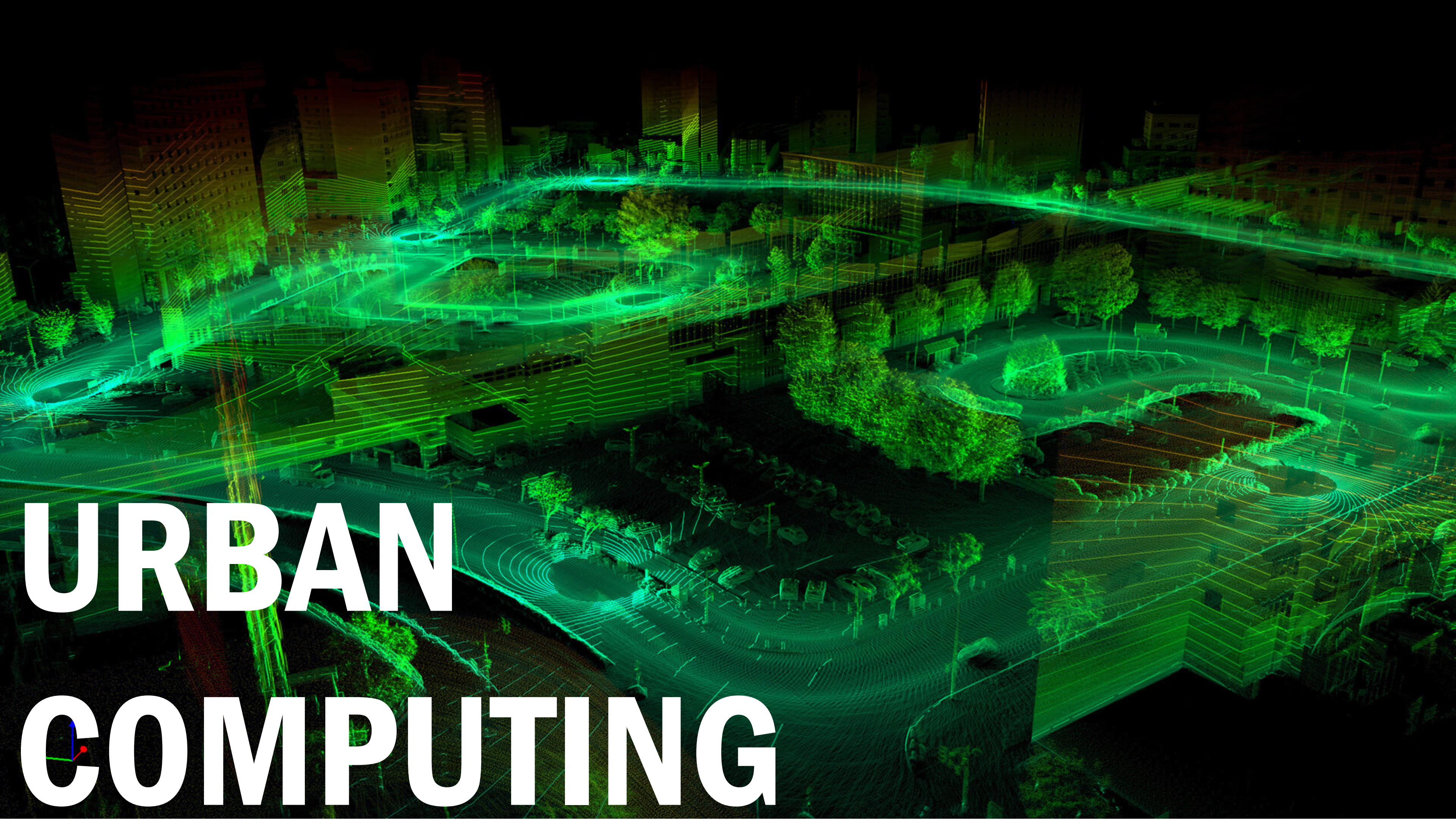


URBAN COMPUTING

Image credit: https://sigmoidal.ai/en/orb-slam-3-a-tool-for-3d-mapping-and-localization/

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

AI6128 Urban Computing

Lecture 5

AIoT

Content

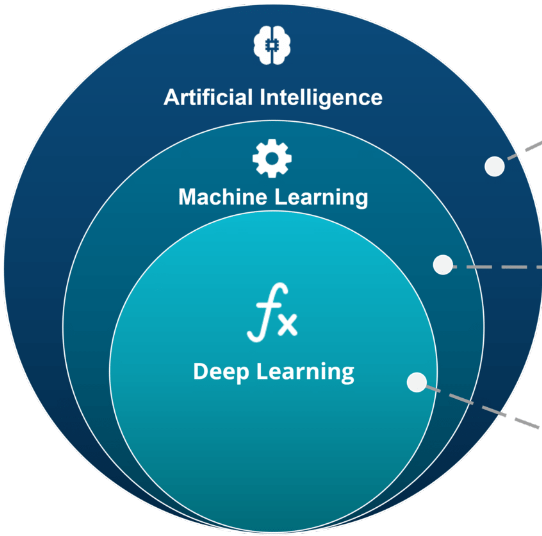
- Artificial intelligence of things (AIoT)
- Challenges in AIoT and solutions
- Simultaneous localization and mapping (SLAM)

AIoT & Its AI Primer

AIoT ≈ AI + IoT

- AIoT is the combination of AI technologies with the IoT infrastructure to achieve more efficient IoT operations, improve human-machine interactions, enhance data management and analytics.
  - AI in cloud
  - AI in fog
  - AI at edge

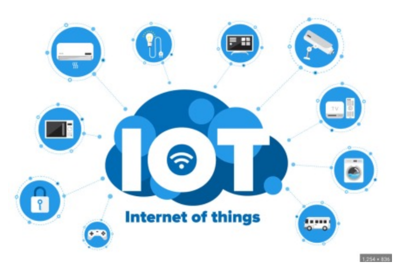
Current Focus of AIoT



ARTIFICIAL INTELLIGENCE
A technique which enables machines to mimic human behaviour

MACHINE LEARNING
Subset of AI technique which use statistical methods to enable machines to improve with experience

DEEP LEARNING
Subset of ML which make the computation of multi-layer neural network feasible



AIoT

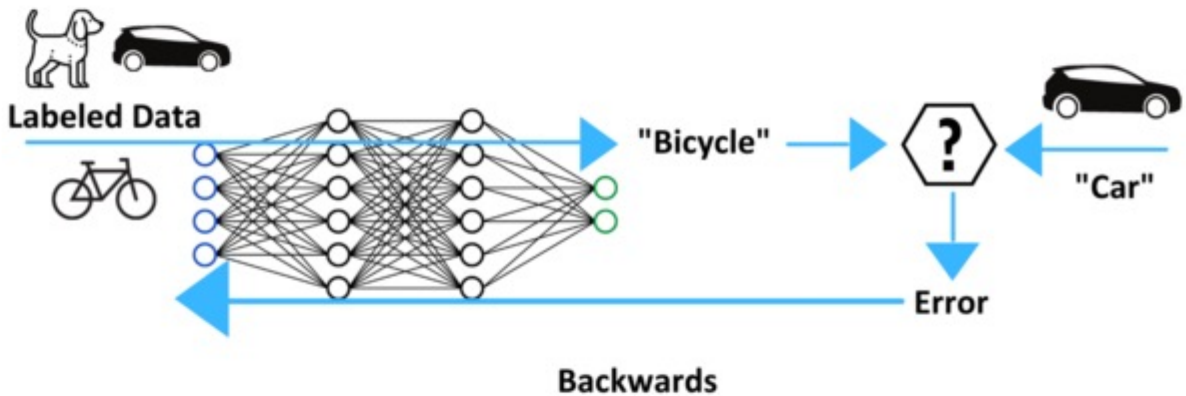
Roadmap

- A big picture of today's machine learning techniques
- Why they are successful for multimedia data?
- Challenges in making IoT artificially intelligent

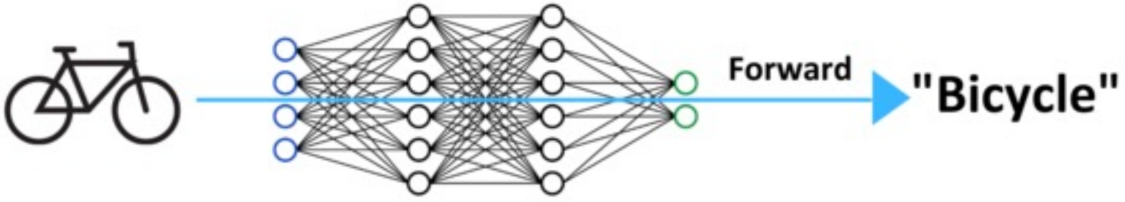
Paradigms of Machine Learning

- Supervised learning (and its deep counterpart)
  - Need labeled training data
  - Learn from feedback of error between predicted label and actual label
- Reinforcement learning (and its deep counterpart)
  - Learn from feedback given by a reward function
  - Reward function provides implicit labels: a relaxed form of supervised learning
- Unsupervised learning
  - Apply to unlabeled data directly
  - No feedback

Supervised Learning



1. Learning



2. Inference

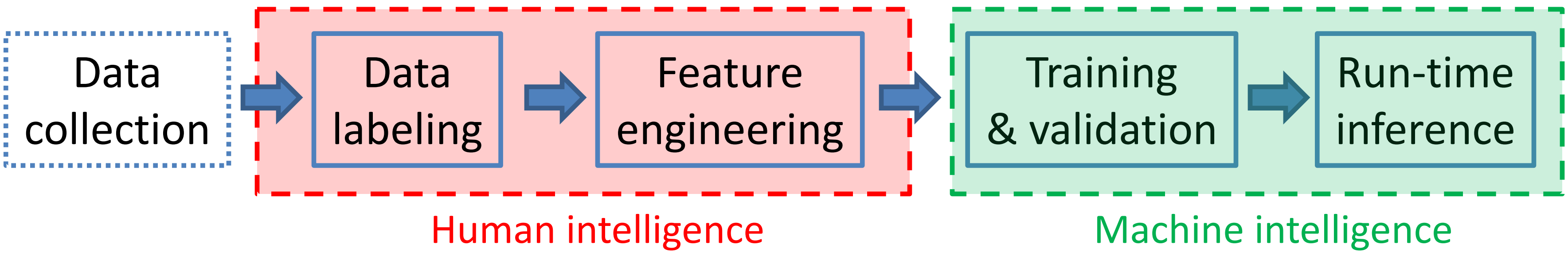
Deep Learning

- Traditional learning process
  - Data collection and labeling, manual feature engineering, training and validation, inference
  - Small-scale neural network models

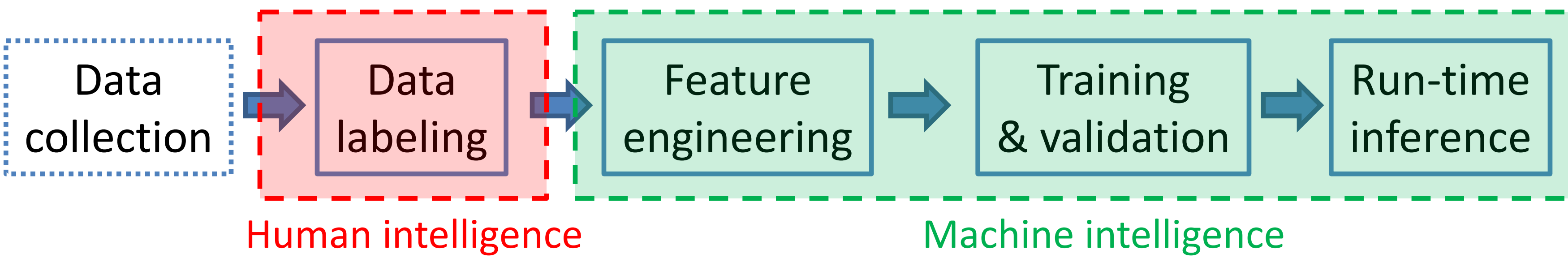
- Deep learning process
  - Manual feature engineering becomes automated feature learning
  - Deep neural network models

Advance of Supervised Learning

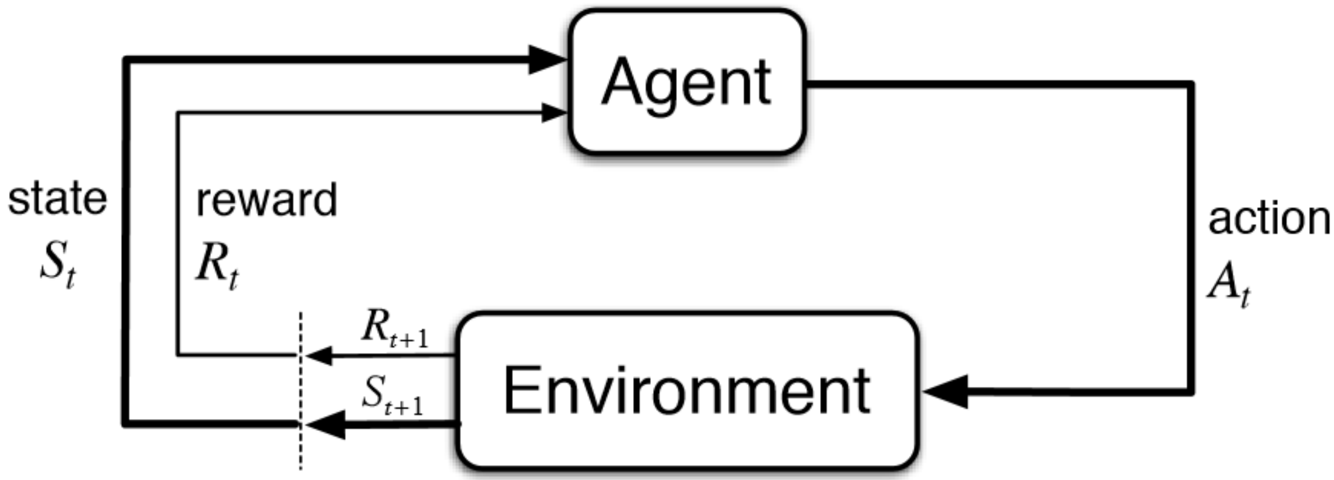
1. Traditional/shallow learning



2. Deep learning



Reinforcement Learning (RL)



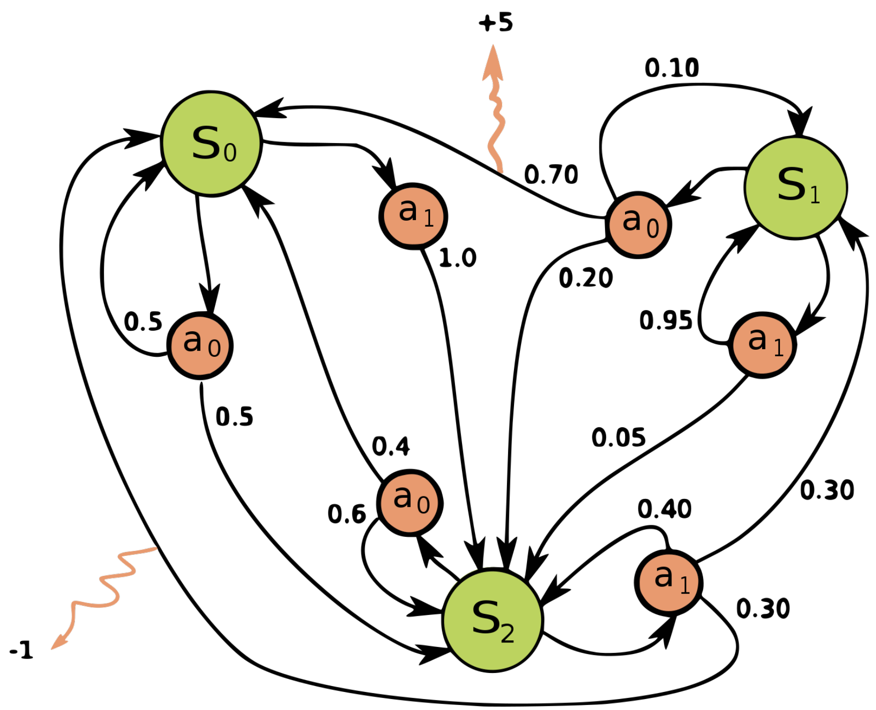
- Agent learns a policy (state → action) to maximize accumulated reward

Supervised vs. Reinforcement

- Supervised learning
  - Classification, prediction
  - No interaction
- Reinforcement learning
  - Decision making
  - Interactions with the environment

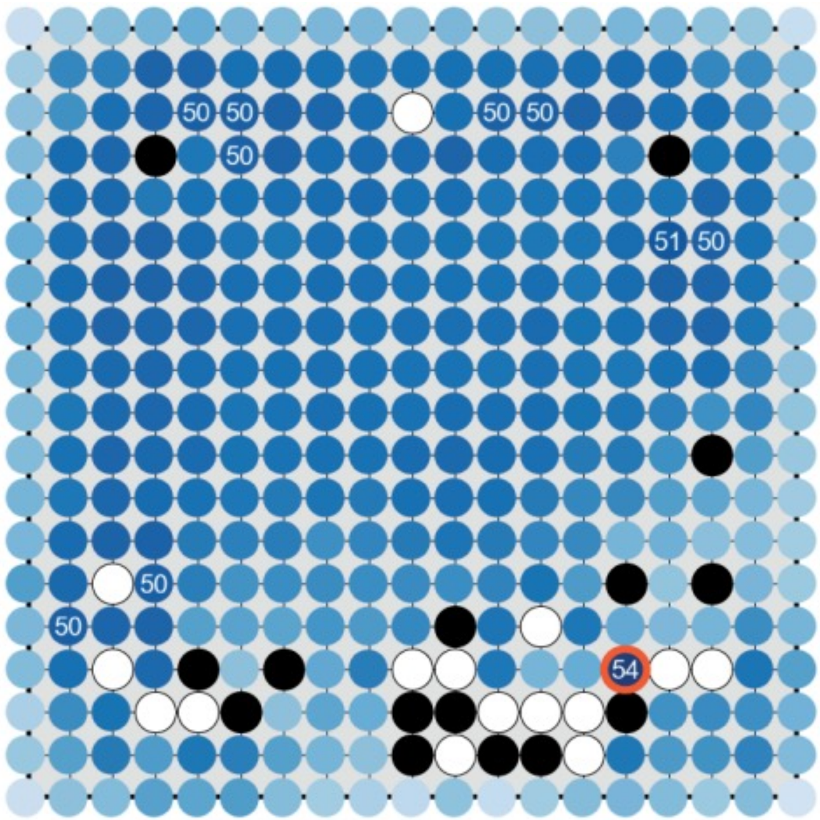
Markov Decision Process (MDP)

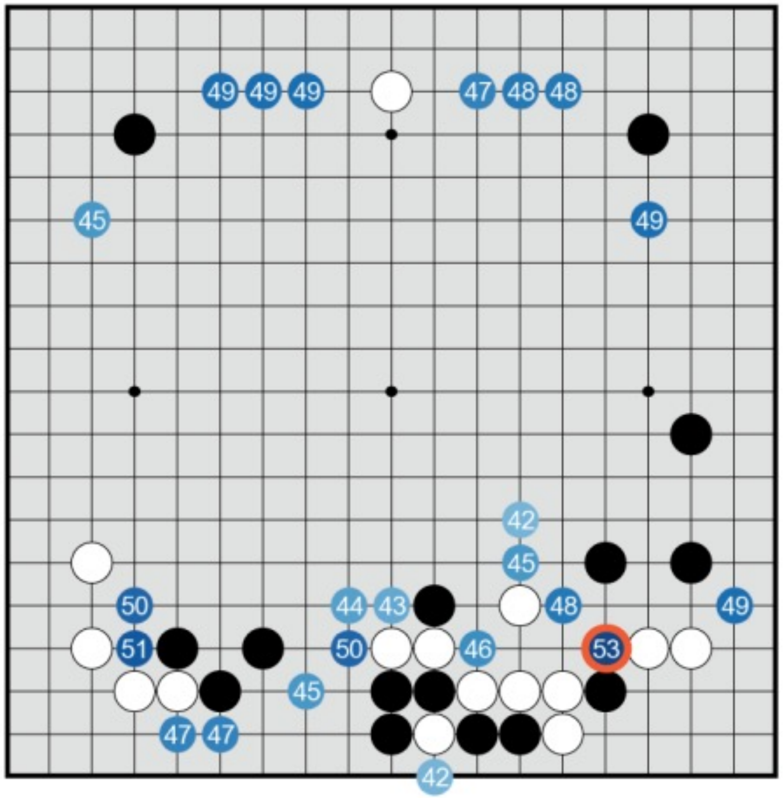
- Rigorously, RL only solves MDP problems
- Markovian: Distribution of transition probability from a state does NOT depend on how the state is reached



A simple MDP with three states (green circles) and two actions (orange circles), with two rewards (orange arrows).

Example: AlphaGo





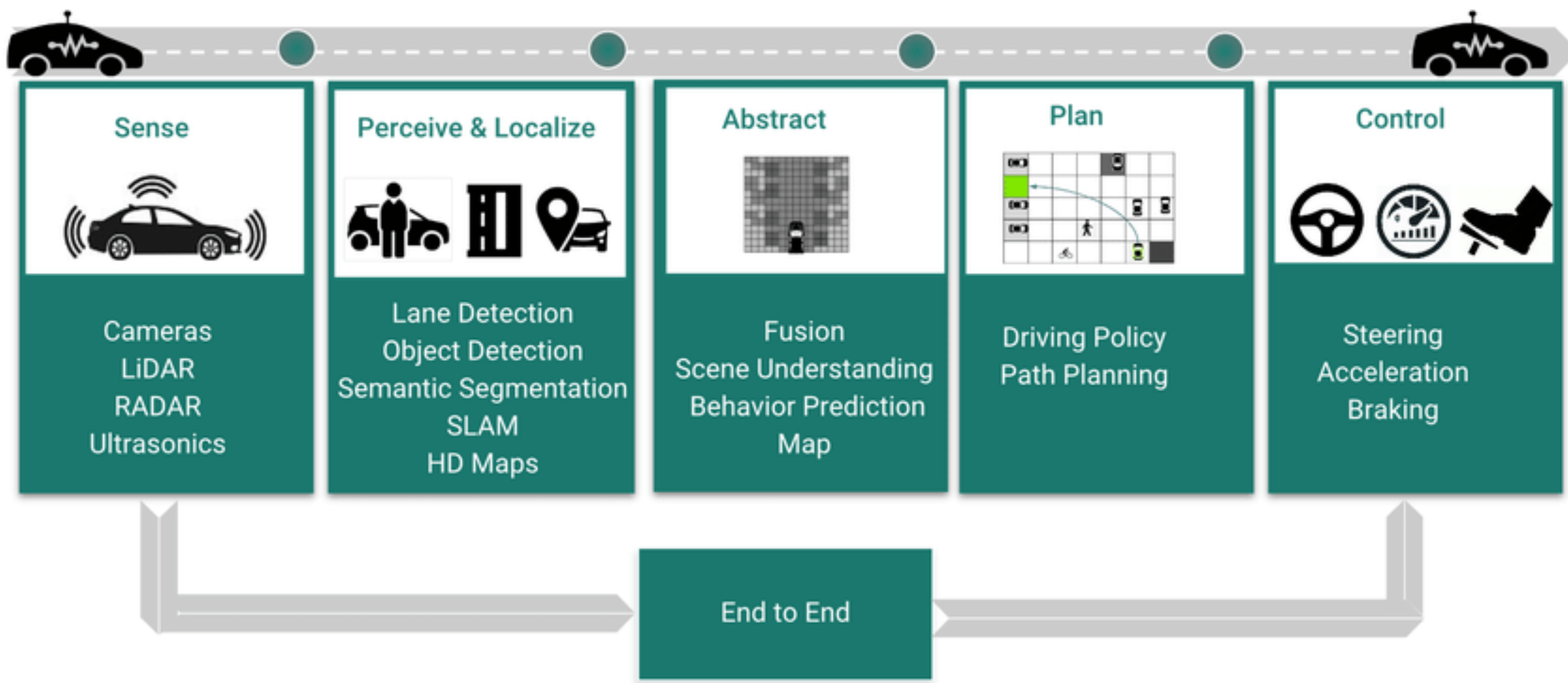
Traditional vs. Deep RL

- Traditional RL
  - Policy represented by a lookup table
  - Table is updated / extended / shrunk during learning
  - Not scalable with dimension of state

- Deep RL
  - Policy represented by a deep neural network
  - Deep neural network is updated during learning
  - Good scalability with dimension of state

Example: Autonomous Driving

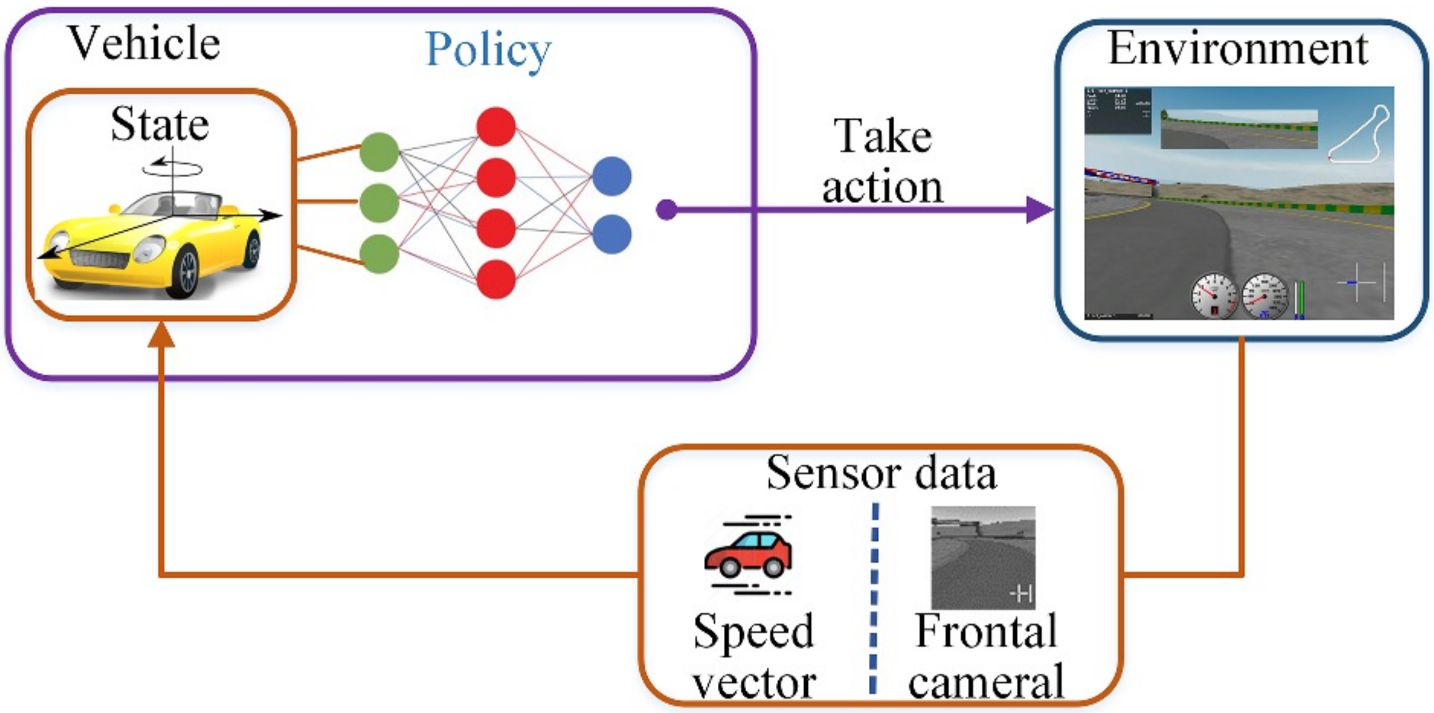
- From pipeline to end-to-end



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Example: Autonomous Driving

- End-to-end driving agent



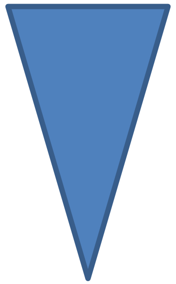
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Automated Feature Engineering

- Common key to supervised deep learning and deep reinforcement learning

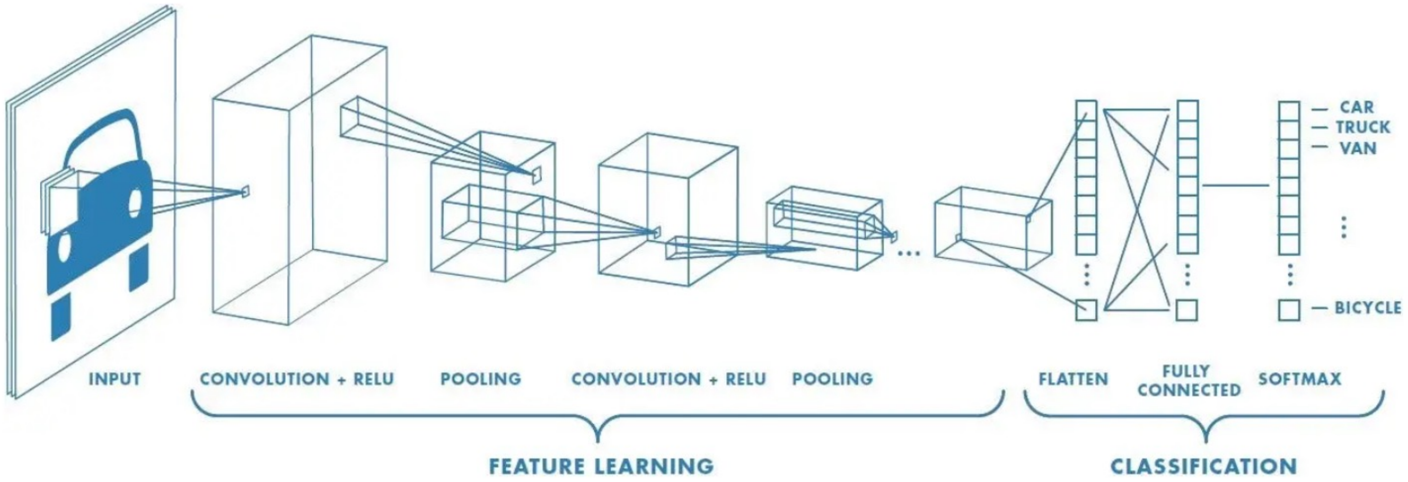
- Approaches
  - Supervised feature Learning
  - Contrastive feature learning
  - Self-supervised feature learning

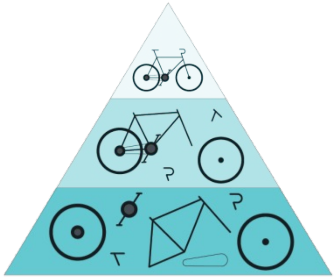
Requirement on labels



Supervised Feature Learning

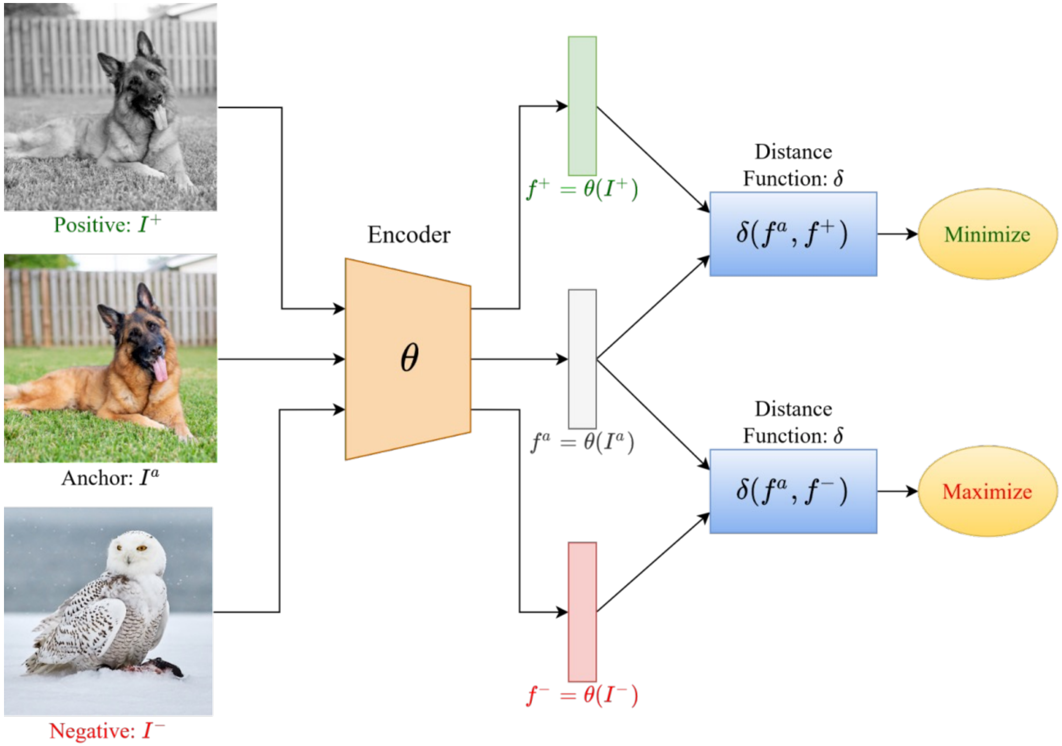
- Example: convolutional layers





Contrastive Feature Learning

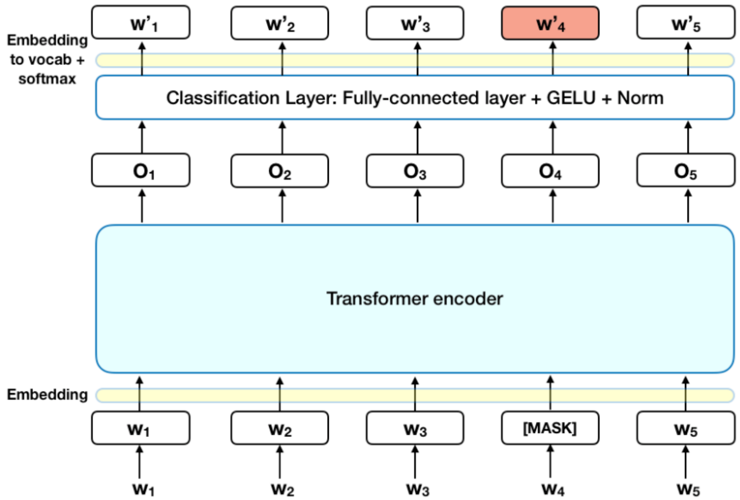
- Require contrastive information of whether any two samples are from same class



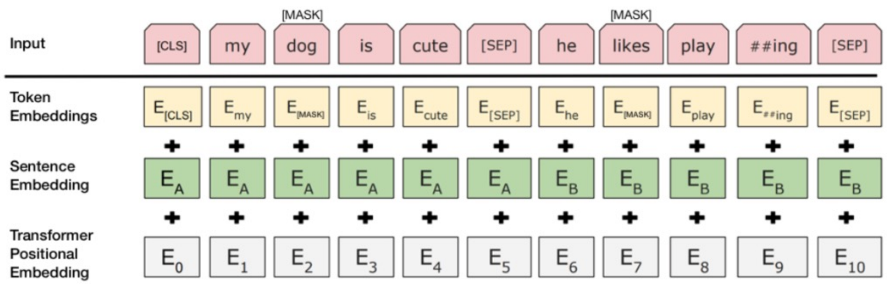
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Self-supervised Feature Learning

- Learn underlying structure of unlabeled data by predicting masked part of the data
  - Example: Google's BERT (Bidirectional Encoder Representations from Transformers)

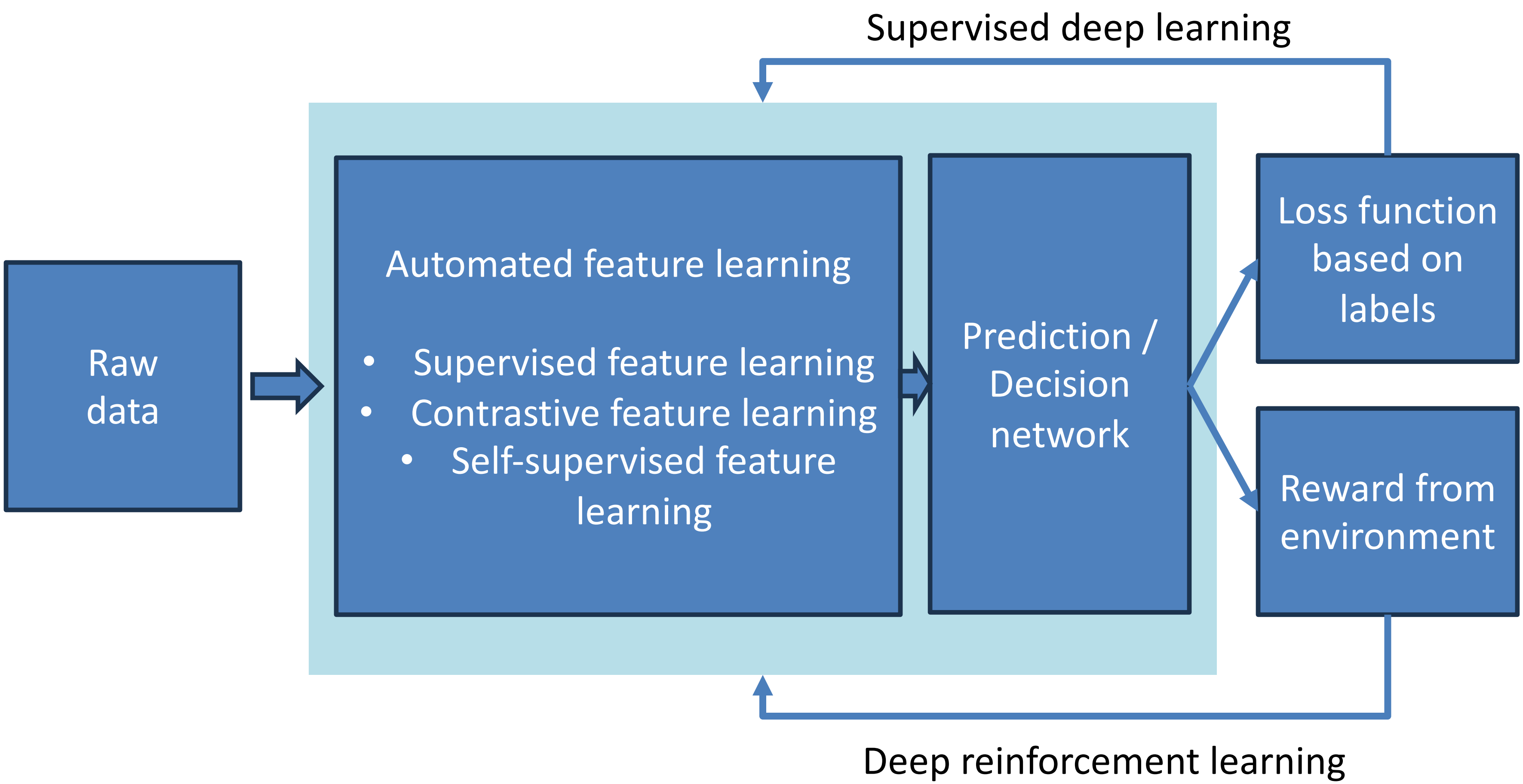


Predict marked words



Predict next sentence

Recap: Deep Learning

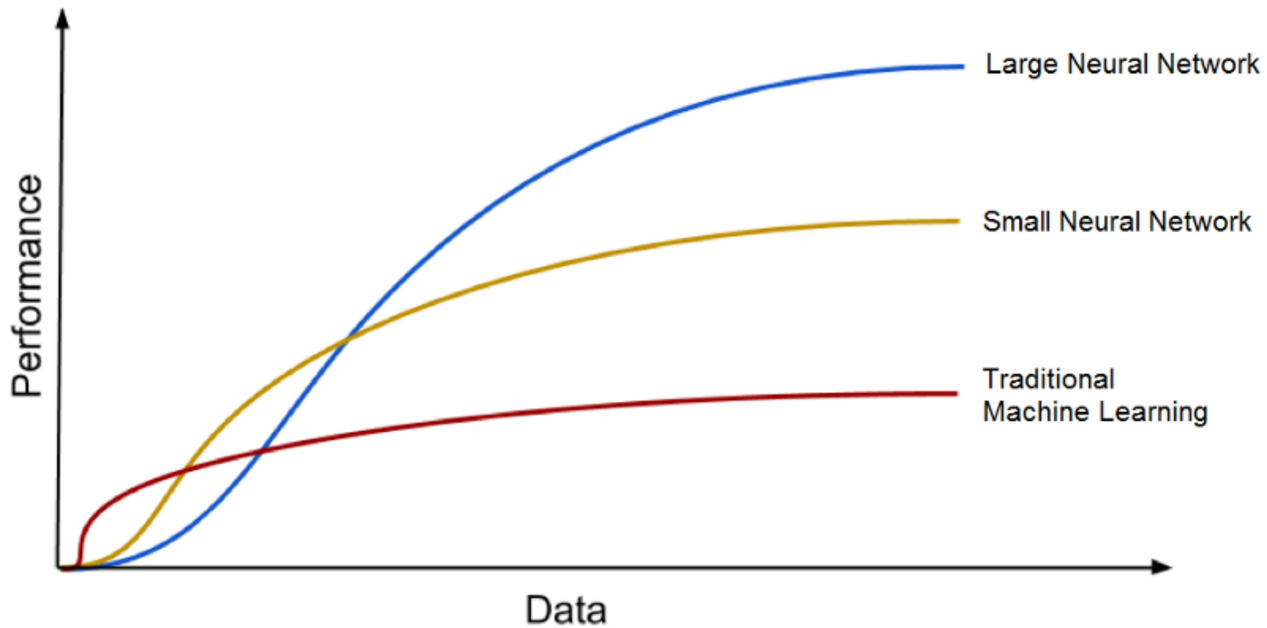


Challenges in AIoT & Solutions

Domain-Specific AI Literacy

- Understand the current capability boundary of AI in a certain domain
  - What can / cannot be done

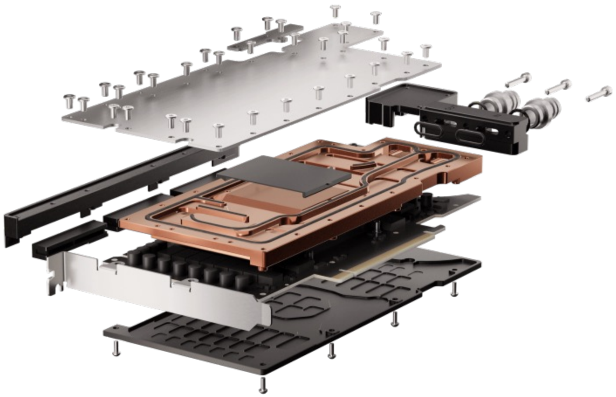
Keys to Success of Deep Learning



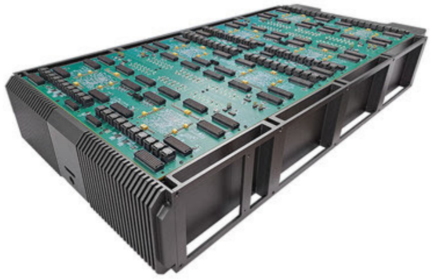
- From manual feature engineering to automated feature learning
  - Demand on labeled training data increases
  - Computation overhead increases

Hardware Accelerators

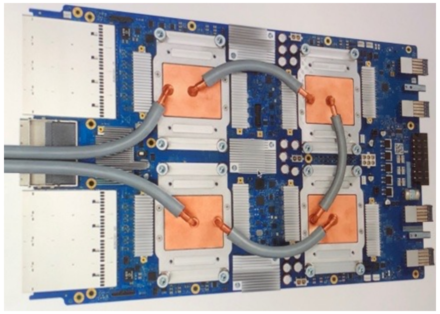
- Three major types
  - GPU
  - Field-Programmable Gate Array (FPGA)
  - Application-Specific Integrated Circuit (ASIC)
    - e.g., Google's Tensor Processing Unit (TPU)



H100 GPU



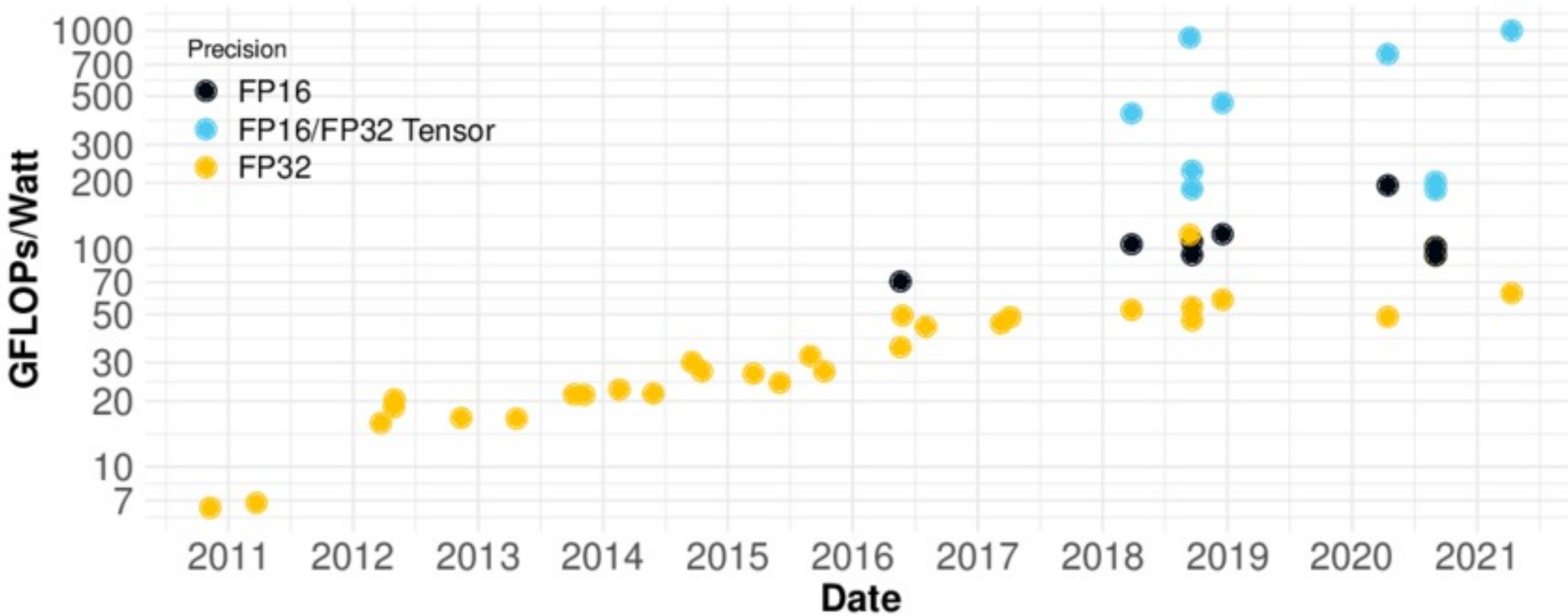
FPGA



TPU 3.0

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

GPU's Performance/Watt



Desislavov, Radosvet, Fernando Martínez-Plumed, and José Hernández-Orallo. "Compute and energy consumption trends in deep learning inference." *arXiv preprint arXiv:2109.05472* (2021).

Multimedia Data

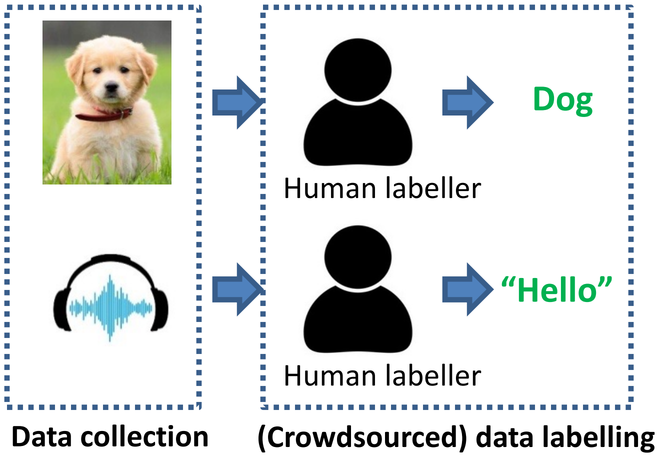
- Information perceived by human's five sensory systems
  - Sight
    - Image, video
  - Hearing
    - Audio, speech, (text)
  - Smell
  - Touch
  - Taste

- Major sources of information
- Human-made sensors comparable to human's sensory systems

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

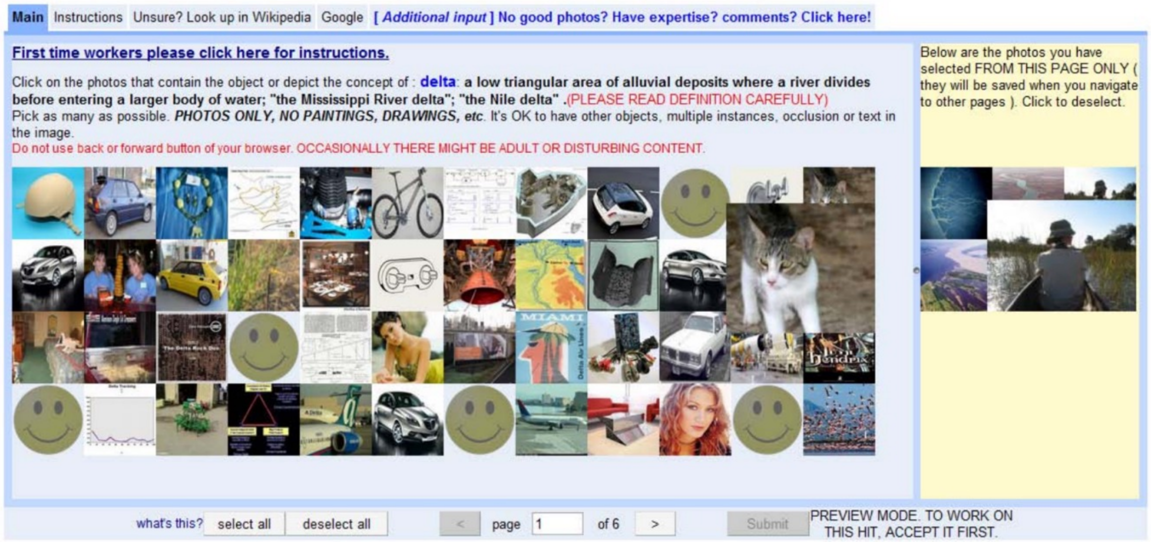
Success of ML for Multimedia

- Hardware accelerators
- Large-scale labeled multimedia data
  - Human-interpretable
  - Data collection and labeled can be separated



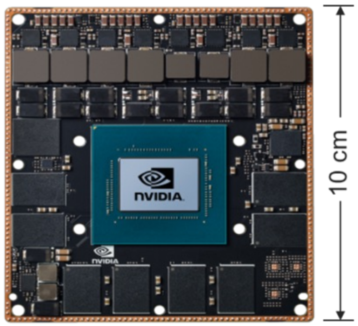
Example: ImageNet

- 14+ million images labeled by paid humans on Amazon Mechanical Turk (a crowdsourcing platform)

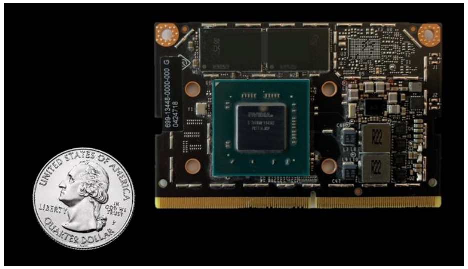


Now, how about AIoT?

- Hardware accelerators also available

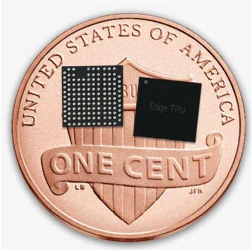


Embedded GPU:
NVIDIA Jetson AGX
8-core CPU, 512 tensor cores, 16GB mem



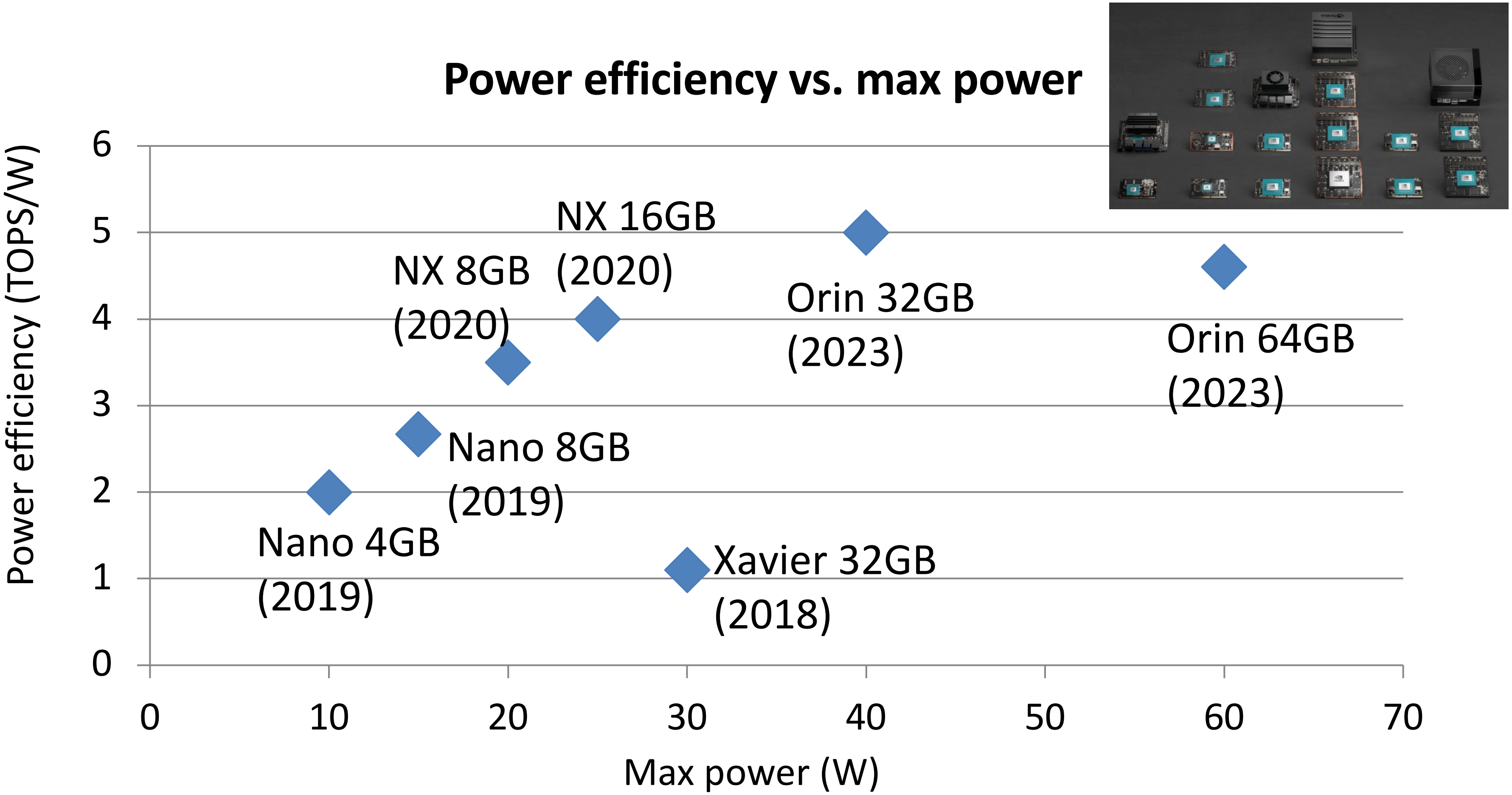
Embedded GPU:
NVIDIA Jetson Nano
4-core CPU, 128 tensor cores, 4GB mem

FPGA
(ICE65L)



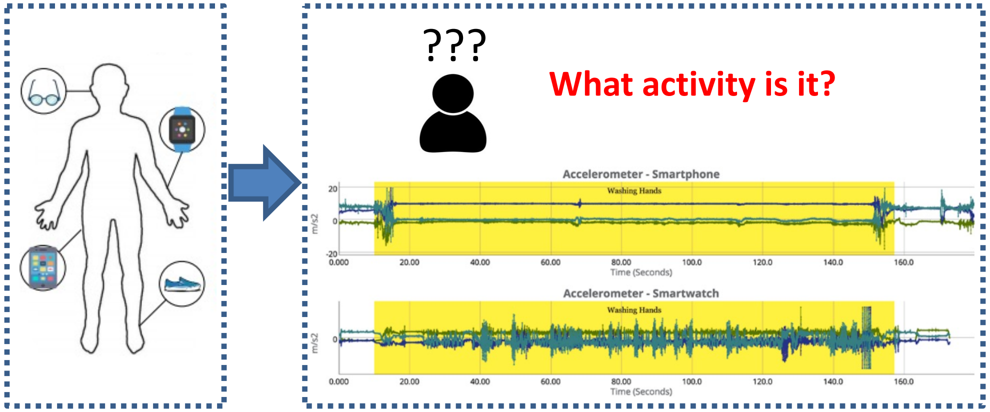
Edge TPU

Embedded GPUs



Multimedia Data vs. IoT Data

- Multimedia
  - Few modalities (sight, hearing)
  - Human-interpretable
- IoT
  - Many modalities (motion, magnetic, RSS, etc)
  - Human-uninterpretable

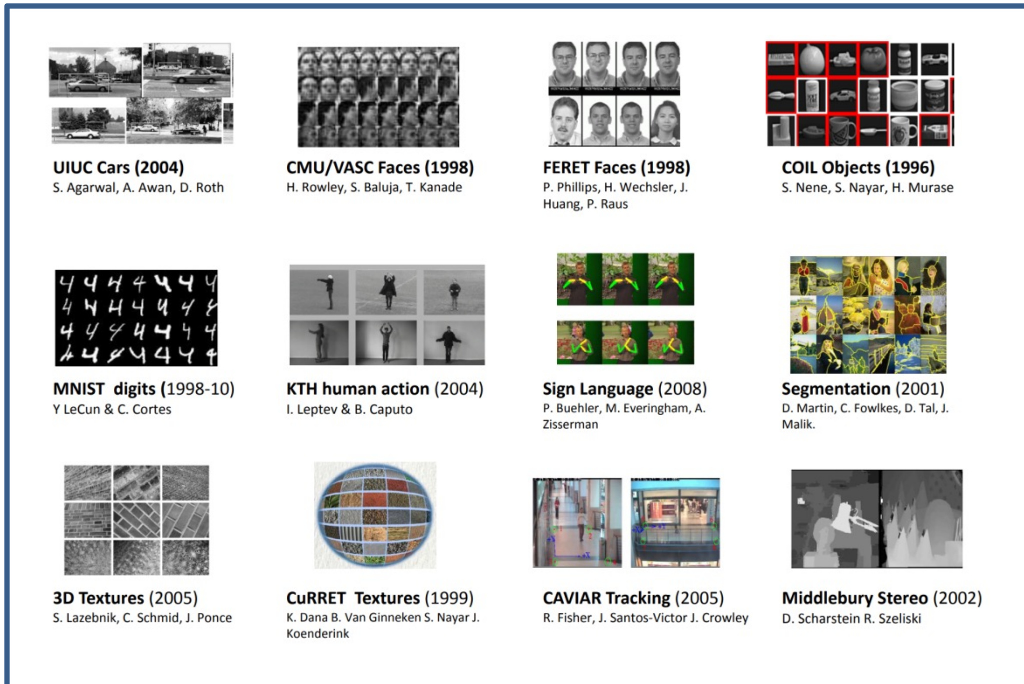


Data collection

Separated data labelling

Fragmented IoT Data

- Modality diversity leads to fragmentation



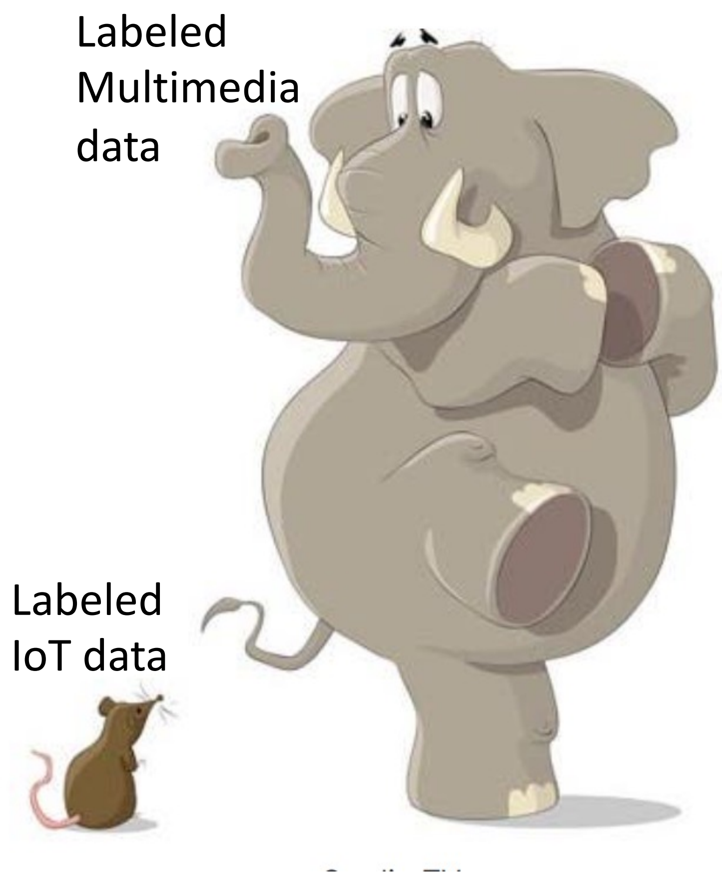
Nightmare for a single modality (before ImageNet)

Not to say diverse modalities

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Small IoT Data

- Inseparability of data collection and data labeling
  - IoT data must be labeled by the data collector (e.g., user) *in situ* immediately
  - Large-scale crowdsourced labeling impossible



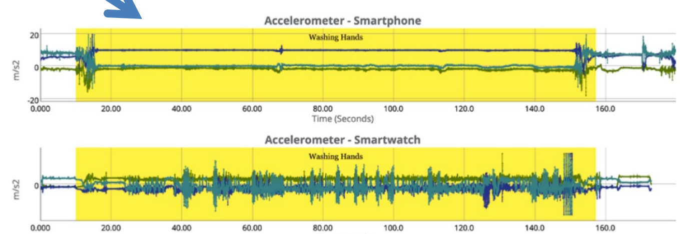
Prerequisite Bottleneck

2. Deep learning

Data collection → Data labeling → Feature engineering → Training & validation → Run-time inference

Human intelligence

Machine intelligence

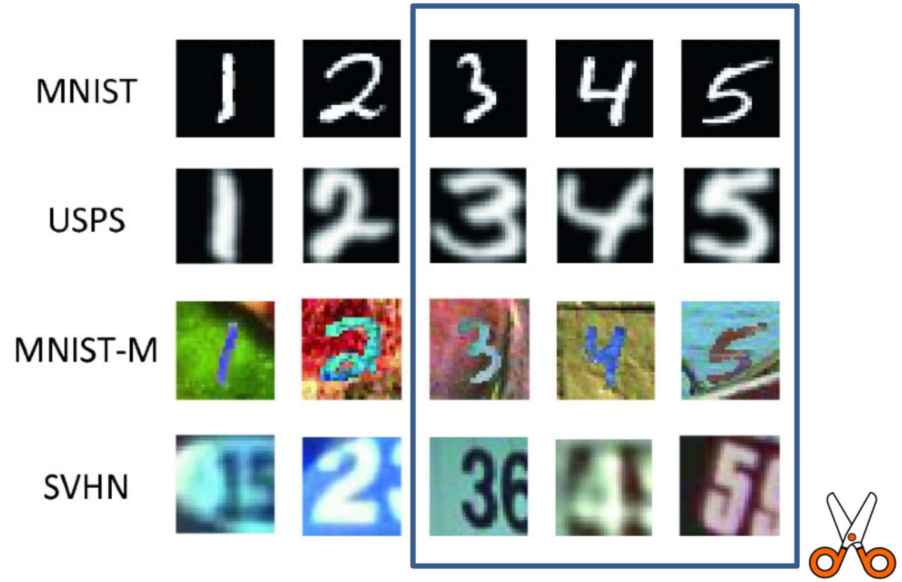


Human-interpretable data:
Human intelligence **can** be applied at large scales

Human-uninterpretable IoT data:
Human intelligence **cannot** be applied at large scales

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

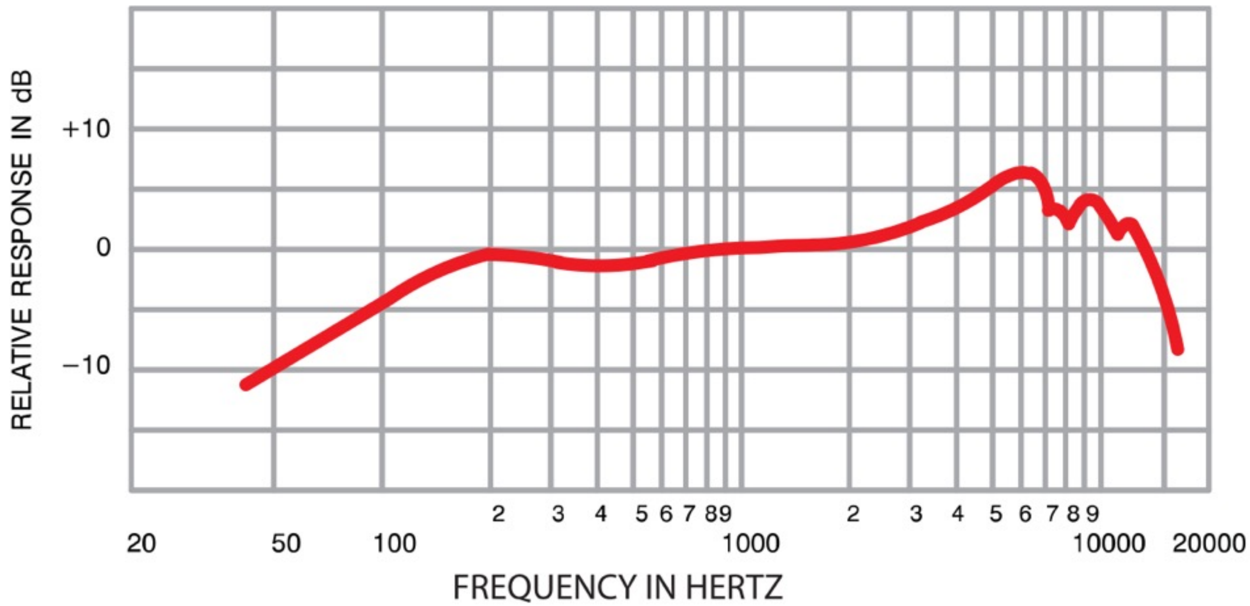
Small Data Problems



- Ubiquitous domain shifts
  - Heterogeneity in sensor, environment, user, etc
- Incomplete data
  - Limited class coverage

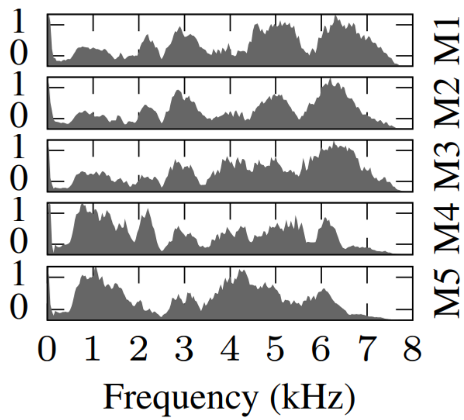
Ubiquitous Domain Shifts

- Sensor heterogeneity-induced domain shifts
  - Example: Microphones have different frequency response curves

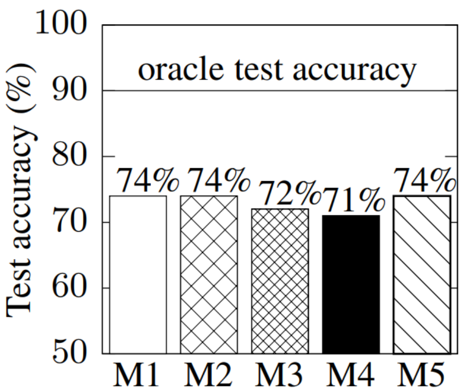


Ubiquitous Domain Shifts

- Microphone heterogeneity affects speech recognition

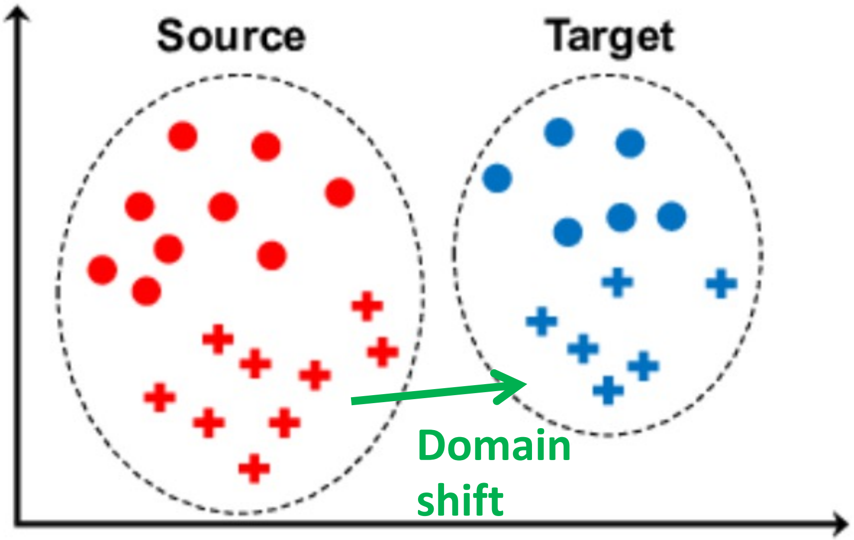


Five microphones' frequency responses



Their voice command recognition accuracy

Domain Shift and Solutions



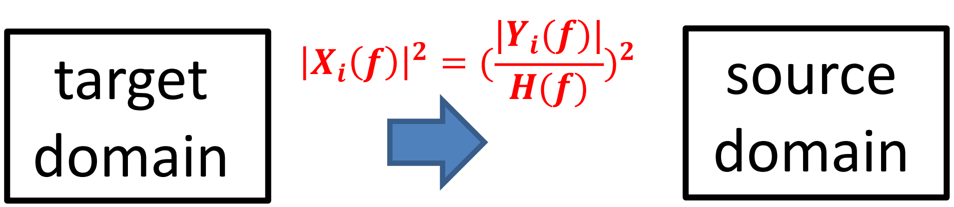
- Domains definition
  - Source domain: speech recognition model is trained for
  - Target domain: not seen during training
- Solutions
  - Sensor calibration / domain adaptation, data augmentation, domain-invariant features

Sensor Calibration

- Backward calibration: convert target-domain sensor data back to source-domain sensor data

This is not an AI based approach

- Forward calibration: source → target



Backward calibration

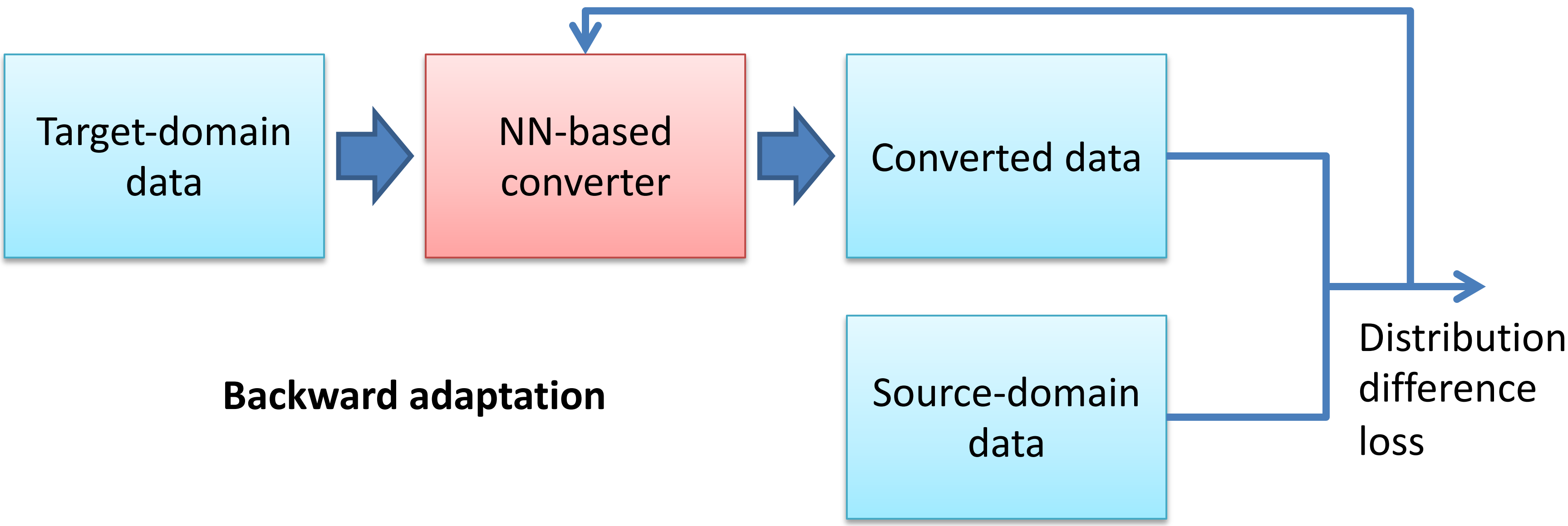
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Domain Adaptation

- Backward adaptation: Train a neural network to convert the target-domain data to source domain
- Forward adaptation: source → target

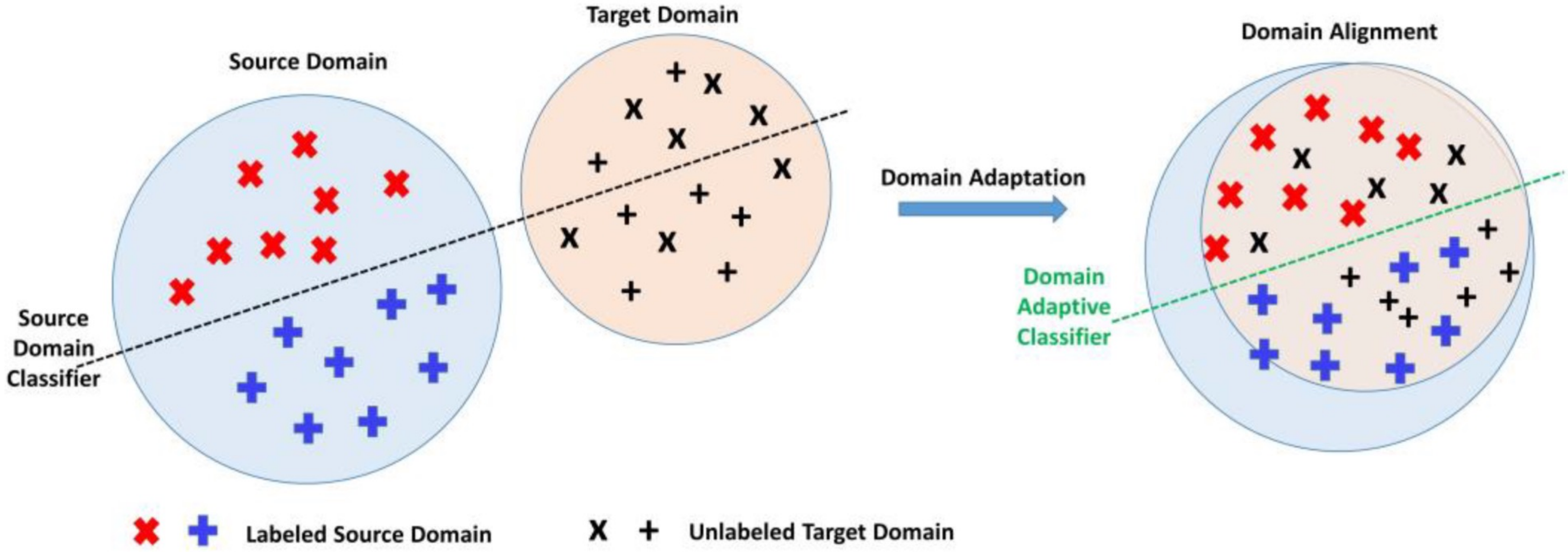
Cant train a simple regressor, since we dont know the mapping from target domain to source domain.

This scheme is more flexible, since only a comparioson between converted and source domain data, like similarity, but no sample to sample comparison.



Backward adaptation

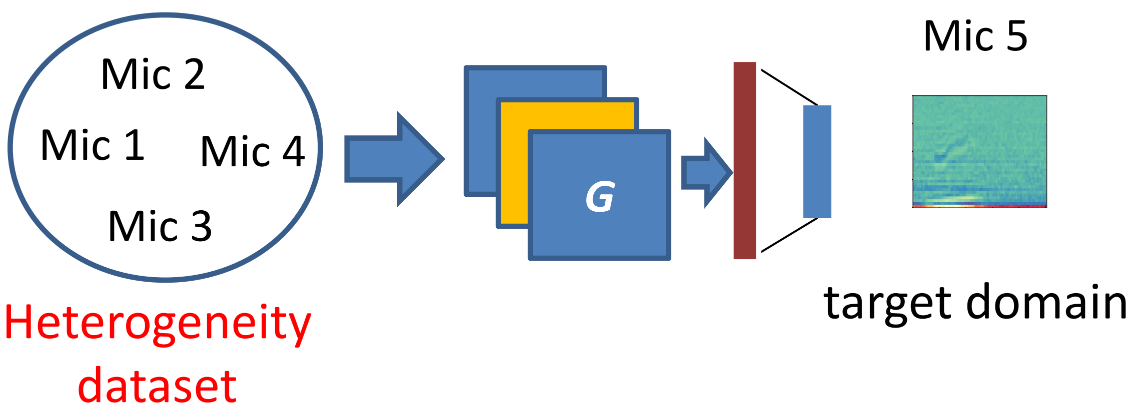
Effect of Domain Adaptation



Data Augmentation

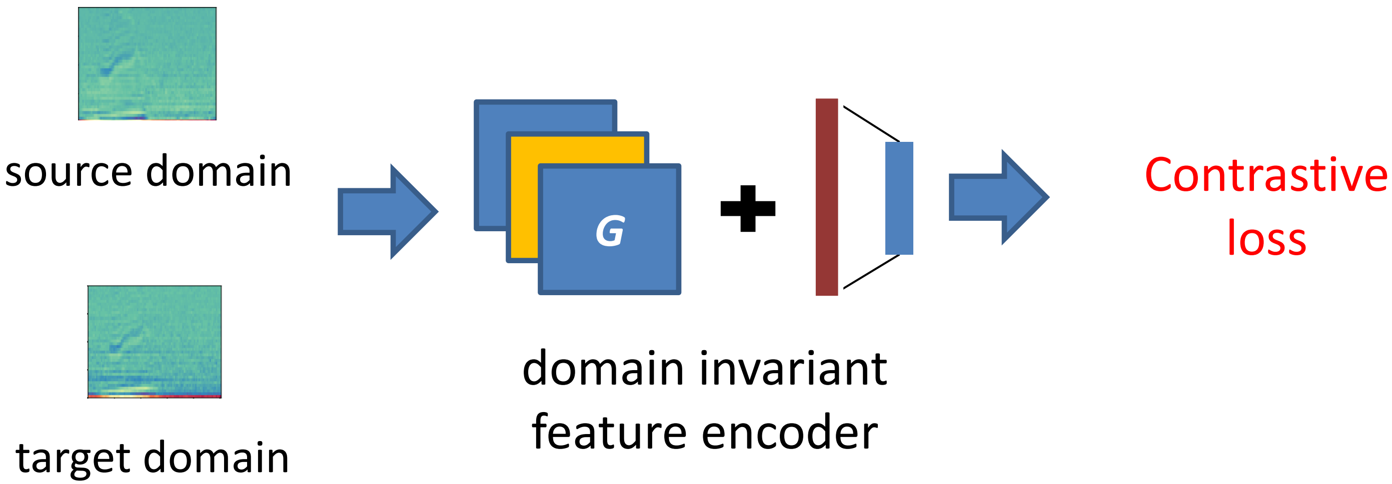
- Create sufficient varied versions of the source-domain data to hopefully cover the target domain

Not guranteed to work!

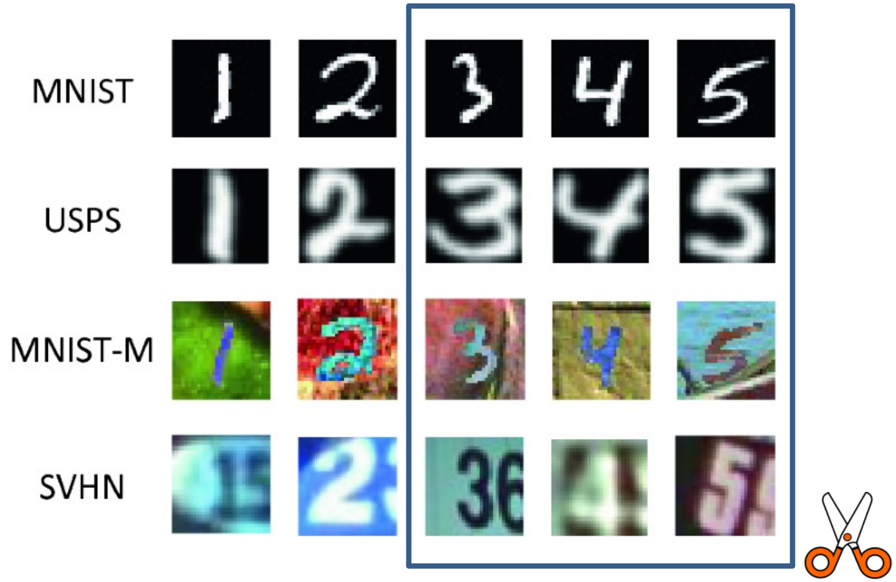


Domain-Invariant Features

- Train a common encoder with contrastive learning to produce domain-invariant features



Small Data Problems

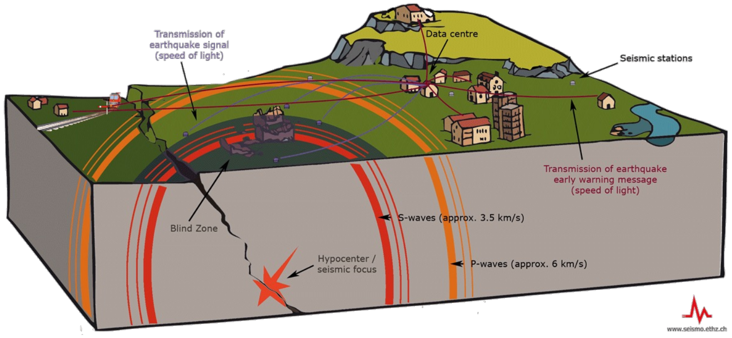


- Ubiquitous domain shifts
  - Heterogeneity in sensor, environment, user, etc
- Incomplete data
  - Limited class coverage

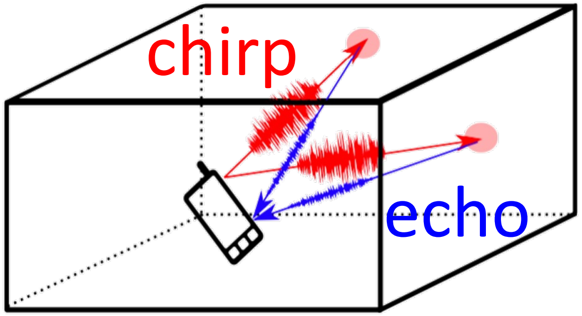
Incomplete Data

- Limited class coverage
  - Can only collect labeled location fingerprint data at limited locations

Seismic events localization



Echo-based smartphone sensing

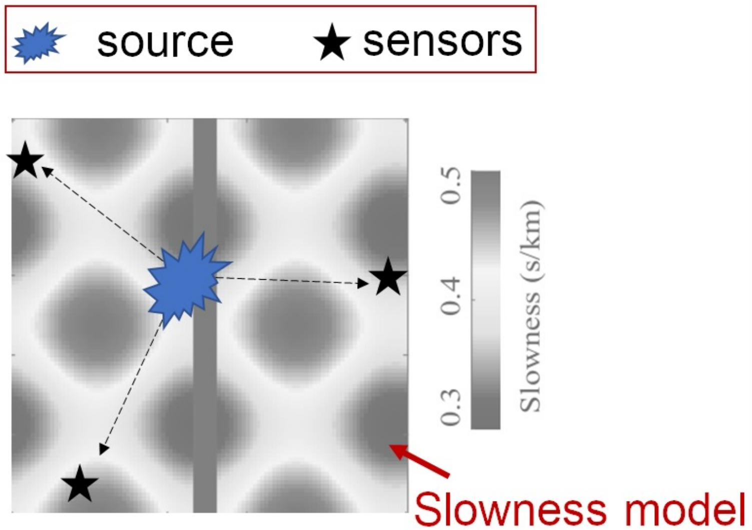


Potential Solutions

- Data augmentation
  - Assisted by physical model
  - Assisted by other sensing channels

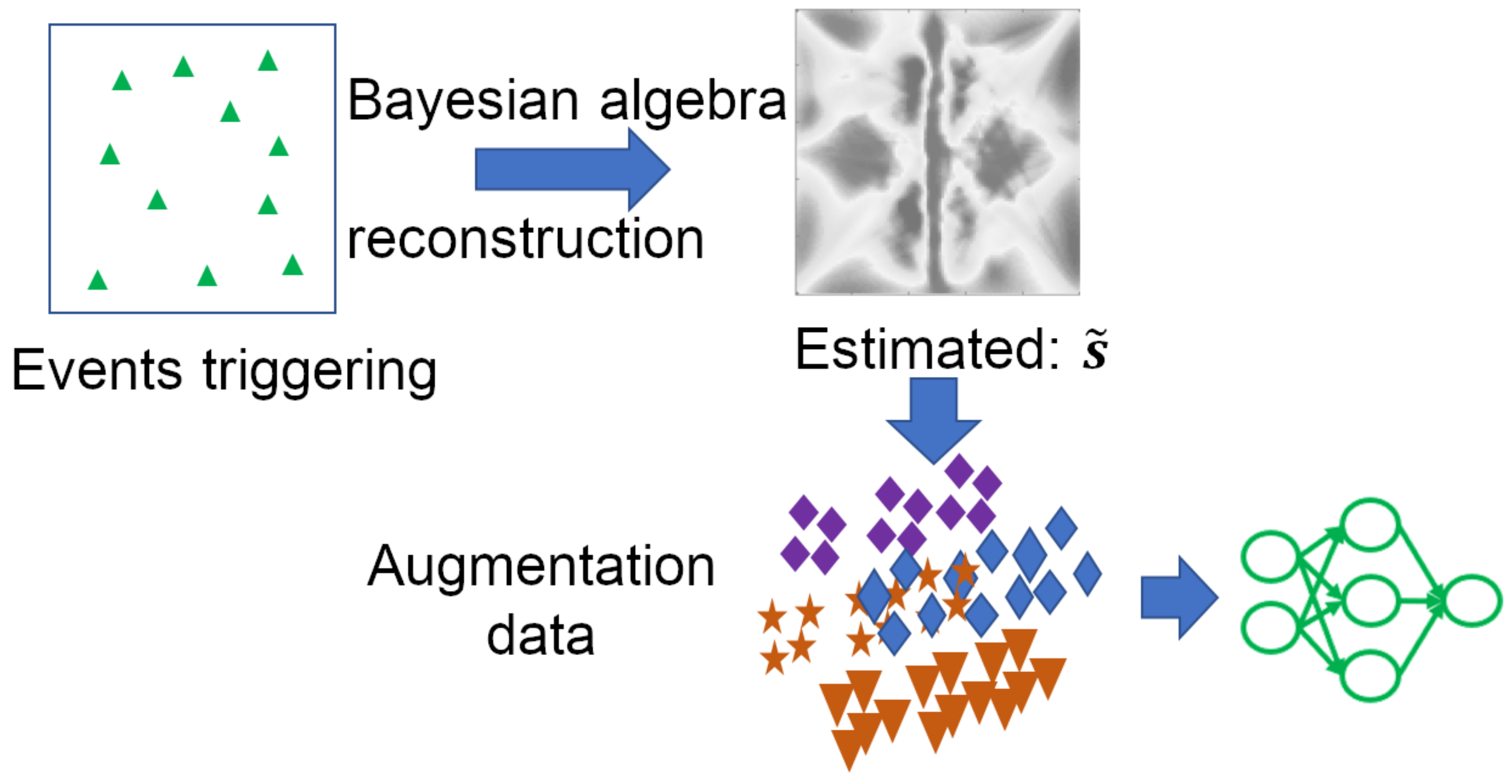
Assisted by Physical Model

- Seismic signal propagation in an uneven medium follows ray tracing with a slowness model



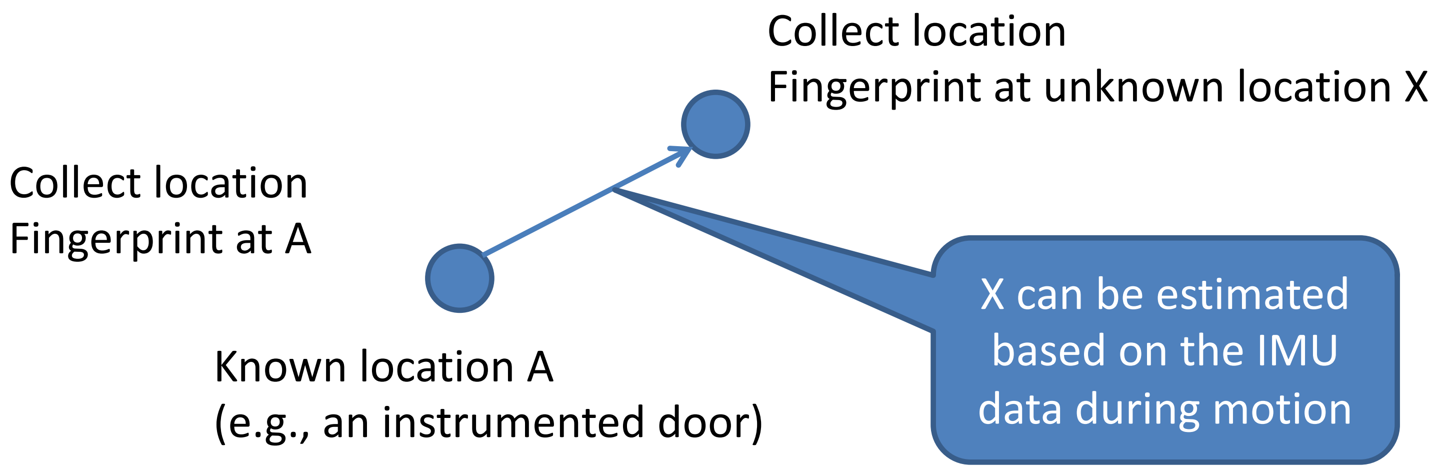
Assisted by Physical Model

- Augment data with physical model



Assisted by Other Sensing Channels

- Physics describing indoor sound / Wi-Fi signal propagation is sophisticated
  - Multi-path effect
  - Complex indoor arrangement

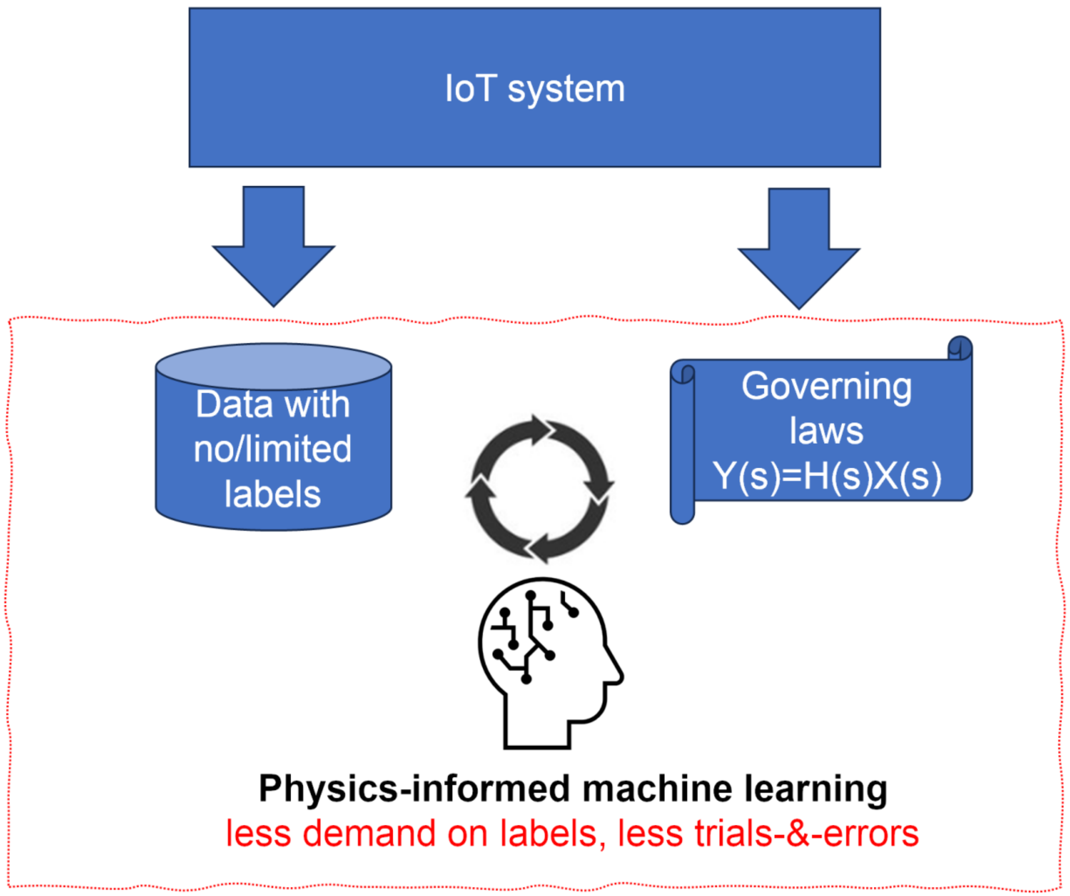


How AIoT Succeeds?

- Deeply customize the solution to exploit application characteristics
  - E.g., data augmentation assisted by physics or other sensing channels

- New machine learning approaches
  - Physics-informed machine learning (PIML)

PIML (2018-)


IoT system
Data with no/limited labels
Governing laws
Y(s)=H(s)X(s)
Physics-informed machine learning
less demand on labels, less trials-&-errors

Physics vs. Data

- Physical law is a nutshell representation of data

Tycho Brahe
(1546-1601)
Astronomical data

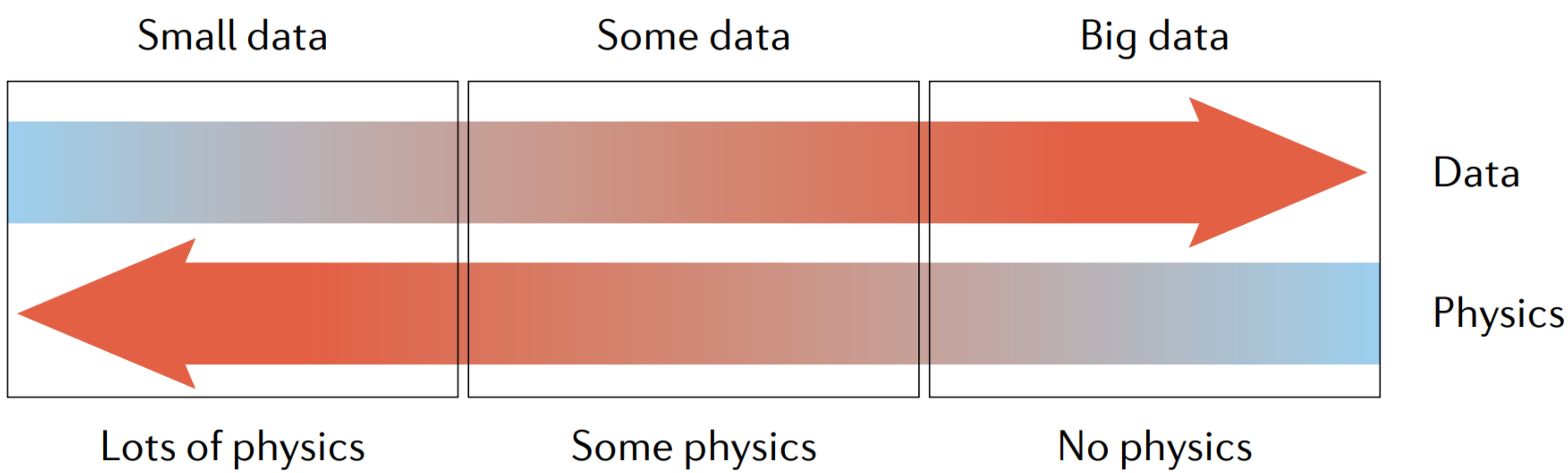
Summarize

Johannes Kepler
(1571-1630)
Laws of planetary motion

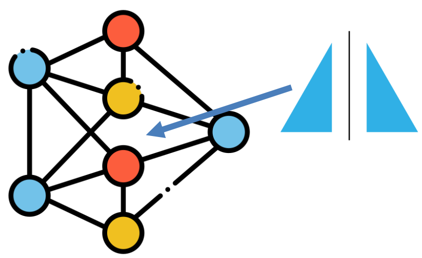
Hypothesize

Isaac Newton
(1643-1727)
Theory of universal gravitation

Physics or Data for ML?



3 Ways to Use Physics



Inductive bias

Neural network architecture reflects high-level properties of physical laws (e.g., symmetry)

Ad hoc (?)

Learning bias

Loss function includes deviations from analytical expressions of physical laws

Good for apps with clearly known laws

Observational bias

Augmented data to represent physical laws

Doesn't change NN design and learning algorithm

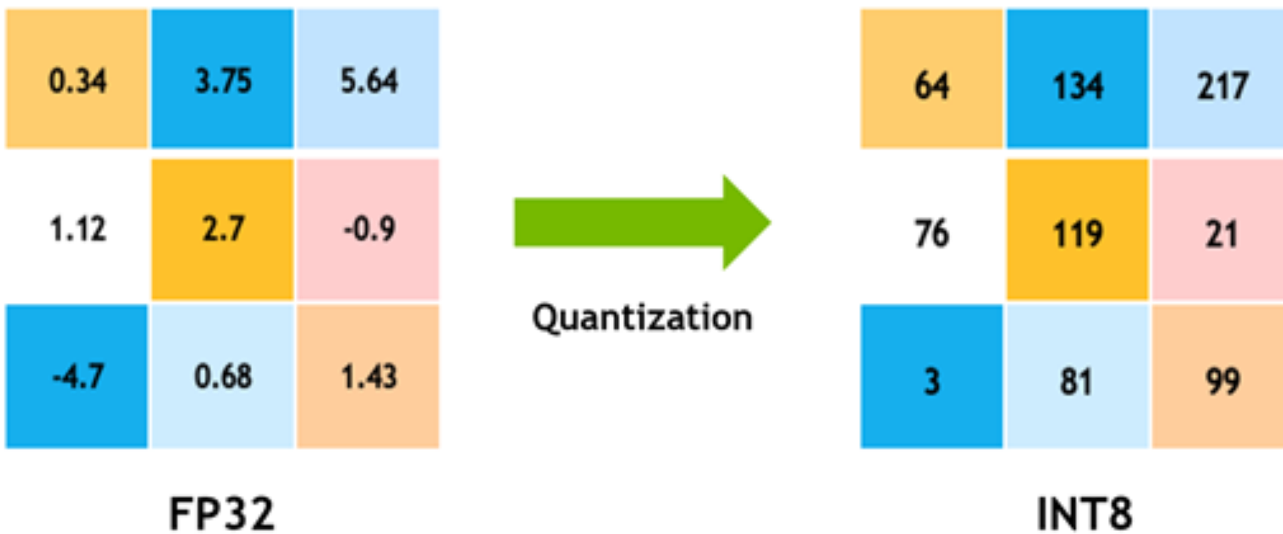
Other Challenges

- Resource constraints of edge devices
- Robustness
- Data privacy
- Carbon emissions
- Etc.

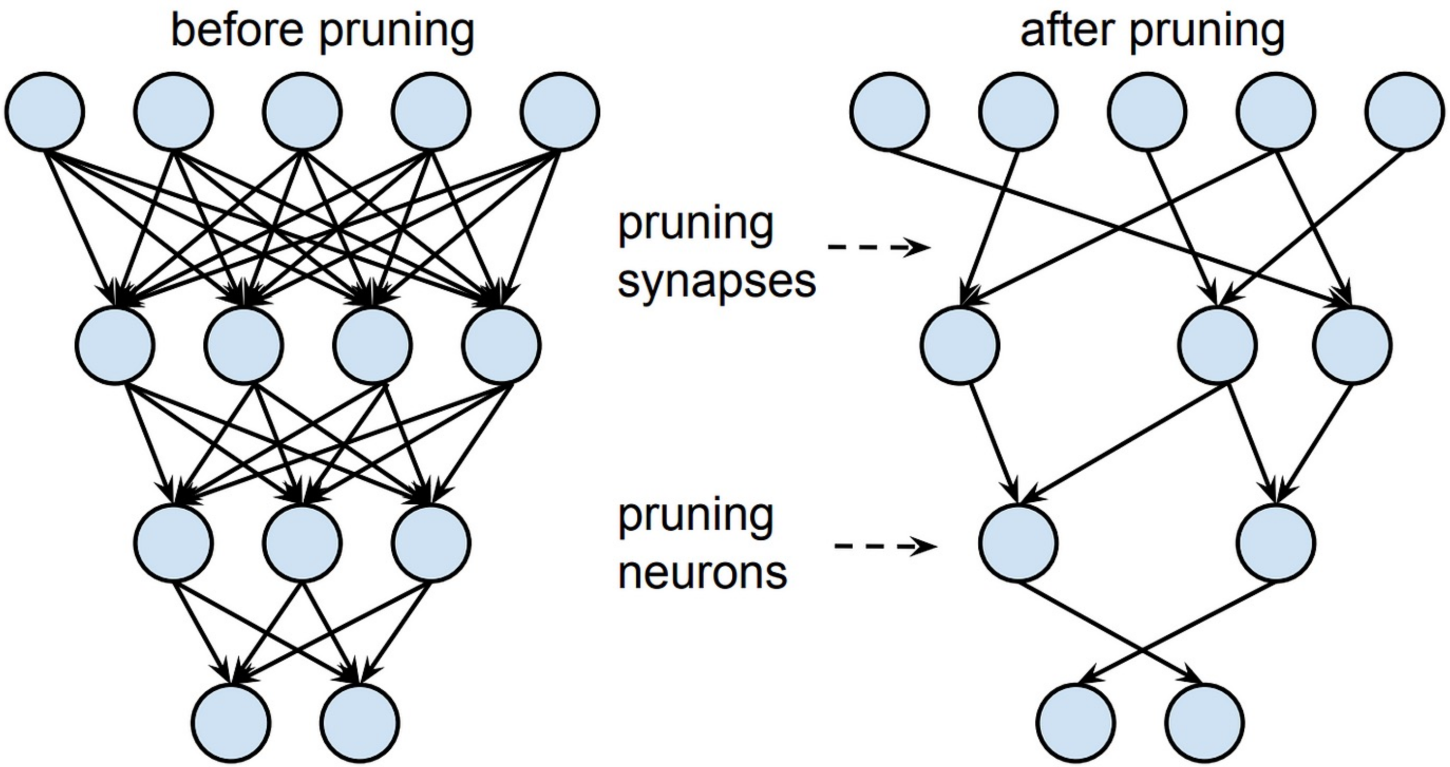
Resource Constraints

- Inference should be lightweight at edge
  - Quantization
  - Model pruning
  - Early exit
  - Cloud offloading
  - Etc

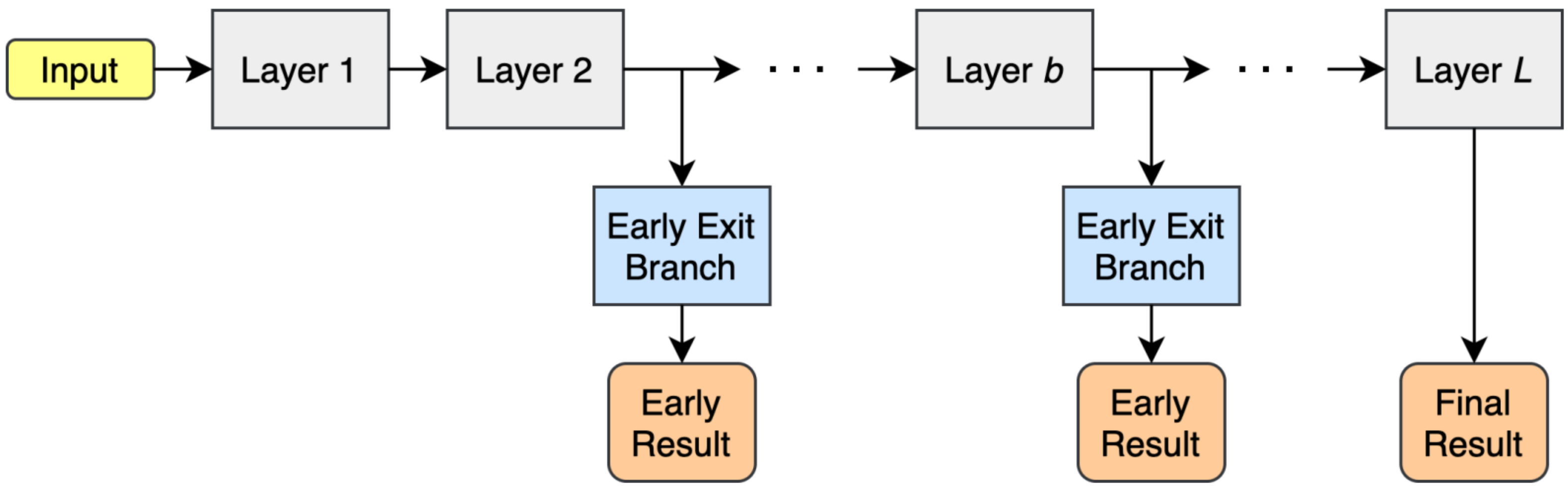
Resource Constraints: Quantization



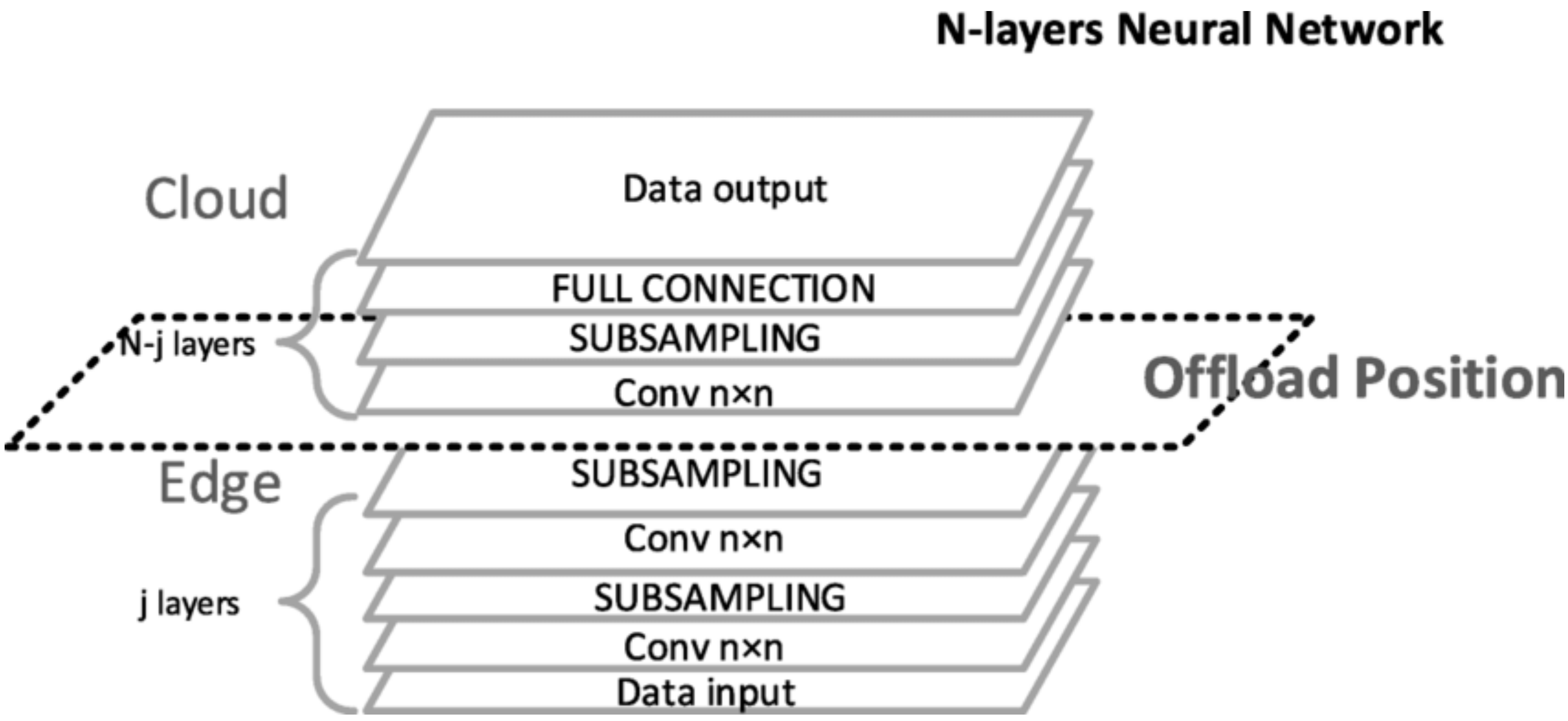
Resource Constraints: Model Pruning



Resource Constraints: Early Exits



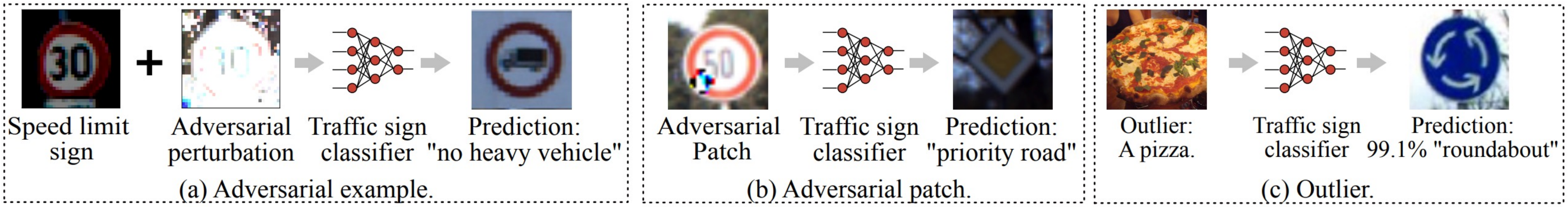
Resource Constraints: Cloud Offloading



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

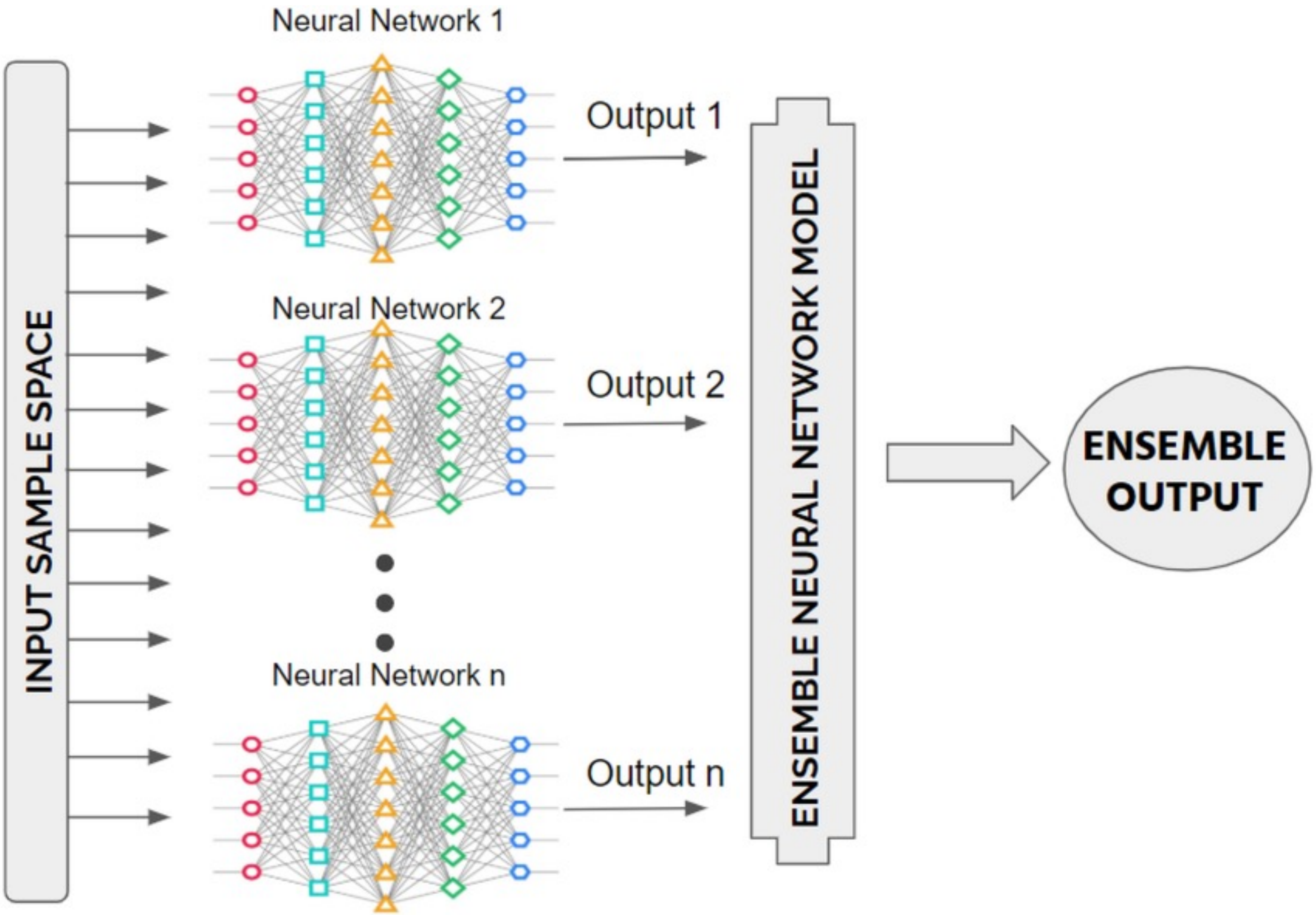
Robustness

- Out-of-distribution (OOD) inputs
- Adversarial examples
  - Small / invisible perturbations mislead the deep neural network
- Possible solutions
  - Deep ensemble
  - Monte Carlo dropout
  - Etc



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

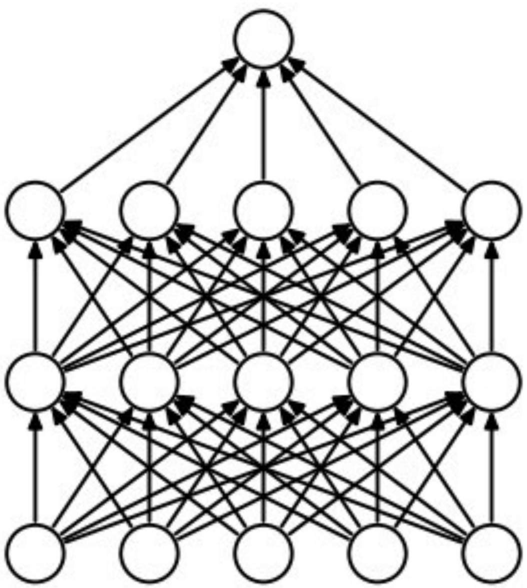
Robustness: Deep Ensemble



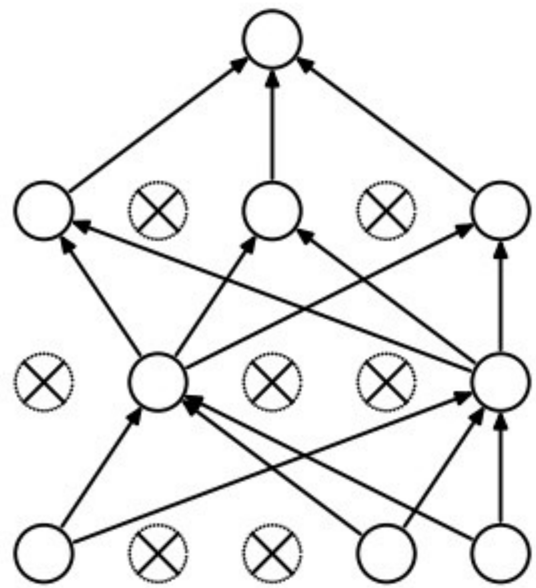
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Robustness: MC Dropout

- Widely used in training to avoid overfitting
- Applied for inference: increase robustness at the cost of some accuracy loss



(a) Standard Neural Net



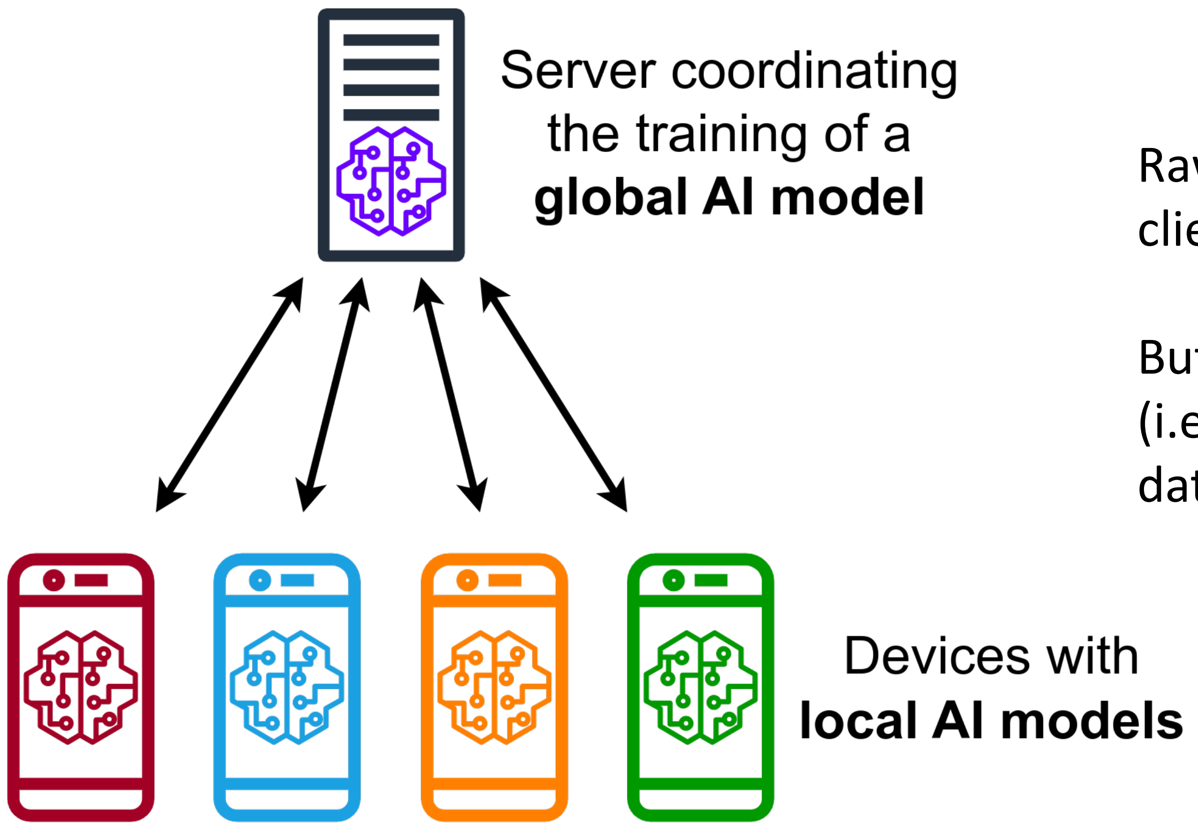
(b) After applying dropout.

Data Privacy

- Problem
  - If training or inference is in cloud, rendering of privacy-sensitive raw data generated by IoT devices
  - Other trickier information leakage, e.g., private attribute leakage

- Possible solutions
  - Federated learning
  - Homomorphic encryption
  - Lightweight data masking

Data Privacy: Federated Learning

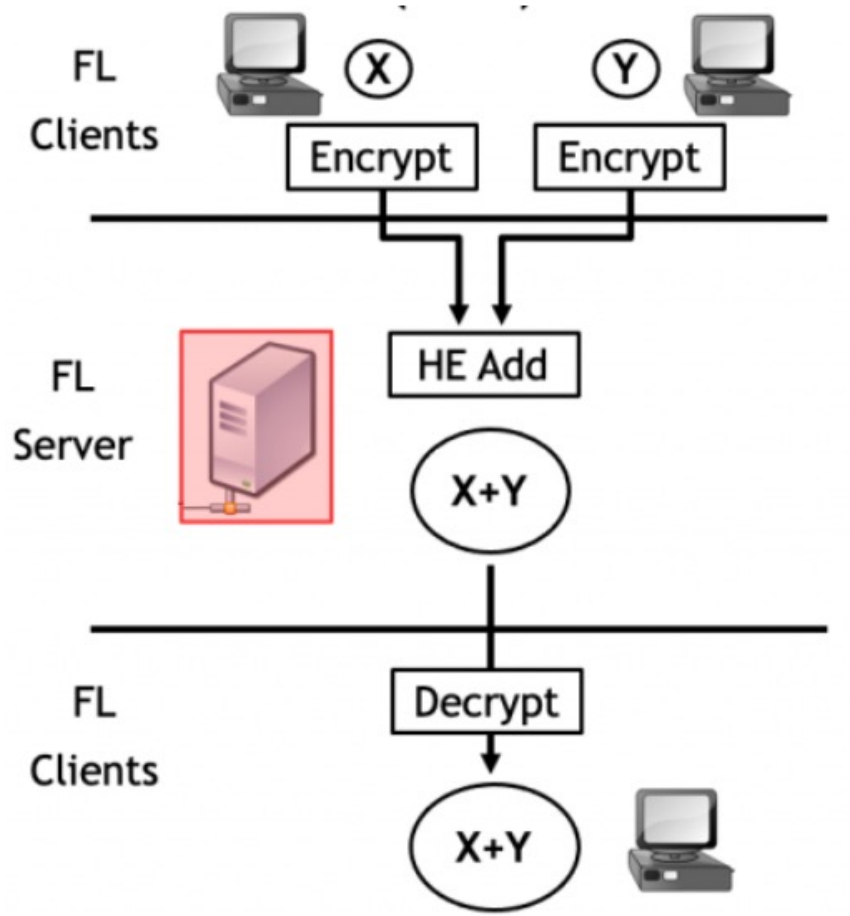


Raw data does not leave clients

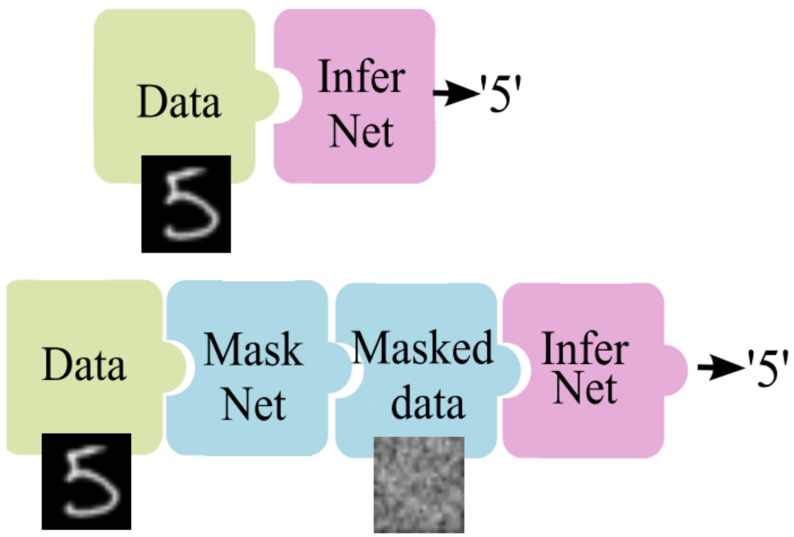
But, exchanged model (i.e., a representation of data) may leak privacy

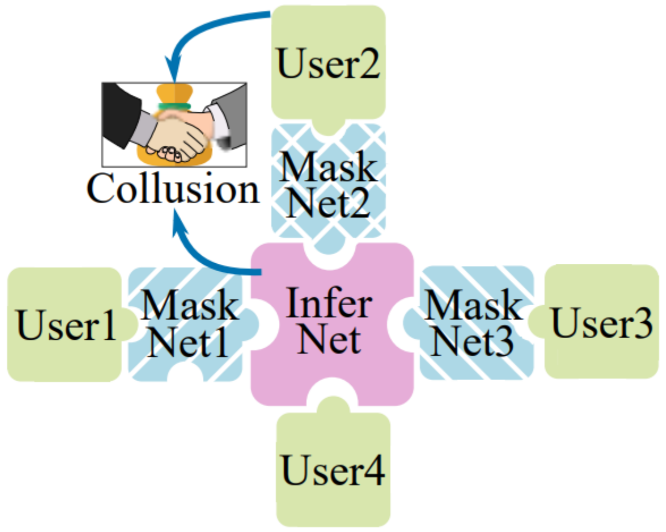
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Data Privacy: Homomorphic Encryption



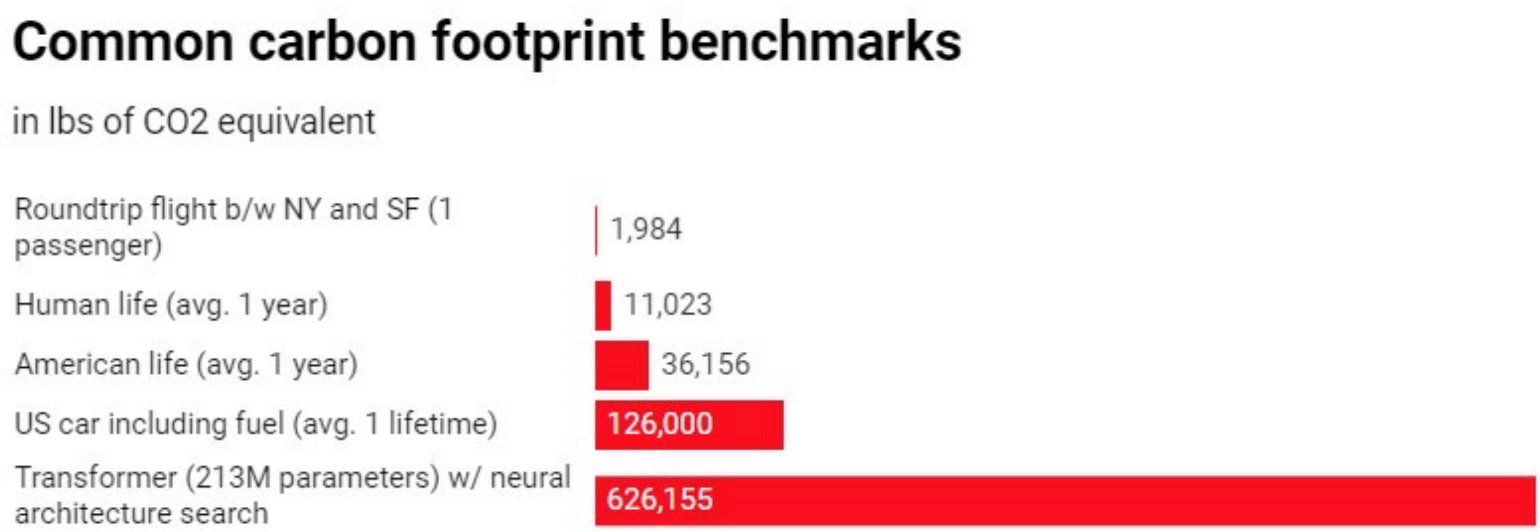
Data Privacy: Data Masking





Carbon Emissions

- "Training a single AI model can emit as much carbon as five cars in their lifetimes"
  -- MIT Technology Review, June 2019



Carbon Emissions

- "Federated learning can emit up to two orders of magnitude more carbon than centralized machine learning."
  -- Qiu et al. A first look into the carbon footprint of federated learning. arXiv:2102.07627, 2021

- Open challenge with no one-size-fits-all solution

SLAM

SLAM

- Mapping: integrating information gathered by robot sensors into a spatial representation, usually over multiple time steps
  "What does the world look like?"

- Localization: estimating the pose of robot in the map
  "Where am I?"

- SLAM: building map while localizing robot within the map

An Example Result

SLAM result by a drone with IMU + LiDAR (GPS is not used)

Red dots: drone locations
3D structures: stitched point clouds sensed by LiDAR

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Sensors for SLAM

- Sensors for robot state
  - IMU (accelerometer, gyro)

- Sensors for environment
  - Camera, LiDAR, Wi-Fi, geomagnetic, etc

Dead Reckoning

- Reconstruct movement trajectory from IMU readings

- "Deak reckoning is a navigational term. It means you're picking a course based solely on your last known position and that becomes quite the metaphor not only for Ethan, but several characters."
-- Movie director

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

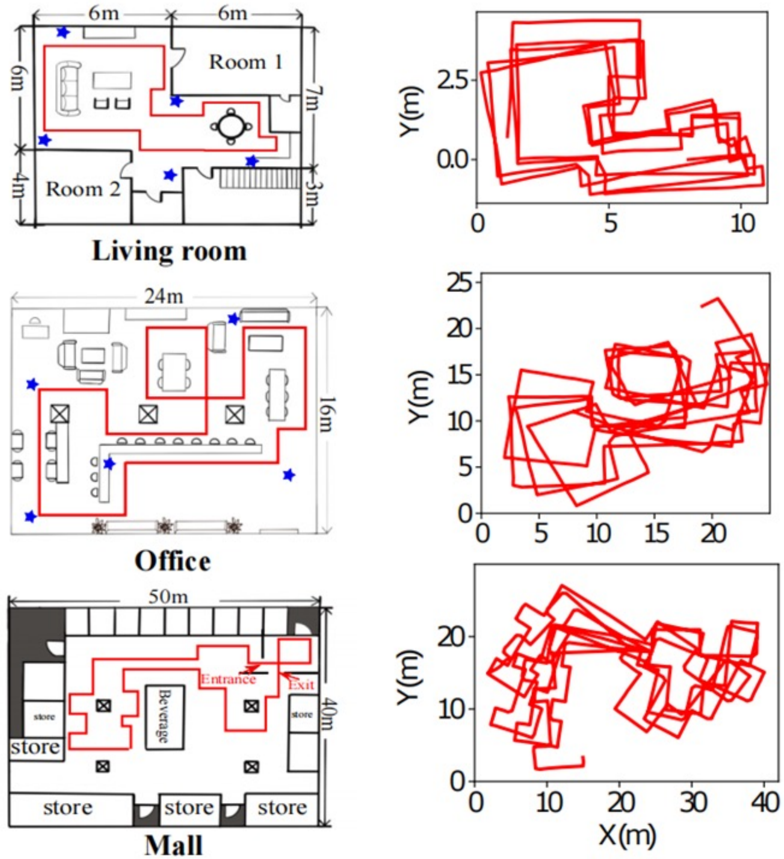
Error Accumulation

- Example: 1D dead reckoning
  - Assume accelerometer has a constant error

$$s(T) = \iint_{t\in[0,T]} \left(a(t) + e\right)dt = \iint_{t\in[0,T]} a(t)dt + eT^2$$

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

2D Dead Reckoning



(a) Floor plans (b) IMU

Error Rectification

- Use environment sensing as feedback to rectify dead reckoning errors

- Loop closure (simplest feedback)
  – Robot visits the same place
  – Assumption: Environment sensor produces same/similar reading
  – Detected by analyzing environment sensor's measurement trace

Optimization Formulation

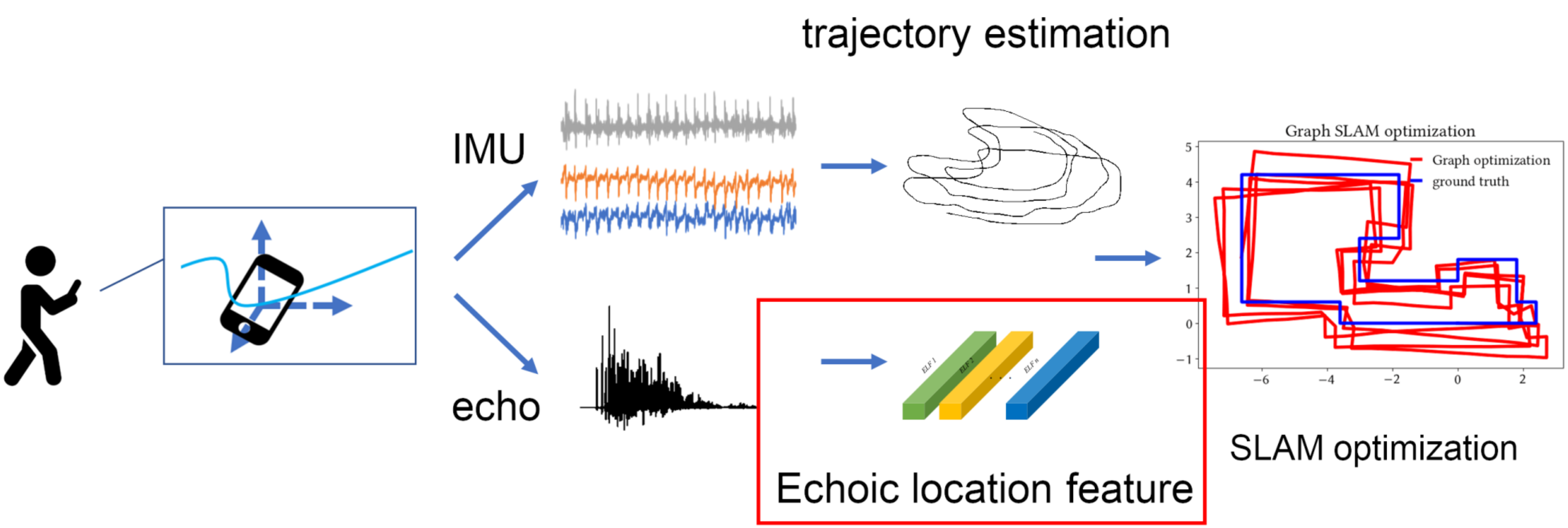
- Find a trajectory $X = (x_1, x_2, \ldots, x_N)$

Dead reckoning algorithm predicting xj from xi and IMU trace spanning from i to j

$$X^* = \arg\min_X \sum_{\forall i \in [1,\ldots,N-1]} \| f(x_i, u_{i,i+1}) - x_{i+1} \| + \sum_{\forall <i,j> \in C} \| f(x_i, u_{i,j}) - x_j \|$$

Set of detected loop closures

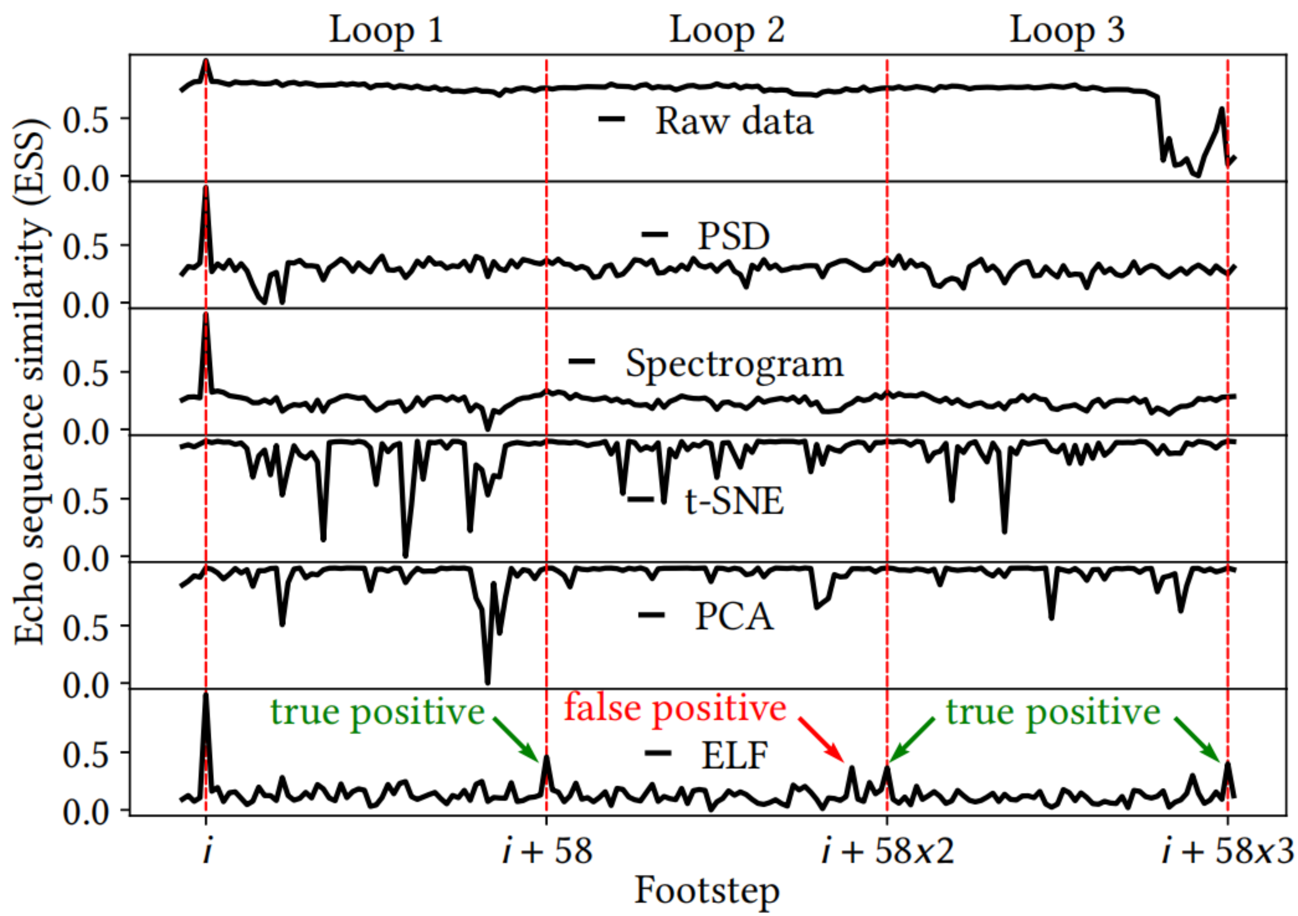
Acoustic Echo as Feedback



How to design an effective feature?

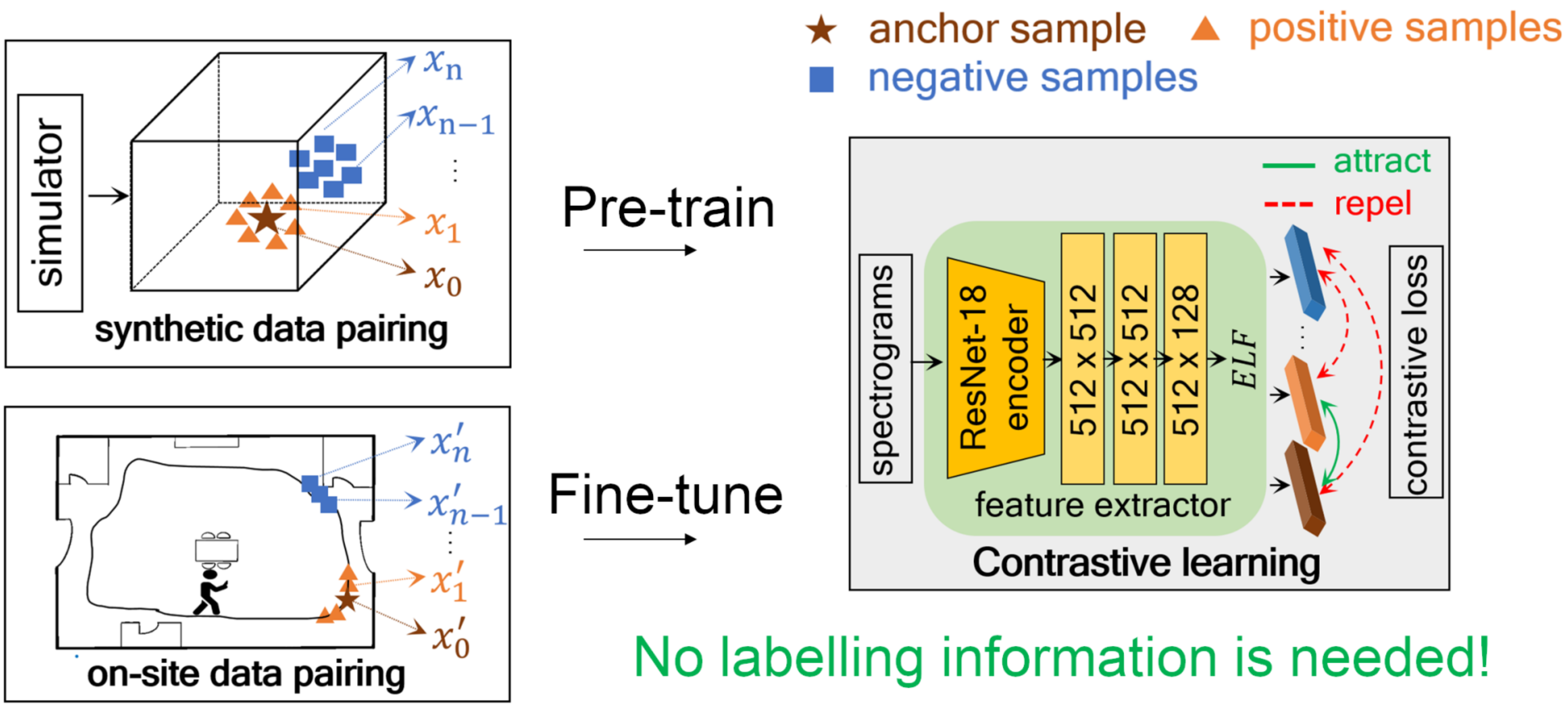
Can't we use the deep model presented in Lecture 3 to detect loop closures?
Ans: SLAM replaces labeled training data collection. Now, we don't have the deep model.

Common Features Ineffective



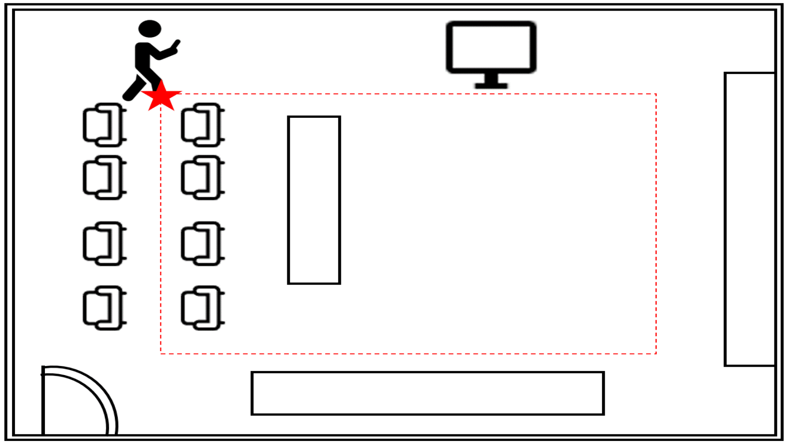
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

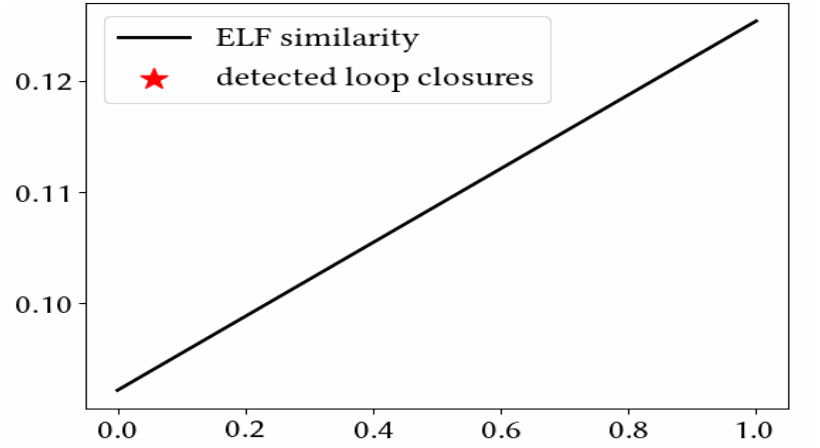
Contrastive Feature Learning



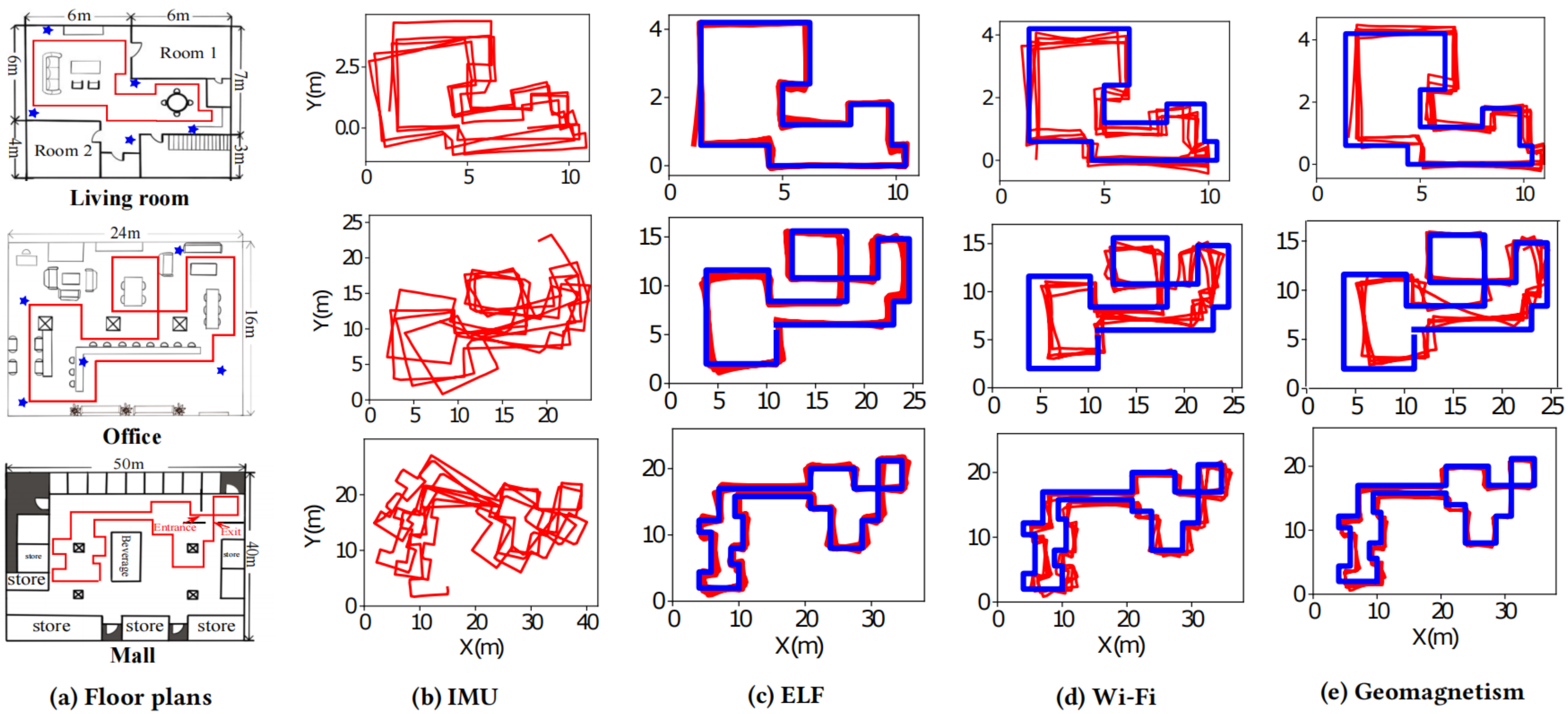
No labelling information is needed!

Loop Closure Detection





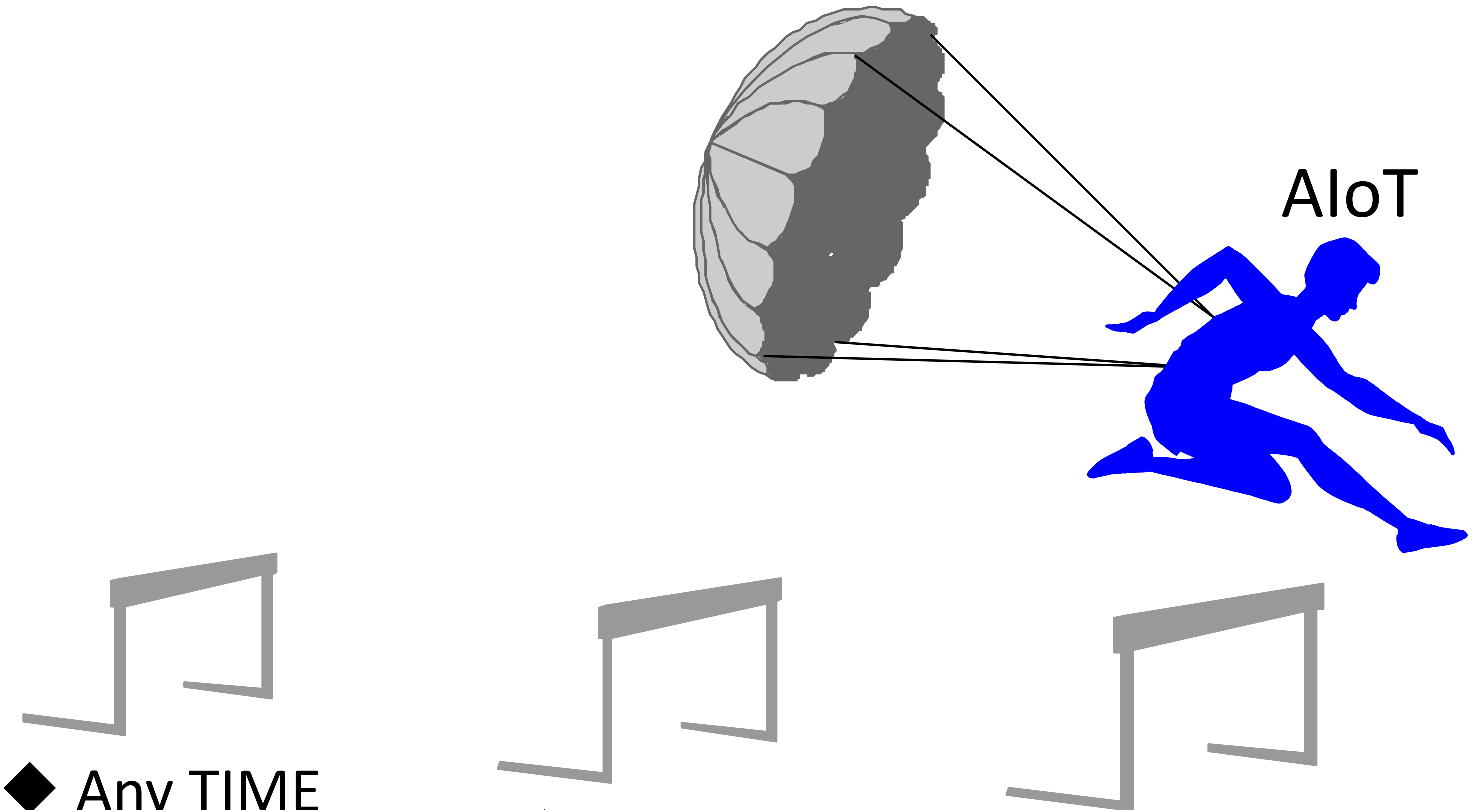
SLAM Performance



Learning Objectives

- What is AIoT?
- Challenges faced by AIoT & solutions
- SLAM

Road is Difficult, Future is Bright



- Any TIME
- Any PLACE
- Any THING

AI6128

- Urban sensing
  - Urban IoT
  - Sensors, sensor networks
  - Time and location acquisition
  - Cloud support
  - AIoT

To Learn More

- NTU IoT Research Group (https://ntuiot.xyz)
  - IoT sensing systems/applications
  - AIoT non-functional aspects (security, privacy, etc)