Imperial College London

Optimizing Real-Time Data Processing: Balancing Resource Constraints and Quality-of-Service

Giuliano Casale

Quality of Service Research Lab (QORE)
Department of Computing - Imperial College London

BDA Conference 2024, Hyderabad, 18-Dec-2024

Real-time data processing

Deep neural networks (DNNs) increasingly used to deliver instant insights

AI traffic monitoring

Industrial IoT

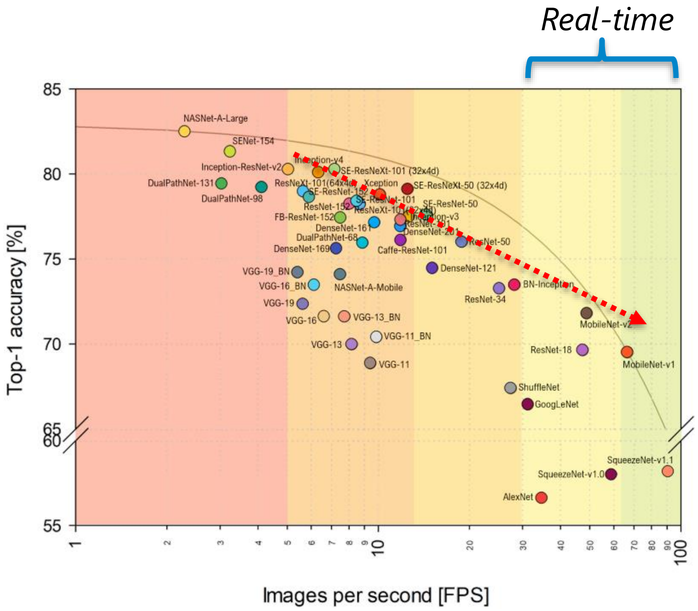
Augmented reality for hospitality

Real-time sport analytics

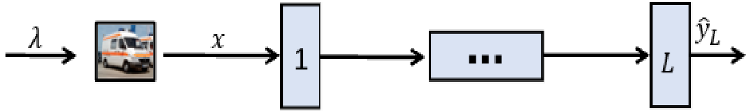
AI-enabled financial trading

Quality-of-Service (QoS) in DNN-based Data Processing

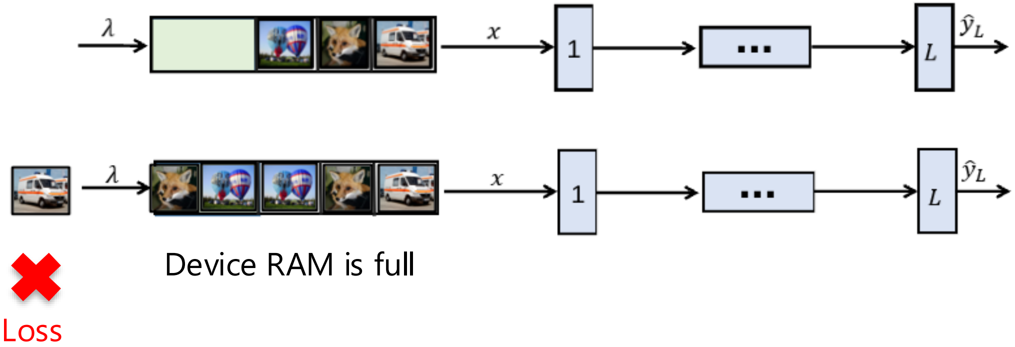
- **QoS tradeoffs**: Performance-Accuracy-Reliability



Periodic inference: DNN latency < time to next arrival



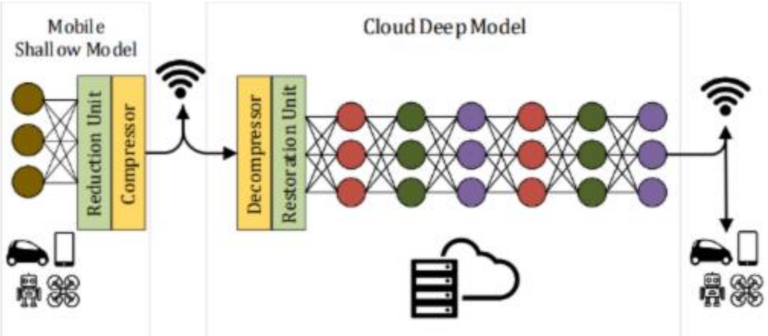
Event-driven inference: buffer losses!



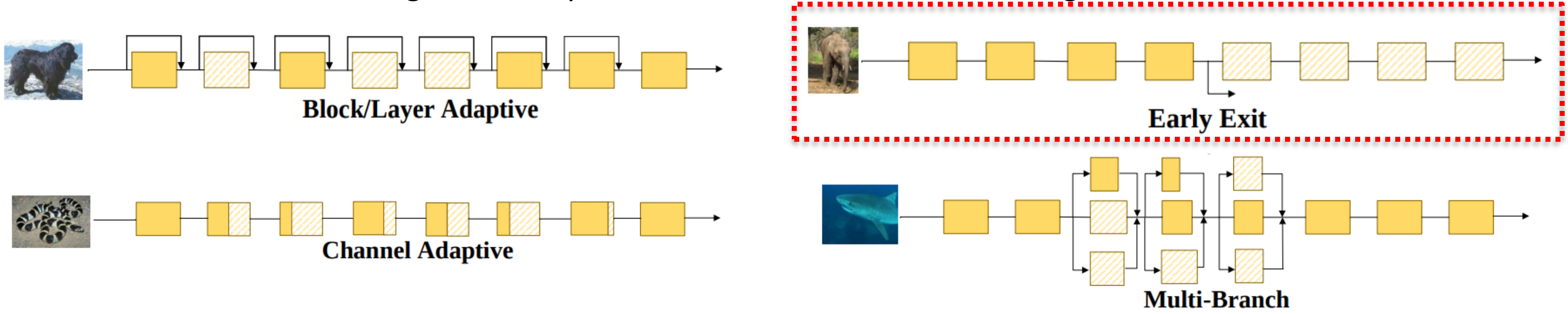
S. Bianco, et al. Benchmark Analysis of Representative Deep Neural Network Architectures, IEEE Access

Model tuning and adaptive architectures

- DNN model tuning
  - Weight pruning
  - Quantization
  - Knowledge distillation
  - Lossy compression
  - Neural Architecture Search (NAS)



- Adaptive DNN models
  - How shall we leverage these capabilities for DNN inference serving?



D Liu, H Kong, X Luo, W Liu, R Subramaniam. Bringing AI To Edge: From Deep Learning's Perspective. Neurocomputing.
Eshratifar, A. E., et al. BottleNet: A Deep Learning Architecture for Intelligent Mobile Cloud Computing Services, IEEE/ACM ISLPED.

Roadmap

1. Controlling QoS tradeoffs in DNN-based data processing — **TC'23**
   - ⇨ Early exits and their scheduling policies

2. Extension: QoS tradeoffs in collaborative DNN inference
   - ⇨ Predicting QoS in distributed DNN deployments — **DSN'24**
   - ⇨ Scheduling early exits in distributed DNN deployments — **INFOCOM'25**

Early exit DNN job scheduling

TC'23

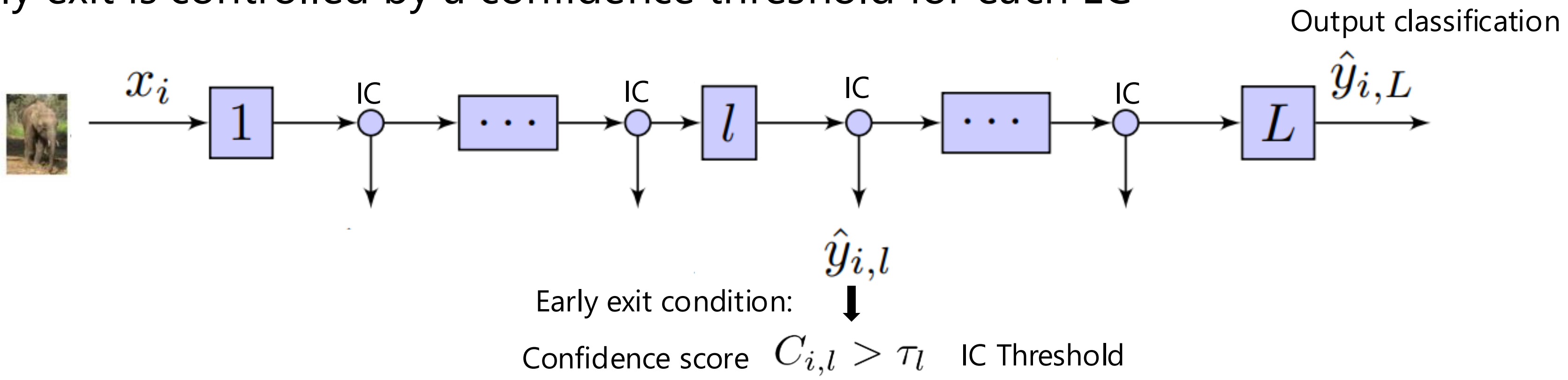
Joint work with:

Manuel Roveri
(Politecnico di
Milano, Italy)

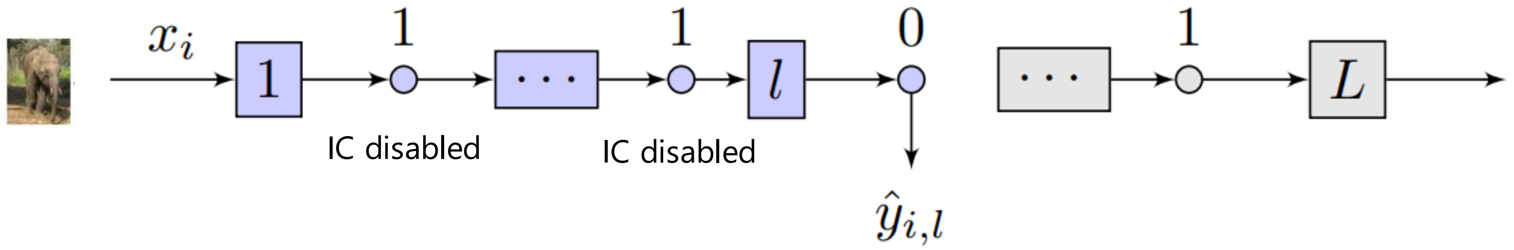
Yichong Chen
(Imperial College
London, UK)

Early exit in CNNs

- Intermediate Classifiers (IC) produce an early classification avoiding "overthinking"
- Early exit is controlled by a confidence threshold for each EC



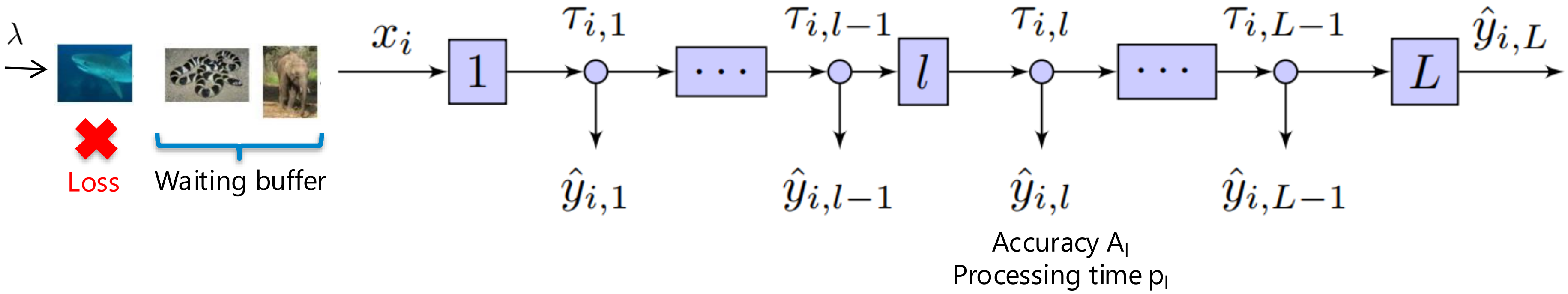
- Example – forcing exit at layer *l*:



- IC thresholds trained with the CNN or decided post-training.

Scheduling early exits for QoS

- How to schedule early exit online to control data loss?



- **Early exit scheduling problem**: choose IC thresholds for each incoming job *i*

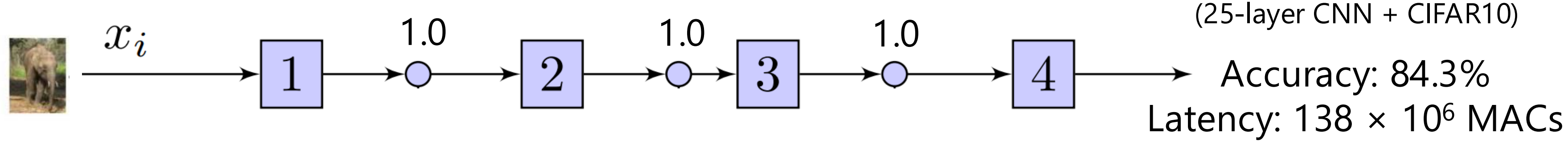
- QoS metrics: latency, accuracy, loss ratio (i.e., fraction of lost jobs).

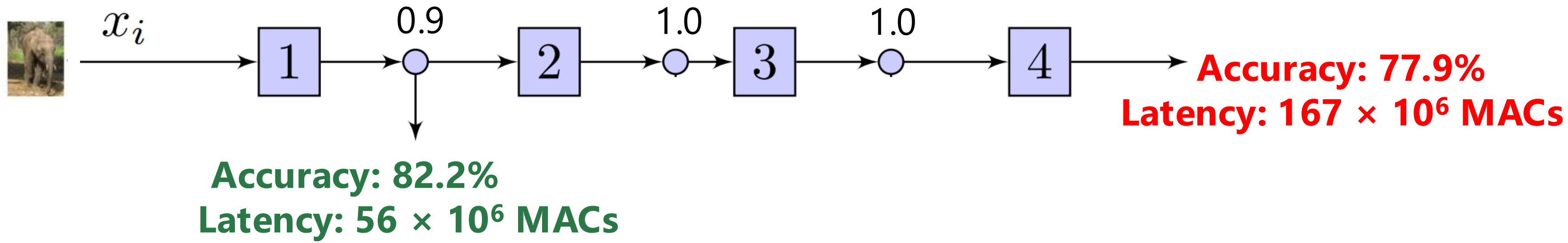
- **Issue**: difficult to predict accuracy and processing time for arbitrary threshold combinations

Casale, G., & Roveri, M. (2023). Scheduling Inputs in Early Exit Neural Networks. *IEEE Transactions on Computers*.

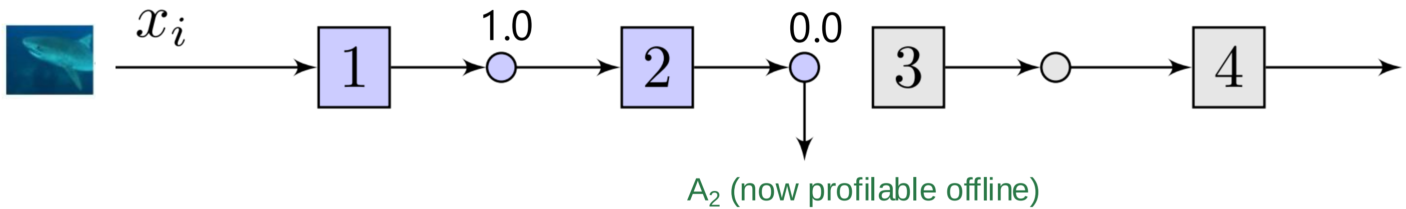
Accuracy in adaptive DNNs

- Accuracy and latency change with the data distribution!





- **Single-exit schedulers**: restrict feasible threshold values to {0,1}



Single-exit scheduling

- Knapsack-based policy:
  - Similar to discrete scheduling with compressible resources (NP-hard)

k jobs in buffer

Time budget B

On-device lookup table

λ/k	*0*	*1*	*2*	...
0-0.5	$\tau_{i,l}$	...	...	...
0.5-1	...	...	...	...
⋮	⋮	⋮	⋮	⋮

knapsack problem

$$\max \sum_{l \neq l_{min}} \sum_{j=1}^{k} A_l x_l$$

Maximize accuracy

$$\text{s.t.} \sum_{l \neq l_{min}}^{L} \sum_{j=1}^{k} p_l x_l \leq B - k p_{l_{min}}$$

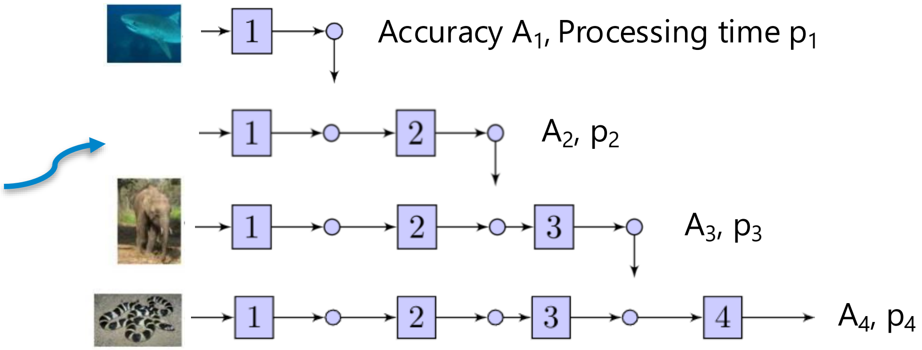
Fit time budget

$$\sum_{l \neq l_{min}}^{L} x_l \leq k$$

Schedule at most k jobs

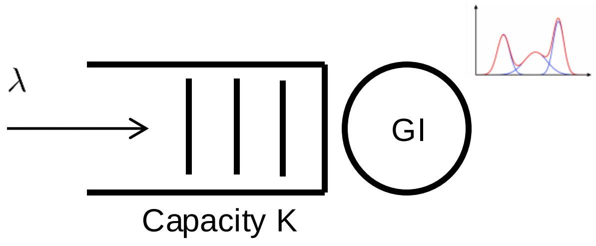
$$x_l \in \{0, \ldots, k\} \quad \forall l$$

Num. jobs to exit at layer l



Single-exit scheduling

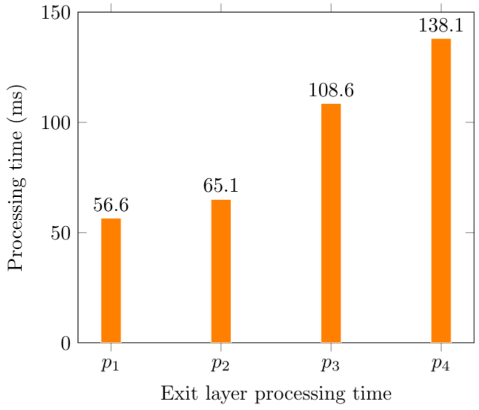
- Queueing model based policy:
  - DNN latency from steady-state M/GI/1/K queue
  - Service seen as a mixture distribution (GI) based on exit layer probabilities

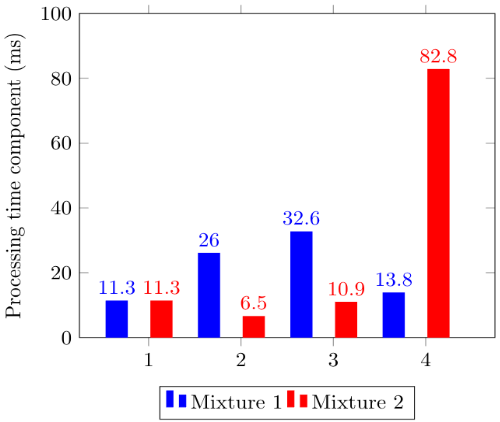


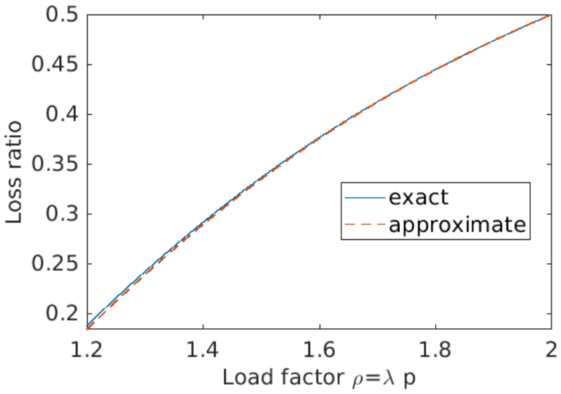
- Optimal schedule obtained via a Linear Program (LP)
  - Maximize accuracy
  - Constraint maximum loss ratio

Loss ratio approximation

$$L = \frac{\rho^{(\sqrt{\rho}s^2-\sqrt{\rho}+2K)/(2+\sqrt{\rho}s^2-\sqrt{\rho})}(\rho-1)}{\rho^{2(1+\sqrt{\rho}s^2-\sqrt{\rho}+K)/(2+\sqrt{\rho}s^2-\sqrt{\rho})}-1}$$

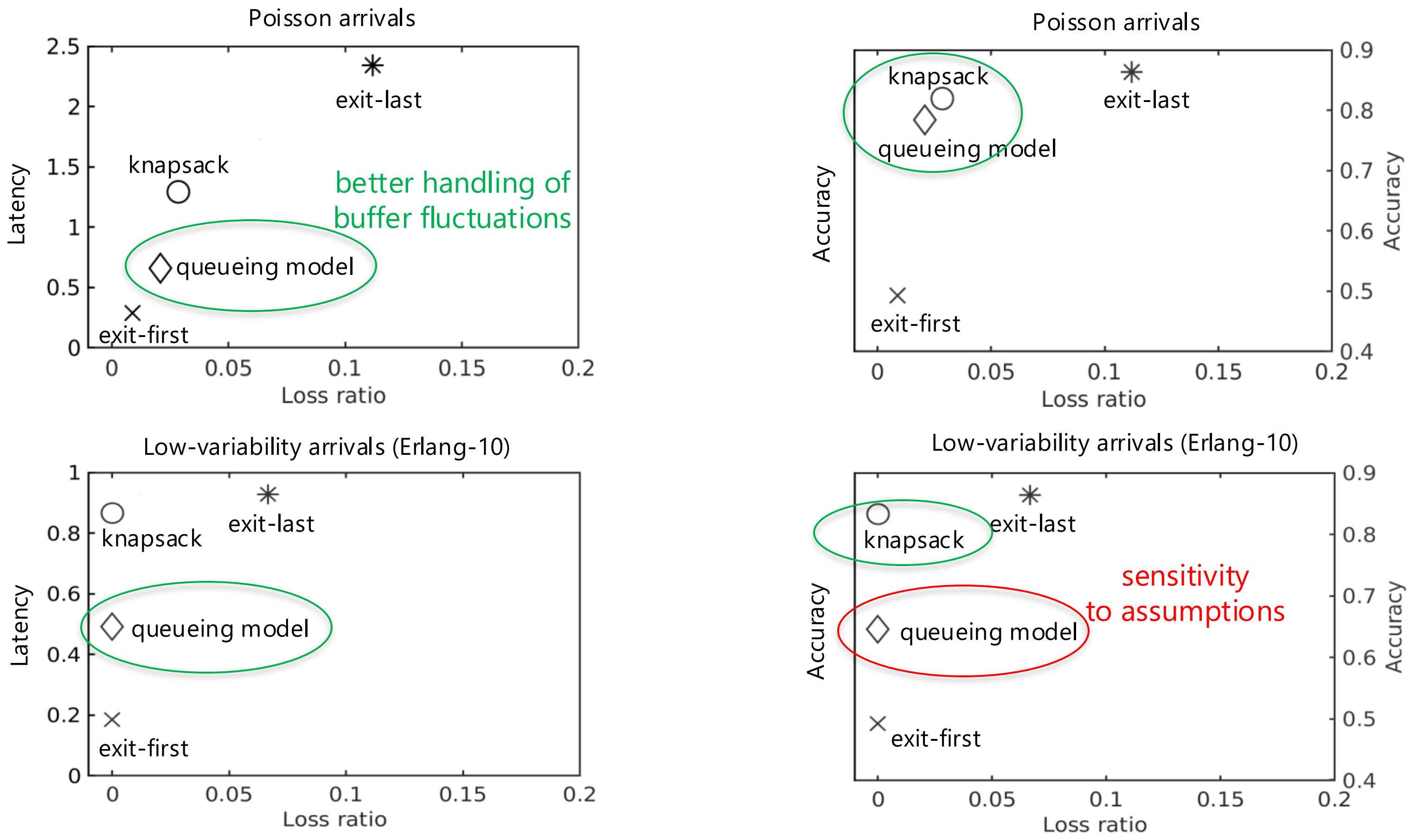






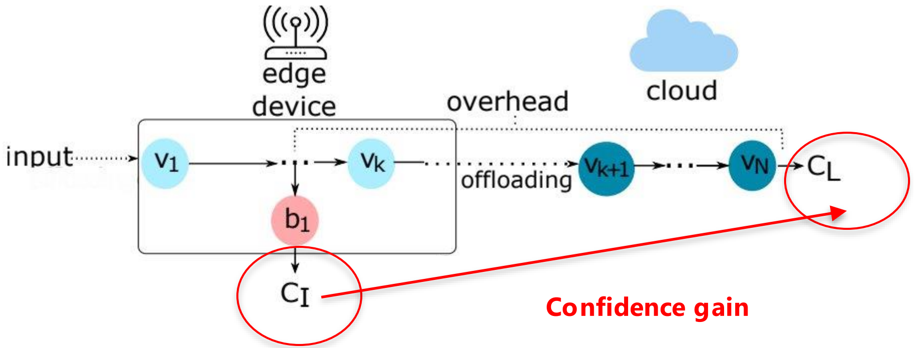
Simulation of real technological scenarios

- 6 CNNs (28-56 processing layers; 8-24 exit points; CIFAR10/100 data)



Extension: dealing with out-of-distribution data

- How to generalize the approach when offline profiling is not viable?
- AdaEE: multi-armed bandit (MAB) to schedule early exits
  - Reward metric: **Confidence gain** - Performance overhead

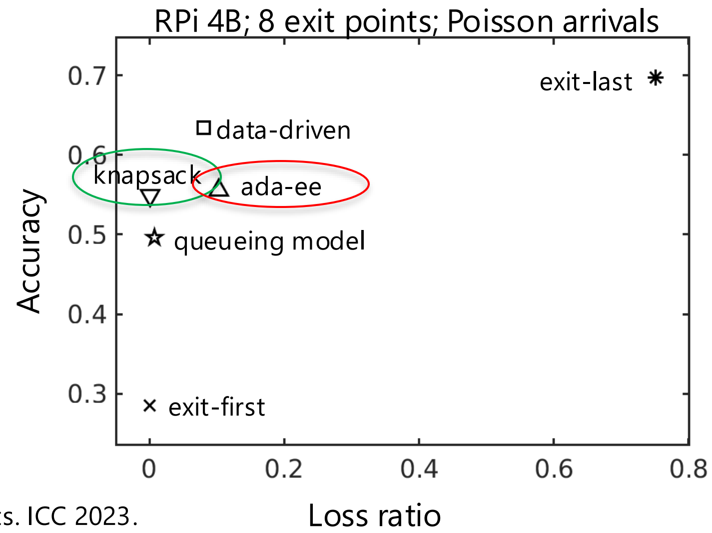


Upper Confidence Bound (UCB):

$$\tau_{i,l} \leftarrow \arg\max_{\tau_{i,l} \in \mathcal{A}} \left(Q_{i-1} + c\sqrt{\frac{\ln(i)}{N_{l-1}}} \right)$$

Exploitation   Exploration

- Driving early-exits with confidence gains
  - Single-exit: <1s to update the policies
  - Data-driven: Bayes optimization based



R. G. Pacheco *et al.* AdaEE: Adaptive Early-Exit DNN Inference Through Multi-Armed Bandits. ICC 2023.

Key takeaways

1. Early-exit ICs a new control knob to dynamically tune QoS trade-offs
2. Knapsack based policy are highly robust
3. Queueing based policy highly effective to reduce latency (but under assumptions)
4. Confidence gain can help dealing with out-of-distribution data

Distributed early-exit optimization

DSN'24

INFOCOM'25

Joint works with:

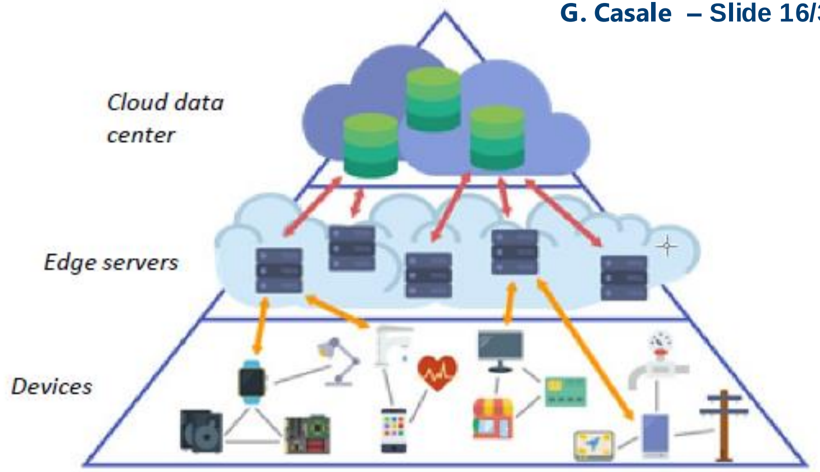
Yichong Chen
(Imperial College London, UK)

Zifeng Niu
(Imperial College London, UK)

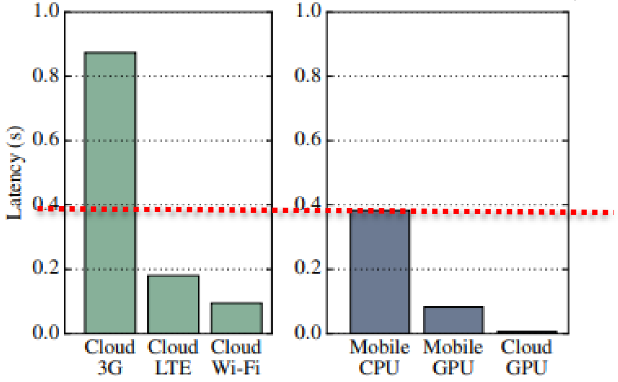
Manuel Roveri
(Politecnico di MIlano, Italy)

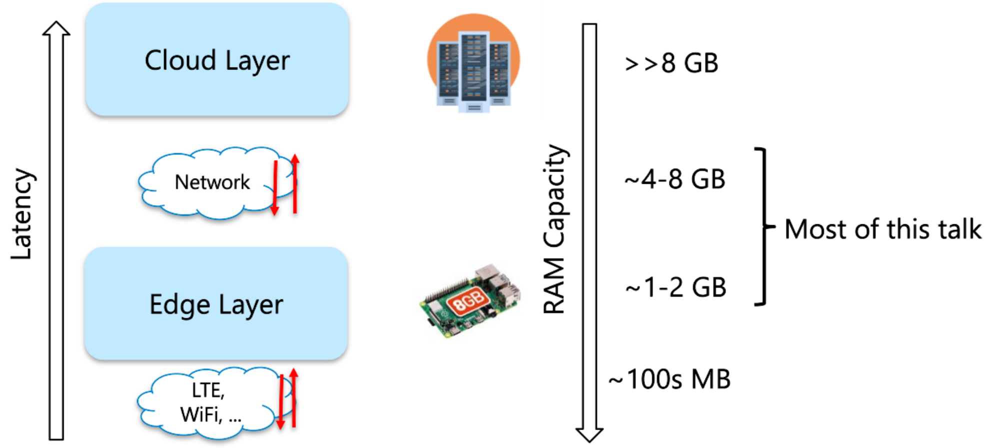
DNNs & Resource Constraints

- Many DNN deployment models
  - Fog, MEC, 3G/4G/5G/Wi-Fi/…, private vs public, Cloud-to-Edge, …
- Common challenges and themes:
  - Processing data closer to where it is generated
  - QoS vs. resource constraints



Data transfer latency vs. Local processing

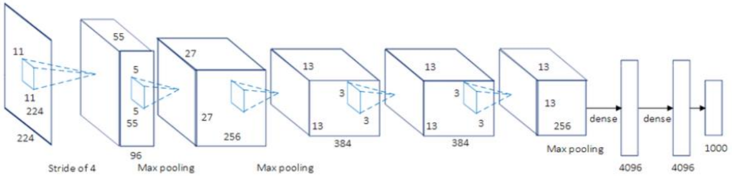


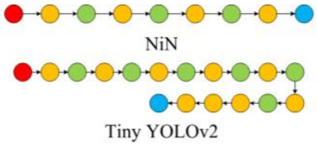


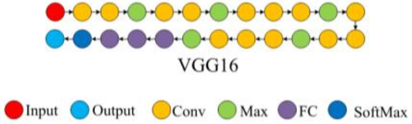
Y. Kang *et al*. Neurosurgeon: Collaborative intelligence between the cloud and mobile edge, Proceedings of ASPLOS.

State-of-the-art: DNN layer-wise partitioning

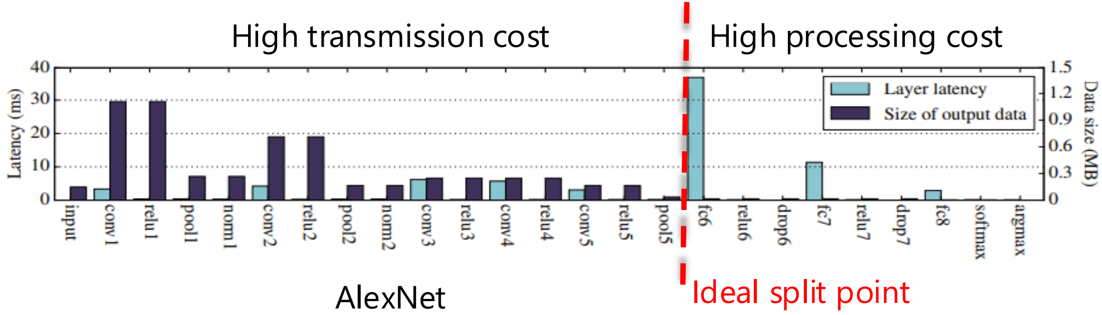
- Many popular DNNs have a linear topology (chain)







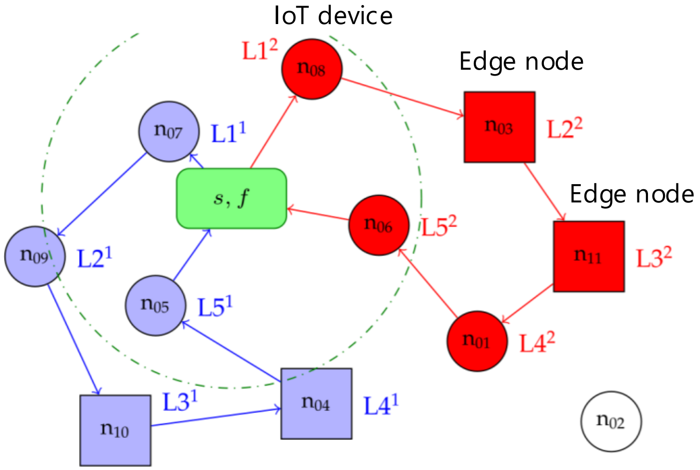
- Ideal split point determined from layer characteristics
  - Convolutional: large output data, Pooling: smaller output data; FC layers: high latency
  - Prediction on processing time on target hardware obtained via regression



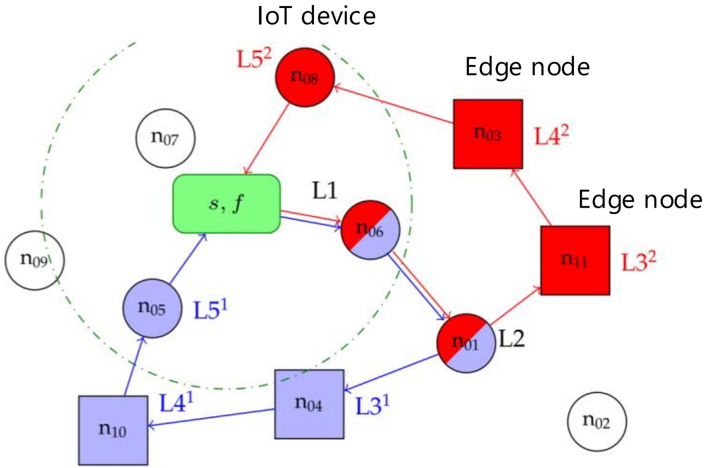
Y. Kang *et al.* Neurosurgeon: Collaborative intelligence between the cloud and mobile edge, Proceedings of ASPLOS.
H. Liang *et al.* DNN Surgery: Accelerating DNN Inference on the Edge Through Layer Partitioning, IEEE TCC 2023.

Designing a DNN-based data processing system

- DNN placement is critical, e.g. IoT devices without on-air update
- State-of-the-art mainly relies on integer-linear programming (ILP)
  - Binary variables map layers to edge & IoT nodes
  - Constraints on memory, processing time, DNN layer dependencies, network range, …



No shared DNN layers

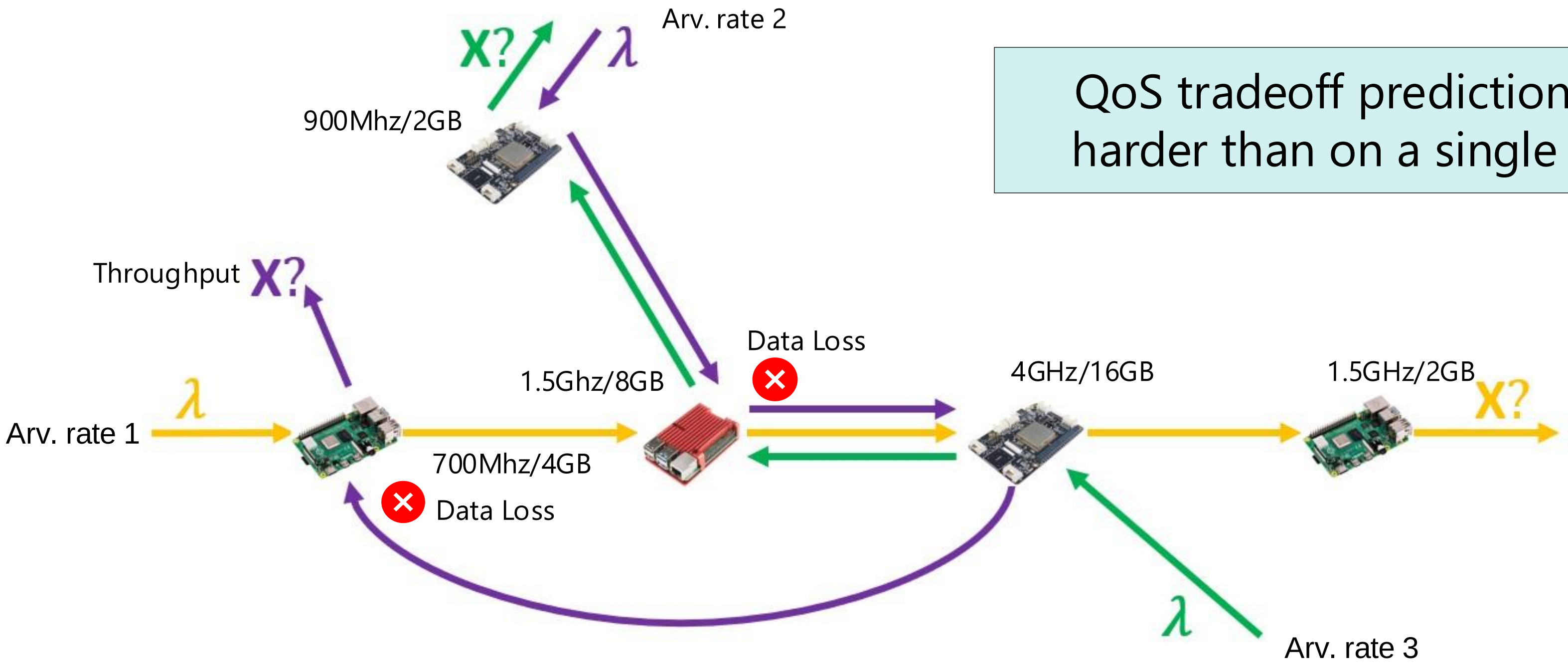


Shared DNN layers

S. Disabato, M. Roveri, C. Alippi. Distributed Deep Convolutional Neural Networks for the Internet of Things. IEEE TC, 2021.

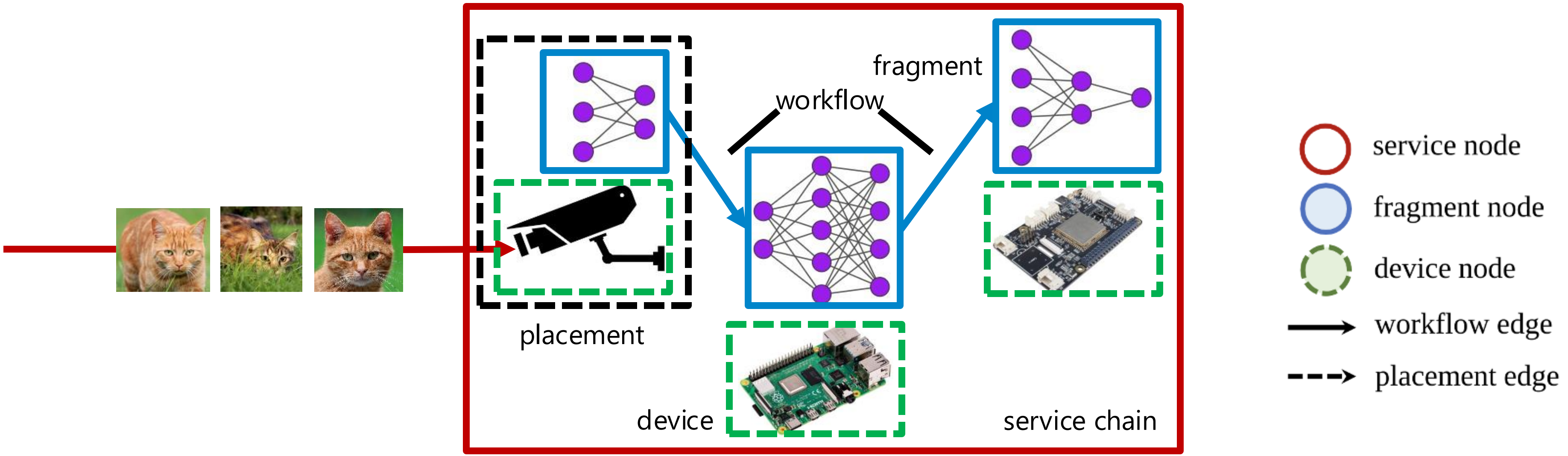
Modelling data loss ratio

- ILP models are appropriate for periodic workloads
- The same approach cannot easily capture stochastic arrivals

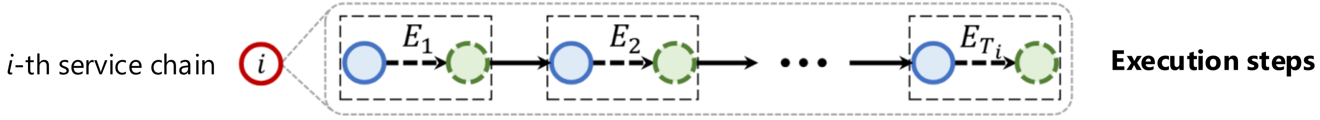


QoS tradeoff prediction much harder than on a single device

DNN-based collaborative inference system as a graph

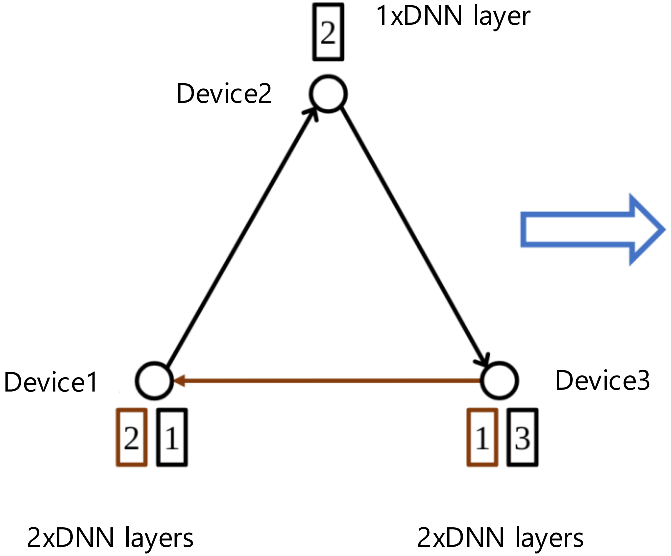


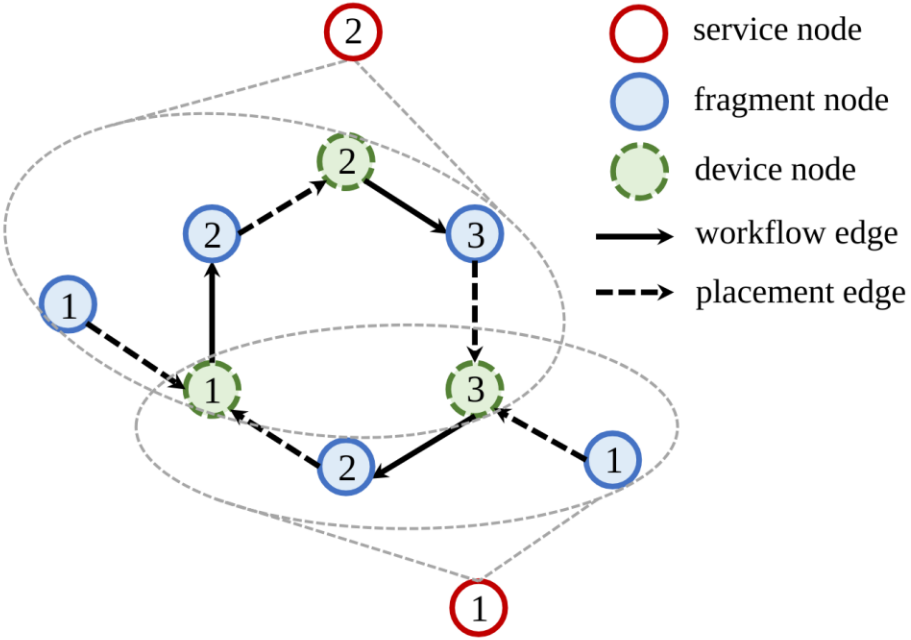
- We focus on linear DNN pipelines (referred to as a service chain)



Example: a shared deployment with two DNN-based services

- DNN deployment described by heterogeneous graph

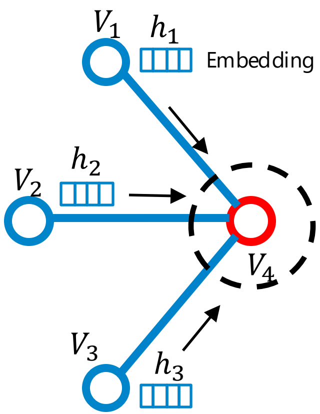




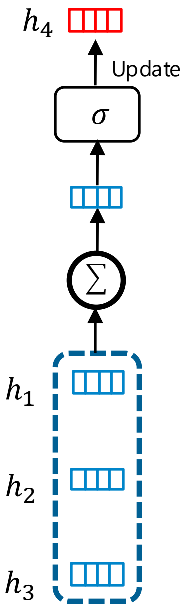
Graph neural networks for QoS trade-off prediction

- GNN surrogates can address the problem
  - Input features: system and workload parameters: arrival rates, RAM size, CPU GHz, …
  - Output features performance metrics: throughputs, latencies, loss ratio, …

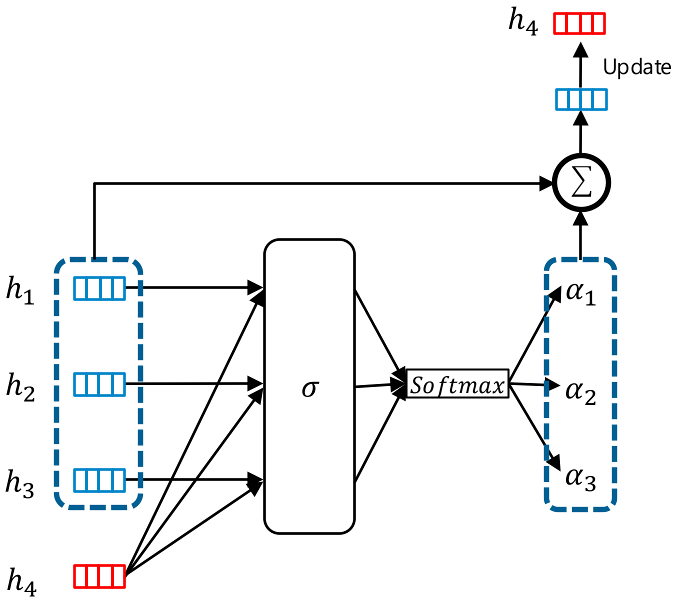
Message passing GNN



Graph Isomorphism Network (GIN)

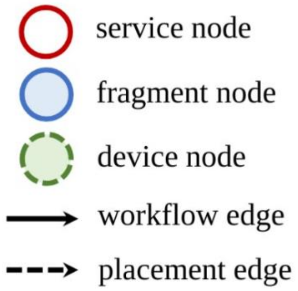


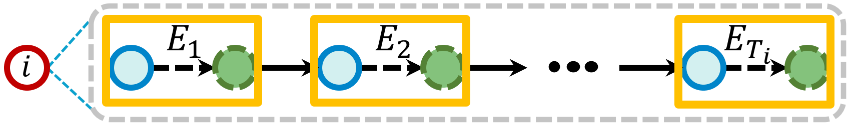
Graph Attention Network (GAT)

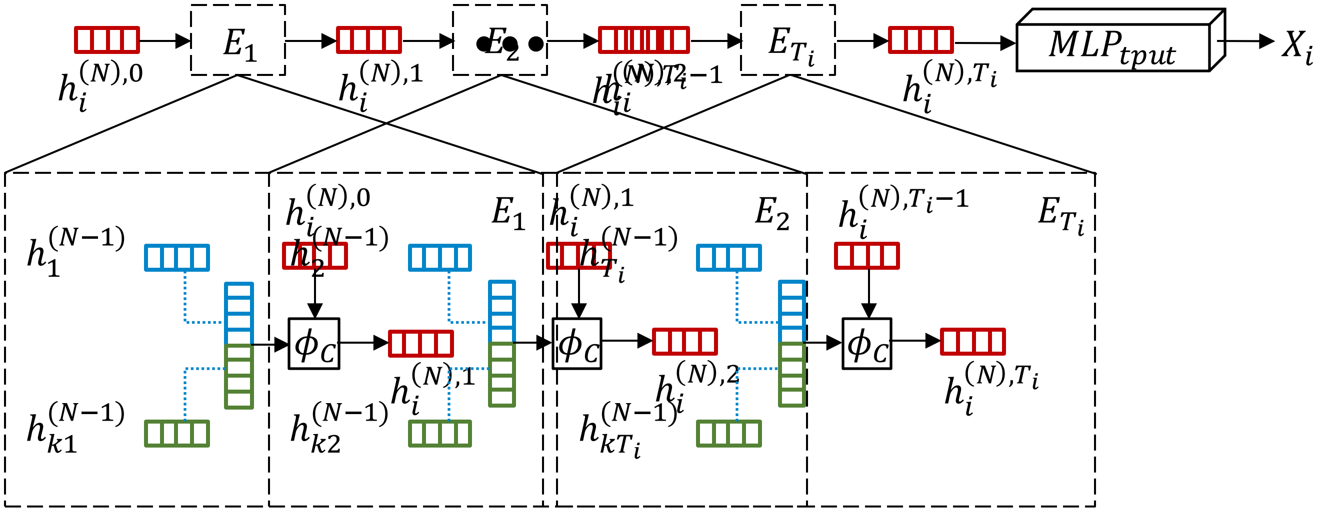


ChainNet GNN: predicting QoS in collaborative inference

- GNN surrogate trained on simulation and/or system data
  - Input features: arrival rates, RAM size, CPU GHz, …
  - Output features performance metrics: throughputs, latencies, loss ratio, …
- Modelling throughput in ChainNet

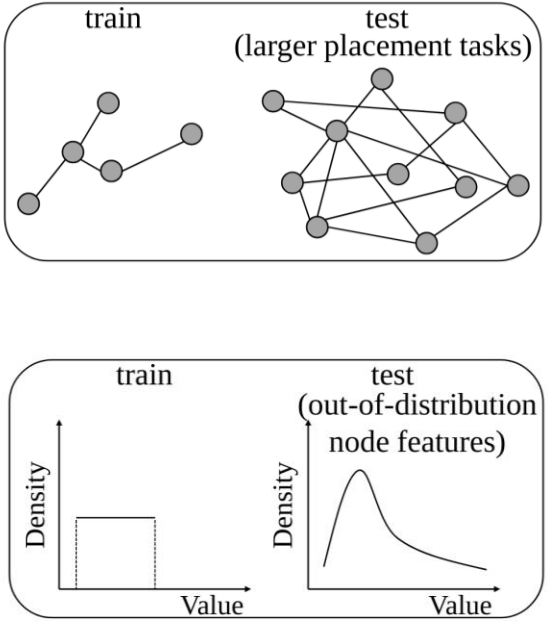




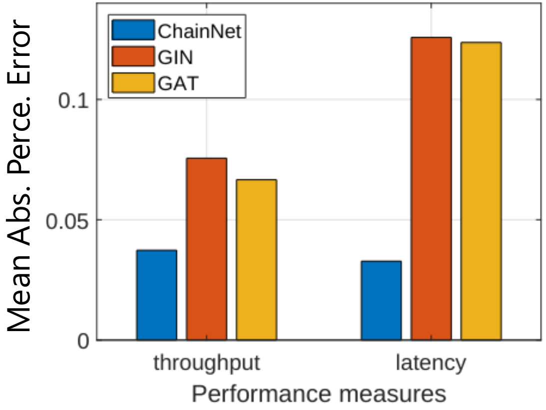


ChainNet: results

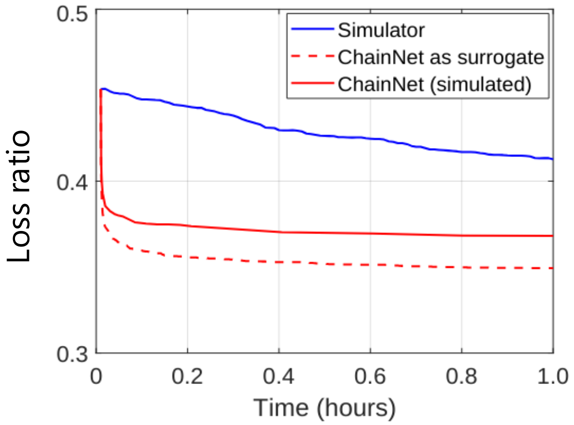
- 71% loss ratio reduction in real-world technological scenario
  - 2×OrangePi Zero, 2×Raspberry Pi A+, and 1×Raspberry Pi 3A+
- Systematic reduction also visible in generalization tests via simulation



50000 training models
10000 test models



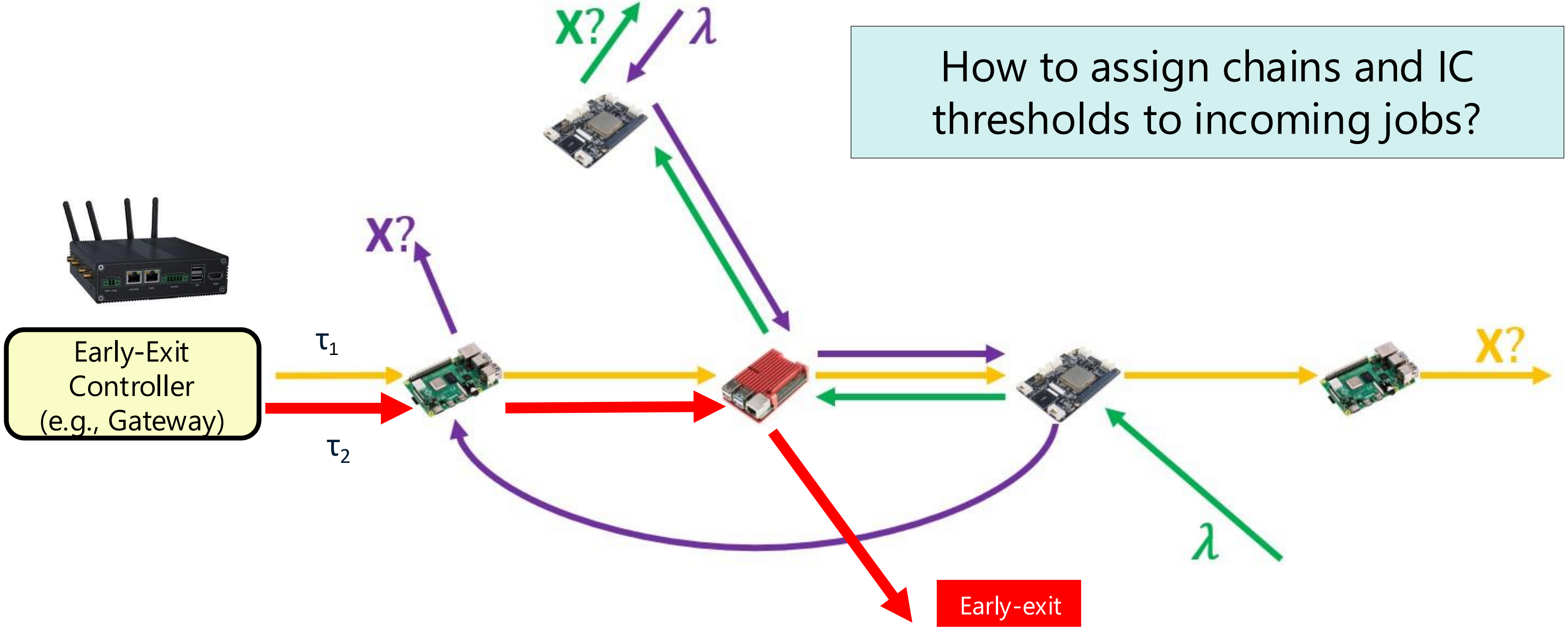
Optimization-based search with GNNs



Extending ChainNet for early-exit scheduling

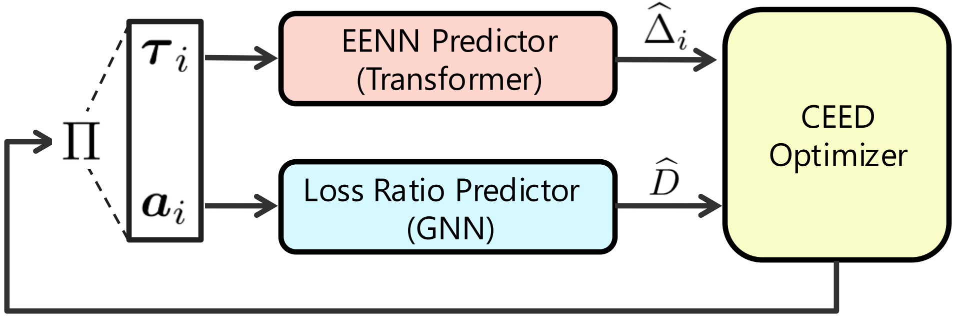
- How can we generalize early-exit scheduling to the distributed setting?
- Jobs assigned upon arrival to a given path (chain) and coupled with (arbitrary) IC thresholds

How to assign chains and IC thresholds to incoming jobs?



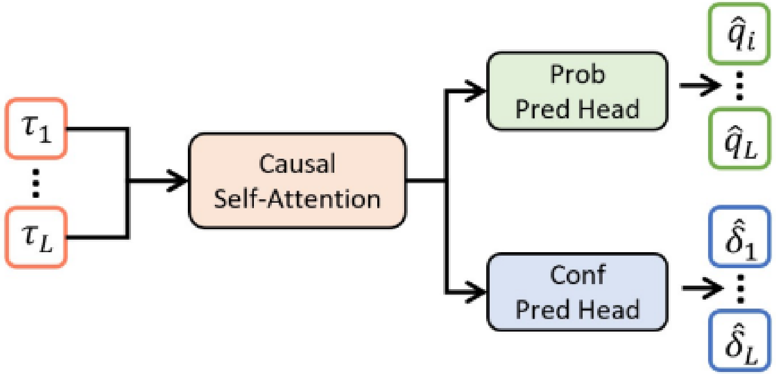
CEEN distributed early-exit scheduler

- Maximize the accuracy of the early-exit DNN while minimizing the data loss
- Control policy:
  - IC threshold configuration
  - chain assignment probability



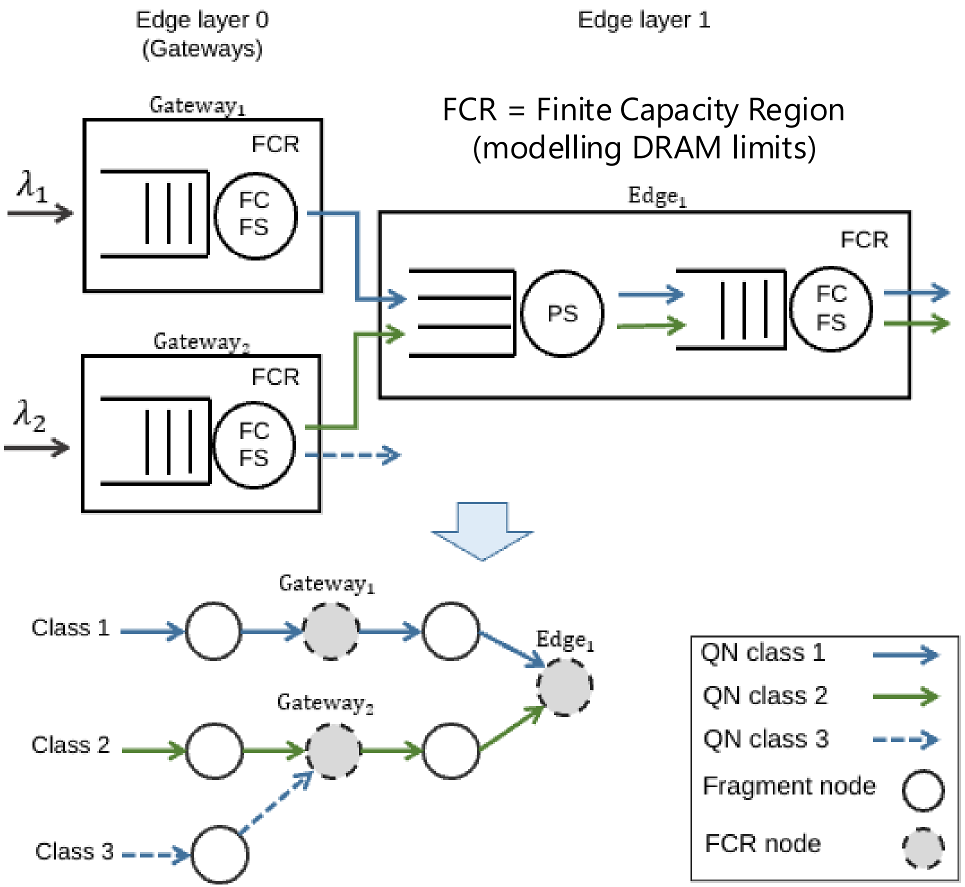
- EENN Predictor
  - Decoder-only transformer
  - Characterizes IC dependencies using empirical data
  - Input: EENN thresholds
  - Output: confidence scores and early exit frequencies

EENN Performance predictor



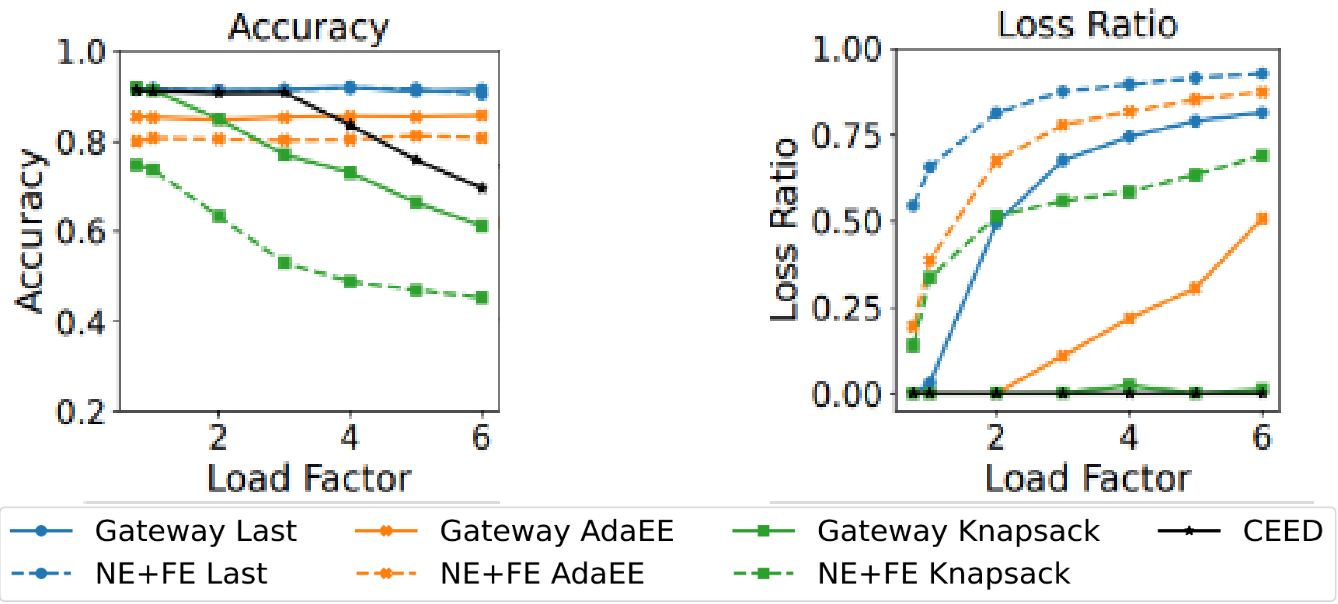
Loss Ratio Predictor with Early Exit

- GNN that predicts throughput and loss ratio
  - Extended version of ChainNet
- System seen as a queueing networ
  - Early-exit modelled as chain
  - Considers blocking and CPU contention
- Memory constraints as limits on queue buffer capacity
  - Fixed-point use of ChainNet



Evaluation

- Multiple ReNet50 deployments, e.g., gateway only, near edge (NE) + far edge (FE)
- Baselines: AdaEE, Knapsack, Exit last
- Load factor = theoretical device utilization (1=100%) without job loss

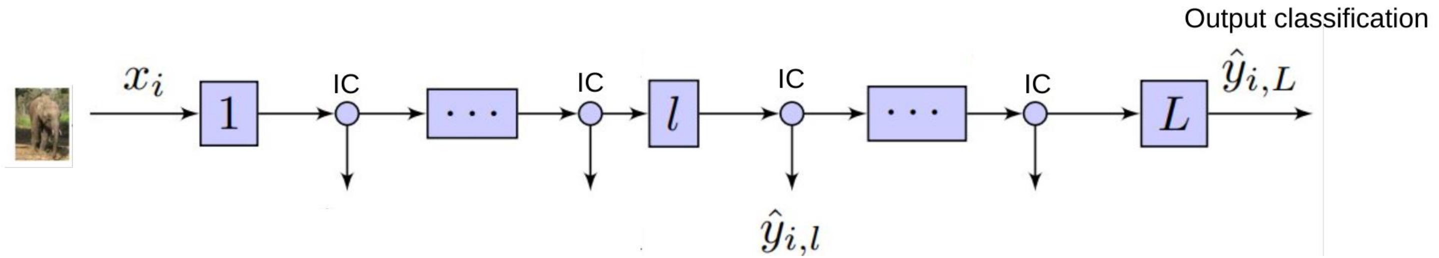


- Similar results when results are considered for other EENNs (e.g., ResNet101)

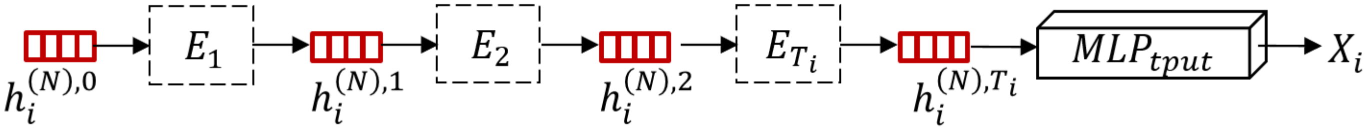
Conclusion

Summary

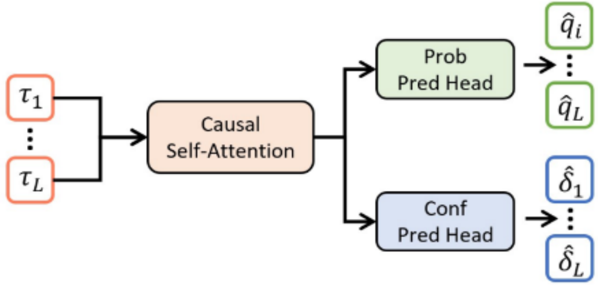
- We can recast early-exits as a mechanism to tune performance and reliability



- We can tailor GNNs to performance prediction tasks: https://github.com/imperial-qore/ChainNet



- Early-exit scheduling in DNN-based distributed data processing



Open challenges

- Generalize early-exit approach to job priorities
- Generalize early-exit approach to cope with non-i.i.d. data and bursts
- Generalize scheduling for QoS to other types of adaptive DNNs
- Early-exit aware DNN topology adaptation and placement reconfiguration