Towards an Intelligent Edge: Wireless Communication Meets Machine Learning

Hai-Liang Zhao
hliangzhao97@gmail.com

October 25, 2018

This slide can be downloaded at ▸ *Link*.

Outline

Outline

Outline

Outline

About the authors

- G. Zhu, D. Liu, Y, Du, C, You and K. Huang are with the Dept. of Electrical and Electronic Engineering at **the University of Hong Kong**, Hong Kong.
  J. Zhang is with the Dept. of Electronic and Computer Engineering at **the Hong Kong University of Science and Technology**, Hong Kong.
- This paper was submitted to IEEE journals in 2018.9. (No publish info found.) The preprinted version can be found at ▸ Link.

Development of the 'edge'

A new research area, called ***edge learning***, emerges, which crosses and revolutionzes two disciplines: wireless communication and machine learning.

The key motivation of **pushing learning towards the edge** is to allow rapid access to the enormous real-time data generated by the edge devices for **fast AI-model training**[1], which in turn endows on the devices human-like intelligence to respond to **real-time events**.

[1] Notice that here we are concerned about **training** an AI model at the edge, not just **inferring**!

Critical points for Edge Learning

Δ How to fully exploit the distributed data† in AI-model training without incurring excessive communication latency‡?

Two critical and coupled aspects:

- learning from **distributed** data [2]
- **reducing communication latency** between edge server and devices

[2] Can we design a method to divide training tasks by meta-information to imrpove the collabotation among edge servers for better orgnization of data?

Cloud, edge, and devices

The coexistence of cloud, edge and on-device learning paradigms has led to a layered architecture for in-network machine learning.

Unique strengths of the architecture

Different layers process different data processing and storage capabilities, and cater for different types of learning applications with distinct latency and bandwidth requirements.

- achieve the tradeoff between the AI model complexity and the model-training speed
- avoid excessive propagation delay and network congestion because of its proximity to data sources
- higer learning accuracy by aggregating distributed data from many devices

Paradigm shift on *communication**

Conventional philosophy in traditional wireless communication

The traditional design objectives of wireless communications, i.e., ***coomunication reliablity*** and ***data-rate maximization***, do not directly match that of edge learning.

$\Downarrow$ A conceptual change

Learning-driven communication

The ***coupling*** between communication and learning in edge learning systems should be exploited.

Components of Learning-driven Communication

This paradigm includes three key communication aspects.

▷ ***multiple access, resource allocation, signal encoding.***

Principle of this paradigm for Fast Intelligence Acquisition

Efficiently transmit data or learning-relevant information to speed up and improve AI-model training at edge serevrs.

Specific research directions

Conventional communication vs. *Learning-driven communication* is summerized in the following table.

Commun. Tech.	Item	Conventional Commun.	Learning-Driven Commun.
Multiple access (Section II)	Target	Decoupling messages from users	Computing func. of distributed data
	Case study	OFDMA	Model-update averaging by AirComp
Resource Allocation (Section III)	Target	Maximize sum-rate or reliability	Fast intelligence acquisition
	Case study	Reliability-based retransmission	Importance-aware retransmission
Signal Encoding (Section IV)	Target	Optimal tradeoffs between rate and distortion/reliability	Latency minimization while preserving the learning accuracy
	Case study	Quantization, adaptive modulation and polar code	Grassmann analog encoding

Outline

Motivation and principle

Upload the orivacy-sentive and large-in-quantity training data from devices to an edge server for **centralized model training** will lead to:

- ▷ a privacy concern;
- ▷ prohibitive cost in communication.

Therefore, we have ⇒

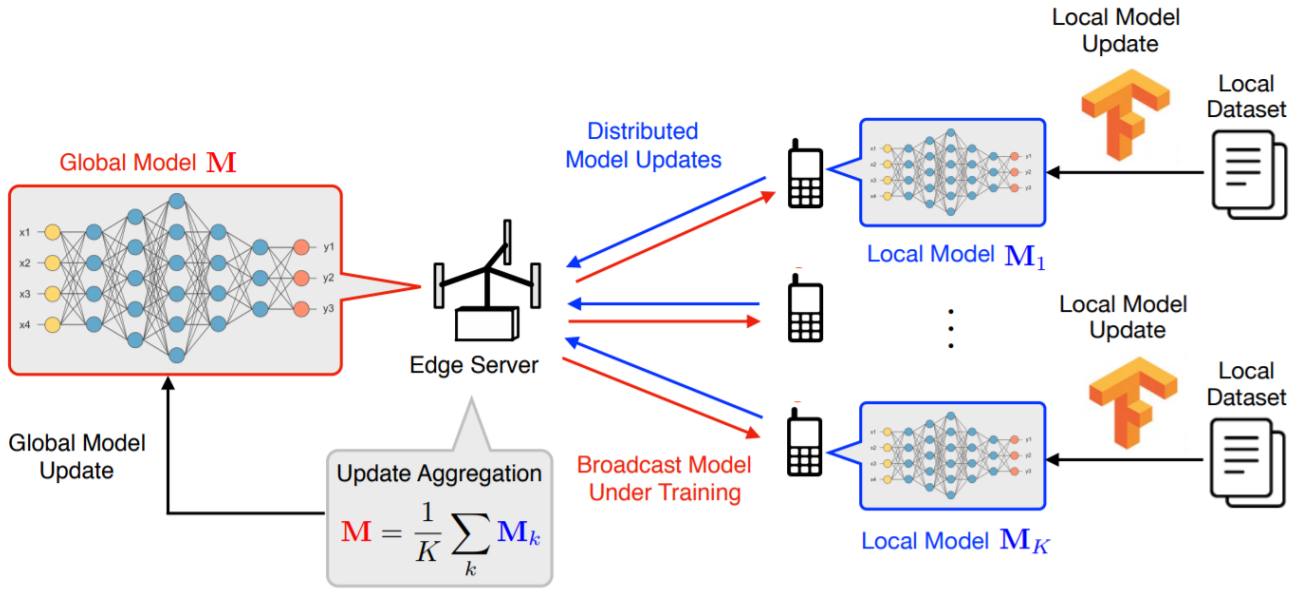
Federated Learning

Federated Learning features ***distributed learning***† at edge devices and ***model-update aggregation***‡ at an edge server.

Schematic diagram for *Federated Learning*

Two phases:

- Aggregate distributed model updates over a multi-access channel and *apply their average*• to update the AI-model
- Broadcast the model under training to allow edge devices to continously *refine their individual versions*• of the model



How to realize *Learning-driven multi-access?*

Existing multiple access technologies such as *orthogonal frequency-division* (OFDMA) and *code division* (CDMA) are purely **rate-driven** and fail in actual learning task when transmitting millions to billions of paras.

We should exploit the computation of aggregating function (e.g., averaging or finding the maximum) over distributed data!

- the **average** of model updates rather than their individual values is required
- the simultaneously transmitted analog-waves by different devices are automatically **superposed** at the receiver but weighted by the channel coeffs

Over-the-air computation (AirComp)

The AirComp[3] tech can dramatically reduce the multiple access latency by a factor equal to the number of users.

What AirComp can do?

By using analog-linear modulation and pre-channel-compensation at the transmitter, the "interference" caused by concurrent data transmission can be exploited for fast data aggregation.

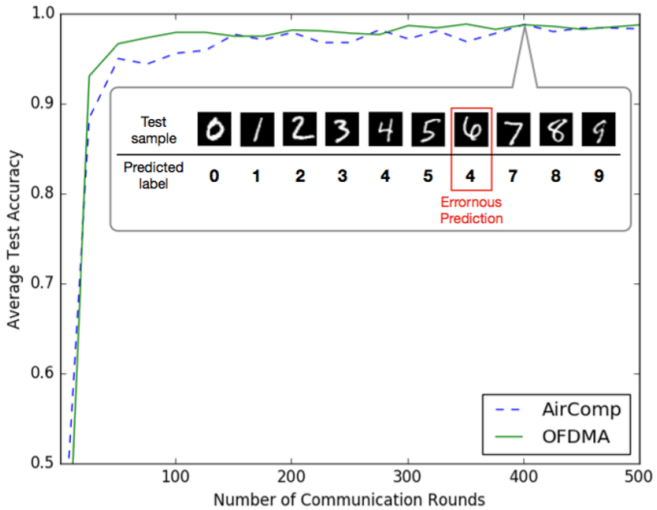
[3] How AirComp works?
In-depth understanding of it will come soon!

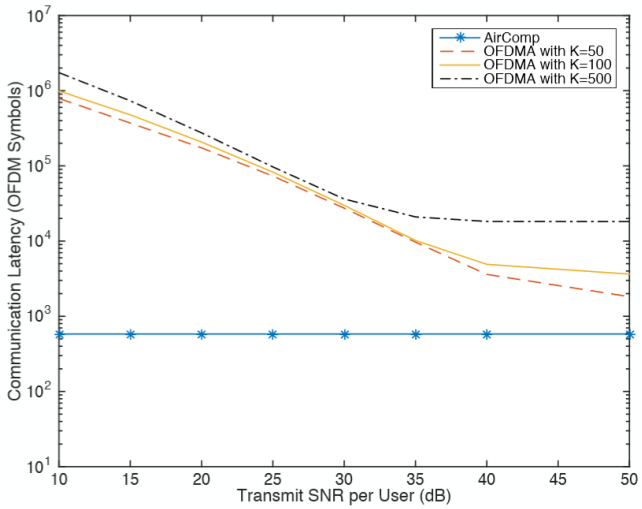
Case study: Handwritten-digit recognition with MNIST

A **shared broadband channel** consisting of $N_s = 1000$ orthogonal* sub channels:

▷ **conventional OFDMA**. 1000 sub channels are evenly allocated to K edge devices. Model average is performed after all local models are reliably received.

▷ **AirComp**. Every device uses the full bandwidth so as to exploit the "interference" for direct model averaging over the air.





Research opportunities

Where to extension?

- **Robust learning with imperfect AirComp.** Pre-equalization of the channel at the transmitting devices is hard to realize, thus the aggregated data is distorted. Can we design a robust algorithm for imperfect AirComp?
- **Asynchronous AirComp.** Strict synchronization between all the participating edge devices is hard to realize.
- **Generalization to other edge-learning architectures.** The AirComp do not have the *generalization ability*. What about more sophisticated computation than simple averaging?

All the listed shortcomes lie in the unmatured technology - AirComp.

Motivation and principle

Existing methods of *radio-resource management* (RRM) are designed to **maximize the efficiency of spectrum utilization** by allocating the scarce radio resources , i.e., power, frequency band, and access time.

Some msgs tend to be **more valuable** than others for training an AI model! Therefore, we can bring in ⇒

Active Learning

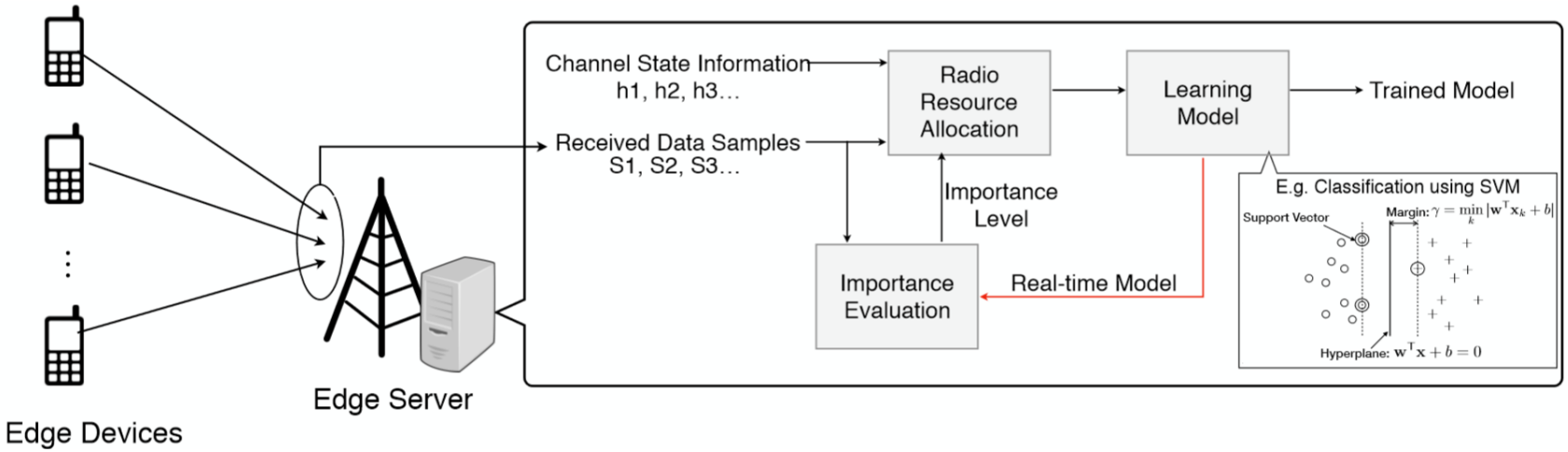
Active Learning is to select important samples from a large unlabelled dataset so as to accelerate model training with a labelling budget.

We can uitilize ***uncertainty*** (such as **entropy**) to measure the importance of data.

Case study: Importance-aware retransmission

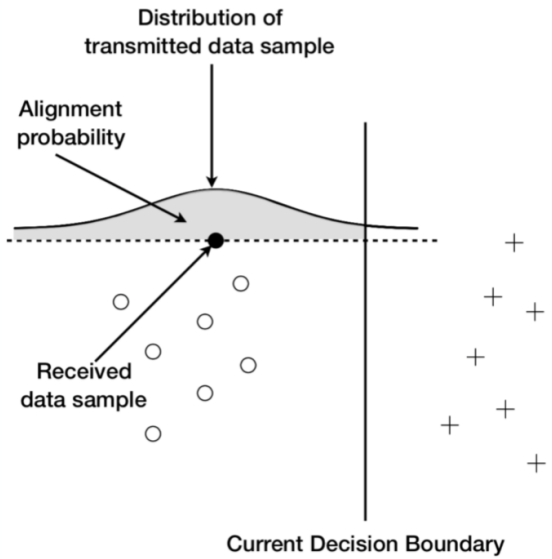
A binary classfier is trained at the edge server based on SVm, with data collected from distributed edge devices.

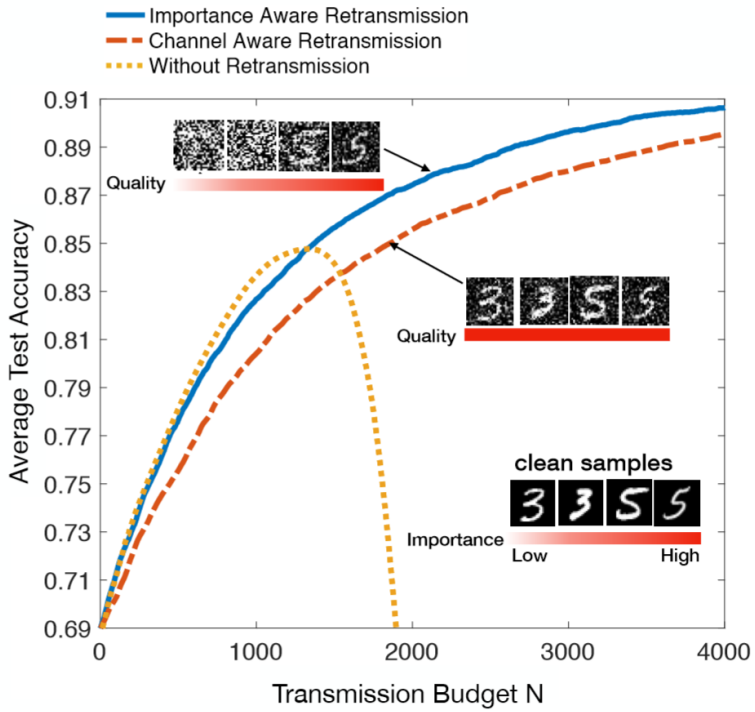
- ▷ **A *noisy* data channel.** For acquisition of high-dimensional training data.
- ▷ **A low-rate *reliable* channel.** For accurately transmitting small-size labels.



Case study: Importance-aware retransmission

Problem formulation: *How many retransmissions instances should be allocated for a given data sample?* In every communication round, the edge server need to decide either *selecting a device for quantity* or *requesting a previously scheduled device for retransmission for quality*.





Research opportunities

Where to extension?

- **Cache-assisted importance RRM.** We can exploit the storage of edge devices to a higher degree. By local cache, edge devices can pre-select important data before offlaoding! How to do that?
- **Multi-user RRM for faster intelligence acquisition.** Redundant information across different users will generated natrually in multiple user scenarios. How to effciently exploit data diversity?
- **Learning-driven RRM in diversified scenarios.** When the offloaded data is not the original data samples but other learning-related contents, the proposed RRM dose not work.

The intro of *Active Learning* is very ingenious! :-D

Feature extraction in Machine Learning

Feature extraction is widely applied in raw data so as to reduce its dimensions as well as improving the learning performance, e.g., PCA, ICA, LDA. However, :

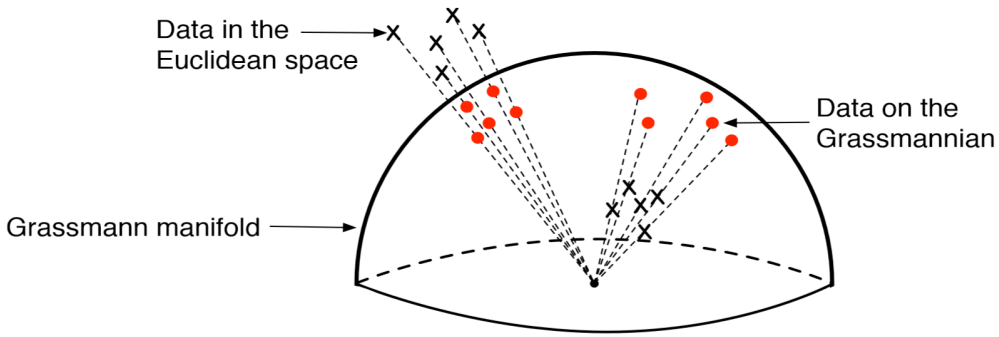
- too aggressive/conservative dimensionality-reduction can both degrade the learning performance;
- choice of a feature space directly affects the performance of the targeted learning task.

Source-and-channel encoding

Source coding *samples, quantizes, and compresses* the source signal such that it can be represented by a minimum number of bits *under a constriant on signal distortion* (rate-distortion tradeoff).

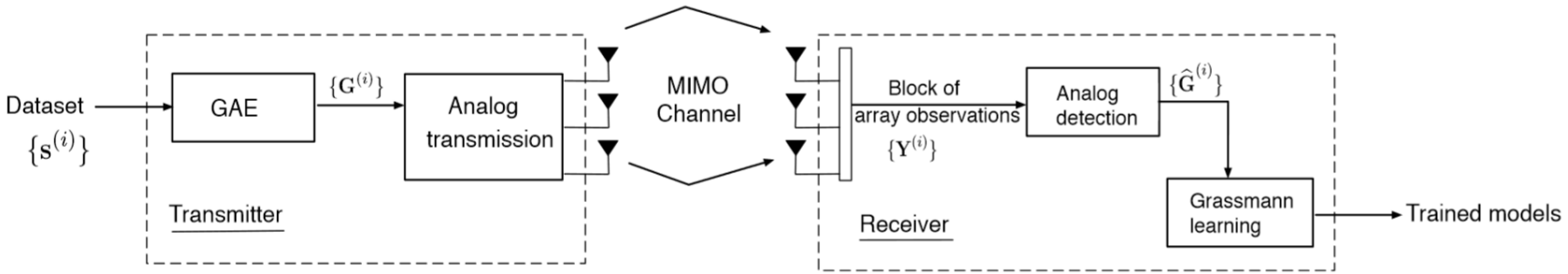
Principle of Learning-driven signal encoding

Signal encoding at an edge device should be designed by jointly optimizing *ferature extraction*, *source coding*, and *channle encoding* to accelerate edge learning.



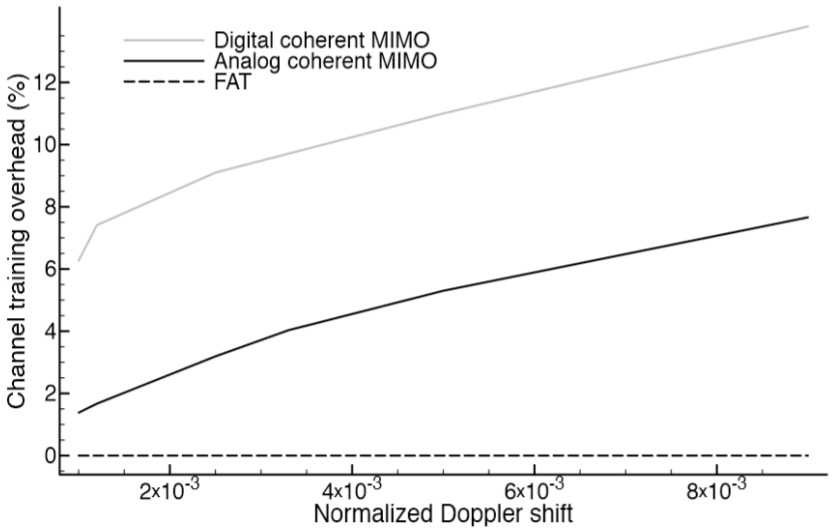
Case study: Fast Analog Transmission Learning

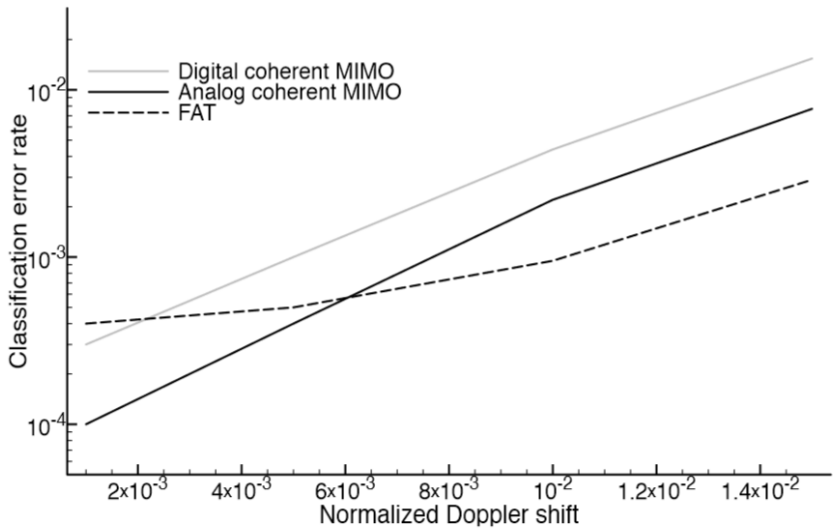
An edge learning system based on ***fast analog transmission*** (FAT) , having *Grassmann analog encoding* (GAE) as its key componet, is propsed for fast edge classification.



Case study: Fast Analog Transmission Learning

An edge learning system based on ***fast analog transmission*** (FAT) , having *Grassmann analog encoding* (GAE) as its key componet, is propsed for fast edge classification.





Research opportunities

Where to extension?

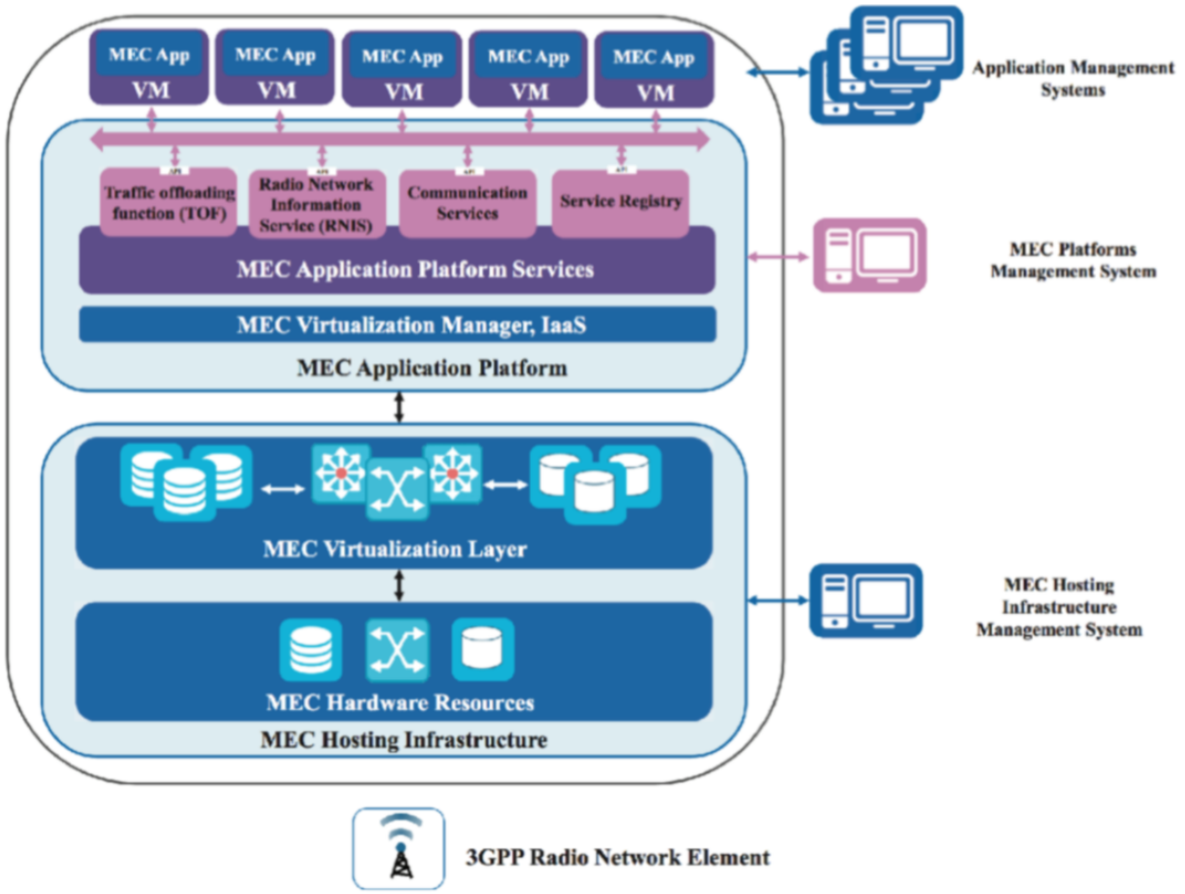
- **Gradient-data encoding.** We urgently need the design of gradient compression technologies to reduce communication overhead and latency.
- **Motion-data encoding.** How to encode a motion dataset for both efficient communication and machine learning? Maybe we can utilize the GAE method?
- **Channel-aware feature-extraction.** How to exploit *the channel characteristics* for efficient feature-extracting?

Foundations of Edge Learning

▷ *The thriving AI chips* and *software platforms* lay the physical foundations for edge learning.
▷ *MEC* and *the upcoming 5G networks* provide a practical and scalable network architecture for edge learning.

- AI chips.
- AI Software platforms.
- MEC and 5G Network Architecture.

Network visualization architecture



Outline

Deficiencies (I am potty-mouthed :-D)

- **The most severe problem** of this paper lies in that the in-depth studied three aspects are *loosely linked*. The scheme proposed in *learning-driven multi-access* can not be applied in *learning-driven resource allocation*, and so on.
- The case studies listed are just toy programs.
- Too little research has been made on AirComp, which is an important component for *learning-driven multi-access*. Unmatured technology can not be qualified for edge learning.
- No long-term reliable network system is considered.
- ...

Motivation for designing Egent

Network models compactable for Egent

Knowledge on Communication in this paper greatly enlightens me the desgin compactable network/communication models, which is a bottomed layer of **Egent**.

- Noise in training-data transmission maybe is not that important.
- Long-term observations and collected data on user profiles can be utilized for joint resource manegment and model-training.
- Collaboration between cloud and edge learning can be in-depth studied.
- Mobility management of users not only effects the offloading decisions, but also incurs frequent handovers among edge servers (not service migration).
- ...