Research Project: Multi-access Edge Computing

Energy-efficient, Scalable, and Reliable Distributed Green Streaming Machine Learning for Edges

PI: Zhang Shuhao (ISTD, SUTD)
Co-PI:
Collaborators: Lu Mian (Fourth Paradigm Southeast Asia Pte. Ltd.)

The greening computing is an emerging trend, but potentially threatened by energy intensive machine learning algorithms that are needed for complex optimizations as well as significant overheads incurred during big data collection and transmission to cloud. The streaming machine learning (streaming ML) is built on the idea of learning continuously and adaptively about the external world. Different from traditional batch-based approaches, streaming ML needs to process data as soon as data arrive. It is gaining more and more attentions recently due to its capability of 1) dropping irrelevant data progressively to reduce data footprints, 2) dealing with concept evolving over time in input workloads, and 3) guaranteeing a low end-to-end training and inferring latency. Streaming ML is particularly suitable for ML tasks at cloud edges due to its reduced computational workloads and memory footprints without affecting the accuracy of training or inferring results. Together with the rapid advancement of reliable wireless communication technology and power-efficient computer architectures, streaming ML can be now largely processed at the edge of the network, instead of waiting for data transmitting to the centric cloud.

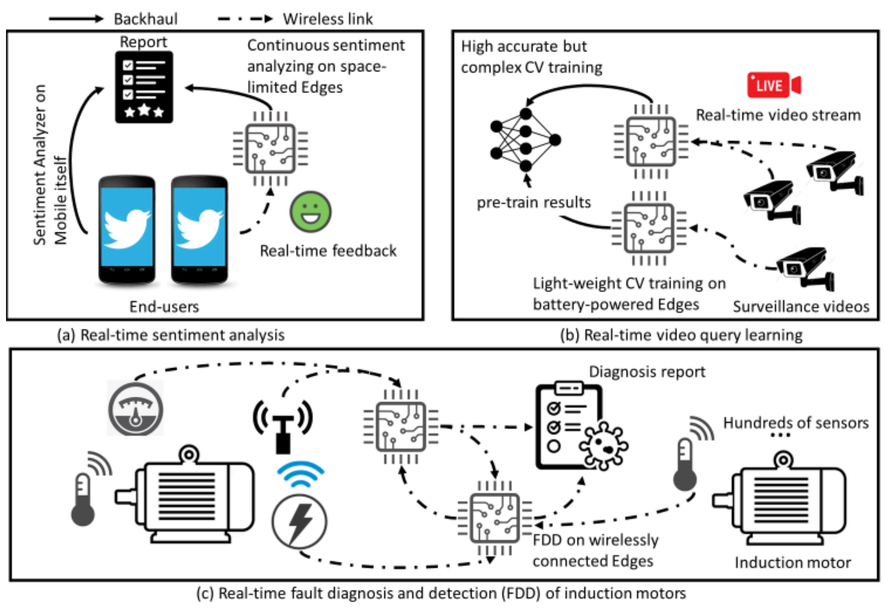

Figure 1 Typical ML applications enabled by GSML4E

This project will investigate how to reduce energy expenditure for streaming ML by better utilizing of the modern edge devices, which we name it as green streaming machine learning for edges (GSML4E). Such a paradigm has the potential to address the concerns of response time requirement, bandwidth cost saving, as well as battery life constraints for machine learning on edge devices (e.g., mobiles, battery-powered edge servers). Typical applications enabled by GSML4E are illustrate in Fig. 1, such as real-time sentiment analysis, real-time video query learning and real-time fault diagnosis and detection (FDD) of induction motors. In those applications, ML tasks are primarily or initially conducted at the storage-constrained, batter-powered and wirelessly connected edges, eliminating or reducing the need of data transmission to cloud. In this way, GSML4E makes ML:

- ✓ *more economically sustainable with reduced costly involvement of cloud center*
- ✓ *more responsible and scalable with lower processing latency*
- ✓ *more practically reliable to be adopted in many industries, which does not allow data to be transmitted to cloud by policy*

Due to its practical potential, we envision that GSML4E can be a key application component of the next generation communication ecosystem.

This project has three major work packages: *(1) An AMP-aware parallelization framework for streaming ML tasks; (2) A sustainable and fault-tolerant workload offloading scheme; (3) an AI-driven continuous workload reduction component.*

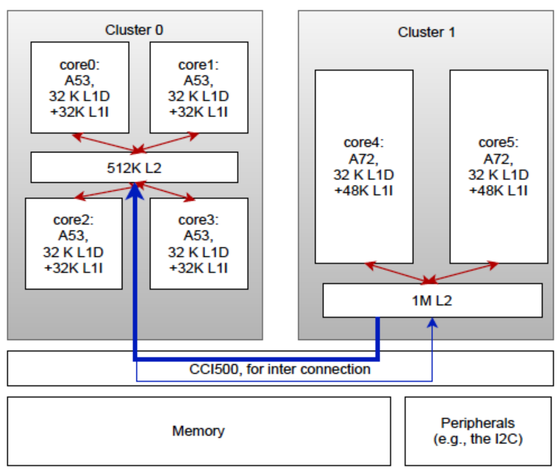

Figure 2 Asymmetric multicore processors (AMPs)

WP1: Modern ARM machines are typical choices for edge devices, which are often equipped with asymmetric multicore processors (AMPs). Typical AMPs include the ARM big.LITTLE architecture, which couples relatively batter-saving and slower processors with relatively more powerful and power-hungry ones, sharing the same Instruction Set Architecture (ISA), as shown in the Fig.2. Such AMP architectures provide the edge devices (e.g., mobile and IoT sensors) the opportunities to support ML tasks close to the data source in order to reduce data transmission overhead (i.e., in-situ learning), while keeping a low power consumption. Due to its limited storage space, traditional batch-based ML algorithms are hardly applicable as they require large data sets being collected and stored before training. If we can fully utilize AMP for streaming ML, we may be able to achieve energy-efficient aspect of GSML4E. Unfortunately, utilizing AMPs is generally non-trivial due to two key challenges. First, different algorithm components (e.g., the arithmetic calculation or model lookup) can contain varying task-core affinity in terms of both performance and energy consumption. Second, the communication penalty between AMP cores is non-deterministic with asymmetry. In particular, the communication cost between two cores varies due to 1) distance between cores, 2) the communication direction, and 3) frequency of each core. Hence, our first step is to address those issues by designing **<u>energy efficient streaming ML leverage on power-efficient AMP architectures</u>** at edges.

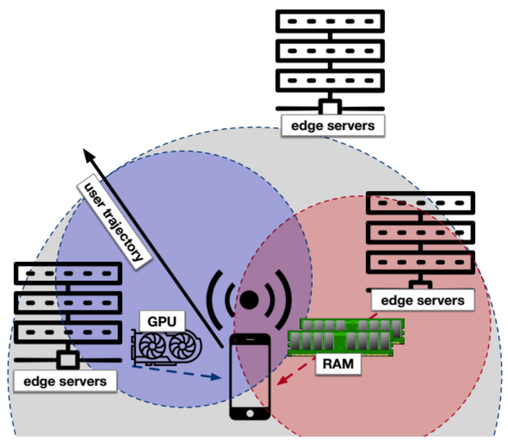

Figure 3 Coalescent Computing*

WP2: GSML4E conducts machine learning tasks over data streams for low-latency training and inferring. It needs to process streams of growing volume and velocity, while the heterogeneity of data sources from edge devices yields unpredictable input rates. This is especially challenging since edge devices are particularly constrained by its by power, storage, and computing resources. Advances in low-latency, wireless networking technology (e.g., 5G and 6G) will allow service providers to blur the distinction between local and remote resources for commodity computing. Aligned with the recent envision of "coalescent computing" shown in Fig.3, the edge devices will no longer have fixed computational power, but rather will appear to have flexible computational capabilities that vary subject to the shared, disaggregated computing resources available in the physical proximity. As such, we can transparently leverage these ephemeral resources to provide a better computing experience at edges. Furthermore, GSML4E must deal with dropped connections gracefully. In any case, techniques applied to achieve resilience should avoid centralized coordination given the ephemeral proximity of resources. We hence aim to develop **<u>a proper and rigorous framework to improve fault tolerance as a constraint on workload offloading among wireless connected devices for streaming machine learning</u>**.

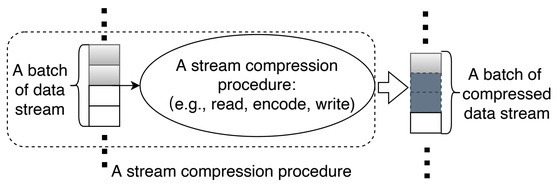

Figure 4 Real-time data compression

WP3: Data compression is an effective way to boost data processing and is a necessity for machine learning on storage-limited edge devices. Despite that there is a very rich literature of existing data compression algorithms, few can be directly applied in GSML4E to bring performance benefits. This is because compression algorithms have diverse suitability and compression itself brings additional (de)compression overheads. Further, various compression schemes including light/heavy-weights, lossy/lossless compression algorithms are applicable but brings non-trivial trade-offs between accuracy and latency when they are applied to GSML4E. Similarly, the selection and combination of other system components such as applied machine learning models brings further design challenges for trading off among energy consumption and latency. We propose **<u>an AI-based system design to automatically involve suitable real-time data compression schemes to continuously reduce workloads in GSML4E</u>**.

*: Reference to "Coalescent Computing", by Kyle C. Hale, APsys'21

FCP FUTURE COMMUNICATIONS R&D PROGRAMME
INFOCOMM MEDIA DEVELOPMENT AUTHORITY
NATIONAL RESEARCH FOUNDATION PRIME MINISTER'S OFFICE SINGAPORE
SUTD SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN