Considerations of deploying AI services in a distributed method

draft-hong-nmrg-ai-deploy-05

Y-G. Hong (Daejeon Univ.), S-B. Oh (KSA), J-S. Youn (DONG-EUI Univ), S-J. Lee (Korea University/KT), S-W. Hong (ETRI), H-S. Yoon (ETRI)

nmrg Meeting@IETF 118 – Prague
November 10, 2023
History and status

- 00 : draft-hong-nmrg-ai-deploy-00 (Mar. 2022)
- 1\textsuperscript{st} revision : draft-hong-nmrg-ai-deploy-01 (Jul. 2022)
  - 1\textsuperscript{st} presentation
- 2\textsuperscript{nd} revision : draft-hong-nmrg-ai-deploy-02 (Oct. 2022)
  - 2\textsuperscript{nd} presentation
- 3\textsuperscript{rd} revision : draft-hong-nmrg-ai-deploy-03 (Mar. 2023)
  - 3\textsuperscript{rd} presentation
  - Updated by comments by Alexander Clemm, Jeff Tantsura, Jeferson Campos Nobre
- 4\textsuperscript{th} revision : draft-hong-nmrg-ai-deploy-04 (Jul. 2023)
  - Updated to reflect the use case of digital twin networks
- 5\textsuperscript{th} revision : draft-hong-nmrg-ai-deploy-05 (Oct. 2023)
  - 4\textsuperscript{th} presentation
  - Updated to reflect the use case of digital twin networks and self-driving car
Motivations (1/2)

SNMP-managed Configuration

[Source: Oracle]

[Source: StackExchange]
Motivations (2/2)

– Deployment of AI services
  • Focus: training (learning) -> inference (prediction)
  • For inference, not only high-performance servers, but also small hardware, microcontroller, low-performance CPUs, and AI chipsets are optimal target device (due to cost)

– Configuration of the system/network in terms of AI inference service
  • For training: accuracy of the model
  • For inference:
    • Target device: Local, edge, cloud
    • Objectives: Accuracy, Latency, Network traffic, Resource utilization, etc.
    • Considerations: Network configuration, AI model, Serving framework, Communication method, device capacity, inference data, etc.
Intentions of this draft

– Share our experiences and implementation results to find optimal network/system for AI services
  • To find what is import information to provide optimal AI services
  • To find How to deliver these information between related devices

– Find common components to provide optimal AI services
  • Common information (similar to MIB)
  • Common system to provide AI services
  • Common network architecture to provide AI services
  • Common protocols to exchange information for AI services

– Find useful use cases
  • Self-driving cars
  • Digital twin networks
Network configuration structure to provide AI services

Figure 2: AI inference service on Local machine

Figure 3: AI inference service on Cloud server

Figure 4: AI inference service on Edge device
AI inference service on vertical/horizontal servers

Figure 5: AI inference service on Cloud sever and Edge device

Figure 6: AI inference service on horizontal multiple servers
Considerations according to the functional characteristics of the hardware

– The performance of AI inference service varies depending on how the hardware such as CPU, RAM, GPU, and network interface is configured for each cloud server and edge device.

– AI inference service can be deployed in the following locations
  • Distant cloud server: High performance and high cost
  • Near edge device: Medium performance and medium cost
  • Local machine: Low performance and low cost

– AI inference service result in (assumption: same AI model)
  • Distant cloud server: High accuracy, short inference time, and long delay to transmit
  • Near edge device: Medium accuracy, medium inference time, and medium delay to transmit
  • Local machine: Low accuracy, long inference time, and short delay to transmit
Considerations according to the characteristics of the AI model

- AI inference service can be deployed in the following locations
  - Distant cloud server: Heavy AI model, high accuracy, Big size, long inference time
  - Near edge device: Medium AI model, medium accuracy, medium size, medium inference time
  - Local machine: Light AI model, low accuracy, small size, short inference time

- AI inference serving framework
  - Traditional web server: ex) FastAPI, Flask, and Django
    - It can be operated on low performance machines
  - Specialized serving framework: ex) Tensorflow serving
    - It can provide high performance.
Considerations according to the characteristics of the communication method

– AI inference service can be utilized
  • Traditional REST method
    • Common and easily deployed
  • Specified communication method (e.g., gRPC)
    • Better performance but need some works

– AI Inference data can be classified
  • Real-time vs. Batch
  • Secure & non-secure
Use cases

Deploying AI services in Self-driving car

Deploying AI services in DTN
Thanks!!

Questions to NMRG

1. Is it useful and appropriate in NMRG?
2. How to develop this draft?
3. Any feedbacks?