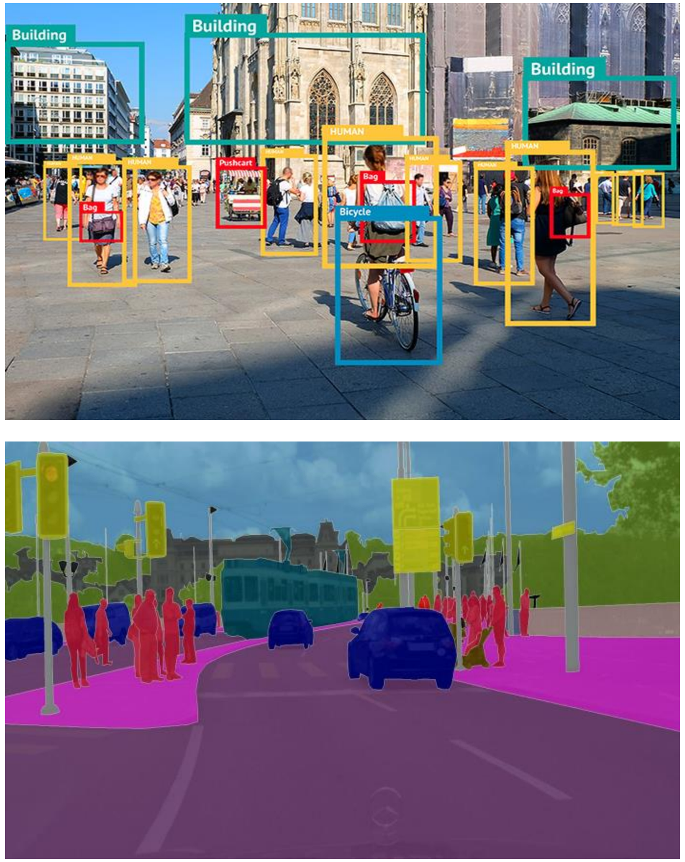
A Brief Introduction to Deep Learning System(DLSys) on Mobile

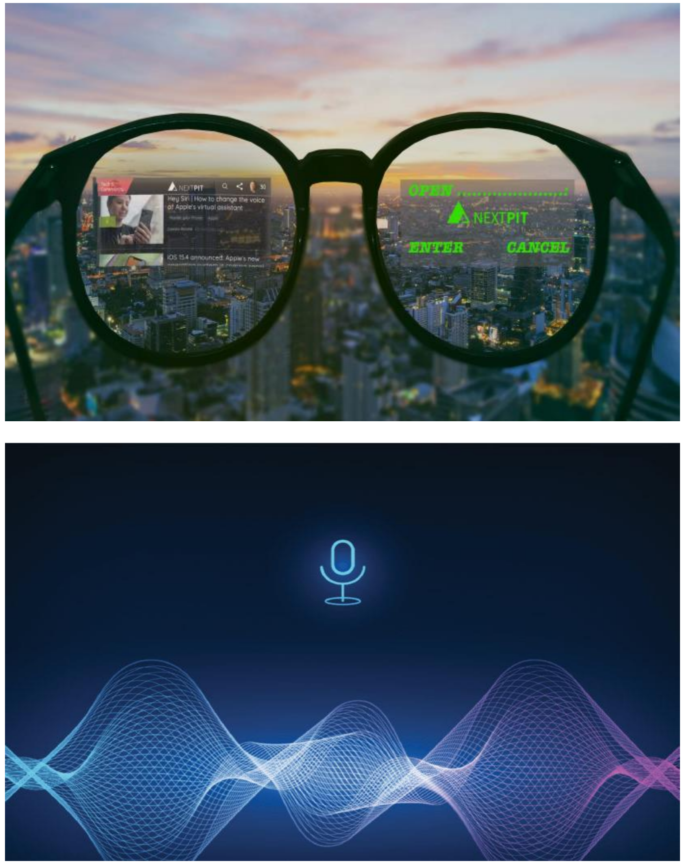
Shengyuan Ye
School of Computer Science and Engineering
Sun Yat-sen University
Contact: yeshy8@mail2.sysu.edu.cn

AI and Deep Learning

- **Deep Learning is all around us.**



Computer Vision



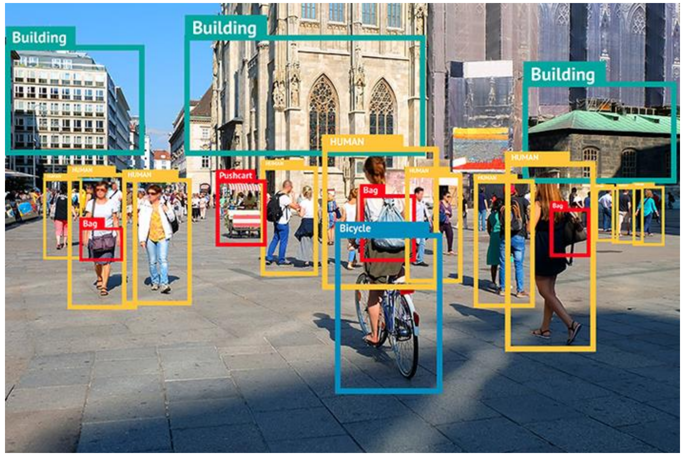
Wearable Agents

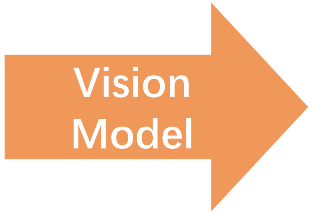
Smart Robotics

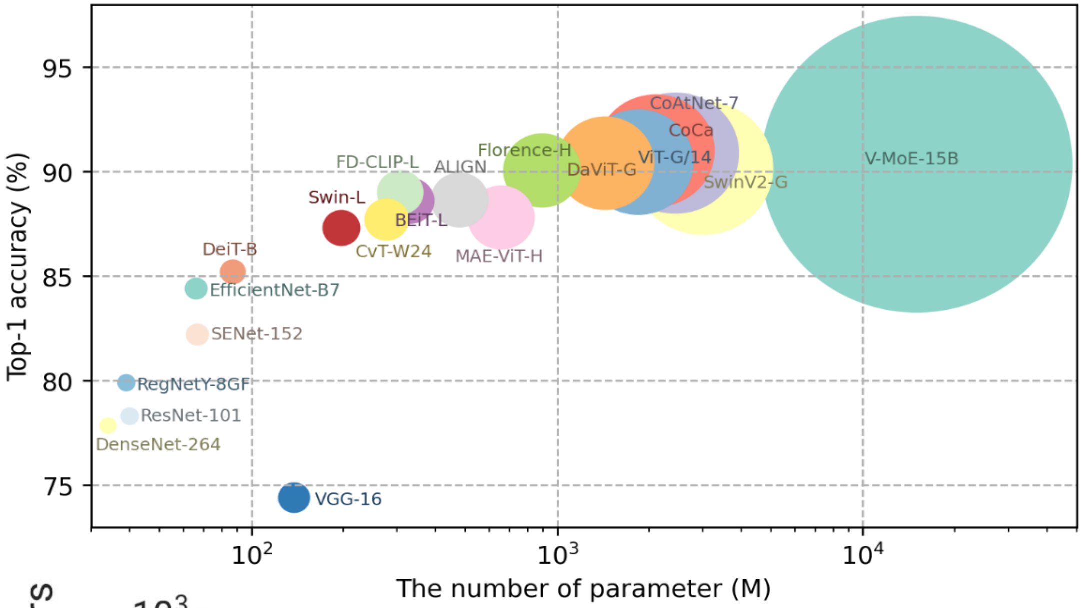
Credit: Google images.

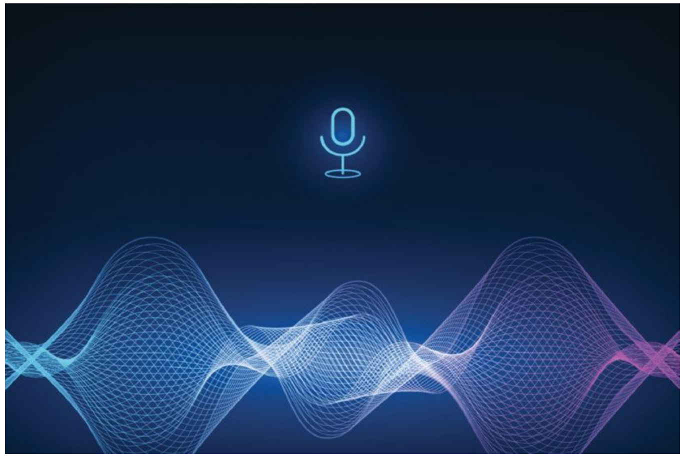
Trend of Deep Learning

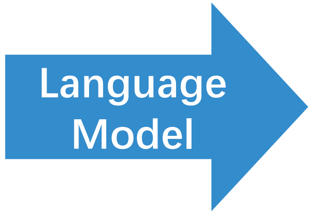
✓ **Both model and dataset are greater and greater!**

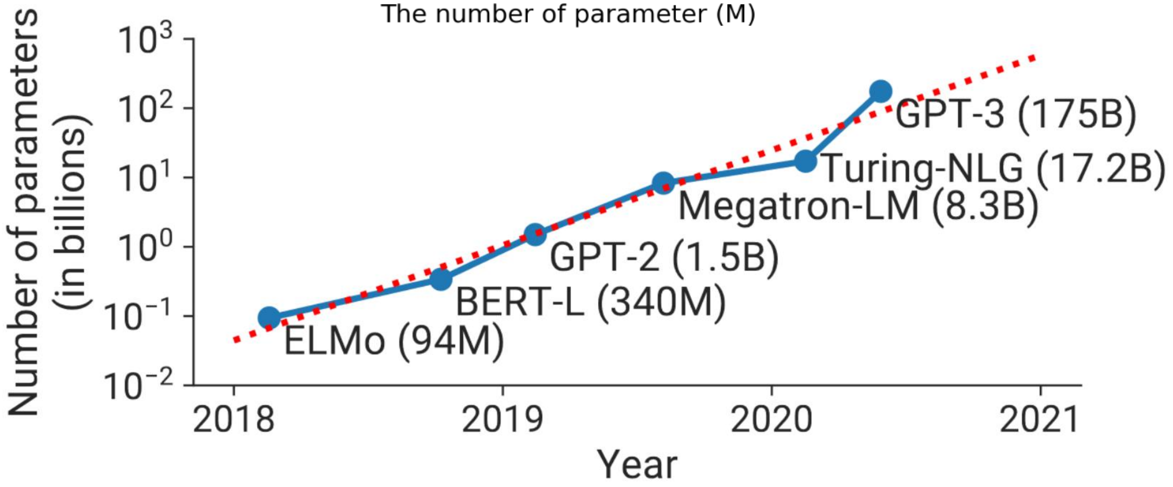






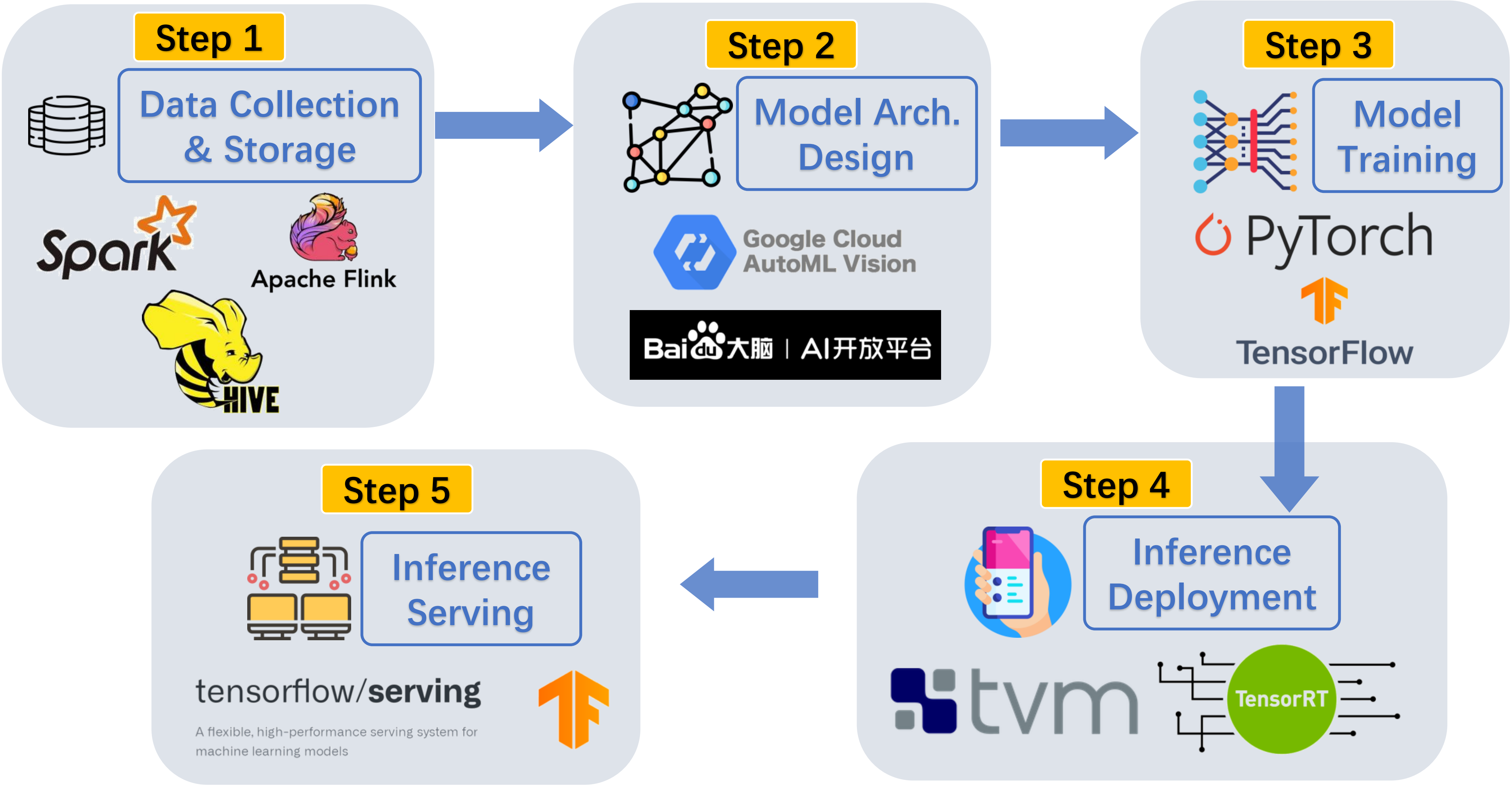






Credit: Google images.

How System Contributes Deep Learning?



Step 1

Data Collection & Storage

Step 2

Model Arch. Design

Step 3

Model Training

Step 4

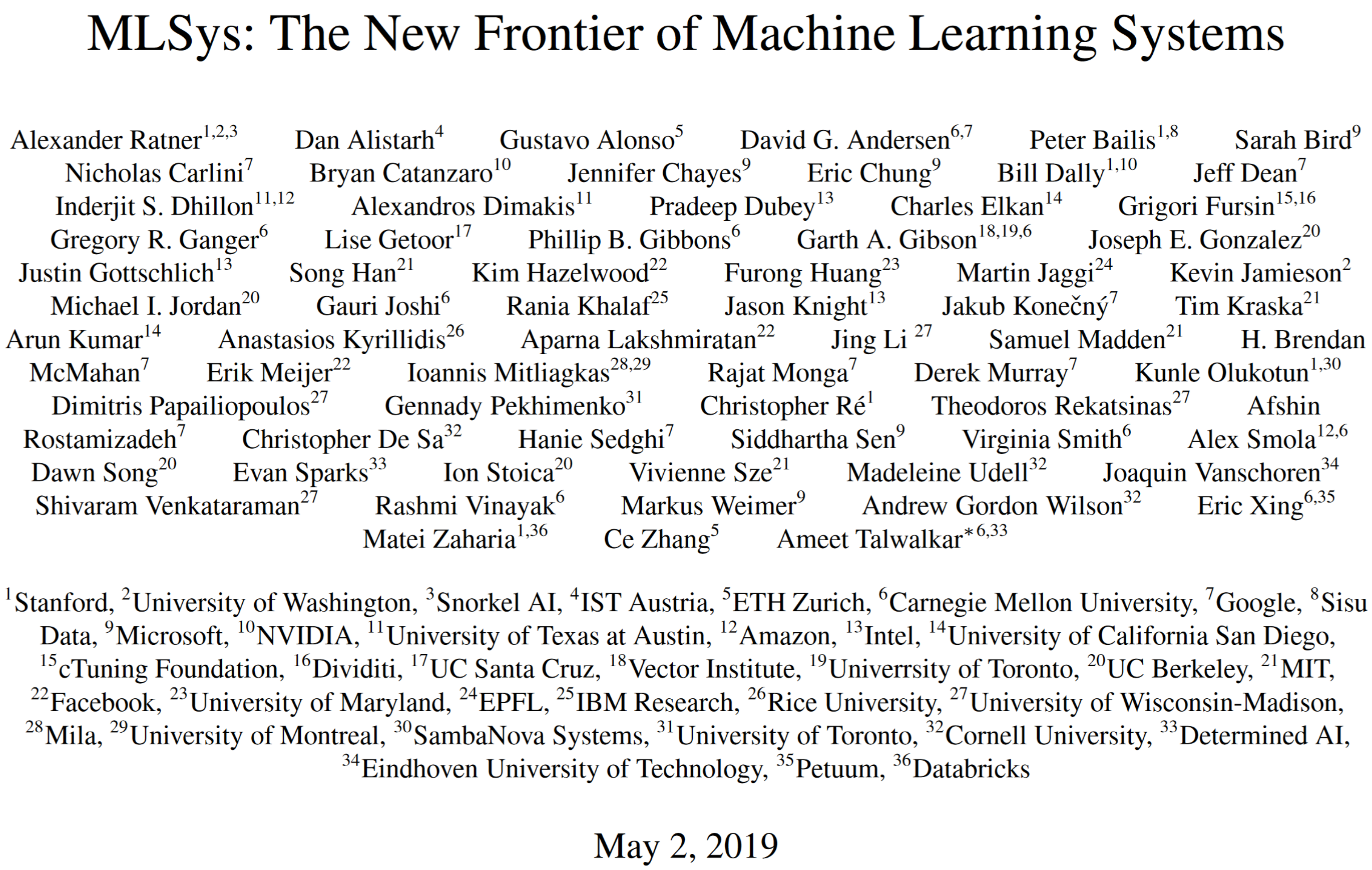
Inference Deployment

Step 5

Inference Serving

Spark
Apache Flink
HIVE
Google Cloud AutoML Vision
Baidu大脑 | AI开放平台
PyTorch
TensorFlow
tvm
TensorRT
tensorflow/serving
A flexible, high-performance serving system for machine learning models

Credit: Google images.



MLSys: The New Frontier of Machine Learning Systems

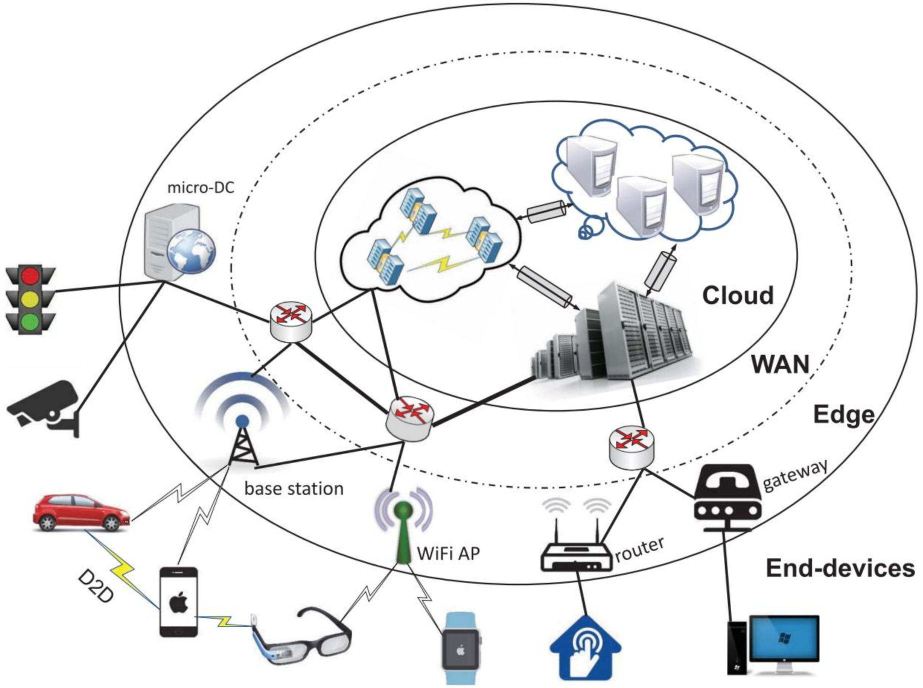
Alexander Ratner[1,2,3] Dan Alistarh[4] Gustavo Alonso[5] David G. Andersen[6,7] Peter Bailis[1,8] Sarah Bird[9] Nicholas Carlini[7] Bryan Catanzaro[10] Jennifer Chayes[9] Eric Chung[9] Bill Dally[1,10] Jeff Dean[7] Inderjit S. Dhillon[11,12] Alexandros Dimakis[11] Pradeep Dubey[13] Charles Elkan[14] Grigori Fursin[15,16] Gregory R. Ganger[6] Lise Getoor[17] Phillip B. Gibbons[6] Garth A. Gibson[18,19,6] Joseph E. Gonzalez[20] Justin Gottschlich[13] Song Han[21] Kim Hazelwood[22] Furong Huang[23] Martin Jaggi[24] Kevin Jamieson[2] Michael I. Jordan[20] Gauri Joshi[6] Rania Khalaf[25] Jason Knight[13] Jakub Konečný[7] Tim Kraska[21] Arun Kumar[14] Anastasios Kyrillidis[26] Aparna Lakshmiratan[22] Jing Li [27] Samuel Madden[21] H. Brendan McMahan[7] Erik Meijer[22] Ioannis Mitliagkas[28,29] Rajat Monga[7] Derek Murray[7] Kunle Olukotun[1,30] Dimitris Papailiopoulos[27] Gennady Pekhimenko[31] Christopher Ré[1] Theodoros Rekatsinas[27] Afshin Rostamizadeh[7] Christopher De Sa[32] Hanie Sedghi[7] Siddhartha Sen[9] Virginia Smith[6] Alex Smola[12,6] Dawn Song[20] Evan Sparks[33] Ion Stoica[20] Vivienne Sze[21] Madeleine Udell[32] Joaquin Vanschoren[34] Shivaram Venkataraman[27] Rashmi Vinayak[6] Markus Weimer[9] Andrew Gordon Wilson[32] Eric Xing[6,35] Matei Zaharia[1,36] Ce Zhang[5] Ameet Talwalkar*[6,33]

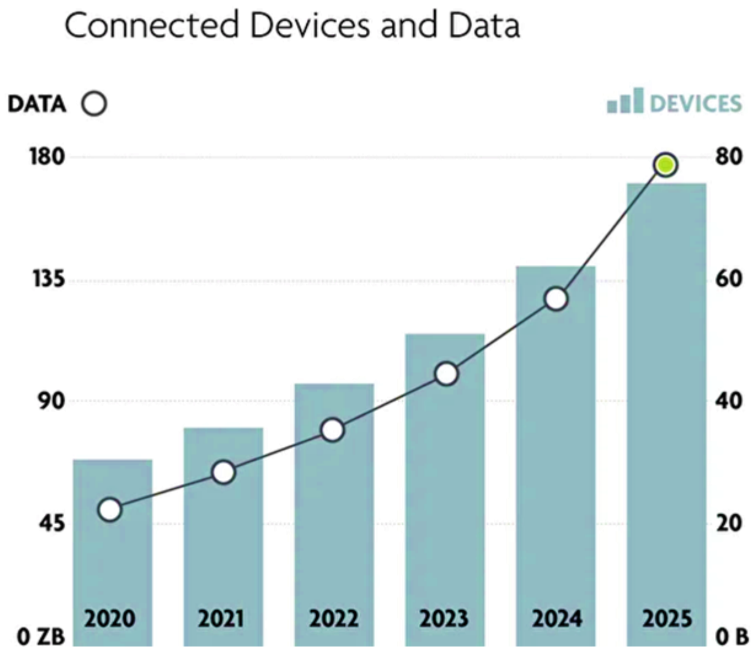
[1]Stanford, [2]University of Washington, [3]Snorkel AI, [4]IST Austria, [5]ETH Zurich, [6]Carnegie Mellon University, [7]Google, [8]Sisu Data, [9]Microsoft, [10]NVIDIA, [11]University of Texas at Austin, [12]Amazon, [13]Intel, [14]University of California San Diego, [15]cTuning Foundation, [16]Dividiti, [17]UC Santa Cruz, [18]Vector Institute, [19]Univerrsity of Toronto, [20]UC Berkeley, [21]MIT, [22]Facebook, [23]University of Maryland, [24]EPFL, [25]IBM Research, [26]Rice University, [27]University of Wisconsin-Madison, [28]Mila, [29]University of Montreal, [30]SambaNova Systems, [31]University of Toronto, [32]Cornell University, [33]Determined AI, [34]Eindhoven University of Technology, [35]Petuum, [36]Databricks

May 2, 2019

Deep Learning System on Mobile Devices

- **Mobile computing** has emerged as a new paradigm
  - Popularization of mobile devices in both magnitude and variety
  - Proliferation of mobile data in both scale and modality

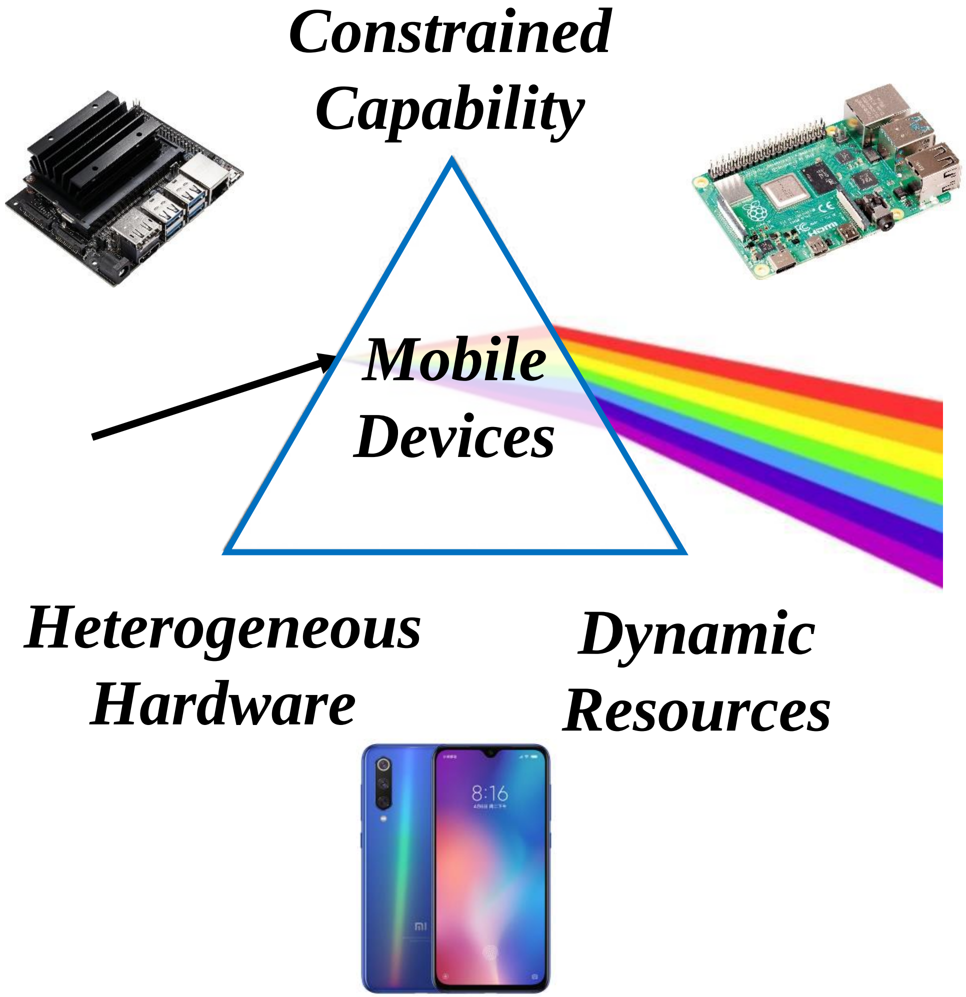




Credit: https://coruzant.com/opinion/the-future-is-edge-computing/

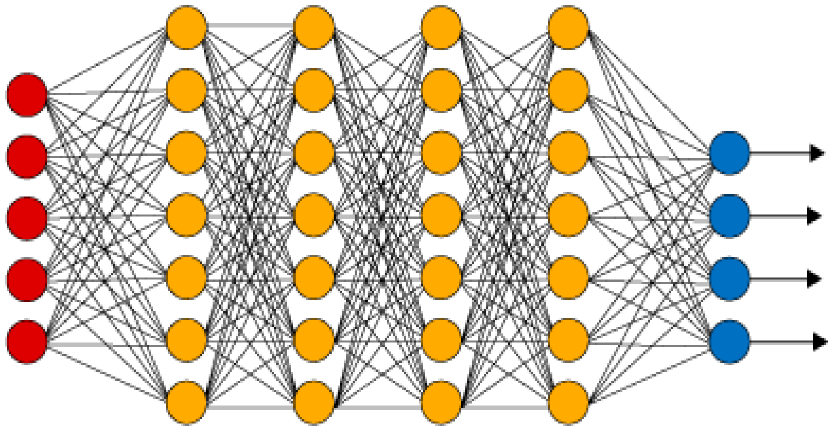
Challenges of Deep Learning on Mobile

How to apply?



Huge Gap

Deep Learning Neural Network



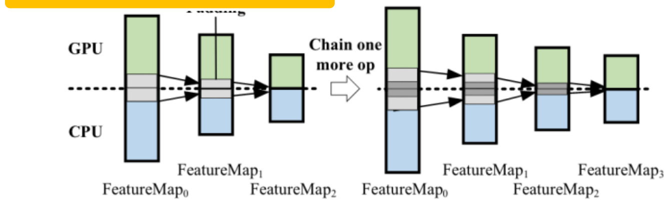
- DNN Computing is extremely computation-intensive and resource-demanding
- Mobile devices are resource-constrained and heterogeneous

Credit: Google Image

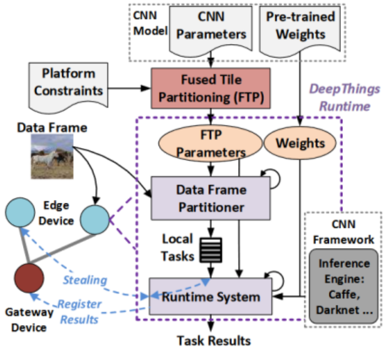
DLSys for Mobile Inference & Training

Inference on Mobile

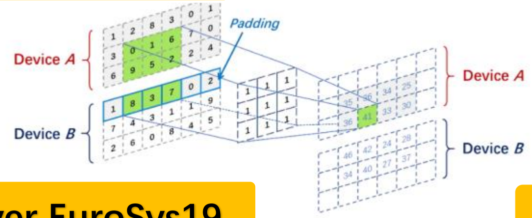
CoDL MobiSys22



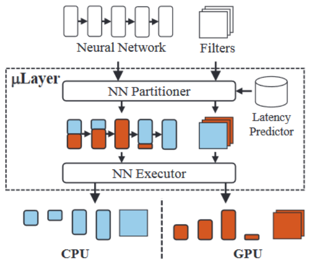
DeepThings TCAD18



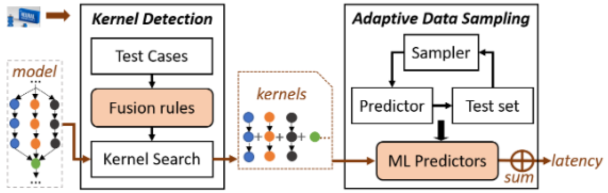
CoEdge TON20



μLayer EuroSys19

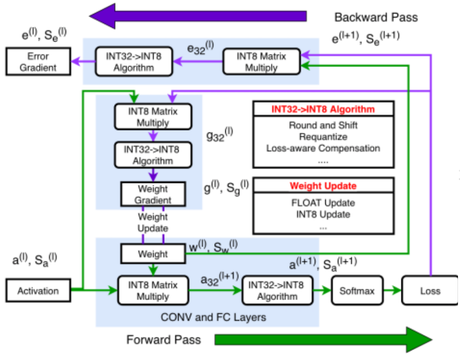


nn-Meter MobiSys21

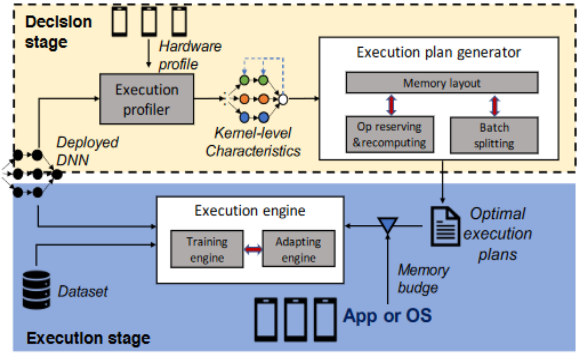


Training on Mobile

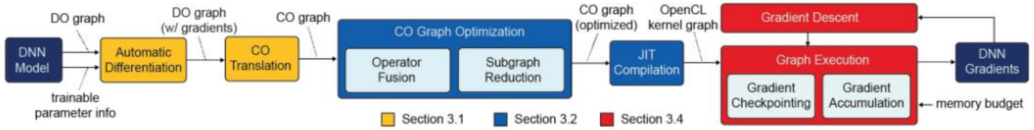
Mandheling MobiCom22



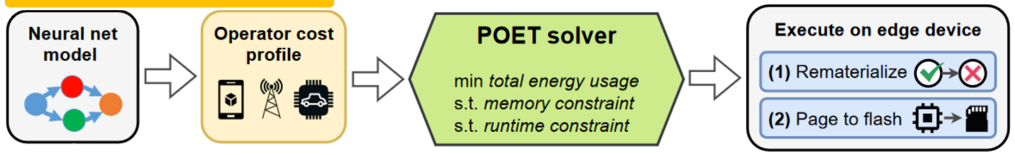
Melon MobiSys22



Sage MobiSys22

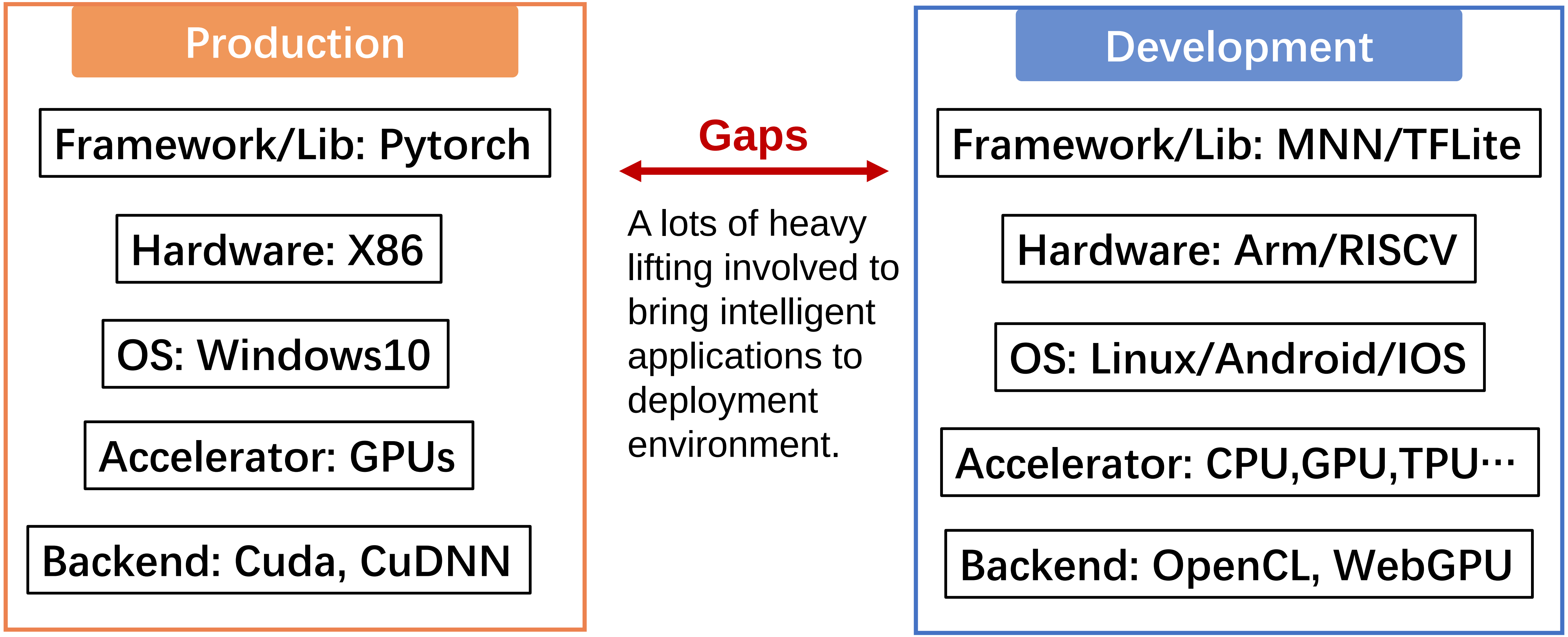


POET NeurIPS22



On-Mobile Inference

- The gap between Production and Development



Production

Framework/Lib: Pytorch

Hardware: X86

OS: Windows10

Accelerator: GPUs

Backend: Cuda, CuDNN

Gaps

A lots of heavy lifting involved to bring intelligent applications to deployment environment.

Development

Framework/Lib: MNN/TFLite

Hardware: Arm/RISCV

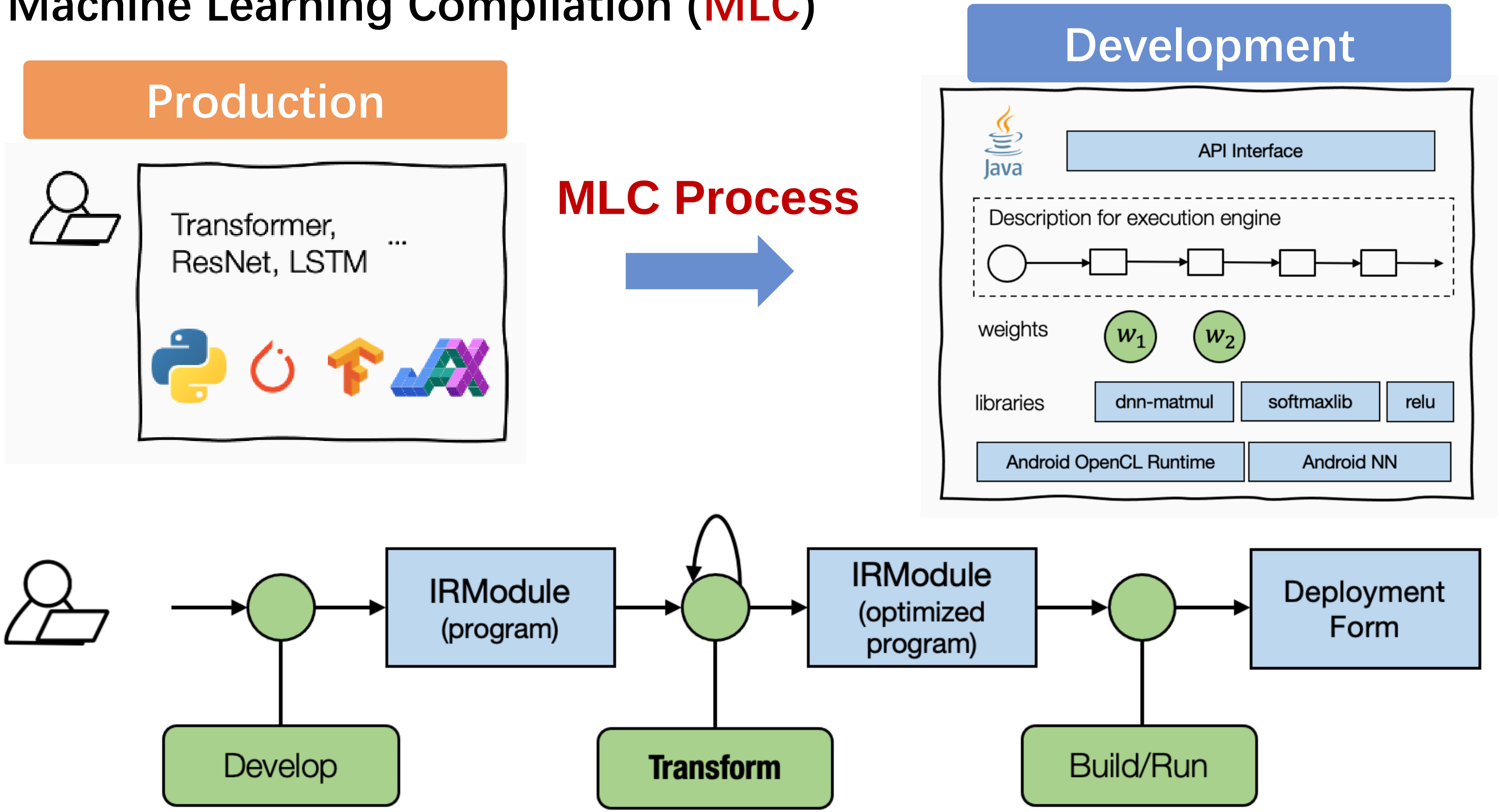
OS: Linux/Android/IOS

Accelerator: CPU,GPU,TPU…

Backend: OpenCL, WebGPU

On-Mobile Inference

- **Machine Learning Compilation (MLC)**



Production

Transformer, ResNet, LSTM ...

MLC Process

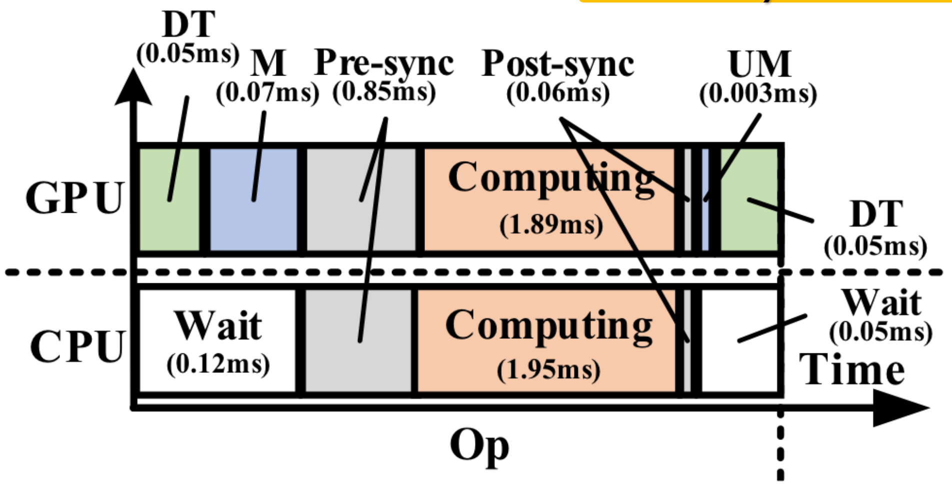
Development

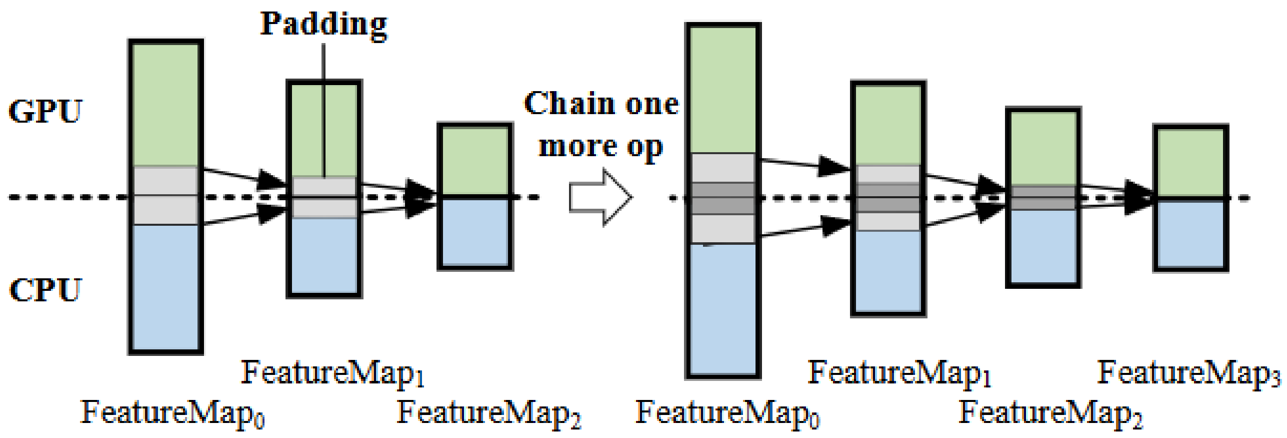
Credit: https://mlc.ai/summer22/.

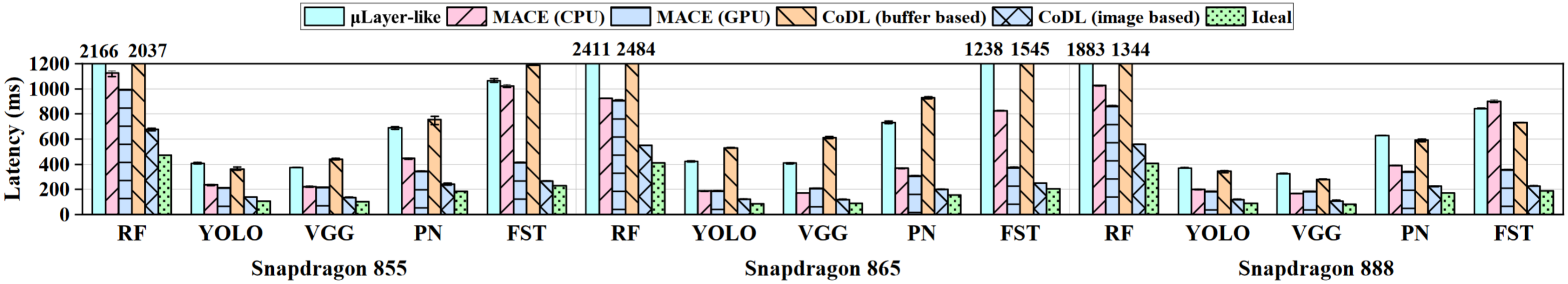
On-Mobile Inference

- **Concurrently Inference on Heterogeneous SoCs**

CoDL, Fuchen Jia et al., MobiSys22





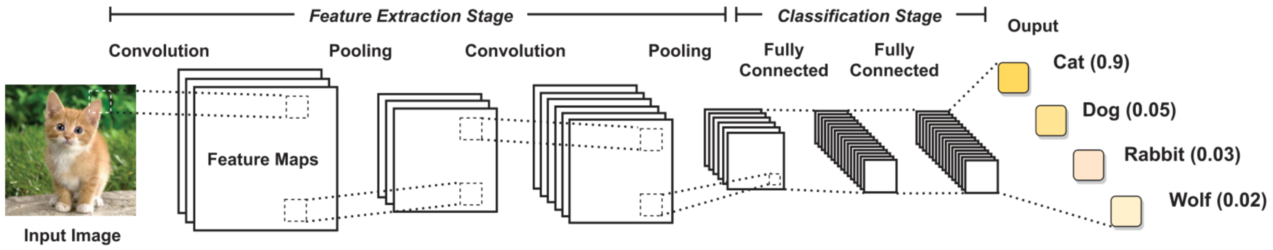


On-Mobile Inference

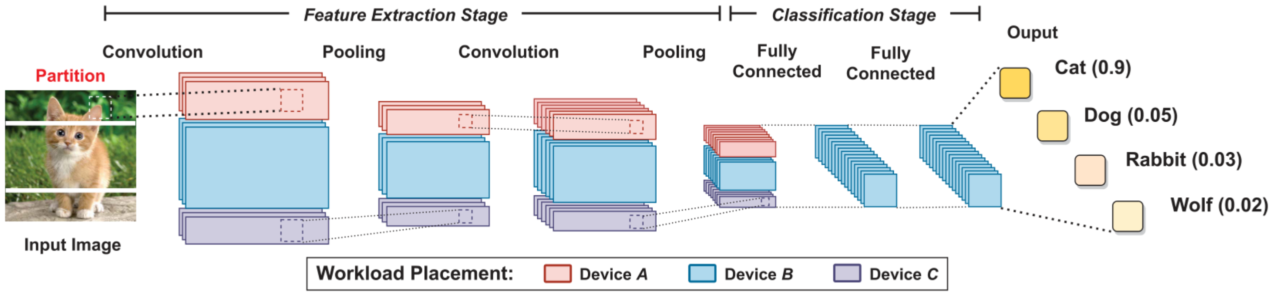
- Collaborative Inference on Mobile Cluster

Coedge, Liekang Zeng et al., TON

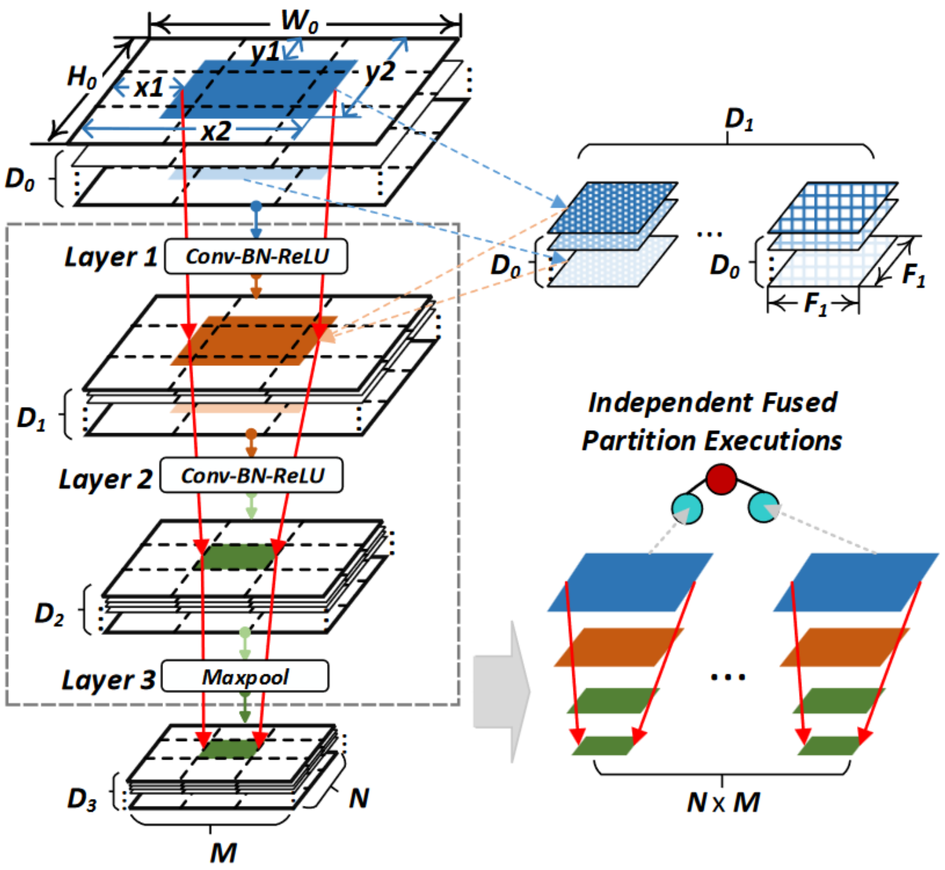
Conventional process



Partitioned Parallelism



DeepThings, Zhuoran Zhao et al., TCAD



On-Mobile Training

- A major bottleneck in on-mobile training is the memory scarcity

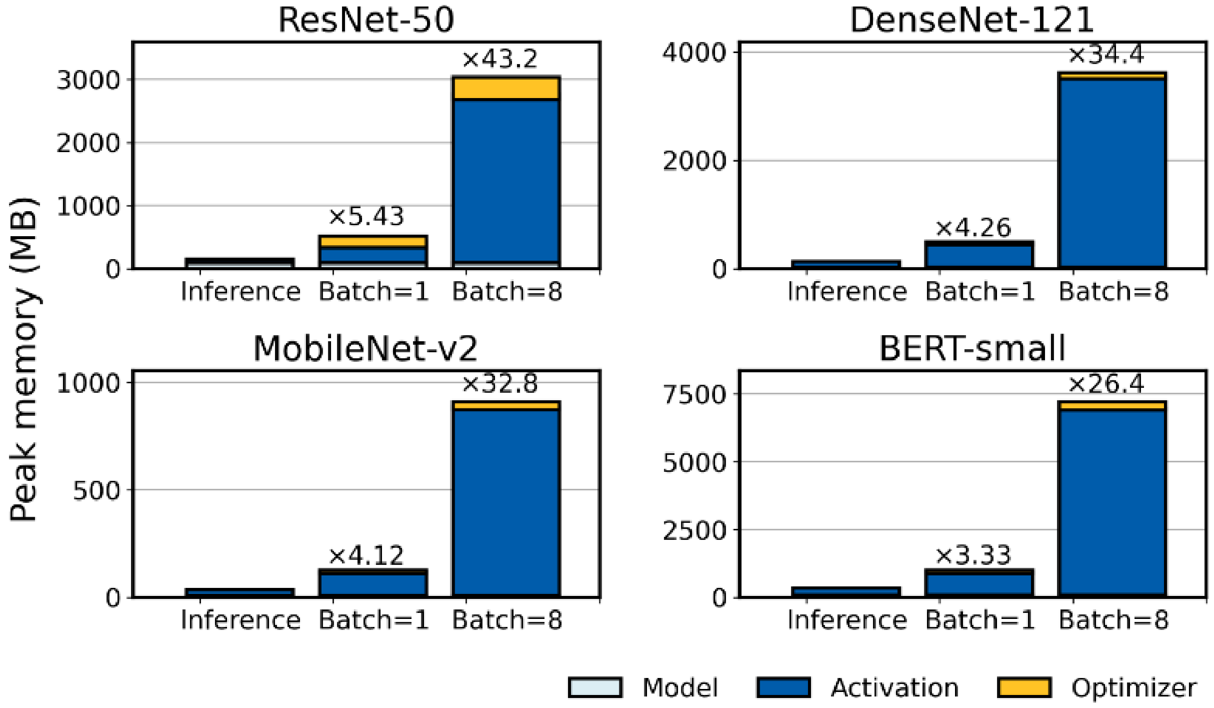


Figure: Memory footprint of DNN inference and training

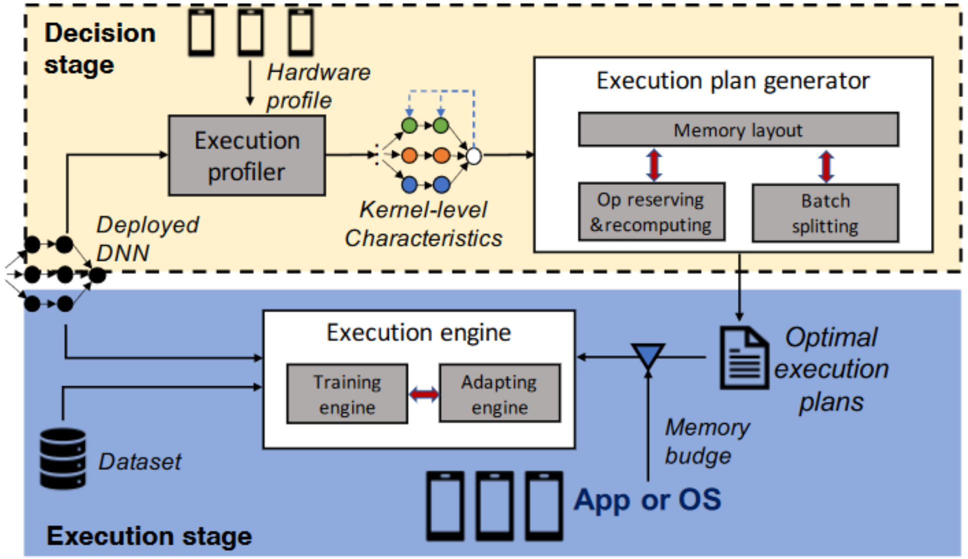
Methods of Memory Optimization:

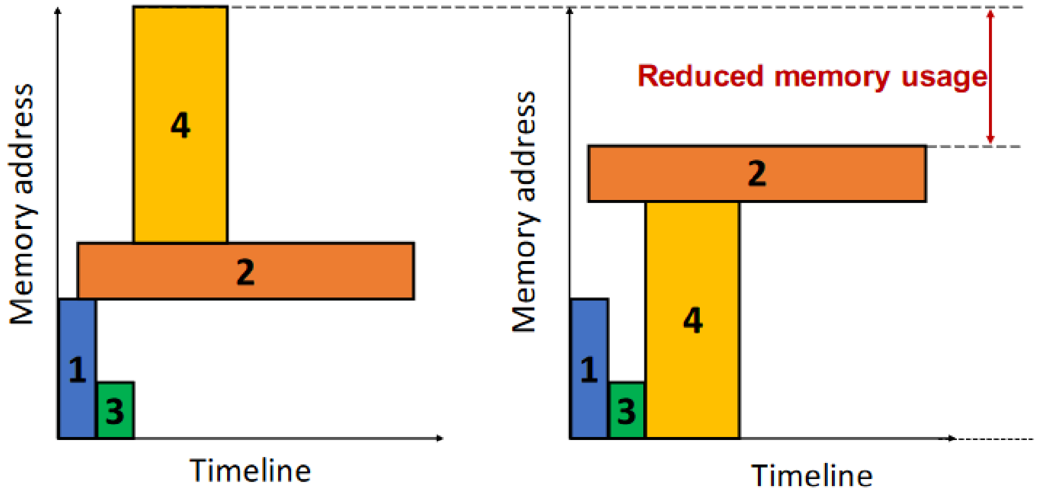
1. Host-device memory swapping.
2. Splitting global mini-batch into micro-batch.
3. Activation recompuation.
4. Model & gradients compression.

Credit: Sage MobiSys22

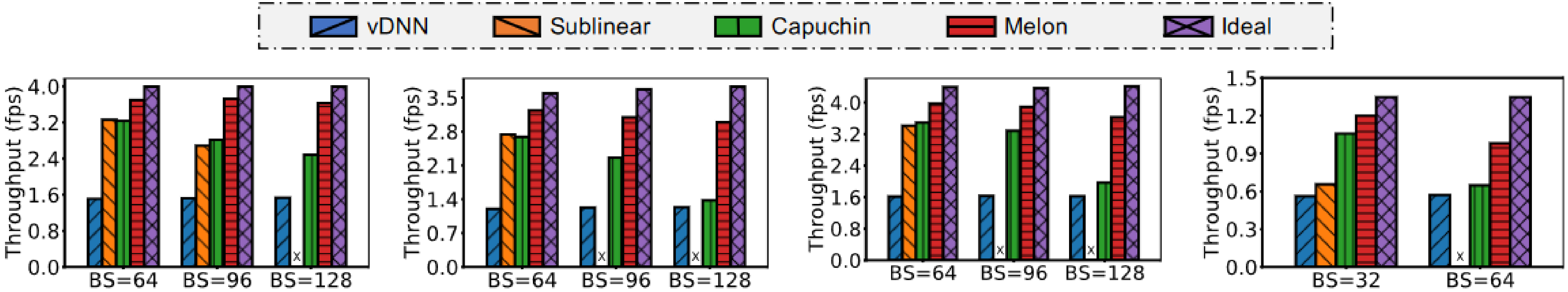
On-Mobile Training

Melon, Qipeng Wang et al., MobiSys22





(a) On-demand memory pool (b) Improved memory pool

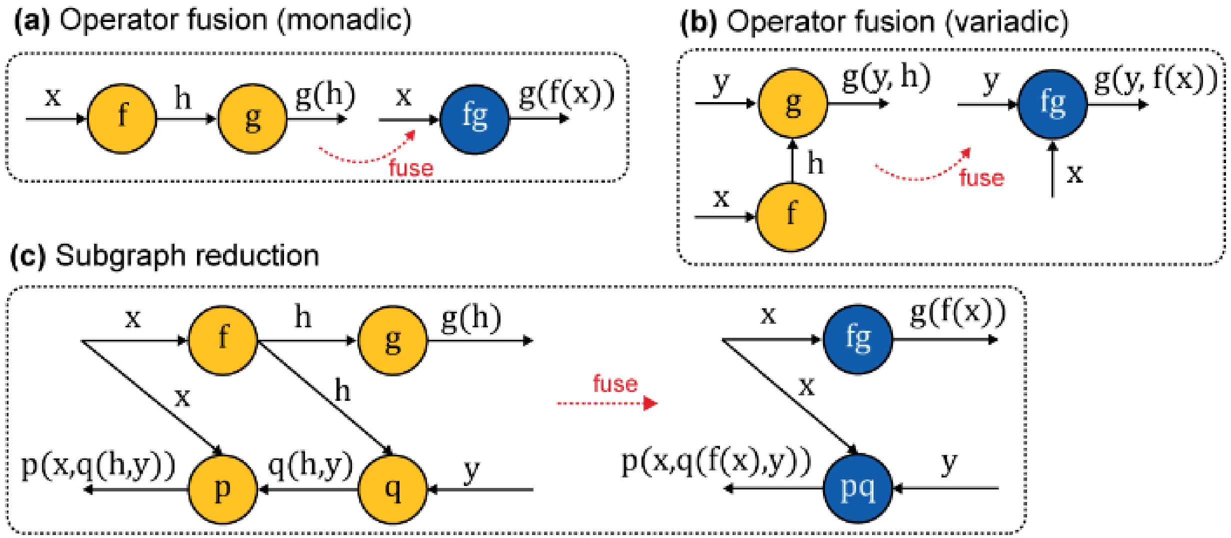


(a) MobileNetV1, SN10 (b) MobileNetV2, SN10 (c) SqueezeNet, SN10 (d) ResNet50, SN10

On-Mobile Training

- Propose a framework support operator fusion and computation graph optimization

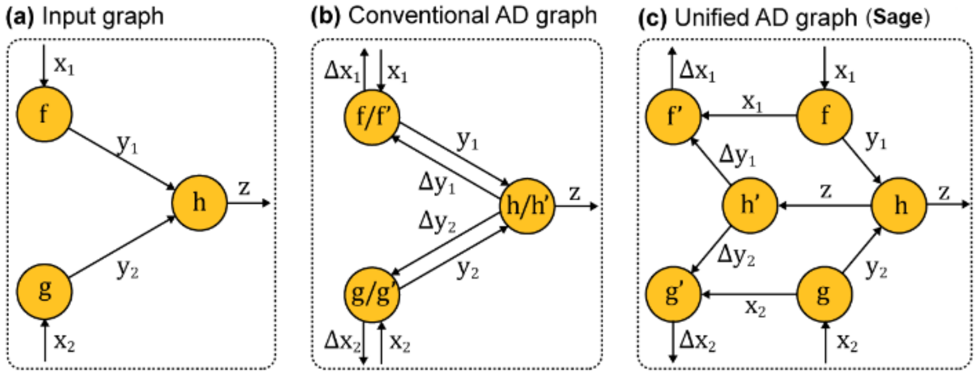
Sage, In Gim et al., MobiSys22



(a) Operator fusion (monadic)

(b) Operator fusion (variadic)

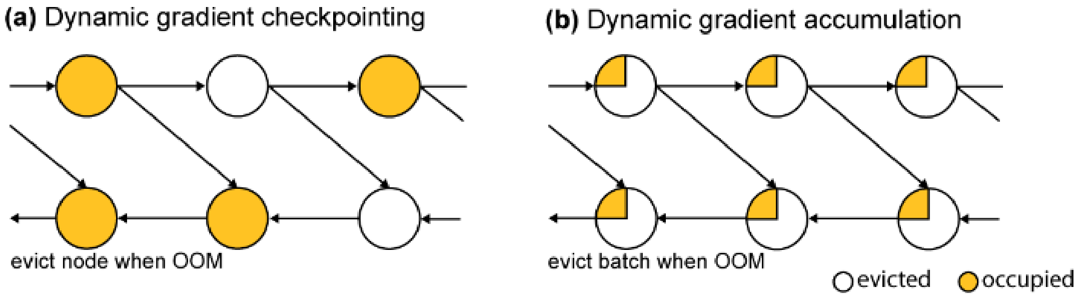
(c) Subgraph reduction



(a) Input graph

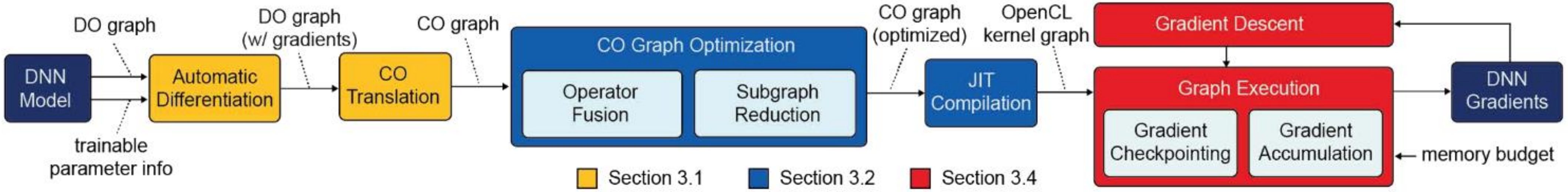
(b) Conventional AD graph

(c) Unified AD graph (Sage)



(a) Dynamic gradient checkpointing

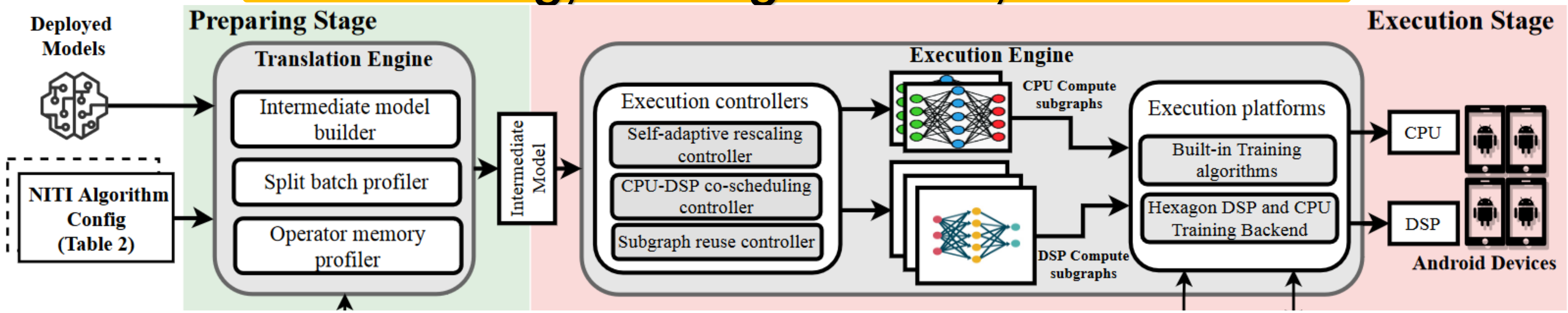
(b) Dynamic gradient accumulation

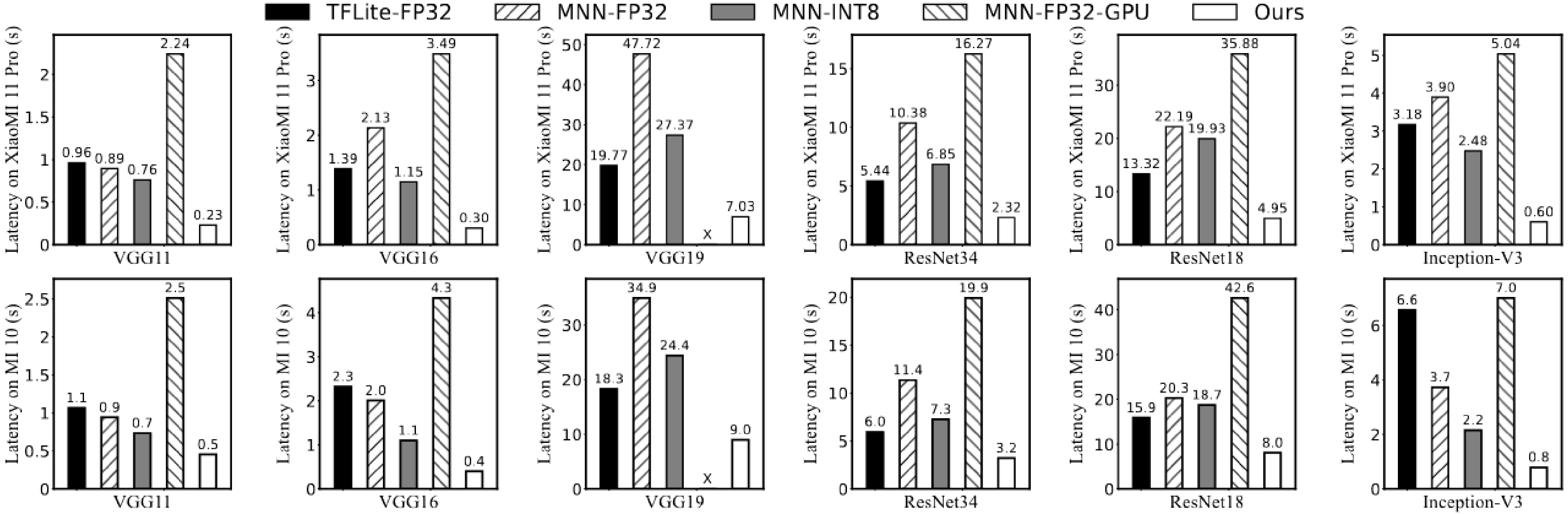


On-Mobile Training

- Training offloading to on-chip Digital Signal Processing(DSP)

Mandheling, Daliang Xu et al., MobiCom22

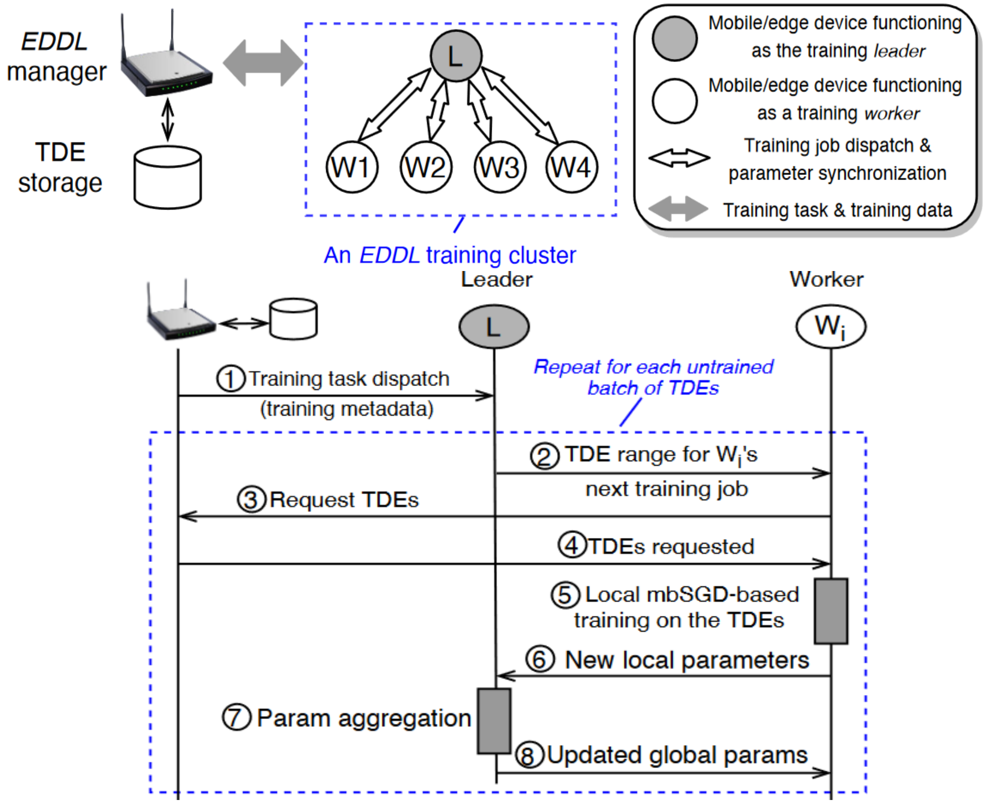




On-Mobile Training

- Collaborative Training on Mobile Cluster

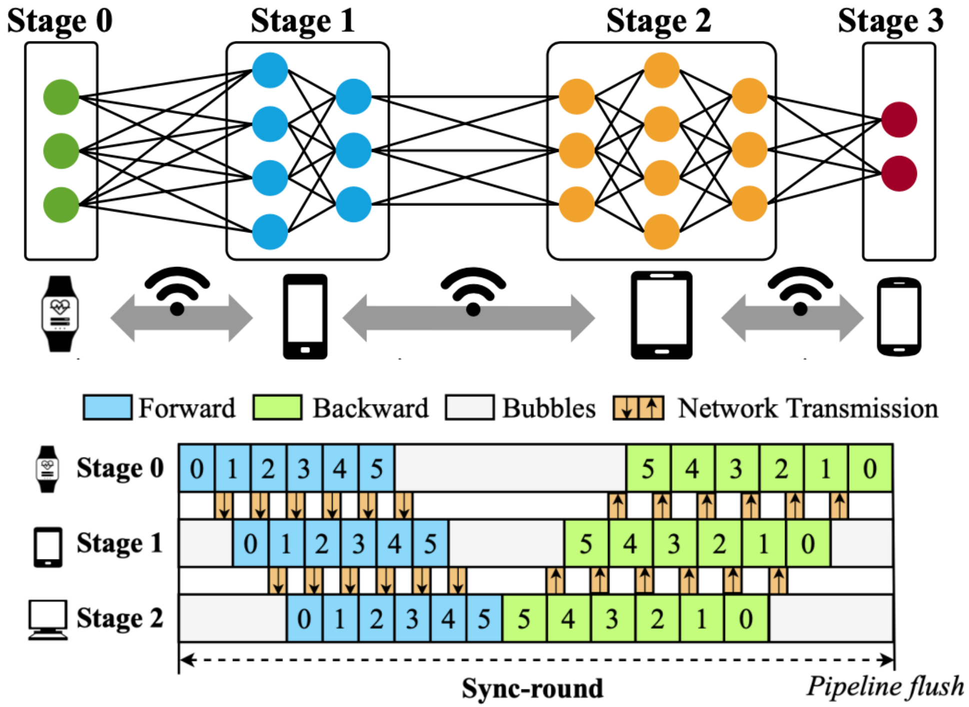
EDDL, Pengzhan Hao et al., SEC21



EDDL manager
TDE storage
L
W1
W2
W3
W4
Mobile/edge device functioning as the training *leader*
Mobile/edge device functioning as a training *worker*
Training job dispatch & parameter synchronization
Training task & training data
An *EDDL* training cluster
Leader
Worker
L
W_i
① Training task dispatch (training metadata)
Repeat for each untrained batch of TDEs
② TDE range for W_i's next training job
③ Request TDEs
④ TDEs requested
⑤ Local mbSGD-based training on the TDEs
⑥ New local parameters
⑦ Param aggregation
⑧ Updated global params

Data Parallelism

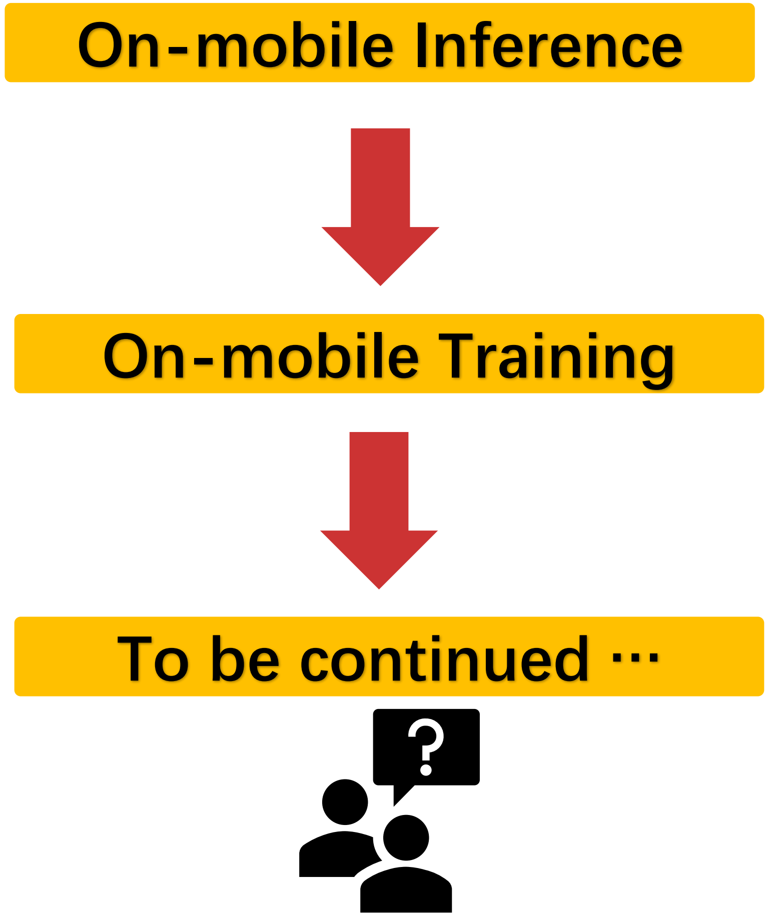
Eco-FL, Shengyuan Ye et al., ICPP22



Pipeline Parallelism

Conclusion & Discussion

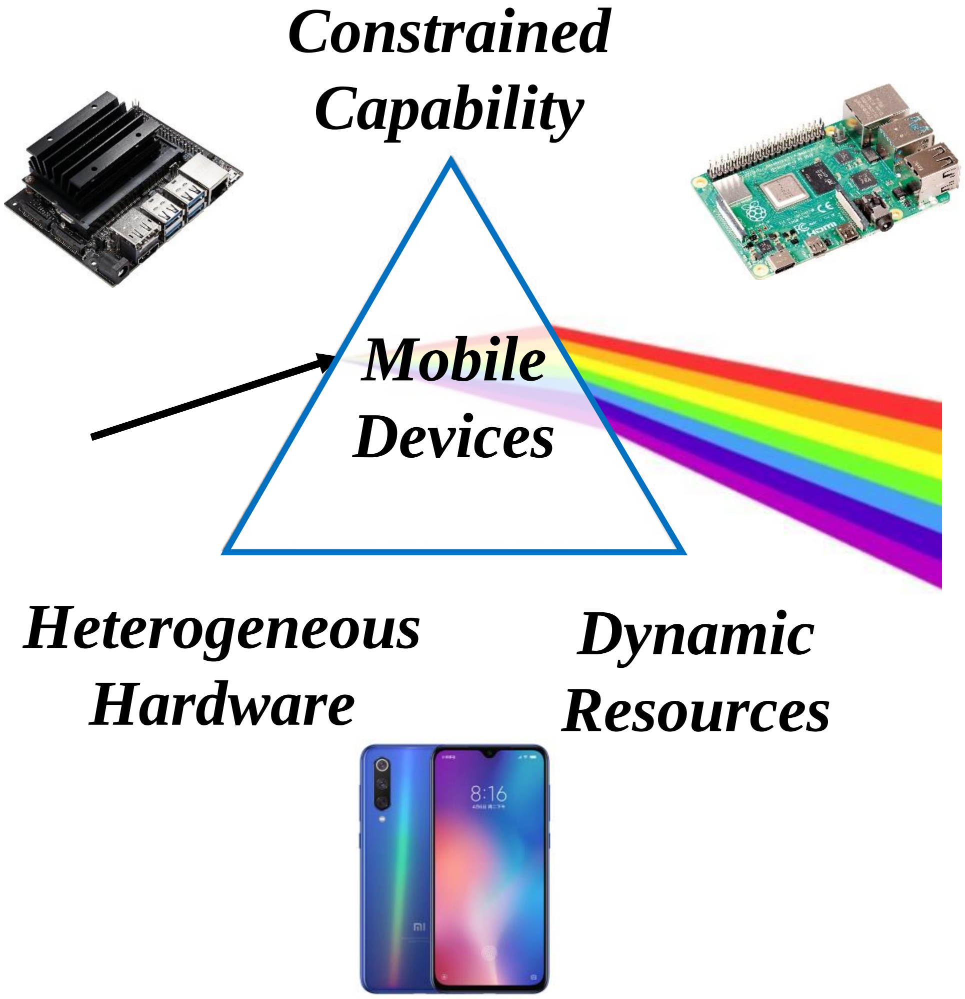
- The **research trend** of DLSys on mobile



On-mobile Inference

On-mobile Training

To be continued …



Constrained Capability

Mobile Devices

Heterogeneous Hardware

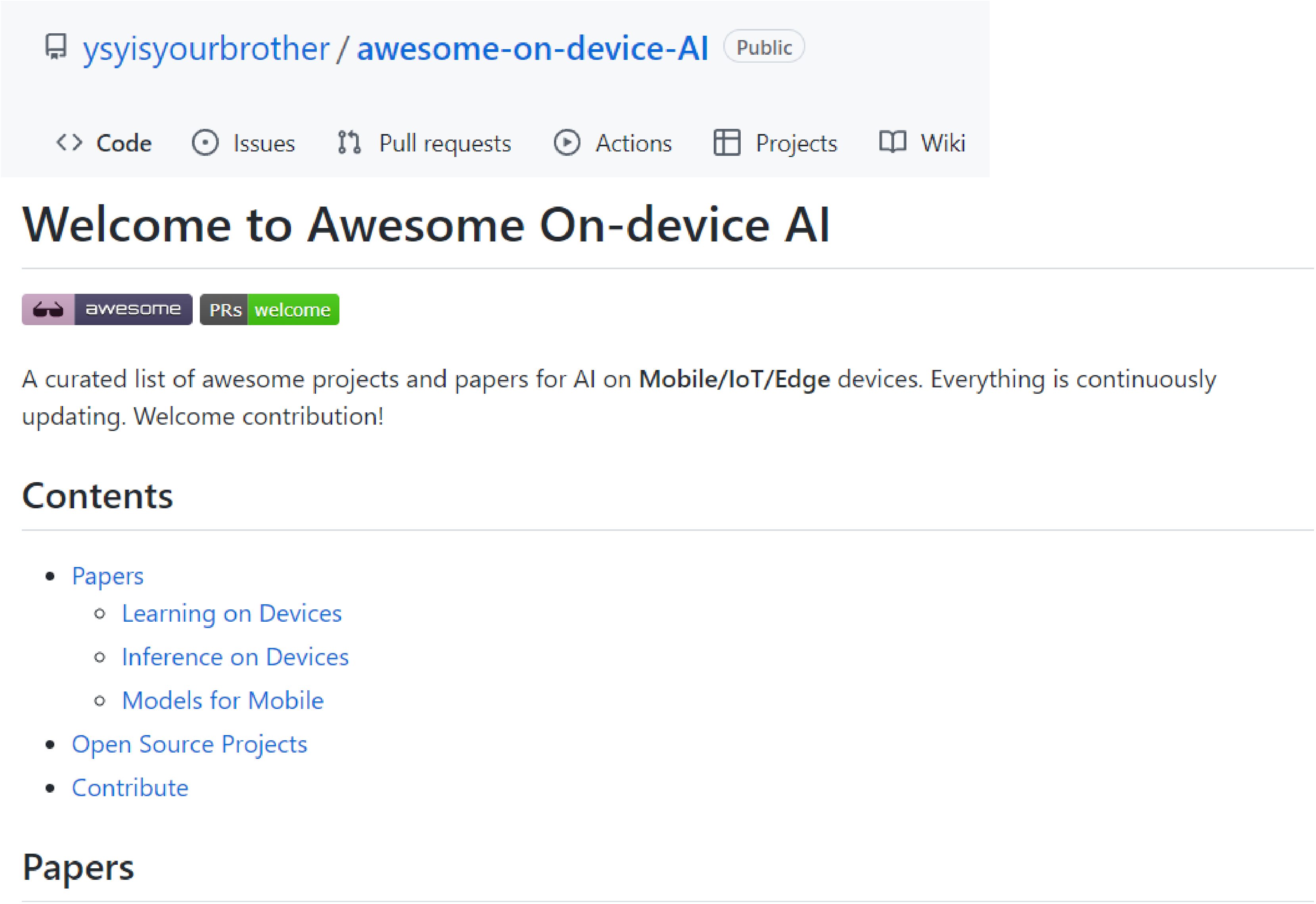
Dynamic Resources

Thanks!

Shengyuan Ye

School of Computer Science and Engineering
Sun Yat-sen University
Contact: yeshy8@mail2.sysu.edu.cn

Link: https://github.com/ysyisyourbrother/awesome-on-device-AI



Welcome to Awesome On-device AI

A curated list of awesome projects and papers for AI on **Mobile/IoT/Edge** devices. Everything is continuously updating. Welcome contribution!

Contents

- Papers
  - Learning on Devices
  - Inference on Devices
  - Models for Mobile
- Open Source Projects
- Contribute

Papers