds to transition between models



# Some Initial Results




Switching between models using different approaches




Utility considering performance and confidence

**39% improved QoS with 0.01 seconds switching time**

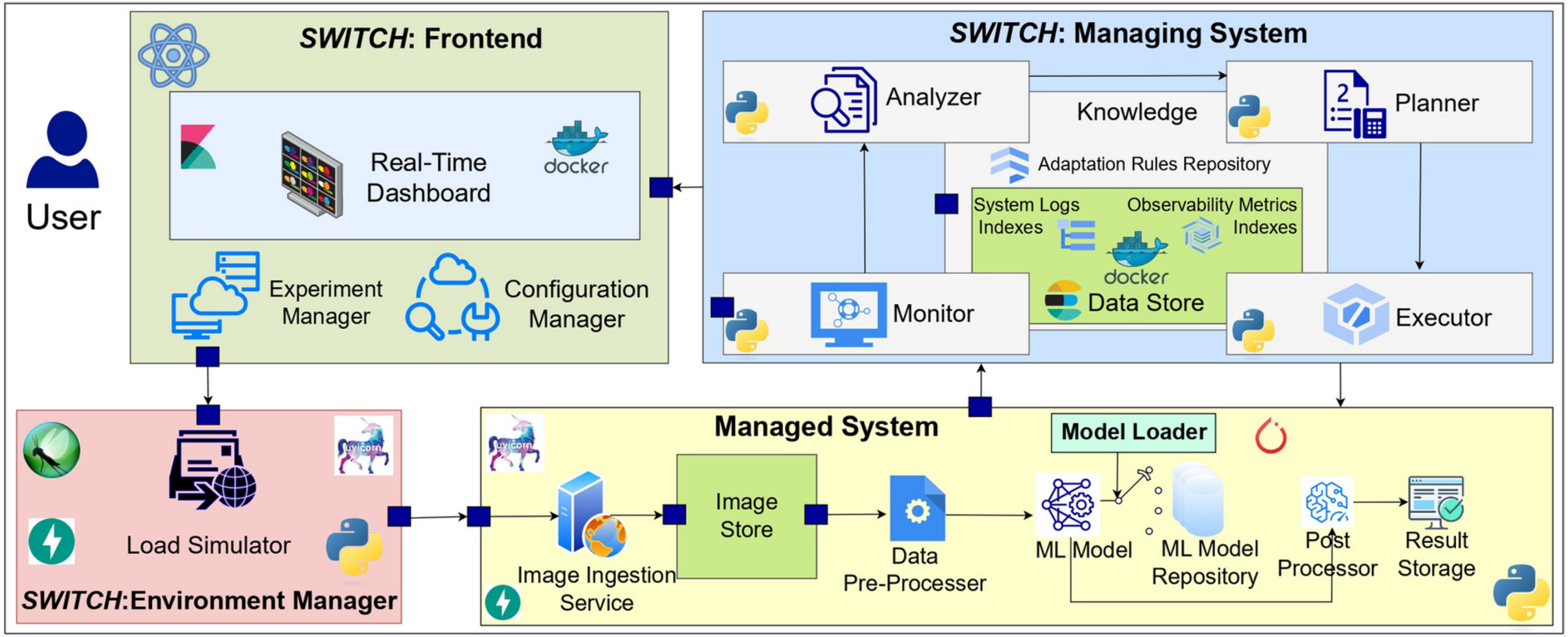


SERC Software Engineering Research Centre

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

# Introducing SWITCH Exemplar

## A tool for practitioners and academic to evaluate switching strategies

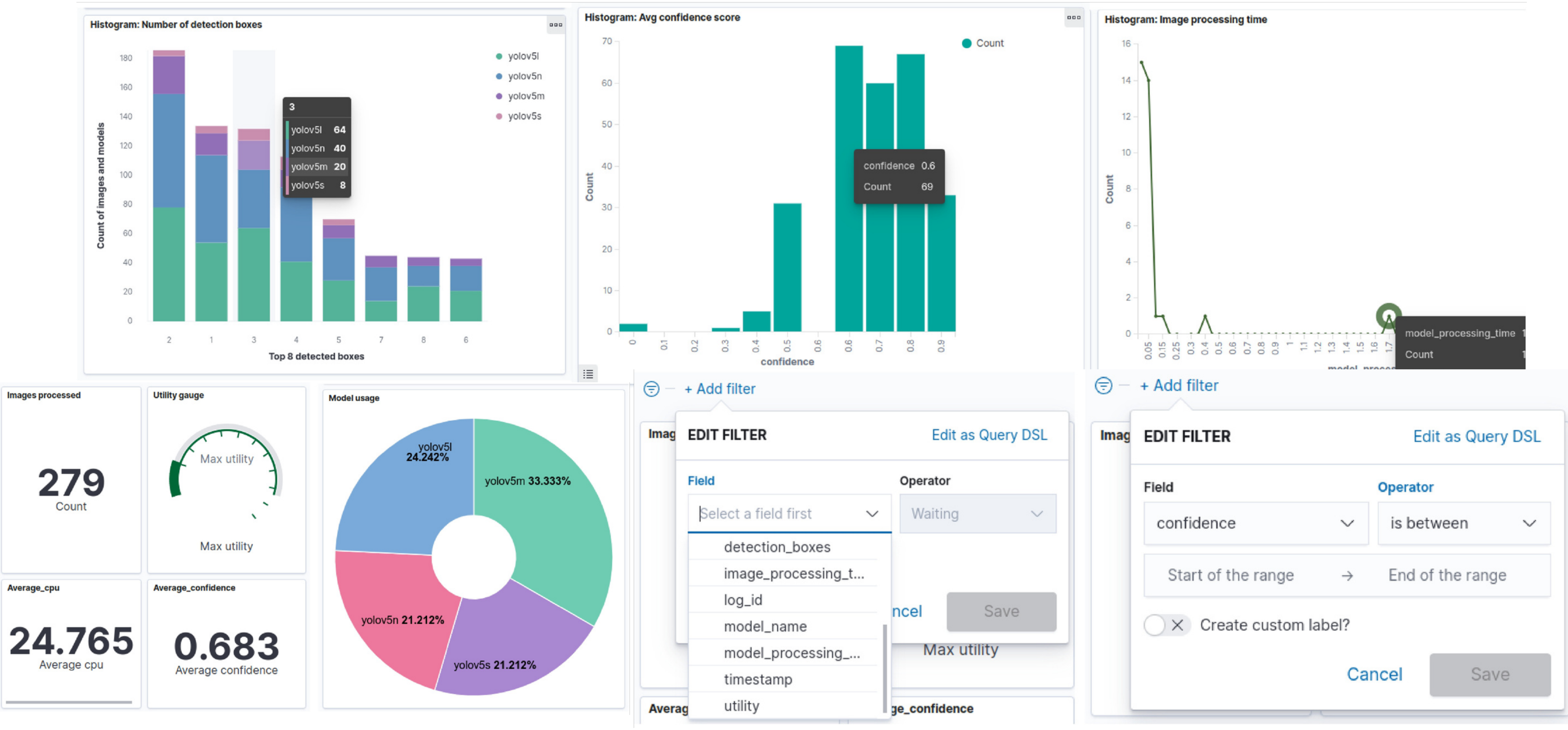



**Scan me**

SEAMS 2024@ICSE 2024

https://tool-switch.github.io/

# Glimpse of SWITCH




Activities
SimpleScreenRecorder
Jul 6 16:12
arya@arya-asus: ~/Documents/DOCKER
arya@arya-asus:~/Documents/DOCKER$ docker-compose up
[+] Building 0.0s (0/0)
WARN[0000] Found orphan containers ([logstash]) for this project. If you removed or renamed this serv
ice in your compose file, you can run this command with the --remove-orphans flag to clean it up.
[+] Running 2/0
✓ Container docker-elasticsearch-1 Created
✓ Container kibana
Attaching to docker-elasticsearch-1,
docker-elasticsearch-1 | {"type": "server", "timestamp": "2023-07-06T10:26:25,761Z", "level": "INFO"
, "component": "o.e.p.PluginsService", "cluster.name": "docker-cluster", "node.name": "1803bef41746",
"message": "loaded module [ingest-common]" }
"message": "loaded module [ingest-geoip]" }
"message": "loaded module [ingest-user-agent]" }
"message": "loaded module [kibana]" }
SimpleScreenRecorder
Recording
Pause recording
Schedule: (inactive)
Activate schedule
Edit schedule
Enable recording hotkey
Enable sound notifications
Hotkey: Ctrl + Shift + Alt + Super + R
Information
Total time: 0:00:00
FPS in: 0.00
FPS out: 0.00
Size in: 1920x1080
Size out: ?
File name: ?
File size: 0 B
Bit rate: 0 bit/s
Preview
Preview frame rate: 10
Note: Previewing requires extra CPU time (especially at high frame rates).
Start preview
Log
[X11Input::Init] Detecting screen configuration ...
[X11Input::Init] Screen 0: x1 = 0, y1 = 0, x2 = 1920, y2 = 1080
[X11Input::InputThread] Input thread started.
[PageRecord::StartInput] Started input.
Cancel recording
Save recording
process.py - obj-service - Visual Studio Code
File Edit Selection View Go Run Terminal Help
Node.py Request_send.py process.py 4 App.py
NAVIE > process.py > start_processing
current_cpu = psutil.cpu_percent(interval=None)
total_processed += 1
detection = response.pandas().xyxy[0]
confidences = detection['confidence'].tolist()
current_conf = sum(confidences)
current_boxes = len(confidences)
if (current_boxes != 0):
avg_conf = current_conf/current_boxes
else:
avg_conf = 0
t = time.time()
current_time = t - current_time #model processsing time
start_time = t - start_time # total time take by image t
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL
bash - NAVIE
arya@arya-asus:~/Desktop/SERC/ArchML-main/obj-service/NAVIE$
arya@arya-asus:~/Desktop/SERC/ArchML-main/obj-service$
Run Testcases
Ln 165, Col 1 Spaces: 4 UTF-8 LF Python 3.9.5 64-bit Go Live

# EcoMLS: Model Balancer for Enhanced Sustainability (Environmental!)

## EcoMLS: A Self-Adaptation Approach for Architecting Green ML-Enabled Systems


Meghana Tedla
*Software Engineering Research Center*
*IIIT Hyderabad, India*
meghana.tedla@students.iiit.ac.in

Shubham Kulkarni
*Software Engineering Research Center*
*IIIT Hyderabad, India*
shubham.kulkarni@research.iiit.ac.in

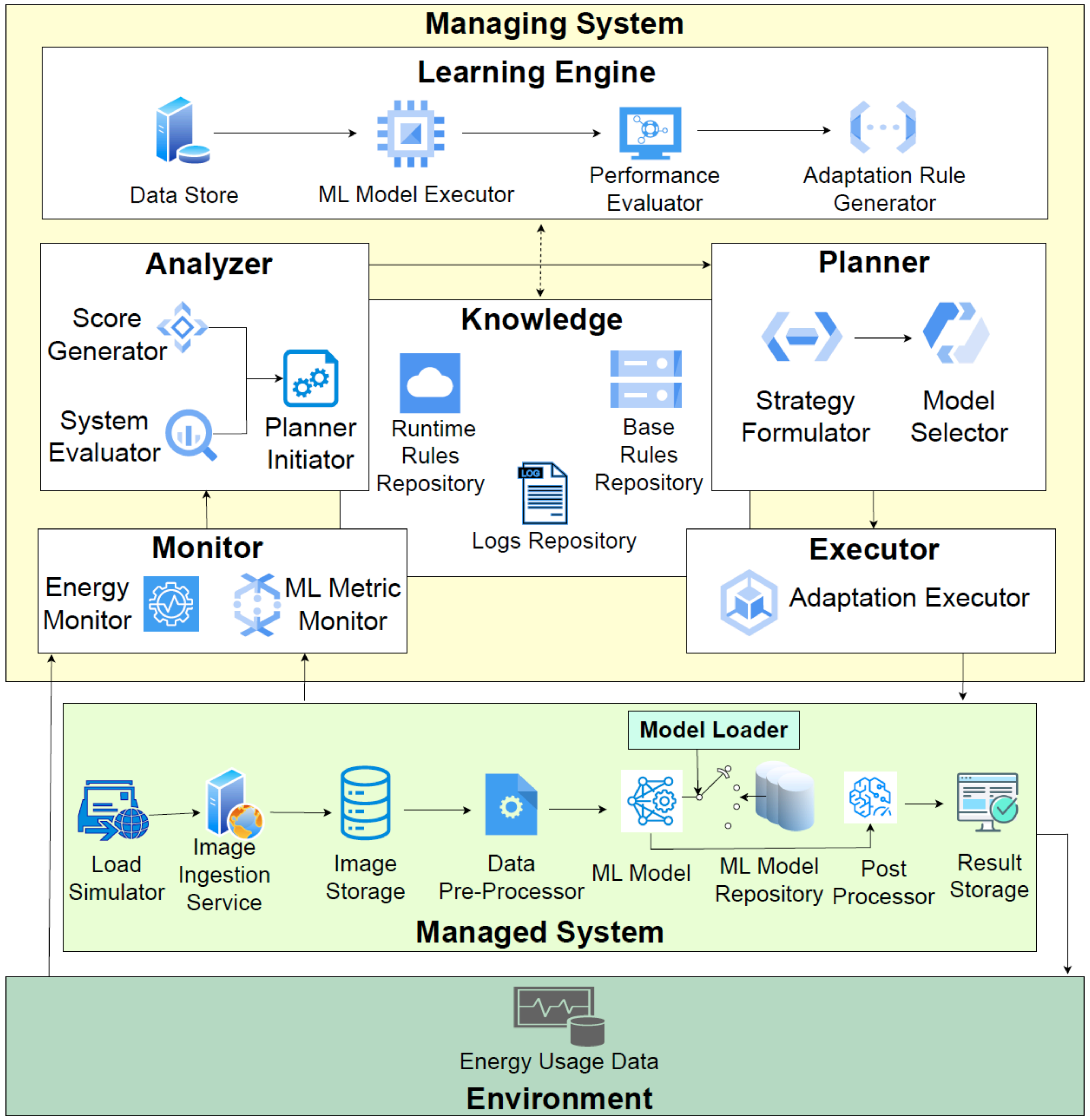
Karthik Vaidhyanathan
*Software Engineering Research Center*
*IIIT Hyderabad, India*
karthik.vaidhyanathan@iiit.ac.in



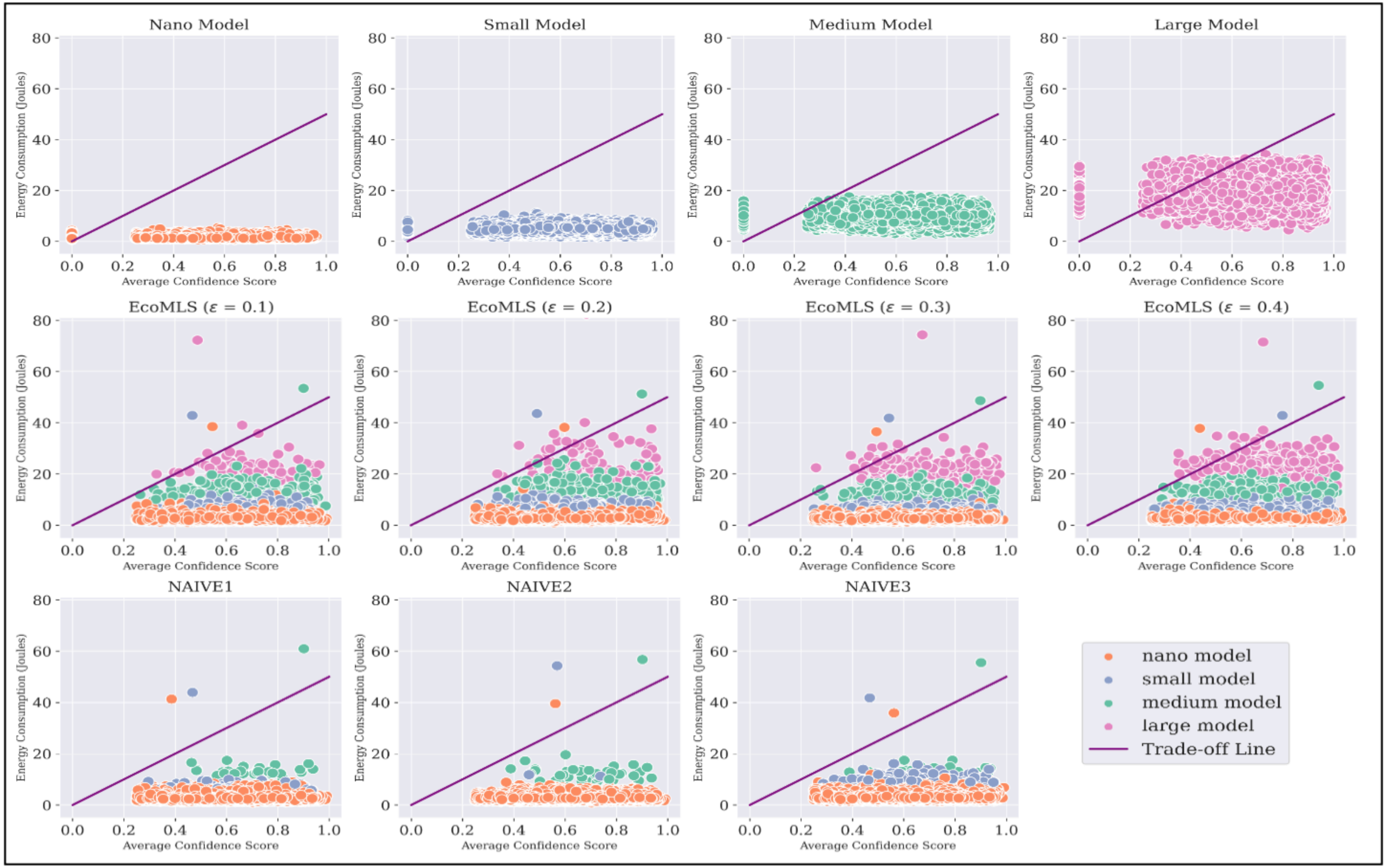
***Abstract*—The sustainability of Machine Learning-Enabled Systems (MLS), particularly with regard to energy efficiency, is an important challenge in their development and deployment. Self-adaptation techniques, recognized for their potential in energy savings within software systems, have yet to be extensively explored in Machine Learning-Enabled Systems (MLS), where runtime uncertainties can significantly impact model performance and energy consumption. This variability, alongside the fluctuating energy demands of ML models during operation, necessitates a dynamic approach. Addressing these challenges, we introduce EcoMLS approach, which leverages the Machine Learning Model Balancer concept to enhance the sustainability of MLS through runtime ML model switching. By adapting**


primarily focused on optimizing the training phase, with less attention given to the energy demands of inference in practical applications [8]–[11]. This gap highlights the need for strategies that reduce energy consumption without compromising performance and can adjust to varying operational demands. The potential of self-adaptation techniques, which balance energy efficiency with QoS, remains largely unexplored in this context [12]. As the ICT sector's energy consumption is expected to increase, creating adaptive, energy-efficient MLS is paramount [13]. Our work seeks to bridge this gap, proposing a self-adaptive approach aiming to ensure MLS

GREENS@ICSA 2024

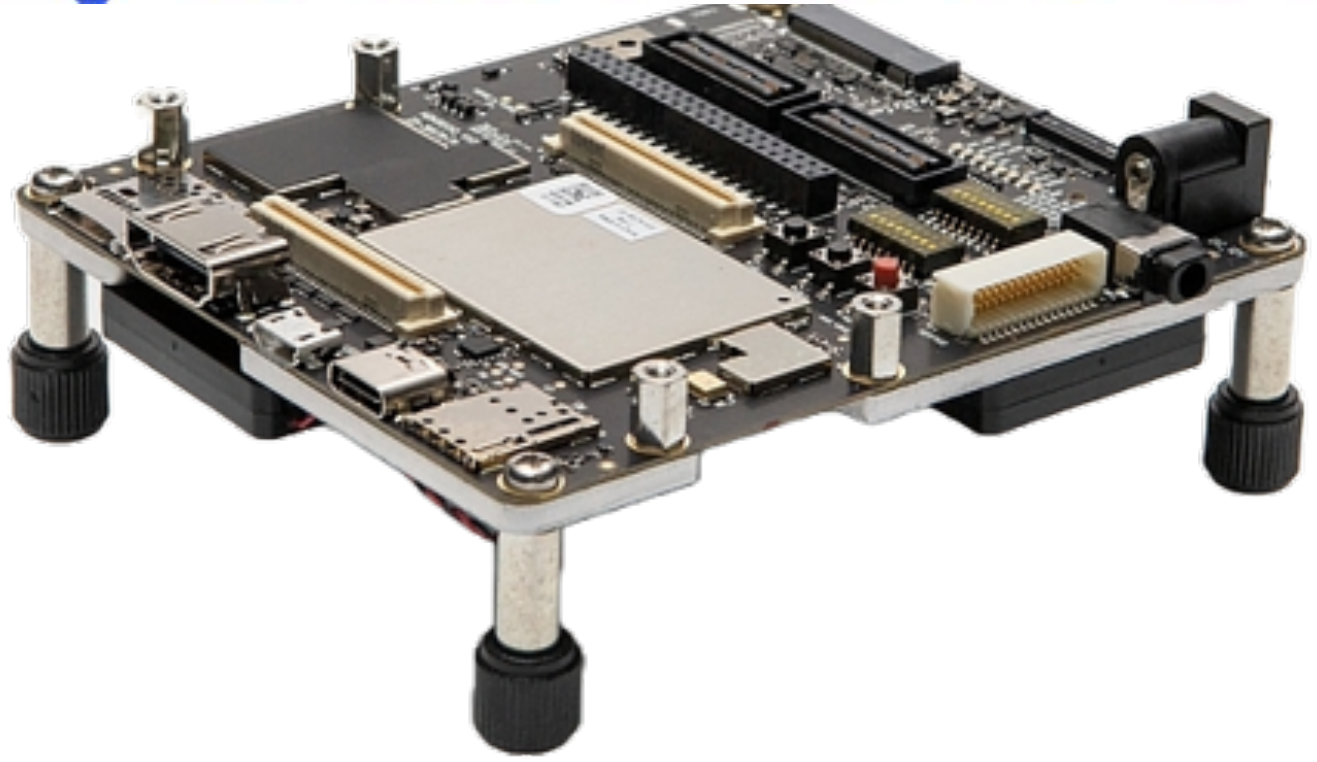


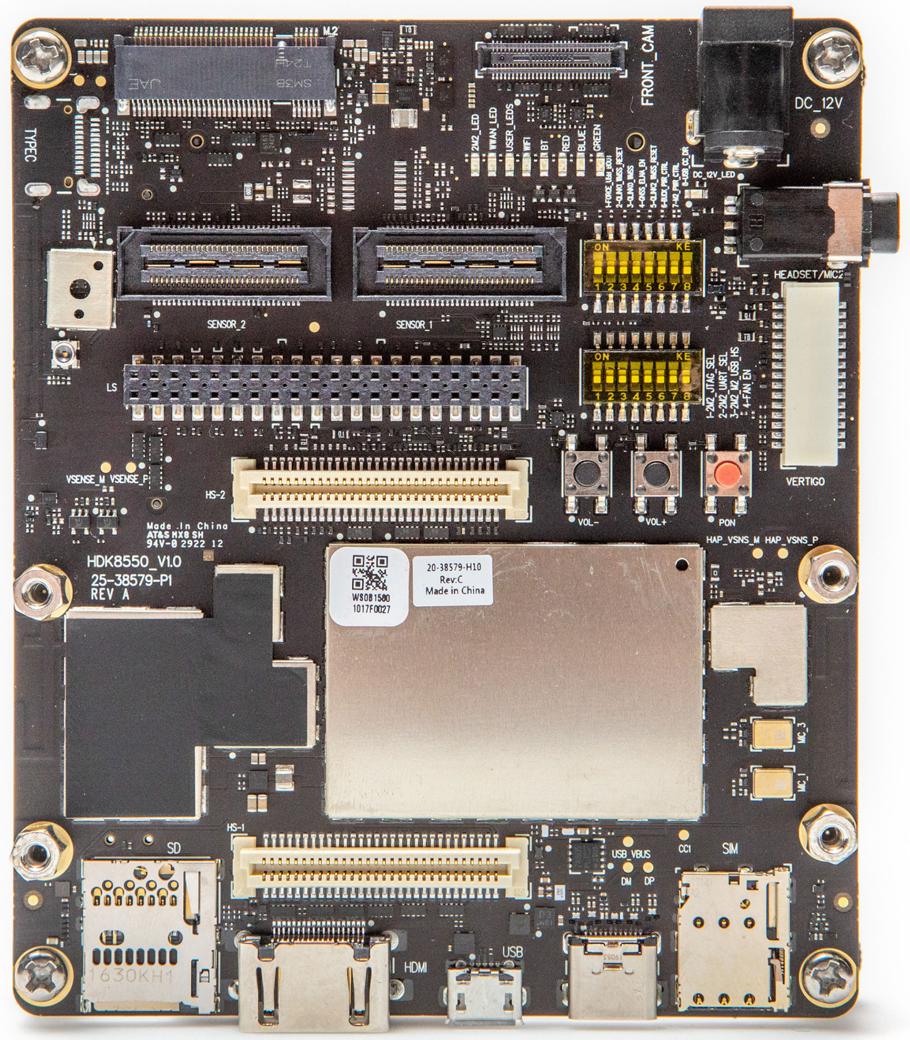
# EcoMLS in Action



# Putting it together on Qualcomm QIDK

- Qualcomm Innovators Development Kit
- Supports prototyping of on-device AI solutions
- Provides access to the premium Qualcomm Snapdragon SoC
- Can be easily deployed to smartphones running Qualcomm snapdragon
- Pre-trained models can be bundled to Android applications
- Part of the Qualcomm EdgeAI labs@IIIT-H

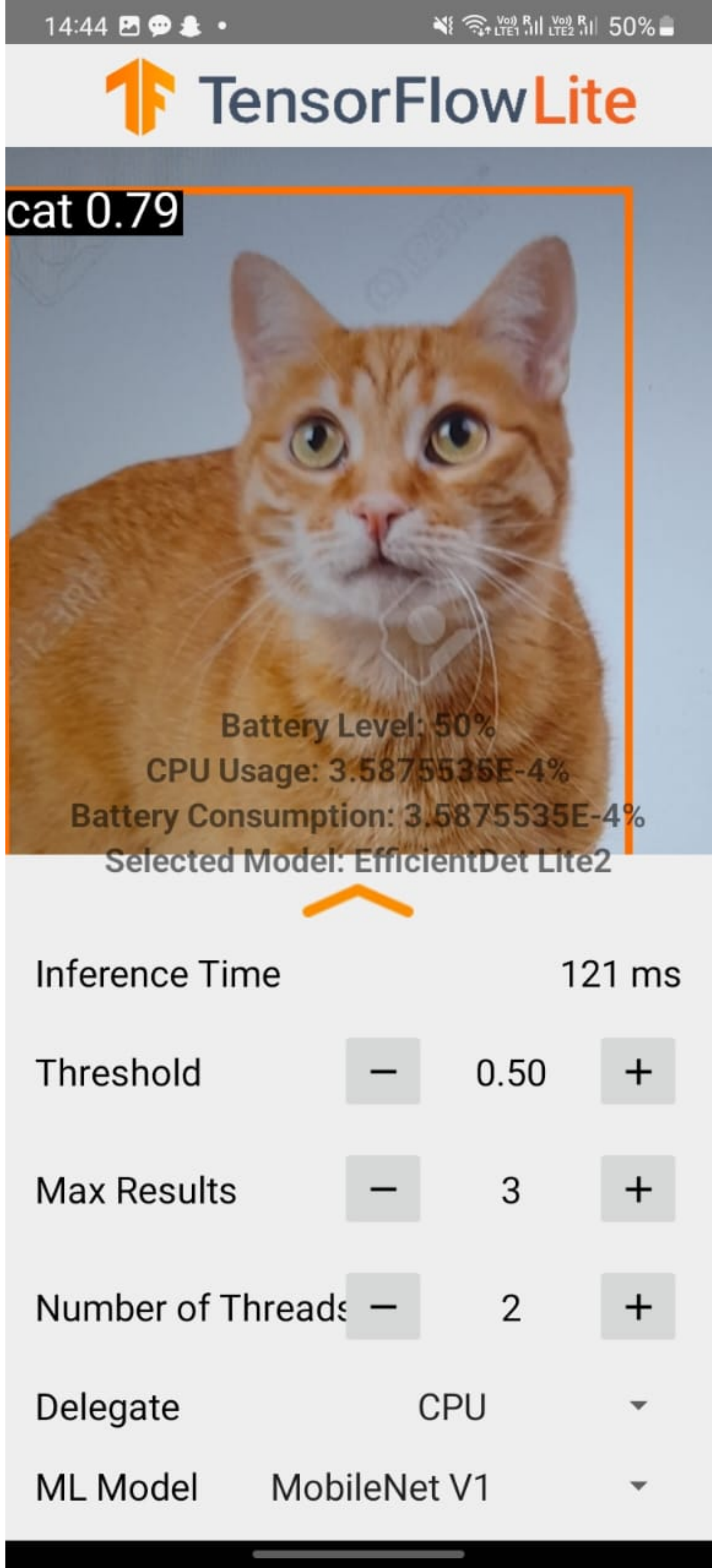




# Model Balancer in QIDK

## Some Results

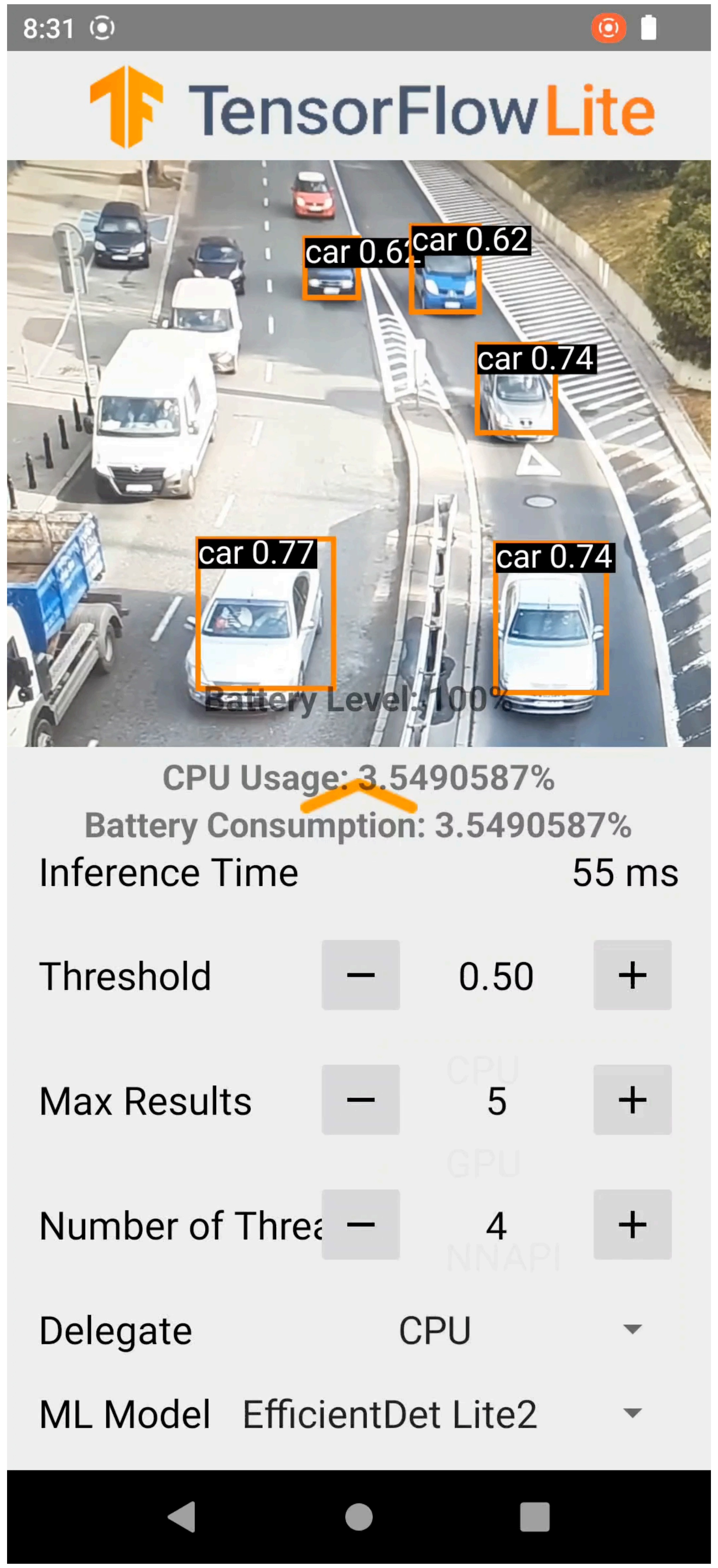
| ML Model | Threads | Inference Time | | |
|---|---|---|---|---|
| | | HDK KIT | Pixel 4 (Data Provided by Tensorflow) | Samsung Galaxy M53 5G |
| MobileNet V1 | 1-4 | 10-20 ms | - | |
| EfficientDet Lite0 | 1-4 | 18-30 ms | 50-36 ms | 50-70 ms |
| EfficientDet Lite1 | 1-4 | 30-50 ms | 49-91 ms | 90-170 ms |
| EfficientDet Lite 2 | 1-4 | 50-84 ms | 69 – 144 ms | 120 – 200 ms |



# QIDK In Action: Demo

### Highlights

- Invited for Qualcomm developer Conference, 2024, Hyderabad, India
- Work presented in the Qualcomm University Platform Symposium - One among 14 universities across the world

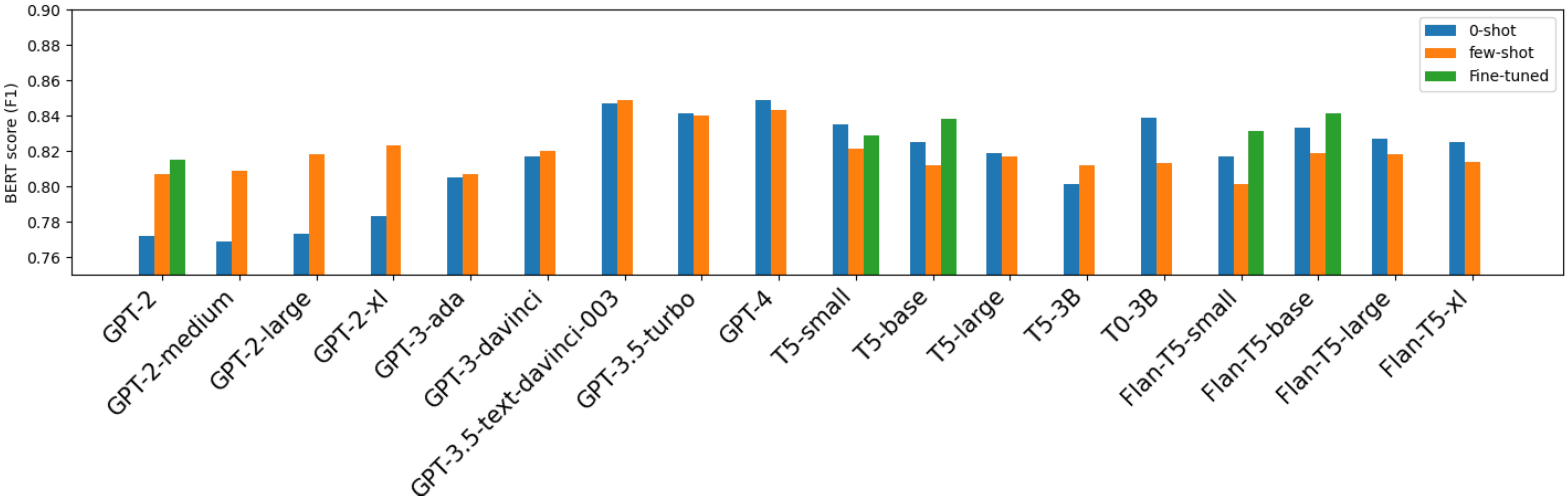



Software Engineering Research Centre

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

# Some more results from the LLM World

## Study on using LLM for generating Architecture design decisions



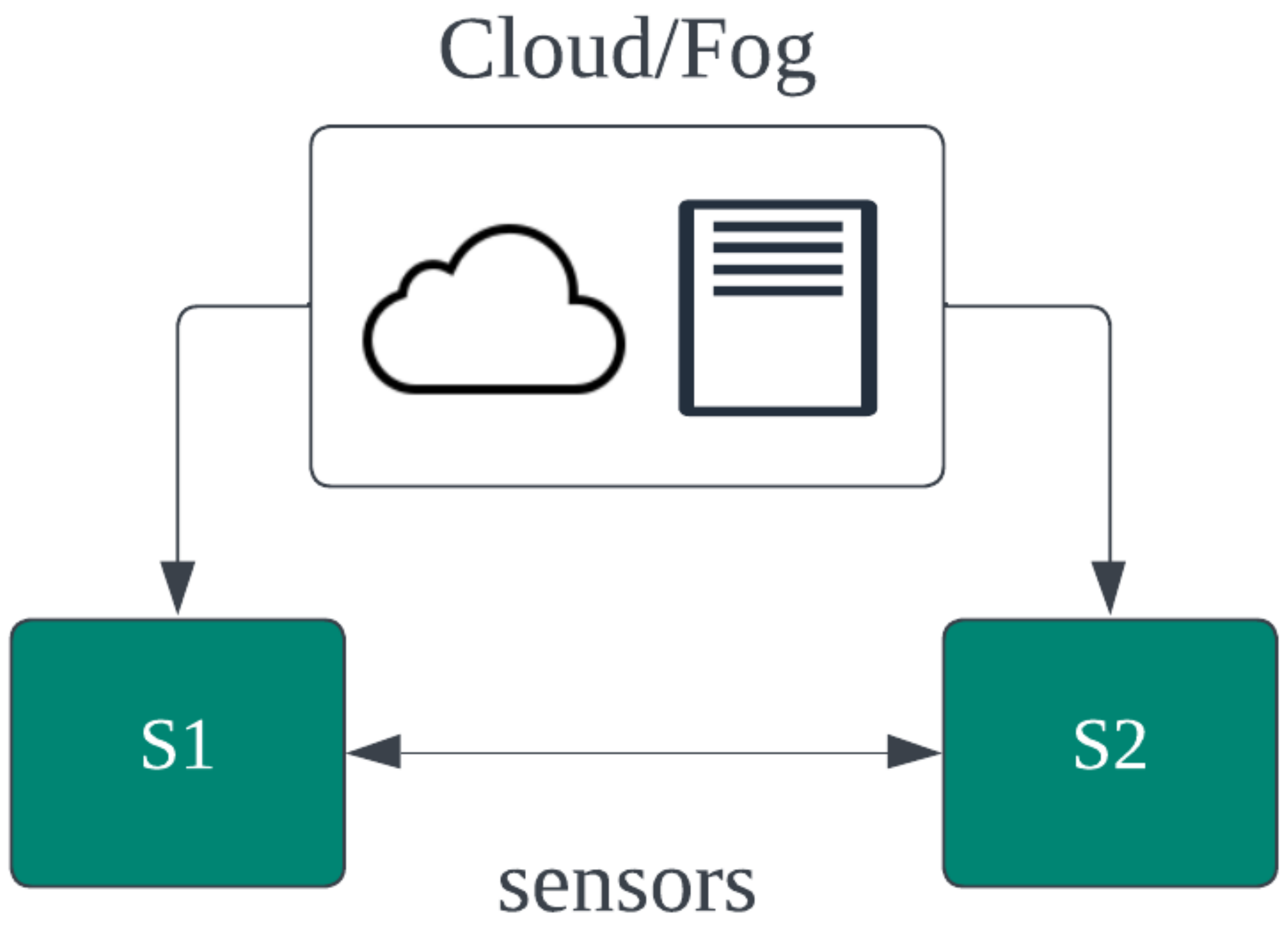

Ongoing works on fine-tuning SLMs for API calling - Runs on edge devices


Dhar, R., Vaidhyanathan, K. and Varma, V. **Can LLMs Generate Architectural Design Decisions? - An Exploratory Empirical study,** ICSA 2024

# Moving Beyond to Edge-Cloud Continuum

- **CloudAI** - Training large AI models (data and compute), Security and privacy, latency, Accuracy of AI models, Scalability, Sustainability (energy/carbon footprint), deployment of large AI models
- **EdgeAI** - Real-time inference of AI models (latency), energy efficiency, scalability, security and privacy, communication overhead, training AI models, Accuracy of AI models, deployment of large AI models
- Various run-time uncertainties affect the performance (resource utilisation, model metrics, hardware constraints, etc).
- How about system having the intelligence to autonomously adapt in edge, in cloud and utilise the edge-cloud continuum?

# Essentially it boils down to!

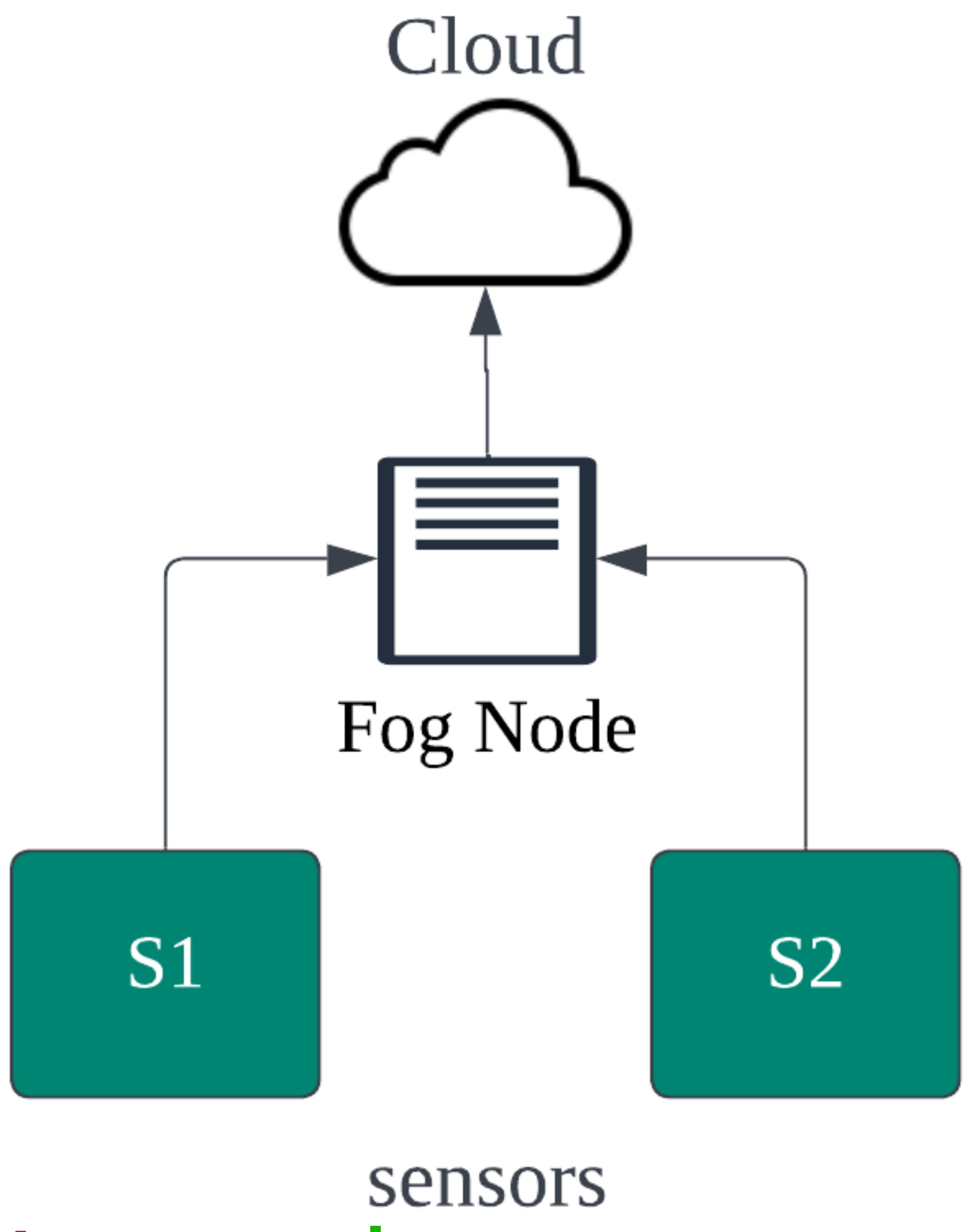



**Edge**

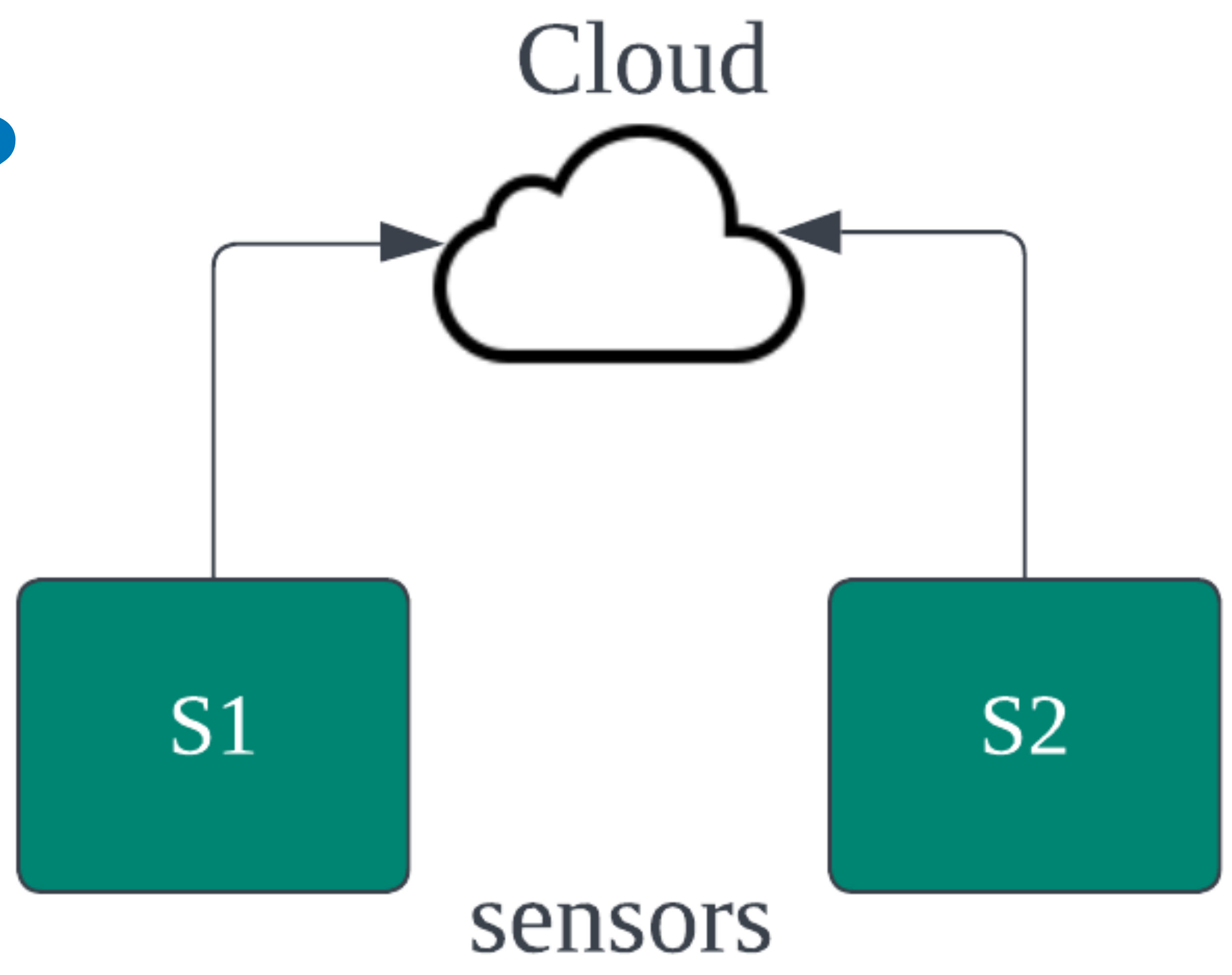
Low latency, high battery

**Can we dynamically adapt?**

**Fog**




Higher latency, less energy, setup cost




**Cloud**

Highest latency, less energy

# Can we reuse?

## Quantitative Verification-Aided Machine Learning: A Tandem Approach for Architecting Self-Adaptive IoT Systems

Javier Cámara
*University of York*
York, United Kingdom
javier.camaramoreno@york.ac.uk

Henry Muccini
*University of L'Aquila*
L'Aquila, Italy
henry.muccini@univaq.it

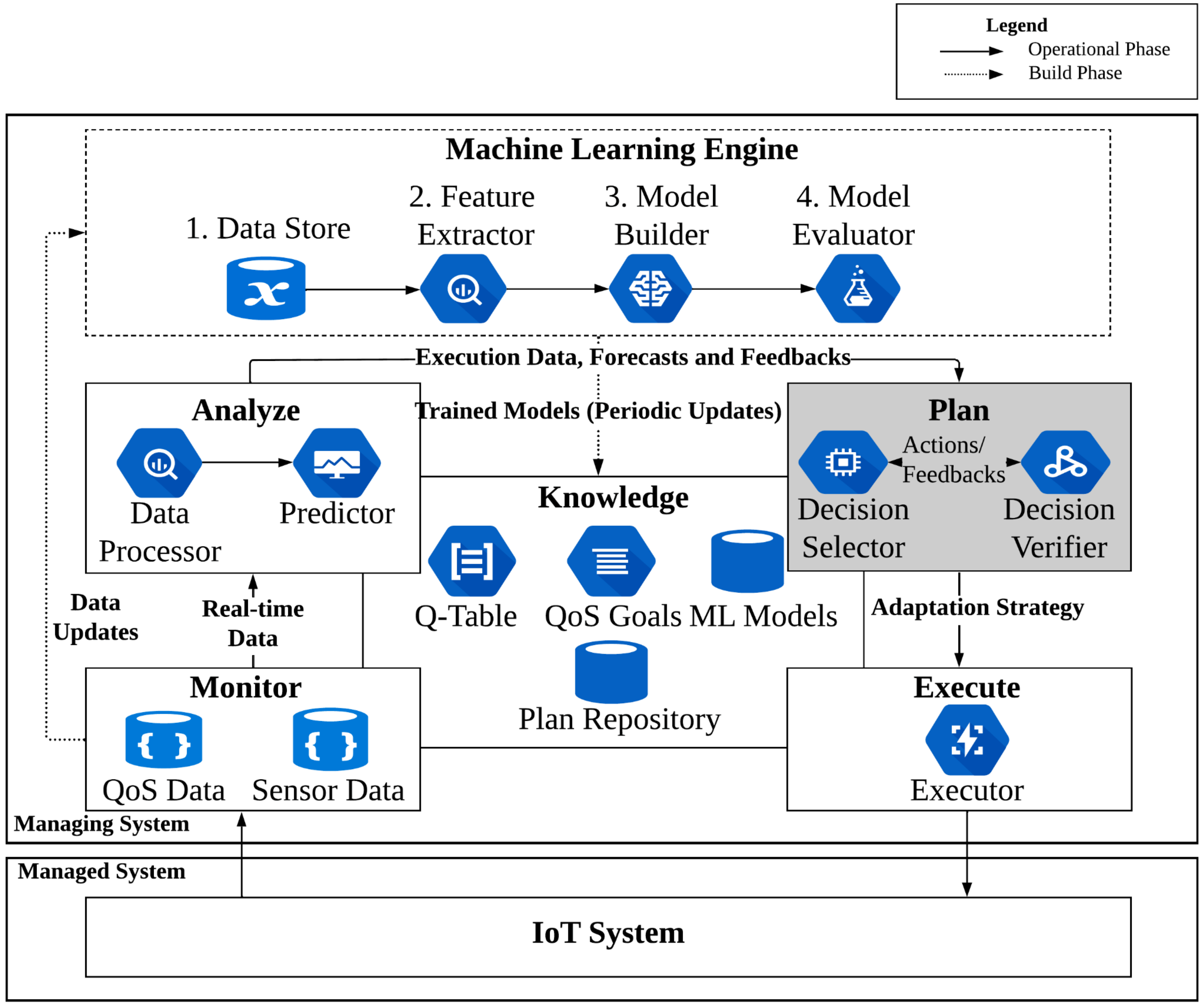
Karthik Vaidhyanathan
*Gran Sasso Science Institute*
L'Aquila, Italy
karthik.vaidhyanathan@gssi.it

***Abstract*—Architecting IoT systems able to guarantee Quality of Service (QoS) levels can be a challenging task due to the inherent uncertainties (induced by changes in e.g., energy availability, network traffic) that they are subject to. Existing work has shown that machine learning (ML) techniques can be effectively used at run time for selecting self-adaptation patterns that can help maintain adequate QoS levels. However, this class of approach suffers from learning bias, which induces accuracy problems that might lead to sub-optimal (or even unfeasible) adaptations in some situations. To overcome this limitation, we propose an approach for proactive self-adaptation which combines ML and**

and physical processes) [12]. Inability to mitigate the effects of these uncertainties can have major implications on the quality of service (QoS) levels offered by these systems [4], [5].

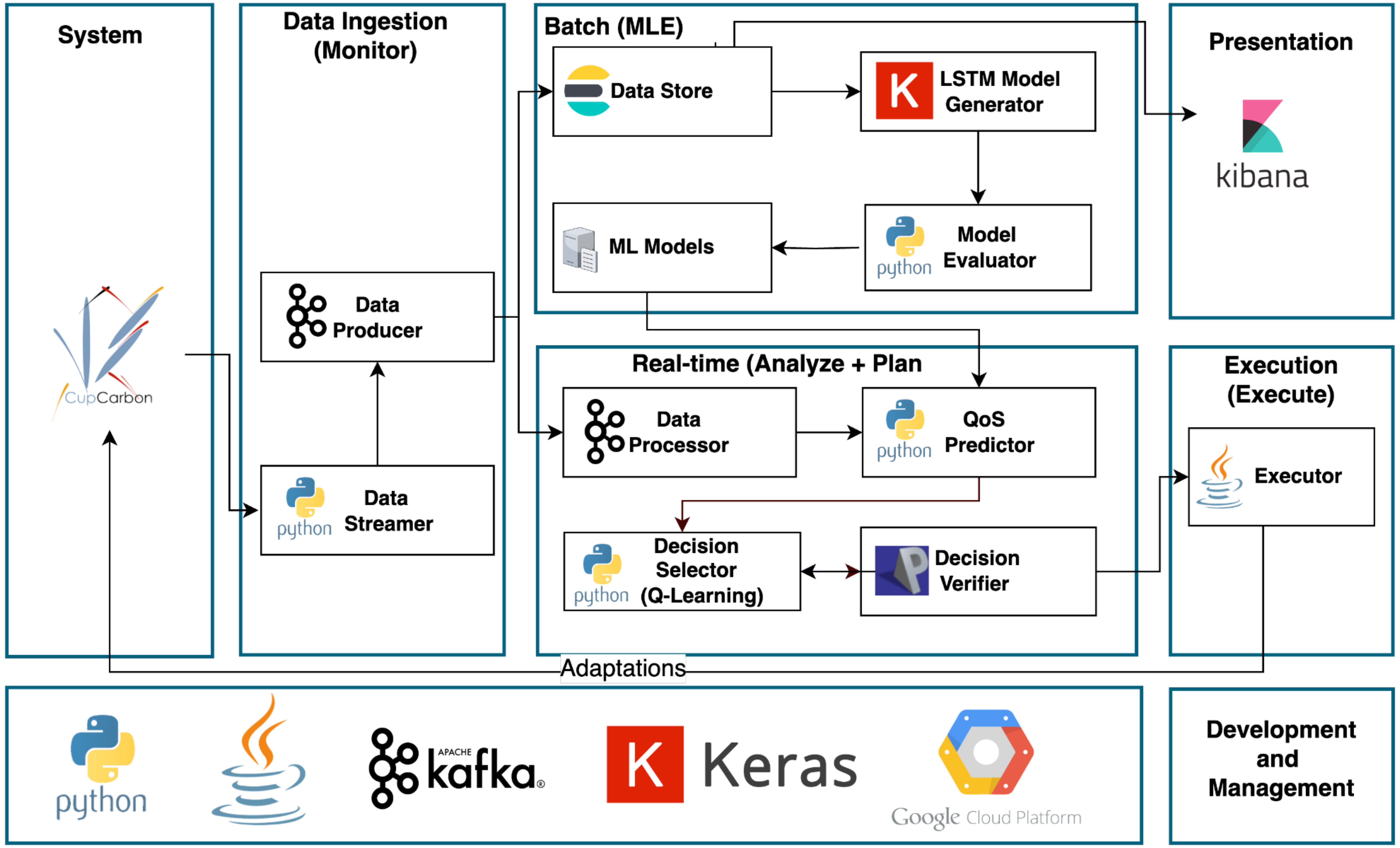
To improve this situation, recent years have seen the emergence of multiple architecture-based self-adaptation techniques aimed at maintaining and guaranteeing improved levels of QoS in applications deployed in different domains [8], [9]. In the specific area of IoT, architectural patterns for self-adaptation recently proposed [14] aim at maintaining accept-

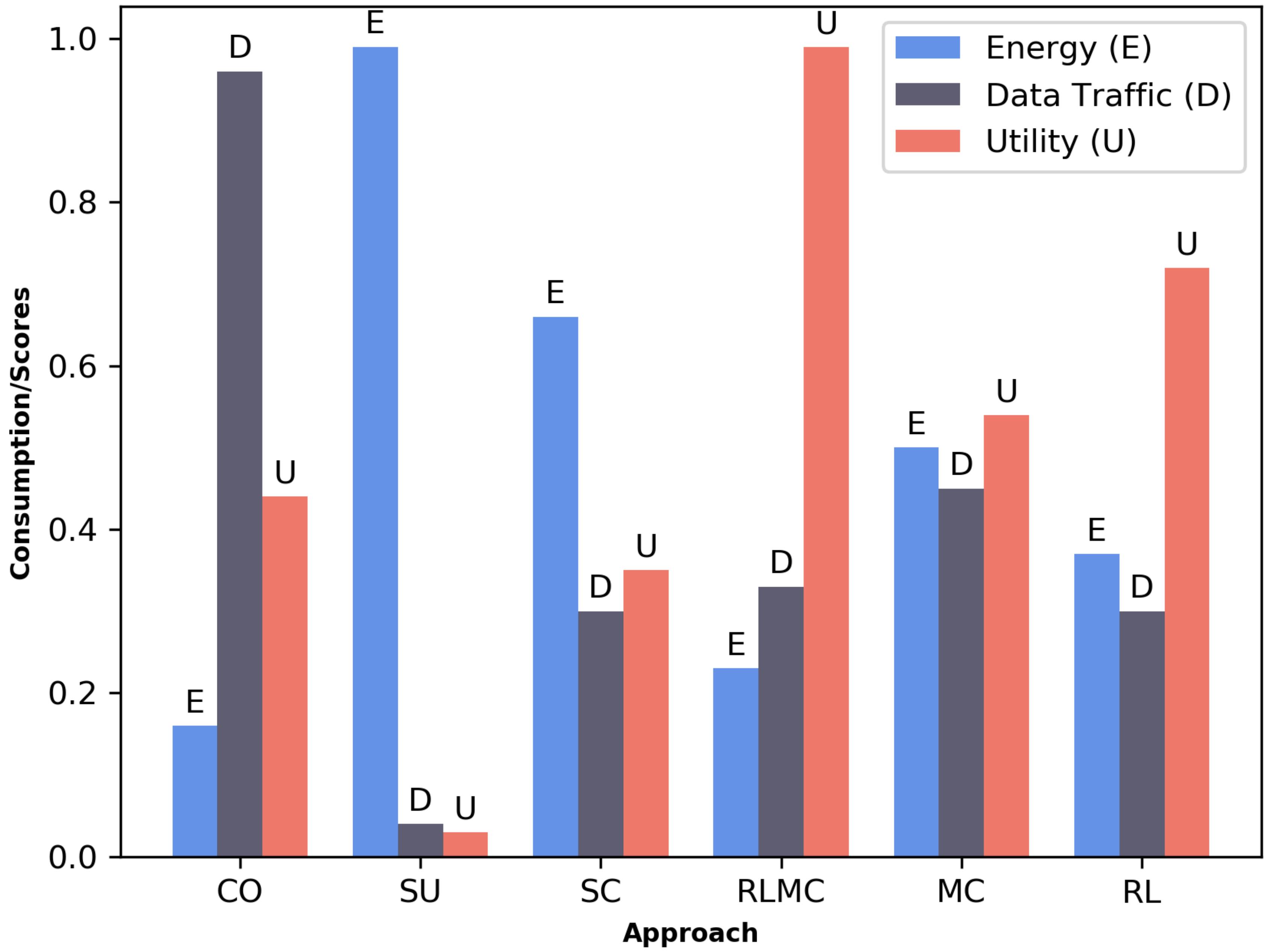
ICSA 2020




As long as ML uncertainties are taken care!

# Implementations and Results



# Going Forward

- **Edge Cloud Continuum for Self-adaptation**
  - Intelligently switch AI processing between edge and cloud
  - Switch could also be between options in edge
  - Proactive approaches for pre-fetch of resources
- **Sustainable EdgeAIOps**
  - Effective management of AIOps on edge
  - Model versioning, governance, automated pipelines

# Going Forward

- **Optimising EdgeML on scale using Dynamic Self-adaptation**
  - Adapt the model in use in edge node in large scale networks
- **Domain specific LLMs on edge**
  - Collection of SLMS running on edge
  - Identification of right LLM for a task
  - Some work on SLMs is on going

# Key Takeaways

## *EdgeAI is here to stay and it can be a game changer - More work is required*

- Increasing focus on Edge systems
- AI models getting bigger is one side of the story
  - We also need models to be smaller and accurate
- Handling uncertainties is the key
  - Self-adaptation can be an enabler: Edge-Cloud
- Need for better ways to architect/engineering EdgeAI systems (maintainability!)
- Efforts on hardware accelerators, efficient neural processing units are some of the way forward

Report from Dagstuhl Seminar 23302

### Software Architecture and Machine Learning


**Grace A. Lewis*[1], Henry Muccini*[2], Ipek Ozkaya[3], Karthik Vaidhyanathan†[4], Roland Weiss*[5], and Liming Zhu*[6]**

1 **Carnegie Mellon Software Engineering Institute – Pittsburgh, US.** glewis@sei.cmu.edu
2 **University of L'Aquila, IT.** henry.muccini@univaq.it
3 **Carnegie Mellon Software Engineering Institute – Pittsburgh, US.** ozkaya@sei.cmu.edu
4 **IIIT Hyderabad, IN.** karthik.vaidhyanathan@iiit.ac.in
5 **ABB – Mannheim, DE.** roland.weiss@gmail.com
6 **Data61, CSIRO – Sydney, AU.** liming.zhu@data61.csiro.au



**Abstract**

This report documents the program and outcomes of Dagstuhl Seminar 23302, "Software Architecture and Machine Learning". We summarize the goals and format of the seminar, results from the breakout groups, key definitions relevant to machine learning-enabled systems that were discussed, and the research roadmap that emerged from the discussions during the seminar. The report also includes the abstracts of the talks presented at the seminar and summaries of open discussions.



**Seminar** July 23–28, 2023 – https://www.dagstuhl.de/23302
**2012 ACM Subject Classification** Software and its engineering → Software architectures; Computing methodologies → Machine learning; Software and its engineering → Extra-functional properties; Computing methodologies → Artificial intelligence; Software and its engineering
**Keywords and phrases** Architecting ML-enabled Systems, ML for Software Architecture, Software Architecture for ML, Machine Learning, Software Architecture, Software Engineering
**Digital Object Identifier** 10.4230/DagRep.13.7.166


#### 1 Executive Summary


*Grace A. Lewis (Carnegie Mellon Software Engineering Institute – Pittsburgh, US)*
*Henry Muccini (University of L'Aquila, IT)*
*Ipek Ozkaya (Carnegie Mellon Software Engineering Institute – Pittsburgh, US)*
*Karthik Vaidhyanathan (IIIT – Hyderabad, IN)*
*Roland Weiss (ABB – Mannheim, DE)*
*Liming Zhu (Data61, CSIRO – Sydney, AU)*




The pervasive and distributed nature of many of today's software systems requires making complex design decisions to guarantee important system qualities such as performance, reliability, safety and security. The practices within the field of software architecture guide the design and development of software systems from its high-level blueprint down to their implementation and operations. While the fundamentals of software architecture practices

* Editor / Organizer
† Editorial Assistant / Collector




Software Architecture and Machine Learning, *Dagstuhl Reports*, Vol. 13, Issue 7, pp. 166–188
Editors: Grace A. Lewis, Henry Muccini, Ipek Ozkaya, Karthik Vaidhyanathan, Roland Weiss, and Liming Zhu
Dagstuhl Reports
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany


SERC
Software Engineering Research Centre

# Thanks to my team - SA4S@SERC


Rudra Dhar


Akhila Matathammal

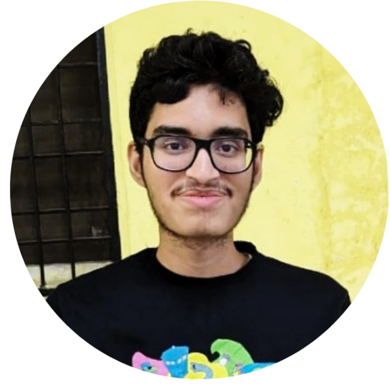

Hiya Bhatt

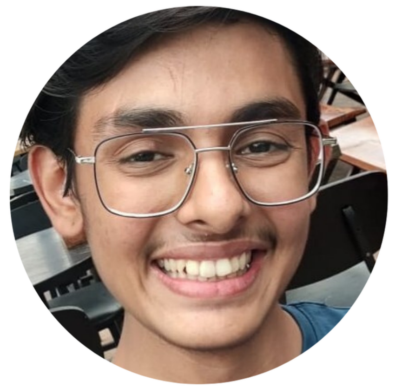

Chandrasekar S

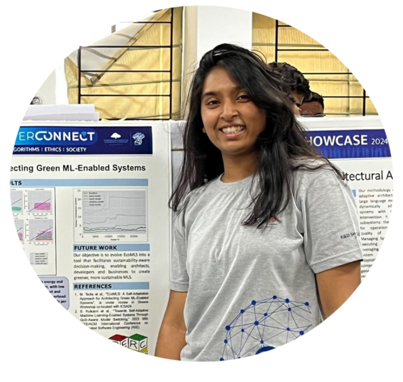

Shubham Kulkarni


Adyansh Kakran


Prakhar Jain


Shrikara A


Arya Pravin Marda


Meghana Tedla


Miryala Sathvika


Prakhar Singhal


Amey Karan


Bassam Adnan


Aneesh Sambu


Shaunak Biswas


Shailender Goyal


Divyansh Pandey


Maddireddy Kritin


Santosh Kotekal


Vyakhya Gupta

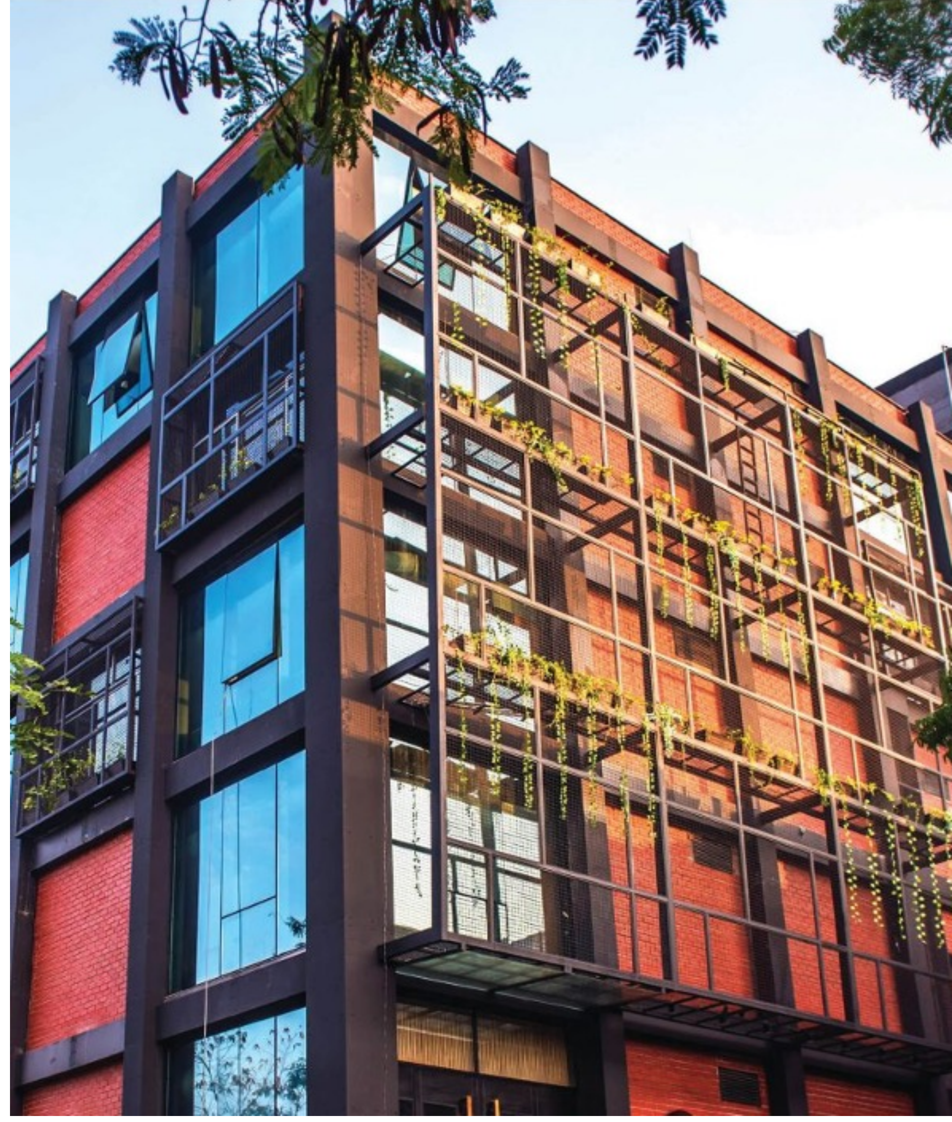
## Team DigIT@IIITH


Deepak Gangadharan


Karthik Vaidhyanathan


Sahil Sahil


Hiya Bhatt



https://serc.iiit.ac.in

Team SA4S

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

SCAN ME

IEEE Software

# Thank you


Web: karthikvaidhyanathan.com
Email: karthik.vaidhyanathan@iiit.ac.in
Twitter: @karthi_ishere


SERC Software Engineering Research Centre

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD