Considerations of deploying AI services in a distributed approach

draft-hong-nmrg-ai-deploy-01

Y-G. Hong (Daejeon Univ.), S-B. Oh (KSA), J-S. Youn (DONG-EUI Univ), S-J. Lee (Korea University/KT), H-K. Kahng (Korea University)
nmrg Meeting@IETF 114 – Philadelphia
July 27. 2022
History and status

– 1\textsuperscript{st} revision : draft-hong-nmrg-ai-deploy-00 (Mar. 2022)

– 2\textsuperscript{nd} revision : draft-hong-nmrg-ai-deploy-01 (Jul. 2022)

• 1\textsuperscript{st} presentation
Motivations

– Change of the 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 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: AI model, Serving framework, Communication method, device capacity, inference data, etc.

– Accelerate the study AI issues in the nmrg
Generic procedure of AI service

Figure 1: AI service workflow

- Data collection & Store
- Data Analysis & Preprocess
- AI Model Training
- AI Model Deploy & Inference
- Monitor & Maintain Accuracy
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 Cloud server and Edge device

Figure 5: AI inference service on Cloud server and Edge device
Considerations of deploying AI services in a distributed approach

– Objectives of AI services
  • Accuracy of model
  • Latency of IoT service
  • Network traffic
  • Resource utilization

– Considerations of deploying AI services
  • AI model (heavy vs. lightweight)
  • Serving framework (Web vs. Serving-targeted)
  • Communication method (REST vs. gRPC)
  • Machine capacity (CPU, RAM, etc.)
  • Inference data (realtime vs. batch, secure & non-secure, etc.)
An example of AI system for Object detection services

Machine requesting AI service

Machine performing AI service
Latency of object detection services in each device

**System Information**
- **Operating System**: Ubuntu 20.04.3 LTS
- **Model**: LG Blackberry 14700PP-G140K
- **Motherboard**: LG Electronics 14700P

**CPU Information**
- **Name**: Intel Core i7-10750H
- **Topology**: 1 Processor, 6 Cores, 12 Threads
- **Base Frequency**: 4.70 GHz
- **L1 Instruction Cache**: 32.0 KB x 4
- **L1 Data Cache**: 32.0 KB x 4
- **L2 Cache**: 256 KB x 4
- **L3 Cache**: 8.00 MB x 1
- **Memory Information**: Memory 16.00 GB

**Geekbench 5 Score**
- **Single Core Score**: 1660
- **Multi Core Score**: 5617

---

**System Information**
- **Operating System**: Ubuntu 20.04.3 LTS
- **Model**: Lenovo V3900
- **Motherboard**: AMD Ryzen 3 3200G 4-core 3.6GHz

**CPU Information**
- **Name**: AMD Ryzen 3 3200G
- **Topology**: 1 Processor, 4 Cores, 8 Threads
- **Base Frequency**: 4.00 GHz
- **L1 Instruction Cache**: 32.0 KB x 4
- **L1 Data Cache**: 32.0 KB x 4
- **L2 Cache**: 256 KB x 4
- **L3 Cache**: 8.00 MB x 1
- **Memory Information**: Memory 16.00 GB

**Geekbench 5 Score**
- **Single Core Score**: 1175
- **Multi Core Score**: 3589

---

**System Information**
- **Operating System**: Ubuntu 20.04.3 LTS
- **Model**: ASUS System Pro X99R
- **Motherboard**: ASRock SUPER COMPUTER POLYMER CARBON D5233 PLUS

**CPU Information**
- **Name**: Intel Core i7-6700K
- **Topology**: 1 Processor, 4 Cores, 8 Threads
- **Base Frequency**: 3.40 GHz
- **L1 Instruction Cache**: 32.0 KB x 4
- **L1 Data Cache**: 32.0 KB x 4
- **L2 Cache**: 256 KB x 4
- **L3 Cache**: 8.00 MB x 1
- **Memory Information**: Memory 16.00 GB

**Geekbench 5 Score**
- **Single Core Score**: 1465
- **Multi Core Score**: 8078
Thanks!!

Questions & Comments