University of Glasgow

School of Computing Science

Knowledge & Data Engineering Systems

AI Engineering at the Edge

Dr Chris Anagnostopoulos

Senior Lecturer in Data Engineering & Distributed Computing

Knowledge & Data Engineering Systems Group

School of Computing Science

WORLD CHANGING GLASGOW

19 MAY 2023 Alan Turing Network for AI in Geotechnics @ Uni of Glasgow

Knowledge & Data Engineering Systems

Brings together the fundamental research areas of **Distributed Computing, Data Science & Distributed ML**

- 3 Academics
- 4 Post-docs & 2 Visiting Research Fellows
- 11 PhD students

Current activities:

Distributed AI: Model training & inference are decentralized

Funding sources

EPSRC Engineering and Physical Sciences Research Council

The DataLab value from data

Horizon Europe

NHS National Services Scotland

Marie Curie Actions

Urban Big Data Centre

European Commission | Horizon 2020 European Union funding for Research & Innovation

BMW

NXP

HT2

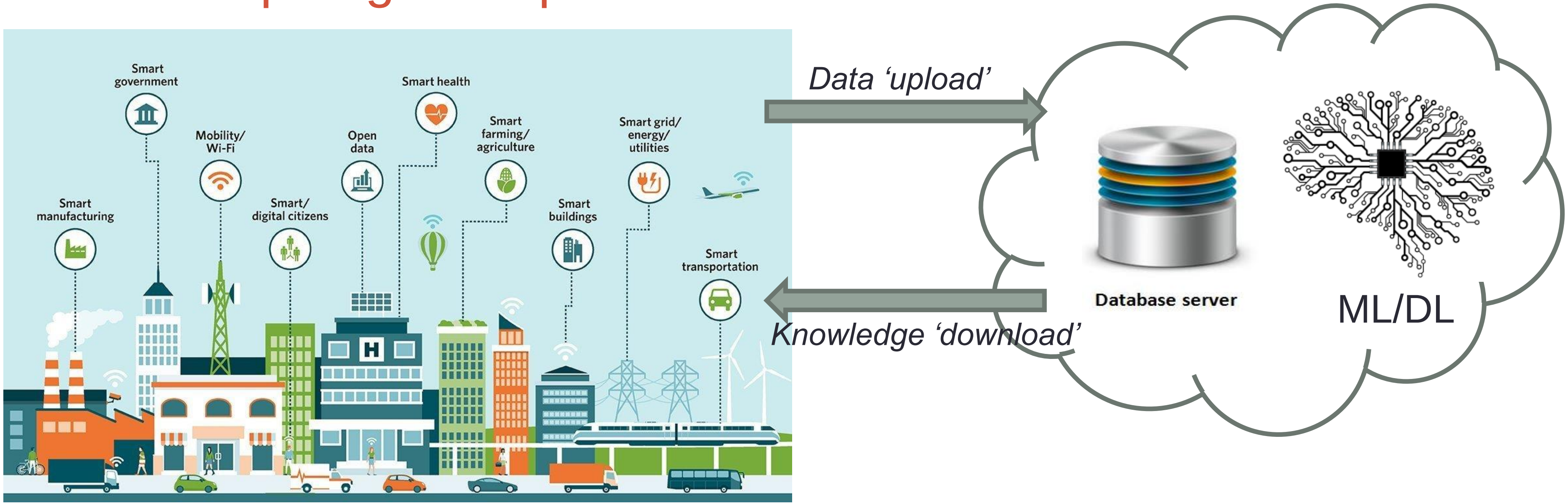
virtualtrip materializing innovation

inCITES Consulting S.A.

repado //biomedical informatics

HUAWEI

Cloud Computing: Principle



Glasgow Smart City: Data collection & urban analytics (**real-time traffic maps, prediction of available parking slots**, **smart waste management**…)

Edge Computing: Paradigm

Data Volume Challenge: Billions of computing devices (e.g., sensors, vehicles, surveillance cameras) produce **~460 Exabytes of data / day!**

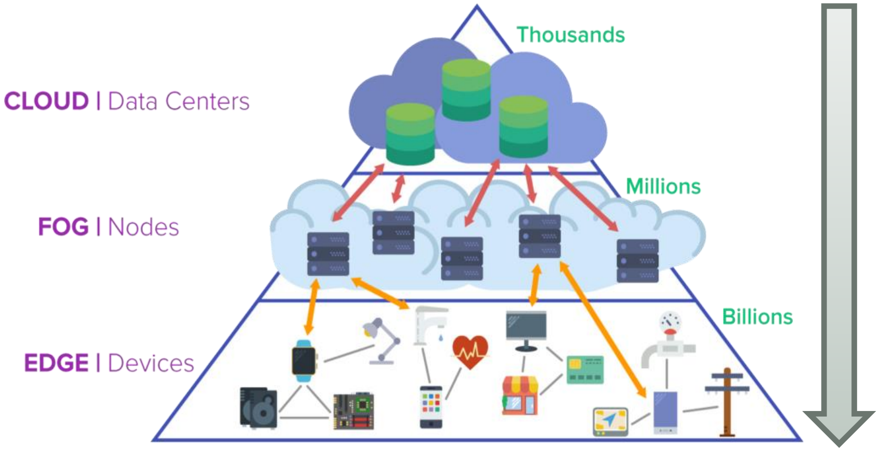
Device Connectivity Challenge: ~30 Billion connected devices, i.e., ~130 new devices *per* second are connected to the Web.

Principle: Push Intelligence (ML/DL models/processing tasks) as **close** to the **data** sources as possible, i.e., **decentralizing intelligence**

Vision: Seamless extension of Cloud for **localized & real-time** data processing & knowledge extraction (ML models)

Fundamental Objectives

- ✓ Minimize Latency (eliminate data transfer to/from Cloud)
- ✓ Minimize Network Load by reducing redundant communication with Cloud
- ✓ Support Real-time Applications, e.g., real-time traffic maps, Augmented Reality, Connected Vehicles, 360° imaging.



Data Sources (e.g., sensors, Smart Cities)

from Data Collectivity *to* Data Selectivity *to* Data Relevance

Context: Unprecedented growth of data *surpasses* models and processing capabilities.

i.e., we generate more data (big data) *than* we want to process (relevant data)

Fact: Gartner[1]: 90% of data are 'useless' or currently 'irrelevant'; *relevance is the new currency*

[1] Gartner; https://www.gartner.com

Rhetorical? Do we *need* all the data? Do we need to *analyse* all the data?

Principle Revisit: Push intelligence close to the source of **relevant** data (and not to any data)

Objective: Data relevance

- Identify relevant/significant data (**where? how? when?**) to **feed** our models
- Develop ML/AI to *learn* the relevant data from *experience*
- Process *only* what & when is needed; *will* be needed in future.
- Be *proactive* in identifying future relevant data; predict our needs?

DATA-CENTRIC AI

Reusable AI

Fact: Redundancy because of *similar* data, thus, *similar* ML/DL models, for even *similar* analytics tasks!

Analytics tasks: e.g., classification (SVM), image recognition (CNN), reinforcement learning (RL), time-series forecasting (LSTM), compression (VAE), outliers detection (OCSVM), …

Reuse existing models
Or, make models *reusable*

Rhetorical? Do we *need* all the models? Do we need to *train* all these models? Do we need all these *redundant* models?

Challenge: Can ***existing*** AI/ML models be ***reused*** or be made ***reusable***?

Benefit: Avoid *building* and *maintaining reduplicative* AI models since reusable models can be 'reused' by other nodes' predictive tasks

EPSRC
Engineering and Physical Sciences Research Council

UK/EPSRC: £3M Grant 'Closed-loop Data Science'

Principles for Reusable AI

Multi-task Learning (MtL): case of **Federated Learning**, which exploits <u>similarities</u> among data and tasks

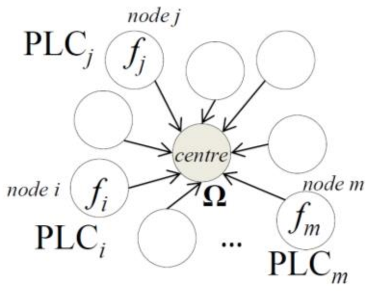
Our Target: Models *useful* in multiple tasks & data, therefore, being *reusable*. Thus, nodes can *reuse* those models <u>*without the need of* training new ones</u>.

Our Idea: Instead of training independent (local) models on nodes with *less* capacity to be reused, we contribute with **distributed-learning models** that learn from *all* of nodes' tasks at once.

Fact: MtL excels when tasks/data have some level of correlation/similarity, <u>*which is the reality in our case*</u>.

Contribution: Distributed AI Framework training *reusable* models.

- Nodes initially train their *local models* & produce their *performances* on *local* tasks.
  - **Learning Curves** (PLC): universal indicators of model performances used for hyper-parameter selection in Deep Learning

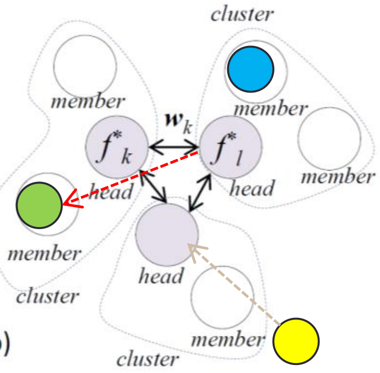


(a)

- Identify **correlations** among models' performances and data via PLCs, thus, *nodes are grouped together*; **cluster-heads** are then selected.

- Distributed AI runs across ***only*** cluster-heads *by* **exchanging model parameters** and **not data**!

$$\min_{\mathbf{W},\Omega}\left\{\sum_{i=1}^{m}\sum_{t=1}^{n_i}\mathcal{L}_i(\mathbf{w}_i^\top,(x_i^t,y_i^t))+\frac{\lambda_1}{2}tr(\mathbf{W}\Omega^{-1}\mathbf{W}^T)+\frac{\lambda_2}{2}\|\mathbf{W}\|_F^2\right\}$$



(b)

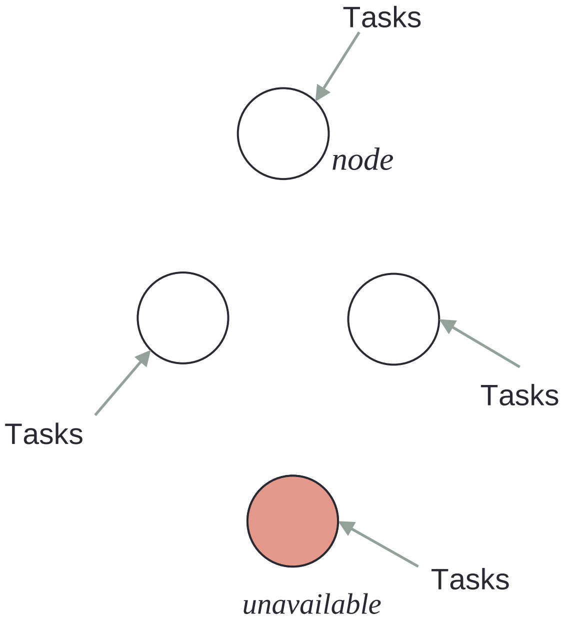
- Cluster-heads generate models, which can be **reused** by *any* member of *any* group.

Resilient AI

Edge Analytics: AI models' *inference* performed near to the data/ on board the nodes.

Fact: When a node's service turns *unavailable* due to e.g., service updates, node maintenance, or even failure or attack, the rest (available) nodes could *not* efficiently replace its service due to e.g., different data, access patterns, and AI models.

Challenge: *Build* and *maintain* the systems' resilience due to node's unavailability by avoiding interruptions of AI services.



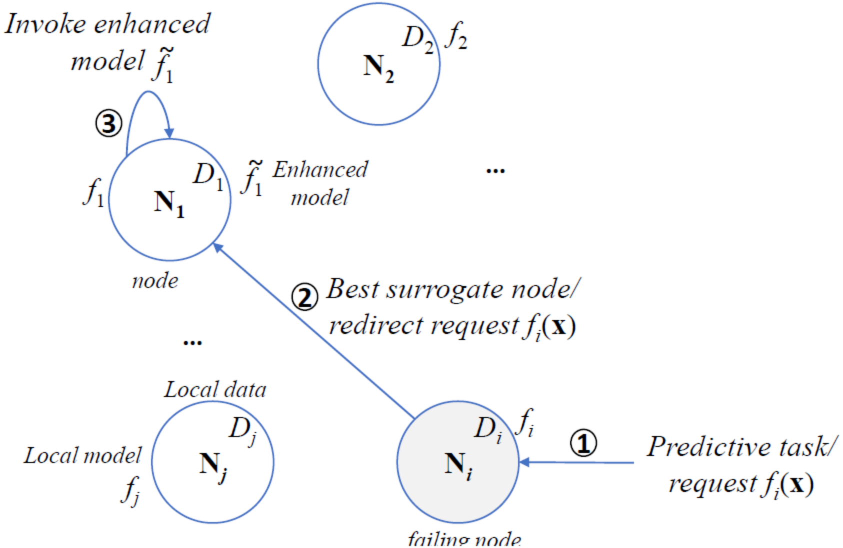
EU Horizon £8.5M TRACE:
Integration and Harmonization of Logistics Operations

Resilient AI

Idea: Make nodes capable of **substituting** failing nodes by building **Surrogate AI models**

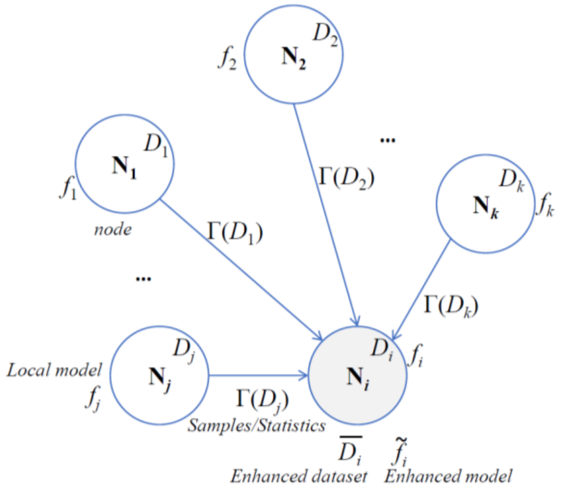
i.e., generalizable AI models trained based on *neighbouring* data.

Benefit: Guide task requests *from* failing nodes *to* the ***most*** appropriate surrogate nodes (principle of reciprocity)



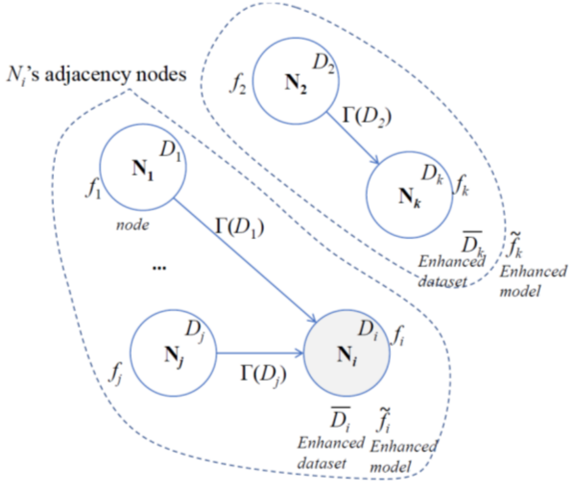
Resilient AI

Challenge: what information is required to be **shared** among nodes to build surrogate AI models with equivalent predictive performance compared to failing nodes?



Contribution: adaptive models to data patterns from **neighboring** nodes by *sharing*:

- Neighboring data samples
- Neighboring **latent data space** (e.g., eigen-basis, KPCA)
- **Generative** AI models from neighboring nodes (e.g., GANs, CVAEs)



In-Vehicle AI: Driver Behaviour & Emotion Identification

Goal: Classify the driving **behavior** & **emotions** in urban driving context

Input: on-board vehicle sensors and cameras ~4GB *per* driver, *per* vehicle, *per route*

Output: driver profiling, e.g., efficient / safe / aggressive / green / happy…

Analytics Tasks: e.g., features extraction, training classifiers, emotion recognition

Challenge 1: Distributed AI Learning under **privacy** sensitive in-vehicle functions, i.e., sharing **only** model parameters and **definitely not** data.

Challenge 2: When to offload tasks to (**road-side units**) servers to **minimize** expected latency delay due to limited communication.

Challenge 3: Which servers to offload tasks for fast processing, i.e., **maximize** the probability of offloading to 'best' servers due to load.

NXP BMW GROUP MINI Rolls-Royce Motor Cars Limited

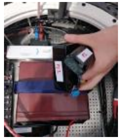
University of Glasgow

School *of* Computing Science
Knowledge & Data Engineering Systems

AI in Swarm Intelligence

Swarm of USVs: USVs are treated as **Autonomous** nodes. One Leader (USV) and Members (USVs) sharing ML/DL models for sea surface environmental monitoring.

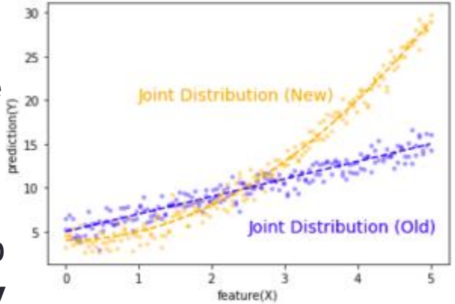


Flock of USVs

Challenge: Decentralized ML/DL Models Update

- **Dynamic data** (concept drifts) make models **obsolete**
- Swarm decides on:
  - <u>when</u> to ***update*** & ***re-train*** ML models
  - <u>when</u> to *share* ML models to Leader/Members to **minimize** models' discrepancy under **energy** budget.



EU/H2020 GNFUV

European Commission | Horizon 2020 European Union funding for Research & Innovation

..more Distributed AI

- **Efficient Federated Learning through Model Pruning** *by* Eric
- **Query-driven Node Selection & Data Relevance** *by* Tahani
- **Multi-Armed Bandits: Sequential AI Learning (RL)** *by* Sham

Efficient **Federated Learning** with Model **Pruning**

Qianyu (Eric) Long

PhD Student
Knowledge & Data Engineering Systems
School of Computing Science

Deep Neural Network (DNN) Pruning

- **Fundamentals: Deep Neural Network Pruning** constitutes a strategic method for ***eliminating superfluous parameters*** (weights) from an already trained NN.

- **Primary Objective**: curtail the model size <u>and</u> computational demands <u>while</u> maintaining its predictive capacity.

- **Significance:** As Deep Learning models continue to evolve, the **dimensions** of NN expands.
- Pruning has become instrumental in enhancing the **efficiency** of DNNs, making them **deployable** in resource-constrained environments or embedded systems.

- **Classification & Techniques**: **Weight Pruning** & **Neuron Pruning**
- e.g., magnitude-based pruning, structured pruning, and tottery ticket hypothesis.

- **Trade-off**: Pruning is effective in reducing the size and computational needs of a model trading-off between *model size* and *performance*.

DNN Pruning in Federated Learning

Parameter Server

Transferred DNN Models

Networked Nodes

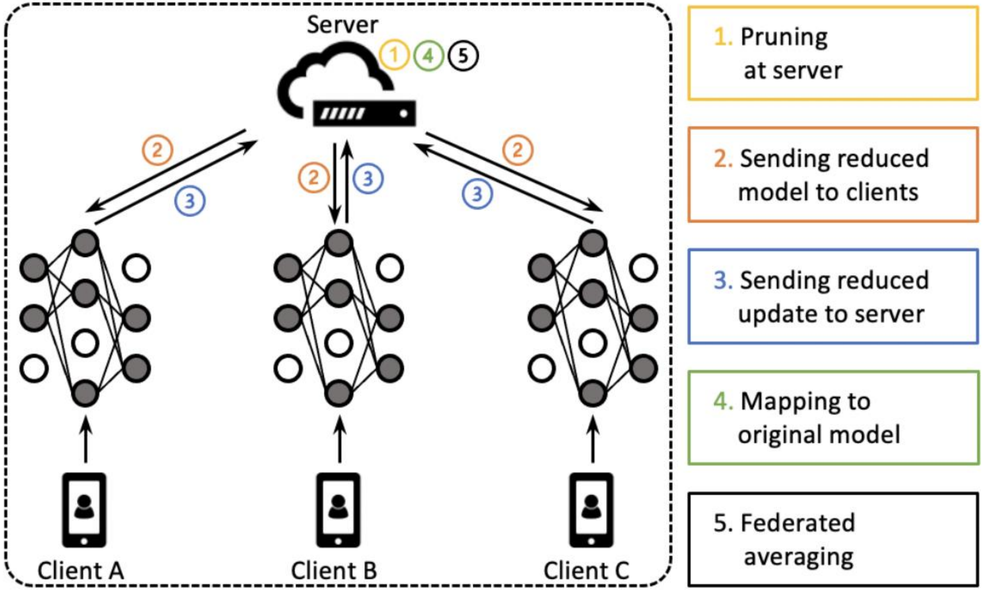


Figure 1: *A federated round of the proposed Federated Pruning. The white circles denote removed parameters.*

DNN Pruning in Federated Learning

Aim: DNN Pruning is essential in Federated Learning (FL), where the primary aim is to **train models** on **decentralized devices** with limited resources.

- **Enhanced Efficiency**: DNN Pruning allows for **large**, **complex models** to be efficiently **deployed** and **executed** on edge devices.
- It facilitates a balance between *model complexity* and *computational demands*, making FL practical and efficient in real-world applications.

- **Methodologies**: Both weight and neuron pruning techniques are used. Each device **independently prunes** its local model creating a **sparse model** that requires *less* computational power and communication bandwidth.

- **Challenges:** Pruning efficiency trade-off & *sparse* structure convergence.

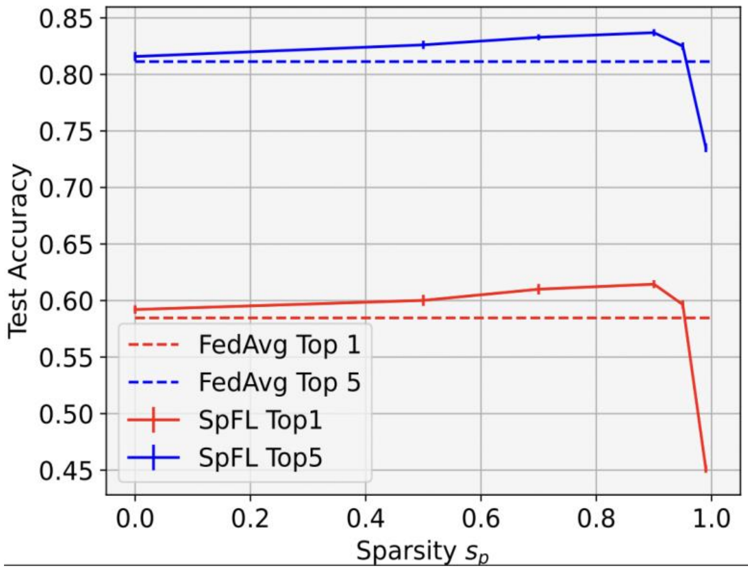
DNN Pruning in Federated Learning with **extreme sparsity**

Aim: Find **extreme sparse models** for devices **with the least drop** in predictive performance.

- Most of existing work adopt Medium Sparsity (0.5-0.8 or 50%-80%)
- **However**, an extreme sparse DNN is **essential** to be deployed on resource-constrained device.

- **Framework**: Fuse pruning methods with FL.
- **Idea**: Add growing regularization and dynamic pruning with error feedback to achieve extreme high sparsity **(0.9-0.99 or 90%-99%).**

- **Huge** improvement over using other SOTA methods, e.g., PruneFL, SNIP, RigL



Applications

- **Mobile Devices:** Pruning in FL optimizes models for efficient execution on smartphones, reducing size and energy consumption.
- **Internet of Things (IoT)**: Pruning ensures that FL models are **compact** and **resource-efficient**, ideal for IoT devices with limited computational capacities.
- **Healthcare:** In FL across hospitals, pruning helps maintain manageable model sizes, ensuring faster training times and lower computational requirements.
- **Autonomous Vehicles**: Pruning in FL enables efficient models that run smoothly on the on-board computers of multiple autonomous vehicles.
- **Edge Computing**: Pruning aids FL in Edge Computing scenarios, delivering **smaller and more efficient models** suited for computation on edge devices.

Query-driven Node Selection & Data Relevance
in Distributed Learning Environments

Tahani Aladwani

PhD Student
Knowledge & Data Engineering Systems
School of Computing Science

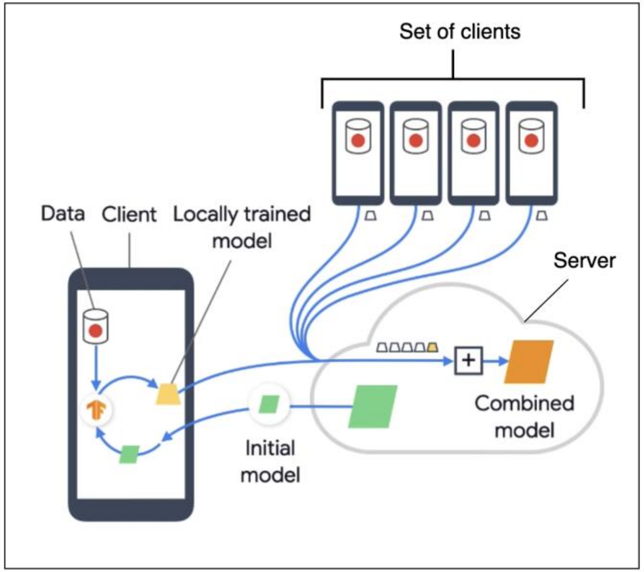
Distributed ML Model Training

Distributed Learning facilitates access to distributed data by training ML/DL models over *disjoint* data by leveraging **nodes' local data and computational resources**.

Aim: Train a ML/DL model efficiently requires training over a set of nodes, a.k.a., **participants**.

However, not all participants play the same role.
This is determined by:

- **Amount** of available data in each participant.
- **Quality** of the data in each participant.
- Percentage of **data overlap** between query's data requirement (analytics task) and participant's available data.



Problem Fundamentals

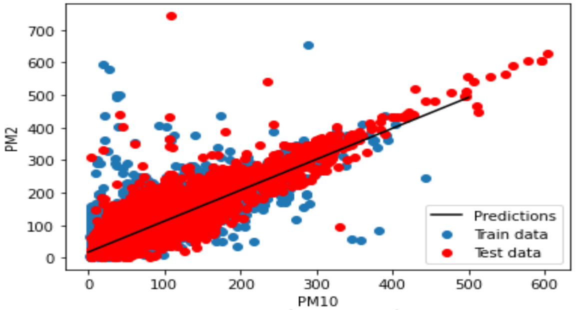
A network of nodes being **heterogeneous** in terms of data distributions.

Set of analytics queries $\mathbb{Q} = \{q_1, q_2, q_3, \dots, q_n\}$

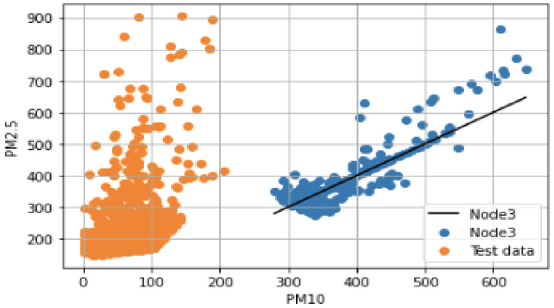
Each query q is a ML/dL learning task that **requires access** to data to be executed.

Given a query q, we engage nodes for the corresponding task. **However**, by selecting not appropriate nodes given a query, it degrades the effectiveness of distributed ML learning.

Problem: Given a query q, **find** and **engage** the most appropriate subset of participants in the ML training task.

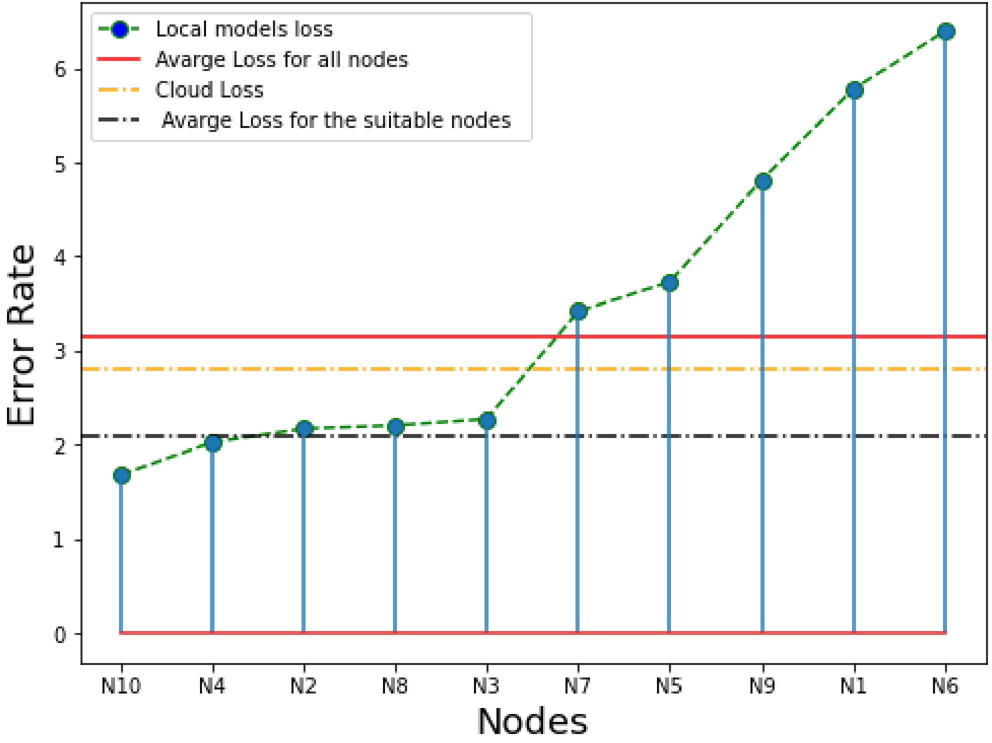


Appropriate node

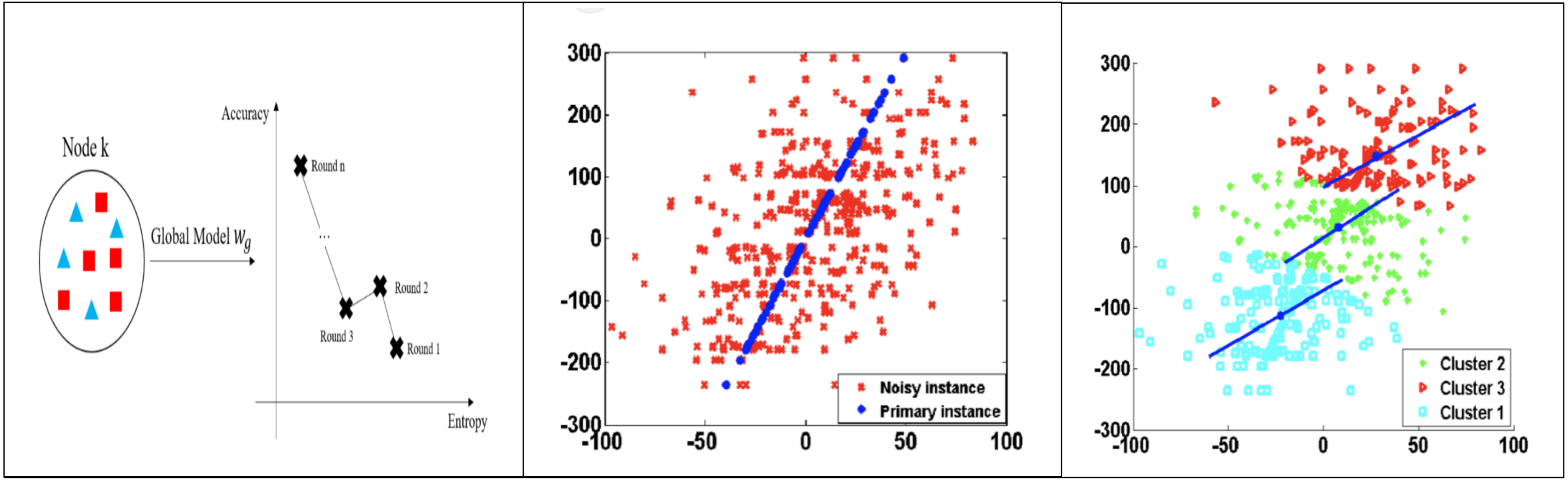


Not appropriate node

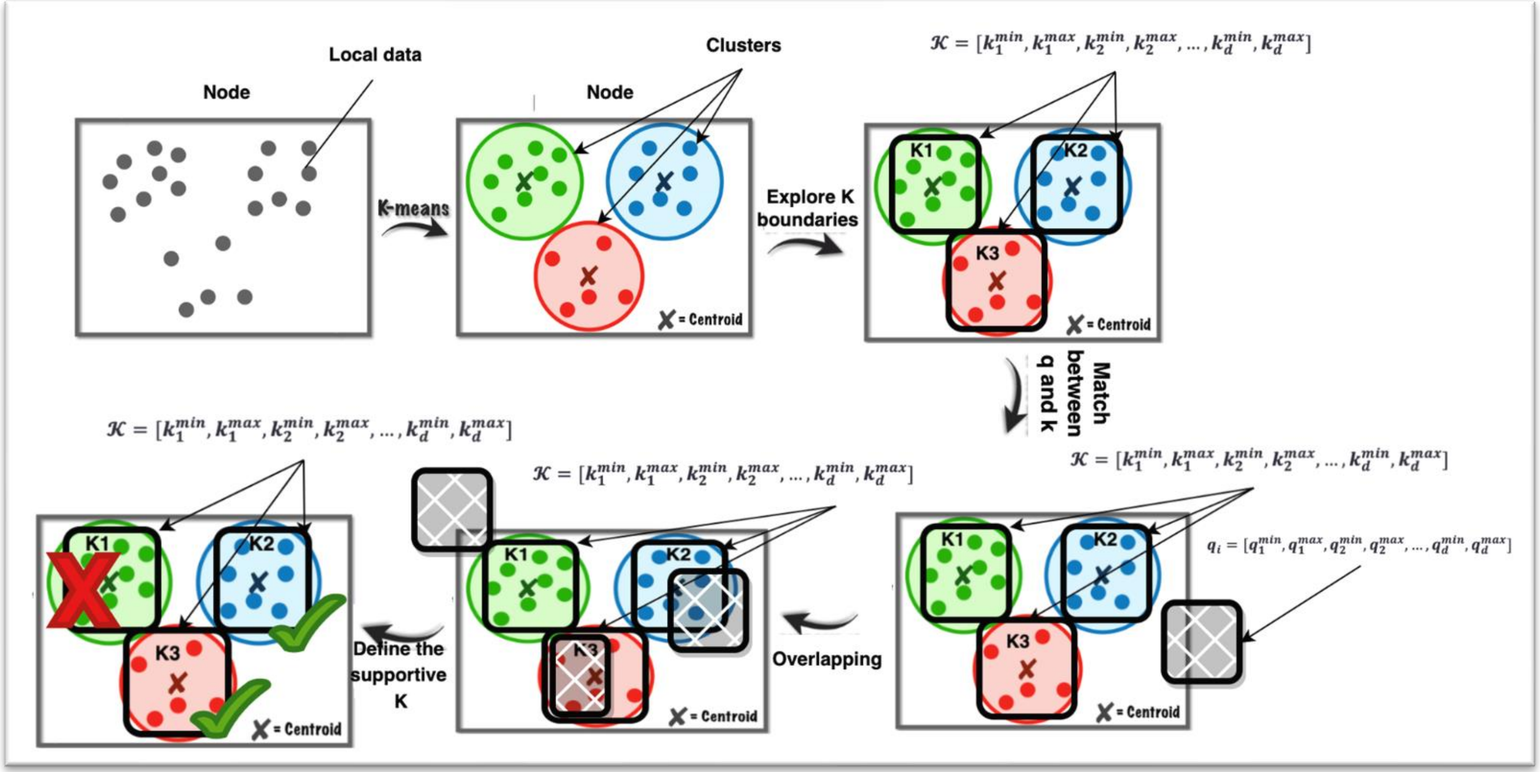
Rationale: Node & Data Relevance



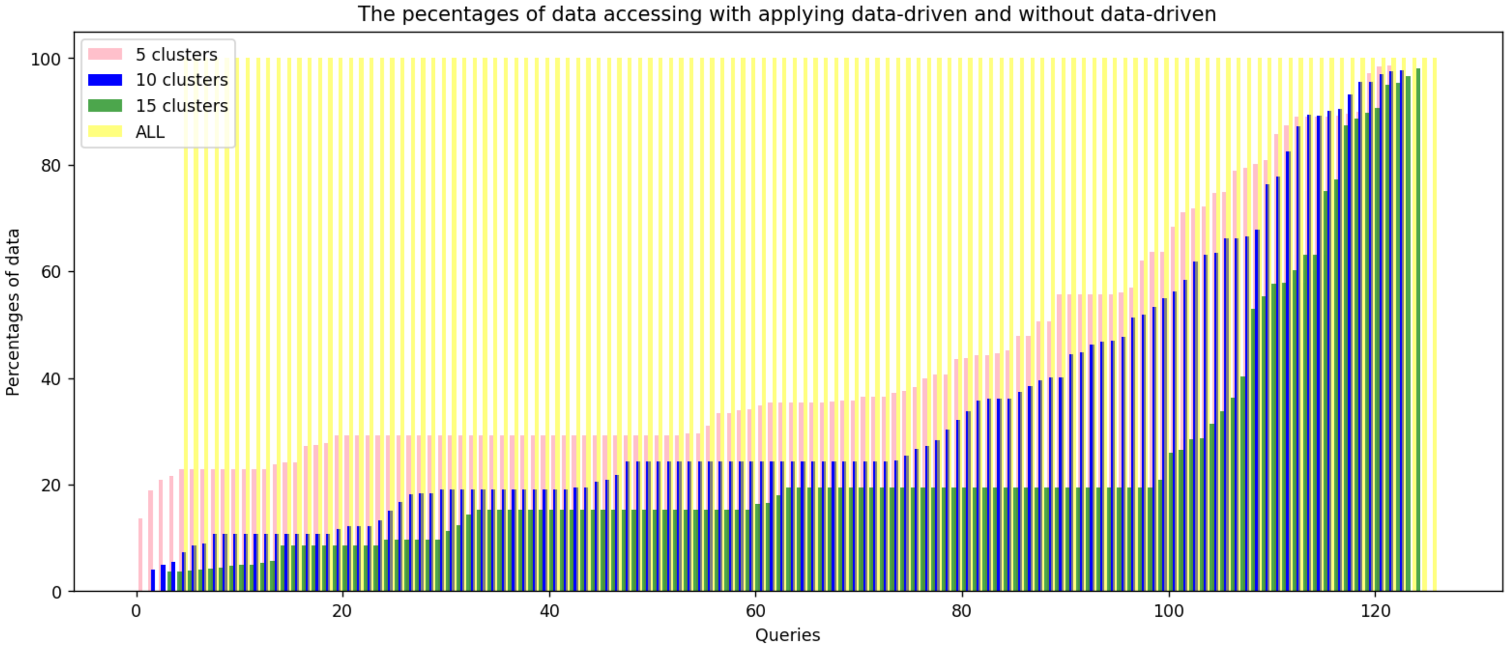
In-node Data Relevance



Data Relevance



Data Relevance



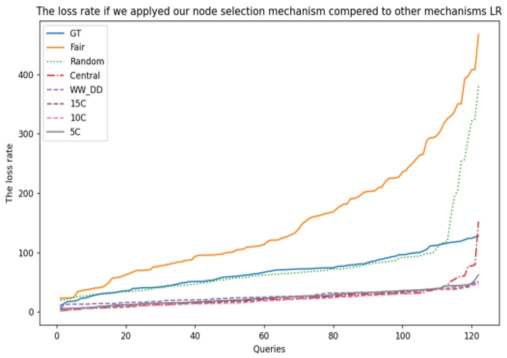
School *of* Computing Science

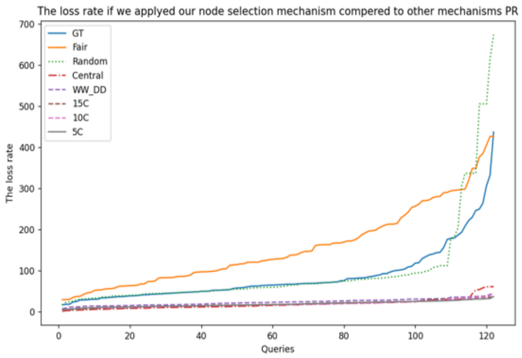
Knowledge & Data Engineering Systems

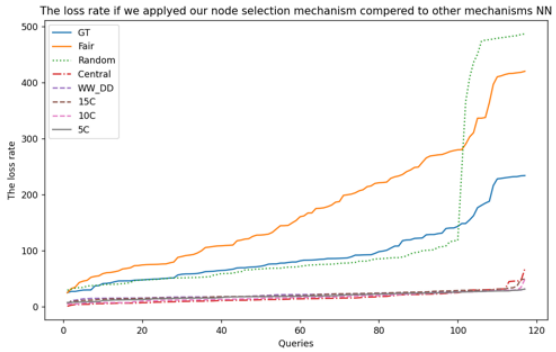
Indicating Results

Comparative Assessment

- **Random selection**: a node (or a subset of nodes) are randomly selected per query.
- **Game Theory (GT) selection mechanism**: nodes are selected based on their pre-trained models performances, i.e., models are built independently of the queries.
- **Fair selection:**







Multi-Armed Bandits: Sequential AI Learning

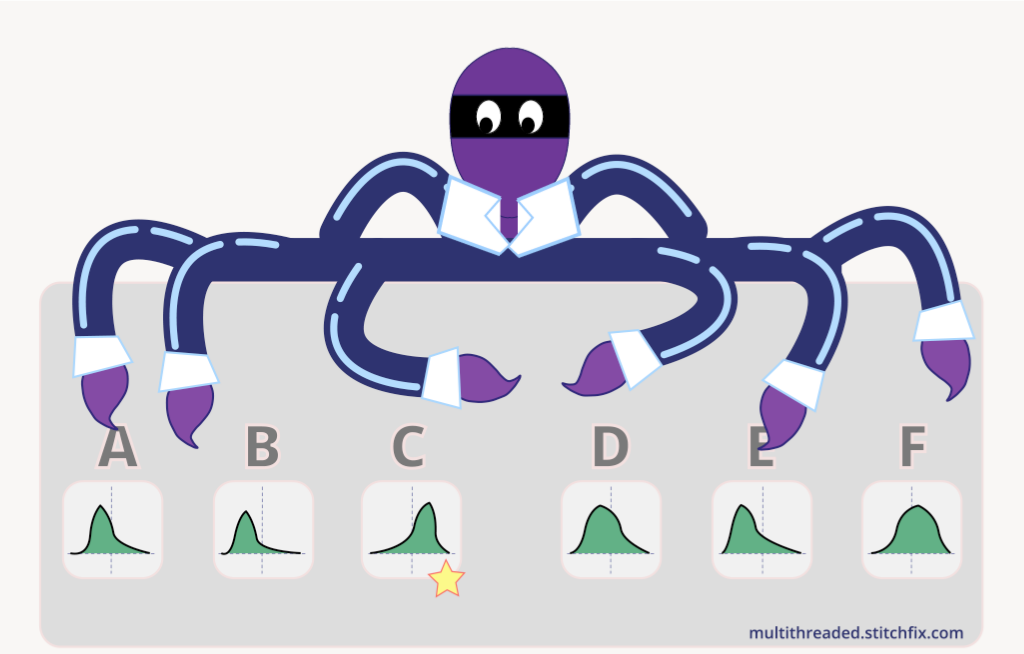
Dr Shameem Puthiya Parambath

Academic Research Fellow in Machine Learning
Knowledge & Data Engineering Systems Group
School of Computing Science

Multi-Armed Bandits

- Multi-Armed Bandits (MAB) is an AI framework for **interactive learning**.
- It can model *uncertainty* over decisions over *very large choices*.
- It is a variant of **Reinforcement Learning** paradigm with a single state.
- **Applications**: recommendations, dynamic pricing, model parameter hyper-tuning, auctions, clinical trials, channel allocation, model pruning, experiment design, etc.

Multi-Armed Bandits



Feedback-Driven Transformer-Based Query Suggestions

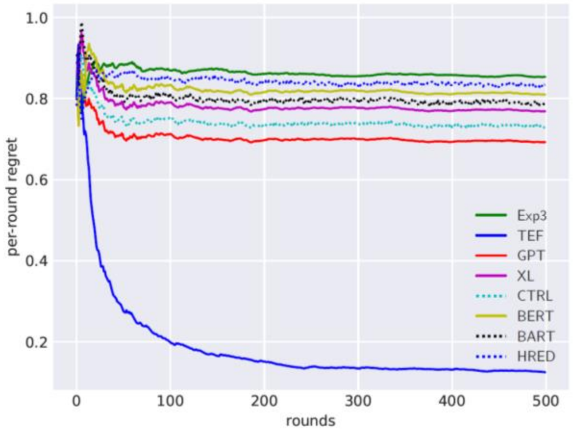
Aim: SOTA algorithms for query suggestions are based on **Transformers**.

- Depending on the initial query, **Transformer can predict the next-query**
- **However**, Transformer models are **not designed** to consider immediate feedback when making decisions.

- **System**: Combine different Transformers and adaptively build a candidate set to make use of immediate user feedback

- **Idea**: selecting the top-*k* queries from different Transformers and weighting them based on the feedback.

- Huge improvement over using a single Transformer model

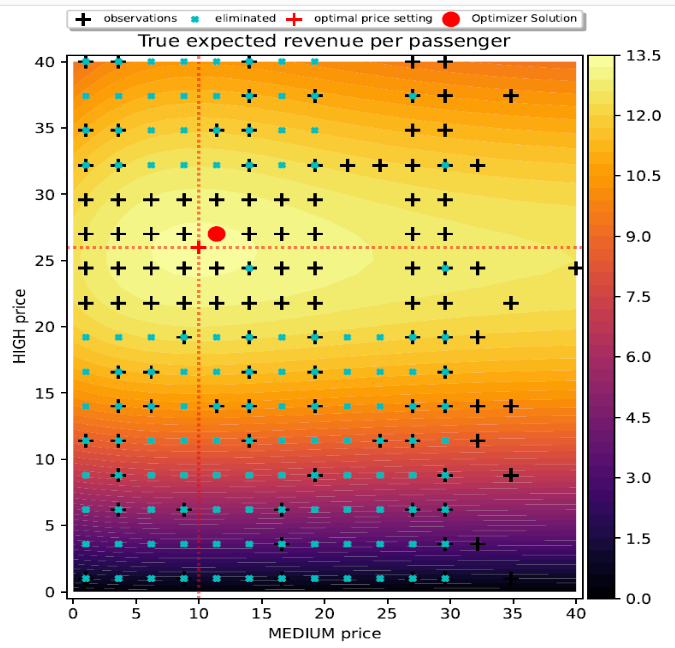


Dynamic Allocation using Bandits

Aim: Important problem in logistic management is finding the optimal prices for different service options.

Similar problems: e-commerce, rideshare

- **Optimal dynamic assignment**: estimate optimal values of different variants of the service that maximise/minimise an objective.
- **System**: Sequential recursive block-elimination MAB that *removes* blocks of values by estimating a confidence interval over the objective.
- **Application**: Validated our methodology on the problem of finding the optimal prices to be assigned to **Standard** and **Express** delivery services in courier services.



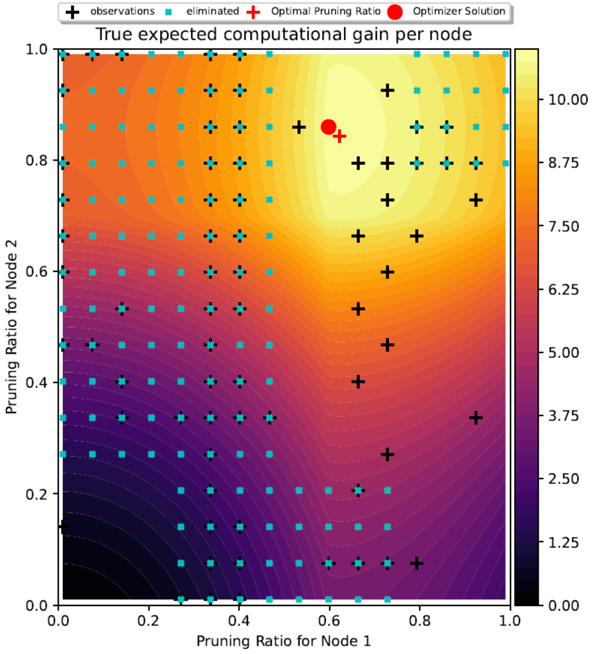
AEGEAN

Dynamic Allocation using Bandits

Aim: **Hyperparameter** tuning is an important part of building efficient machine-learning models. The problem is similar to the dynamic pricing problem discussed earlier. We consider finding optimal pruning ratios in federated learning.

Similar problems: e-commerce, rideshare

- **Optimal dynamic assignment**: find the optimal **pruning ratios** for a global DL model to distribute among nodes.
- **System**: Sequential recursive block elimination in fixed-budget pure-exploration that *removes* blocks of values by estimating a confidence interval over the objective.
- **Application**: Validated our methodology on the problem of finding optimal pruning ratios in Federated Learning.



University of Glasgow

School of Computing Science
Knowledge & Data Engineering Systems

Thank you!

Chris Anagnostopoulos
Qianyu (Eric) Long
Tahani Aladwani
Sham Parambath

christos.anagnostopoulos@glasgow.ac.uk