

Use Case of Computing-Aware AI large model

Qing An

Alibaba Group

anqing.aq@alibaba-inc.com

AI Foundation Model and Customized Model

- AI Foundation Model
 - ✓ Handle multiple tasks and domains
 - ✓ Wider applicability and flexibility, but may not perform as well as customized models in specific domain tasks
 - ✓ Mega-scale parameters
- Customized Model
 - ✓ Trained for specific industries or domains, e.g. medical, finance, etc.
 - ✓ More focused on solving specific problems, but may not be applicable to other domains
 - ✓ Large/Middle-scale parameters

AI Model Training and Inference

- Training

- ✓ **Large dataset input:** Feed AI model with large amounts of data and optimize it to learn and improve its performance
- ✓ **Iteration:** Model is adjusted and refined until it achieves high levels of accuracy and predictive ability
- ✓ High demand on accuracy, computing and memory resource

- Inference

- ✓ **Small data input:** Use the trained AI model to make predictions or decisions based on new input data
- ✓ **One-direction:** Deployed in a production environment where it is given real-world data and make predictions based on that data
- ✓ Focus more on the balance between computing resource, latency and power cost.

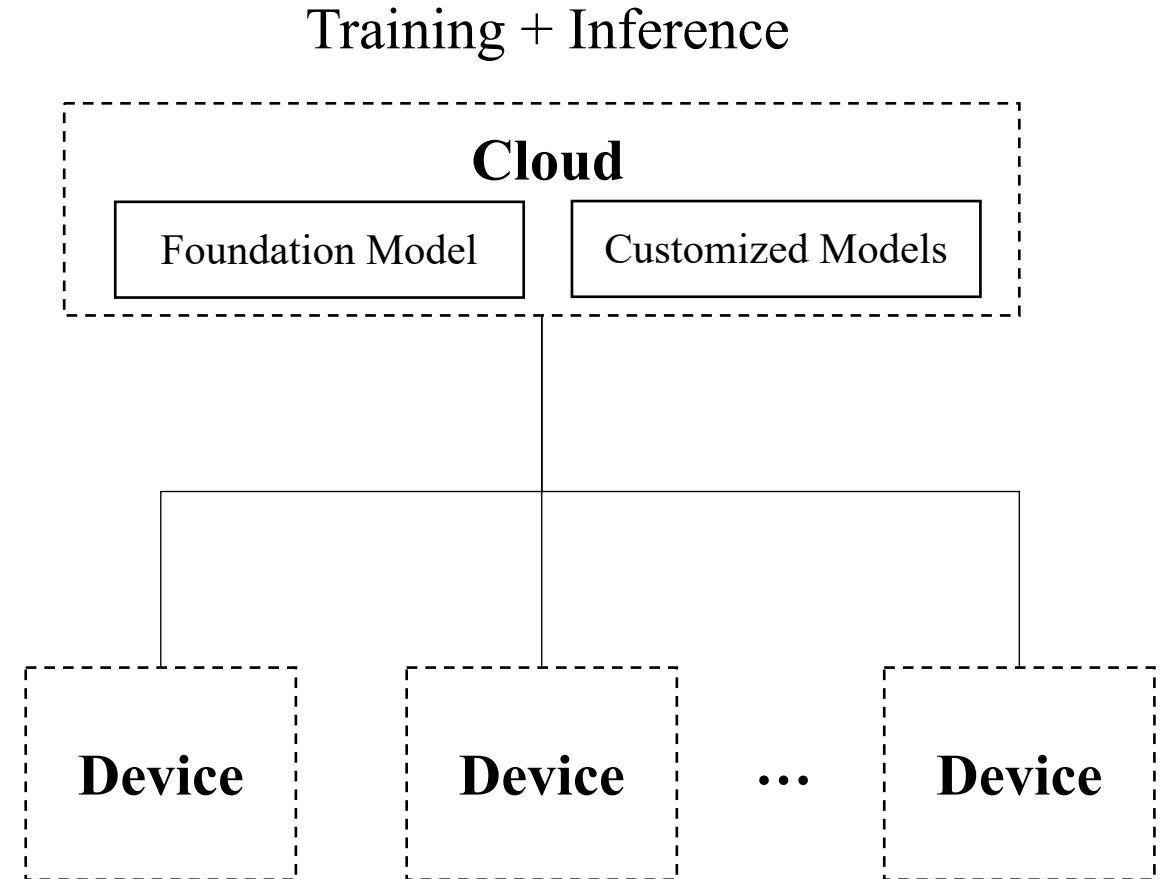
AI Tasks – regarding inference

- Text: text-to-text (conversation), text classification (e.g. sentiment analysis)
- Vision: image classification (label images), object detection
 - ✓ May have high demand on network and computing resource
- Audio: speech-to-text, text-to-speech
 - ✓ May have high demand on network and computing resource
- Multimodal: text-to-image, image-to-text, text-to-video, image-to-image, image-to-video, etc.
 - ✓ May have high demand on network and computing resource

All-in-Cloud

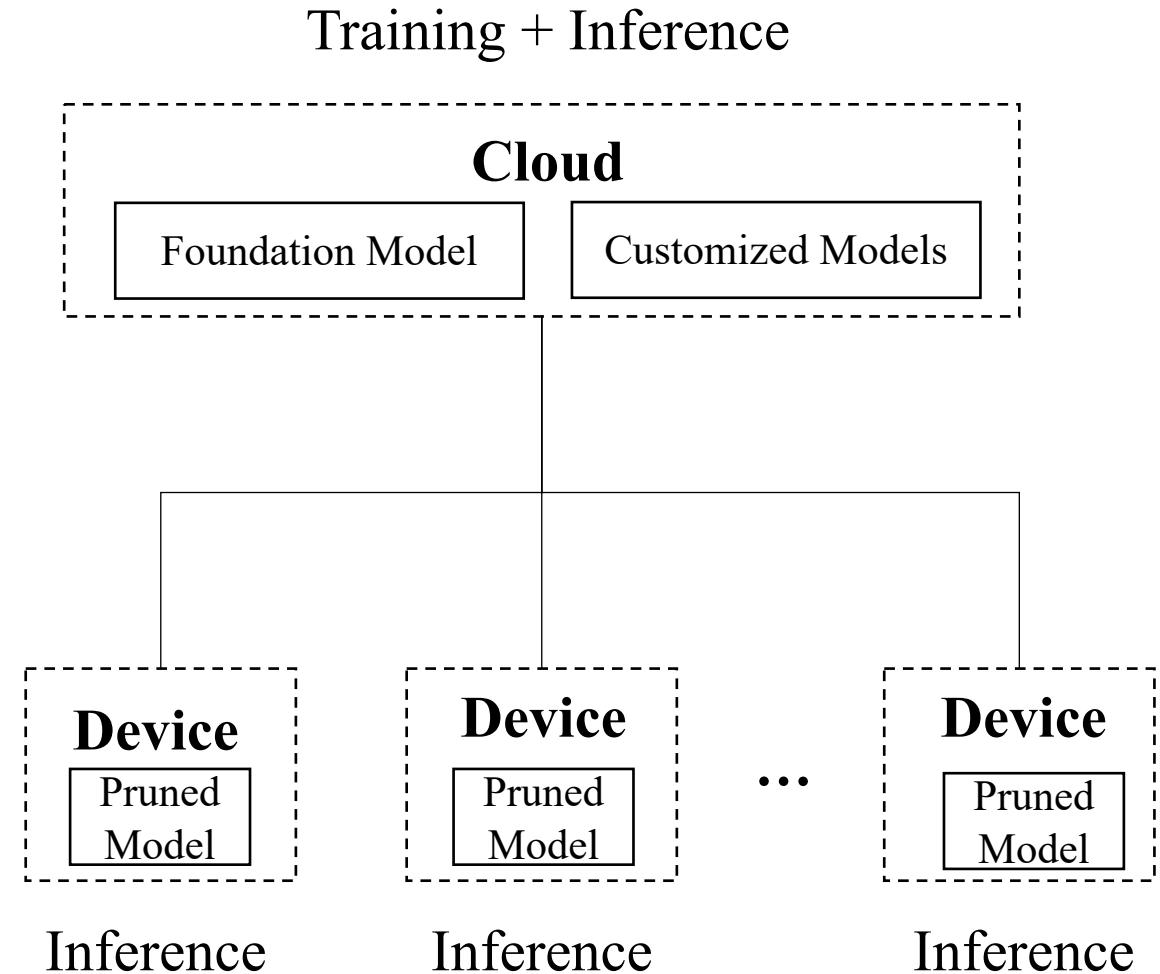
Cloud is highly suitable for training, but may have issues for inference:

- Latency
 - ✓ Especially for delay-sensitive AI applications
 - ✓ Even if real-time interaction is not needed, high latency will affect user experience
- Privacy: high cost to ensure privacy protection in cloud-based inference.



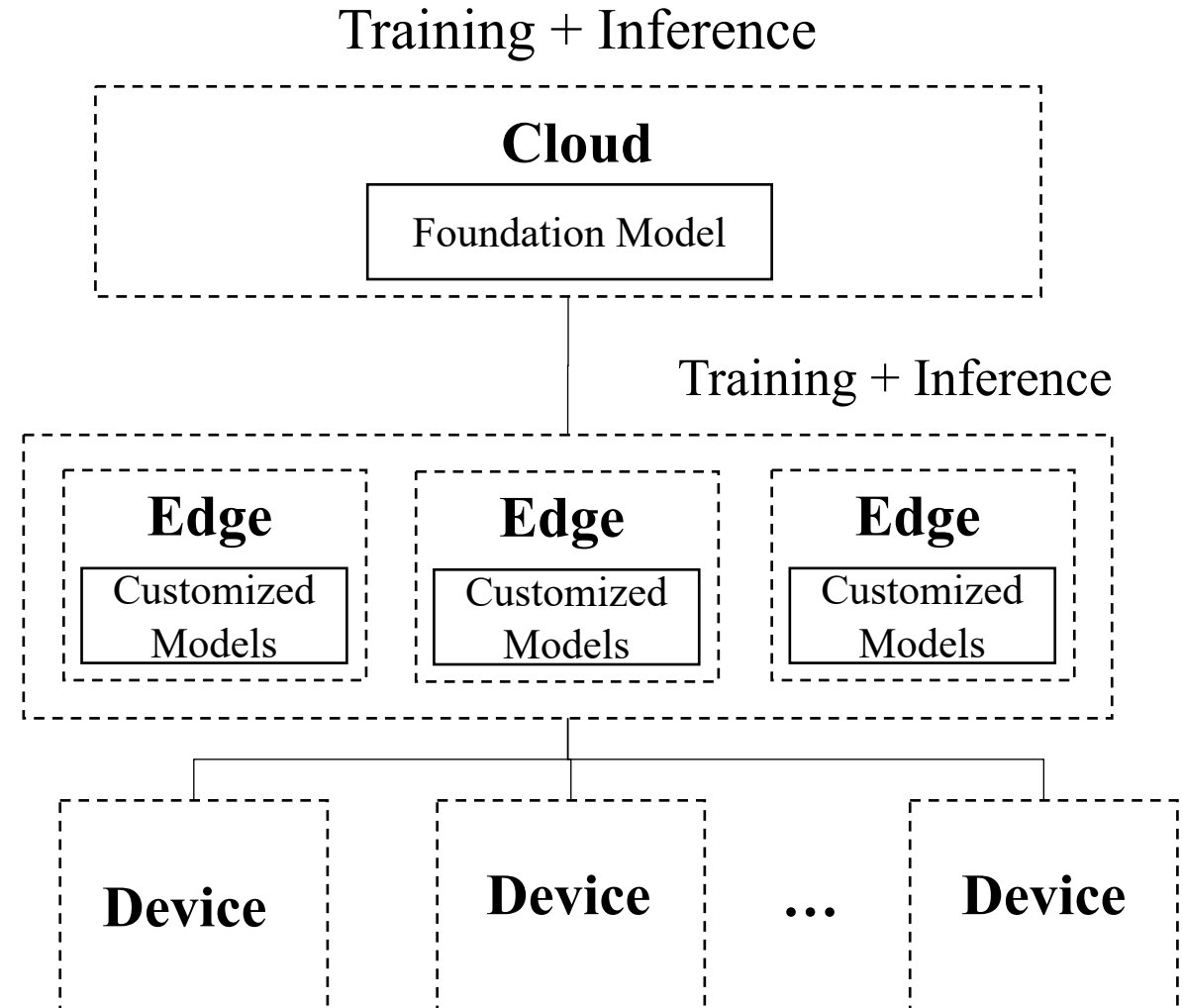
Cloud-device co-inference

- Low latency: deploy inference locally
- But may support only limited AI tasks
 - ✓ In most cases, only compressed, pruned model can be deployed on device



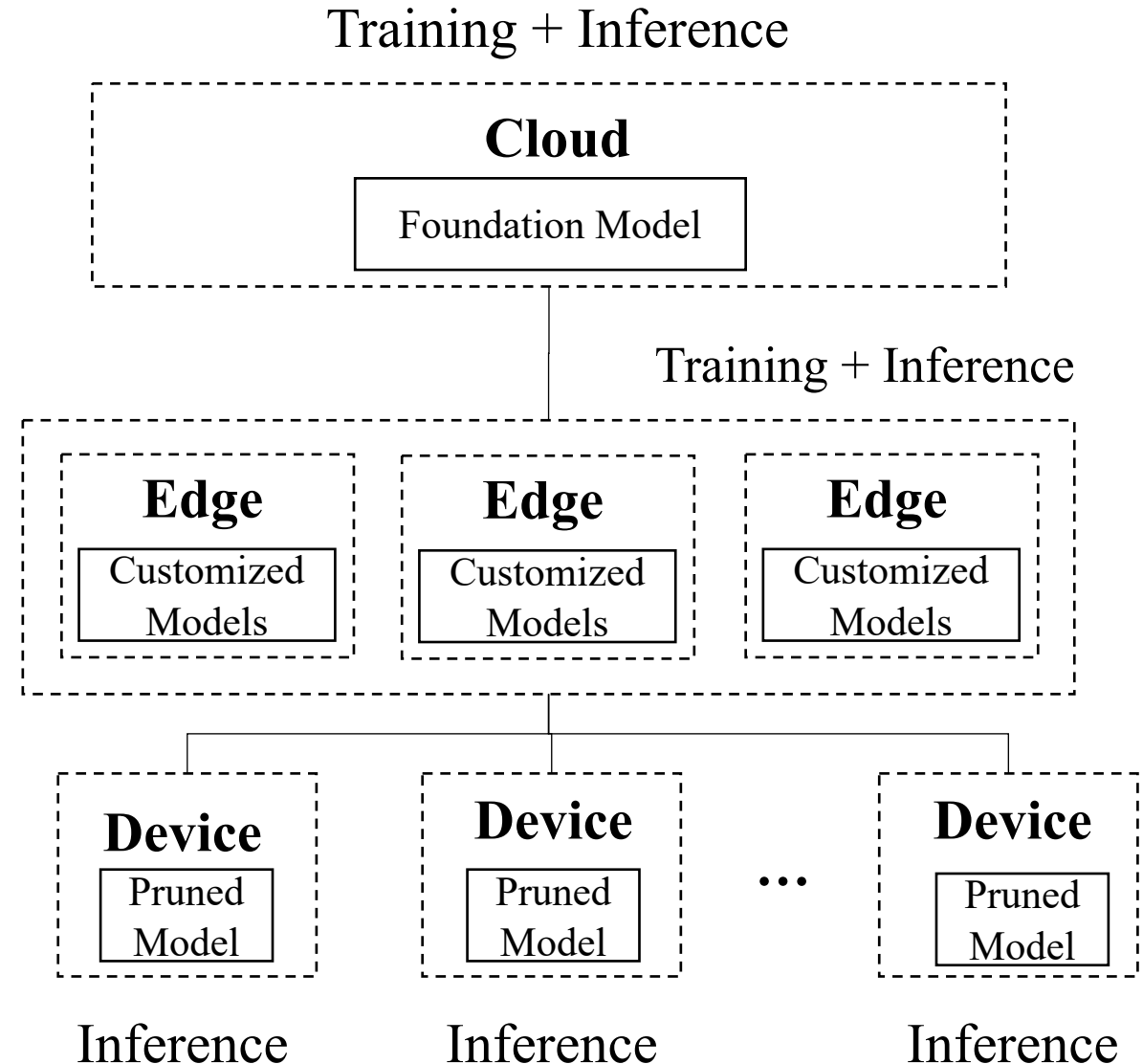
Cloud-edge co-inference

- Low latency: deploy inference near to device
- Low demand on device resources
- But when handling AI inference tasks, if traffic load between device and edge is high or edge computing resource is overloaded, traffic steering is needed to ensure the QoS



Cloud-edge-device co-inference

- More flexible deployment (also more complex): deploy inference locally or near to device
- Device can work when edge isn't available
- Careful consideration to ensure that edge will only be used when the trade-offs are right
- Similar to last scenario, traffic steering is needed



Why traffic steering is needed

- Many AI tasks brings on high demand on network resource and computing resource: **vision, audio, multimodal**
- It is common that same customized model is deployed in multiple edge sites to achieve load balance and high reliability
- The edge site's computing resource and network info should be collectively considered to make suitable traffic steering decision
 - ✓ E.g. If the available computing resource in nearest edge site is low, the traffic of AI tasks should be steered to another edge with high resource.
 - ✓ E.g. If multiple AI tasks, delay-sensitive task (live streaming with AI-generated avatar) and delay-tolerant task (text-to-image) arrive in edge, delay-tolerant task should be steered to another edge if the nearest edge's resource is limited.

Discussion

- Besides the existing defined requirements in draft-yao-cats-ps-usecases-03
 - ✓ Is there a need for the device side application to know and choose where the inference is taking place? (edge's capability)
 - ✓ How can network and computing resource be used optimally? Let device side application declare the AI task constraints (task description, latency requirement, etc.) to the edge, and let edge figure out?

Thank you