mobile AI

AI/Machine Learning on Mobile and IoT Edge Devices

Seoul AI. Emilio Jose Coronado Lopez

paradigm

We’re living in a **Data World**. A **World** of **Sensors, Signals** and **Data Streams.**

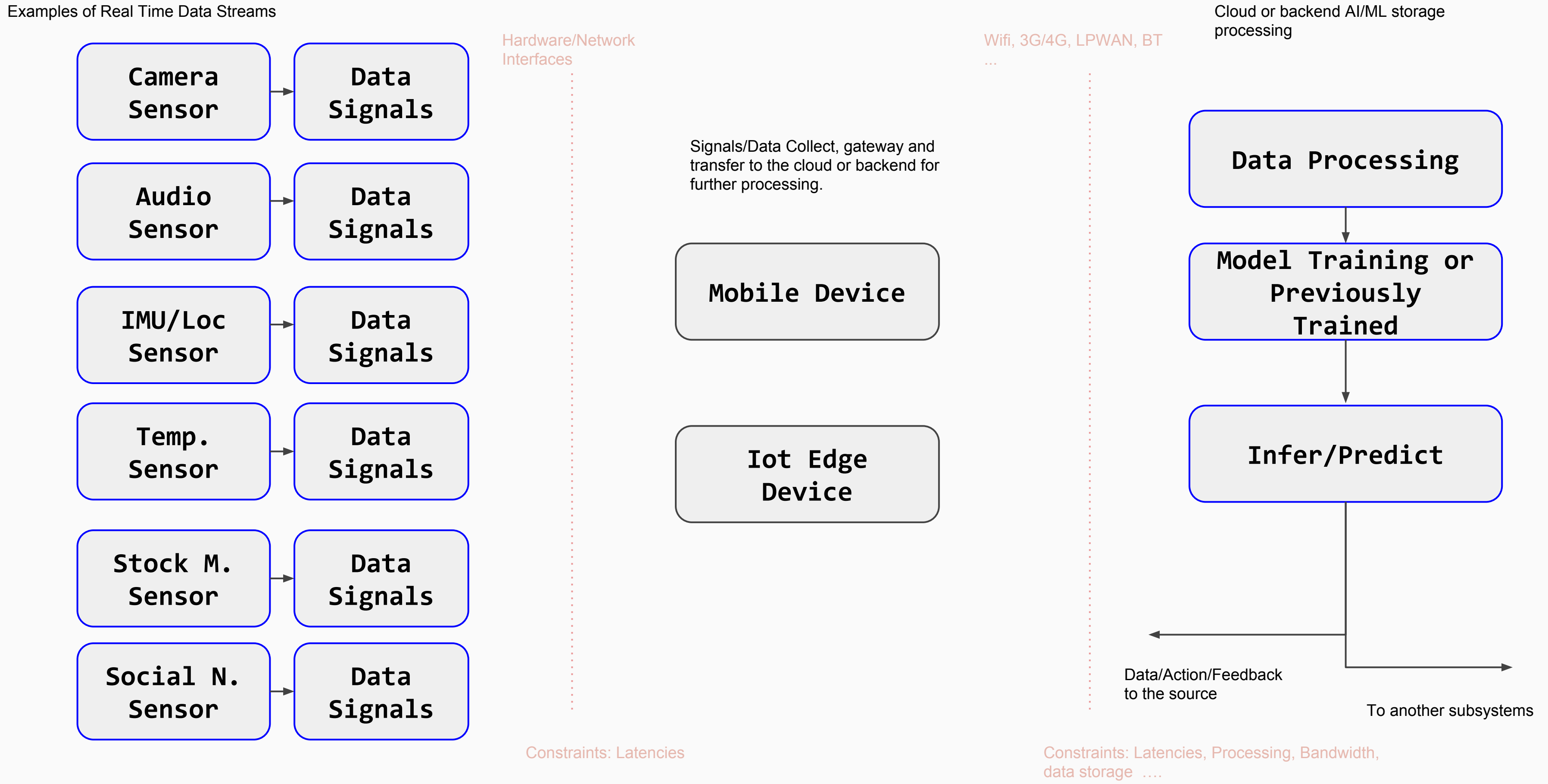
Data Constantly flows and gathers into a form of mobile or IoT Edge device, then processed, transferred to network/clouds backends, storages or analytics AI/ML services.

> **In some cases, a response in the front end or the device where the signal was generated is also expected.**

in real time, in a galaxy far far away ...

Latencies are worst enemies ...

cloud AI architecture



Examples of Real Time Data Streams

Camera Sensor → Data Signals

Audio Sensor → Data Signals

IMU/Loc Sensor → Data Signals

Temp. Sensor → Data Signals

Stock M. Sensor → Data Signals

Social N. Sensor → Data Signals

Hardware/Network Interfaces

Signals/Data Collect, gateway and transfer to the cloud or backend for further processing.

Mobile Device

Iot Edge Device

Wifi, 3G/4G, LPWAN, BT ...

Cloud or backend AI/ML storage processing

Data Processing

Model Training or Previously Trained

Infer/Predict

Data/Action/Feedback to the source

To another subsystems

Constraints: Latencies

Constraints: Latencies, Processing, Bandwidth, data storage ....

cloud AI. notes

A typical big data/analytics pipeline enables top notch applications and user experiences that everyone is used to, mostly it does a lot of pre-training beforehand.

At some point, new uses cases involving more data streams, highest bandwidth demands or stronger hard real time requirements won't be **technically feasible** or operative cost can be **too expensive.**

examples

NLP, **AI assistants** -> **OK: Audio Streaming** is no really hard real time.

Social Network feeds real time analytics -> OK, as soon as data is text, and backend services have enough storage and processing power.

Smart Cameras with low FPS, lower resolution tiers -> OK, however increasing resolution, FPS, with real time features will start creating some hassle.

Connected Cars, UAV feeding images from cameras, service sensors, location services, with real time requirements “mostly” not feasible.

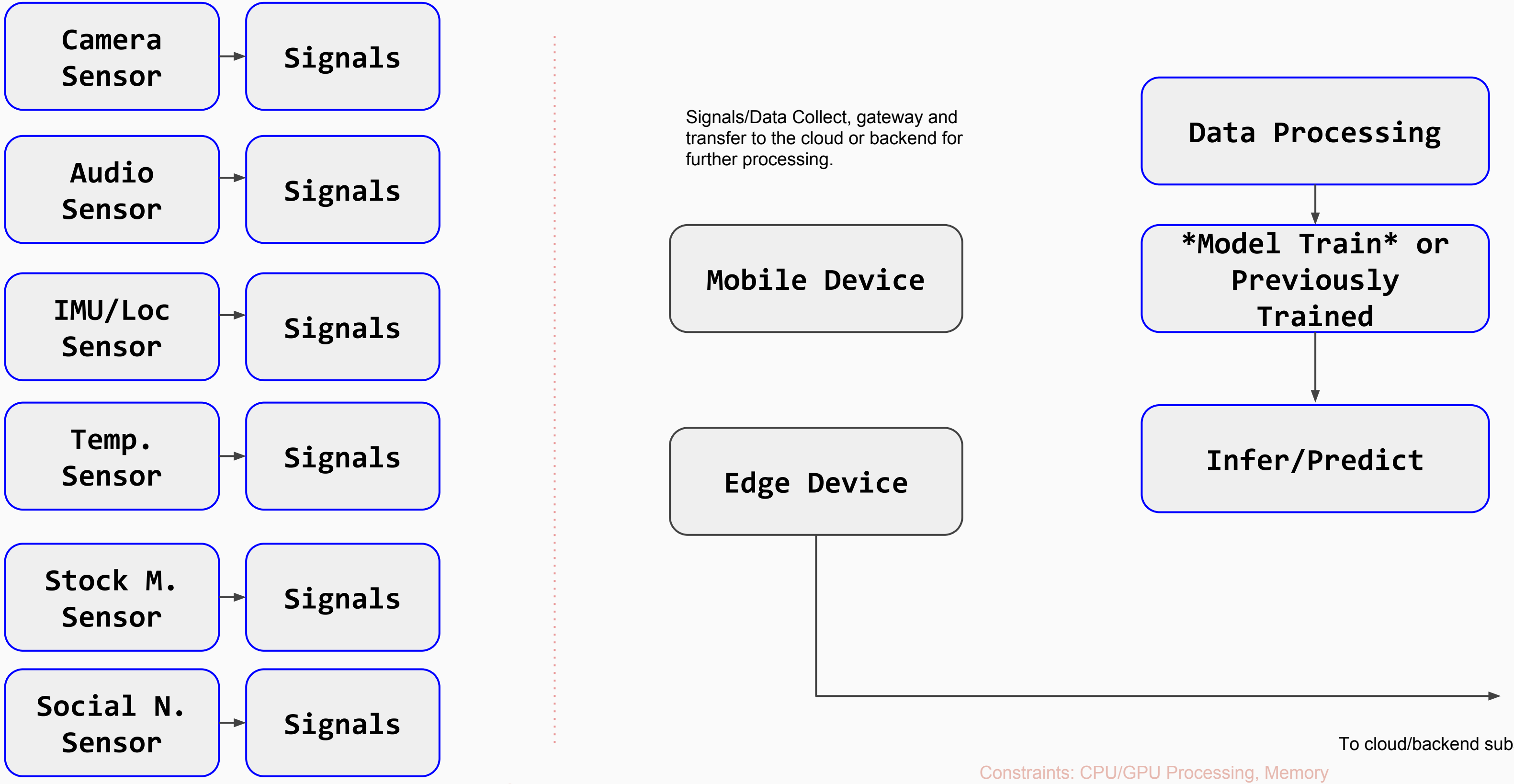
enter AI/ML on ARM

*99.99% of referenced mobile or edge IOT devices uses some kind of ARM CPU many of them paired with a GPU with OpenCL/OpenGL support.

mobile AI. architecture

Examples of Real Time Data Streams

Mobile GPU or attached Neural Compute Unit (Movidius)



Camera Sensor → Signals

Audio Sensor → Signals

IMU/Loc Sensor → Signals

Temp. Sensor → Signals

Stock M. Sensor → Signals

Social N. Sensor → Signals

Signals/Data Collect, gateway and transfer to the cloud or backend for further processing.

Mobile Device

Edge Device

Data Processing

Model Train or Previously Trained

Infer/Predict

To cloud/backend subsystems

Constraints: Latencies

Constraints: CPU/GPU Processing, Memory bandwidths, Batteries

mobile AI. new chipsets

http://i.mediatek.com/p60

4 Arm Cortex-A73 2.0 GHz, 4 Arm Cortex-A53 2.0 GHz NeuroPilot AI tech.

The MediaTek’s NeuroPilot SDK in P60 is compatible with Google Android Neural Networks API (Android NNAPI), and also supports common **AI frameworks, including TensorFlow, TF Lite, Caffe**, and Caffe2. That makes it easy for developers to quickly bring-to-market innovative AI applications. As a partner of the **Open Neural Network Exchange (ONNX)**, MediaTek is working on bringing ONNX support to the chipset in Q2 2018 to provide developers with even more flexibility for designing AI-powered applications.

mobile AI. notes

On mobile AI/ML, constraints are mostly hardware: CPU, GPU, Memories.

Moore’s law, expectation is mobile SOCs more powerful, more power efficient ….

mobile AI. Arm

ARM made available ARM Compute Library for CV/ML last year.

The open source, makers community is starting to play and port some of the typical frameworks: https://ai.stackexchange.com/questions/2854/ssd-or-yolo-on-arm

Interesting devices gives extra punch needed to crunch some of the usual ML/AI frameworks: Intel Movidius Neural Compute Stick

https://ncsforum.movidius.com/discussion/218/tiny-yolo-on-ncs

mobile AI. Google

Google provides mobile versions of Tensorflow: Lite and Mobile

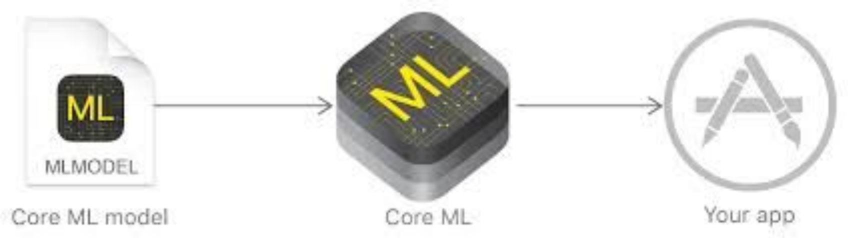
https://www.tensorflow.org/mobile/tflite/

https://www.tensorflow.org/mobile/

https://developer.android.com/ndk/guides/neuralnetworks/index.html,

https://github.com/tensorflow/tensorflow/tree/master/tensorflow/examples/android

mobile AI. Apple



Apple has their coreML framework too:
https://developer.apple.com/machine-learning/

https://www.udacity.com/course/core-ml--ud1038

Examples:

http://machinethink.net/blog/object-detection-with-yolo/

finals.

IOT, and new form of low latency, low power, constantly on sensors means more data points, signals, and streams to process in the gateways/edges

5G will connect more devices, demanding higher bandwidths, by instance 4K/8K, AR, VR streams.

Connected cards, and AUV will add millions of sensors feeds with real time requirements to backend and cloud services.

finals.

I recommend to start exploring with some hardware running Linux, widely supported by the community: Beagle Boards, Raspberry Pi's, ODroid's.

Use software architectures and solutions that can scale with mobile hardware chipsets developments.

If someone is interested, it seems Intel Movidius Neural Computer Stick is also available in Korea.

https://kr.mouser.com/new/Intel/intel-movidius-stick/

thank you ...

Description:

Machine Learning/AI frameworks and pipelines are used to be executed or deployed in a kind of PC+GPU hardware solution. Recently Google, Amazon, Microsoft, Facebook, etc. are opening their internal AI infrastructure to the public, offering ML/AI cloud computing as a service, mostly running NVIDIA’s GPU’s or dedicated NCU (Neural Computing Units) in their servers.

This solution works pretty well on applications and services that does offline, analytics, defers computing, or does not have hard real time requirements, however, most of the mobile first applications, upcoming IOT, UAV applications, are pushing for more connected sensors, more data flowing into the cloud, and UX with none or close to zero latencies.

With more powerful mobile and embedded chipsets with specific GPU’s, AI/ML in mobile or IOT edge devices become possible, doing most or “enough” processing into the device itself, offloading some cloud computing processing and bandwidth for specifics.

This is a quick introduction, current SOC’s and platform status are probably not enough for the most advanced applications or uses cases, but those are first steps, there is already good open source and community support and is good enough to start playing with it.