IEEE Hong Kong 6G Wireless Summit

13-14 September 2023
Hong Kong, China

IEEE ComSoc IEEE Communications Society | IEEE

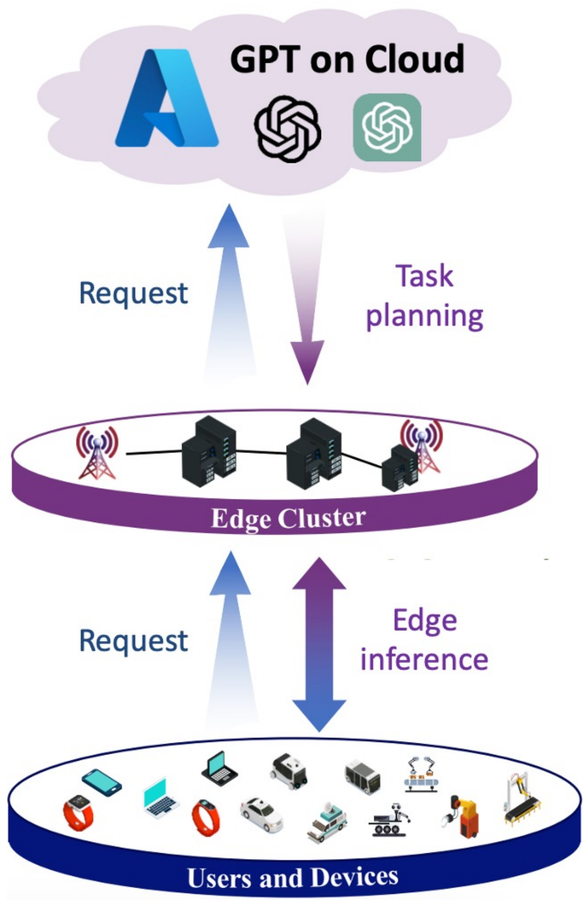
Hong Kong University of Science and Technology

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

EdgeGPT: GPT-Empowered Autonomous Edge Inference

Jiawei Shao and Jun Zhang

- **Edge AI** emerges as a promising solution to achieve connected intelligence by delivering high-quality, low-latency, and privacy-preserving intelligent services at the network edge.
- We introduce an **autonomous** edge AI system that automatically organizes, adapts, and optimizes itself to meet users' diverse requirements.
- The system employs a **cloud-edge-client architecture**, where the large language model, i.e., GPT, resides in the cloud for **task planning**, and other AI models are co-deployed on devices and edge servers for **edge inference**.



Step 1: Task planning

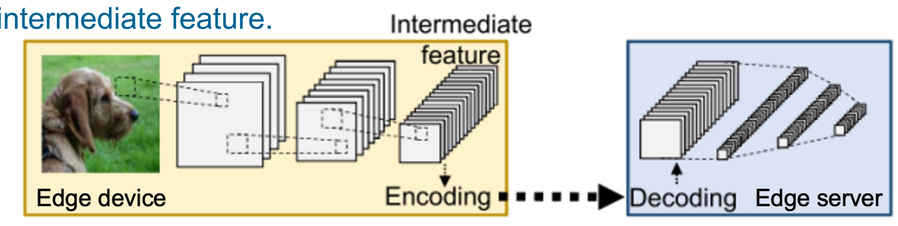
Prefix:
The AI assistant schedules the edge inference according to the *{{Request}}*. It should decompose the request into the tasks in *{{Available tasks}}*. The available resources include *{{Edge device list, Edge server list, Expert AI model list}}*. Here are several cases for your reference: *{{Demonstrations}}*.

Request:
Please monitor the user's emotions.

Task planning (Response of GPT):
{Task 1: Monitor the heart rate, {AI model: NA}, {Edge device: Wearable device}}
{Task 2: Predict mood from the heart rate, {AI model: Mood classification}, {Edge device: Mobile phone}}

Step 2: Edge Inference

Client-edge co-inference: Partition the AI model into two parts. Deploy them at the device and edge server, respectively. Compress and Transmit the intermediate feature.



Case study on various tasks

Request: I have captured an animal. What kind of animal is in the image?

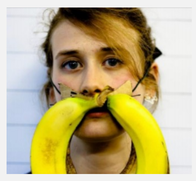
Task planning:
{Task 1: image classification, {AI model: ViT model}, {Edge device: Mobile phone}}
Inference result: Dog

Request: What's in the front of my car?

Task planning:
{Task 1: Capture image, {AI model: NA}, {Edge device: Vehicle}}
{Task 2: image caption, {AI model: blip-image-captioning-base}, {Edge device: Vehicle}}
Inference result: A horse carrying a large load of hay and two people sitting on it.

Request: An interesting image has appeared on the browser. Tell me what the mustache is made of in this image.

Task planning:
{Task 1: Visual question answering, {AI model: blip-vqa-base}, {Edge device: Tablet computer}}
Inference result: Bananas



Request: Detect how many objects are in the picture.

Task planning:
{Task 1: Object detection, {AI model: detr-resnet-101}, {Edge device: Mobile phone}}
{Task 2: image Classification, {AI model: ViT model}, {Edge device: Mobile phone}}
Inference result: One giraffe and two zebras.

Experiments

Setup

- Consider three vision tasks at the network edge, including image classification, image caption, and visual question answering.
- Edge device: Jetson Nano board. Edge server: GeForce RTX series. Cloud server: Microsoft Azure

Evaluation for task planning

- Compare the communication overhead and performance of different inference schemes.

Method	Image classification Cost↓	Image classification Accuracy↑	Image caption Cost↓	Image caption BLEU↑	Image caption CIDEr↑	Image caption SPICE↑	Visual question answering Cost↓	Visual question answering Test-dev↑	Visual question answering Test-std↑
Edge-only inference with lossless data compression	224.41 KB	**84.16%**	249.40 KB	**39.66**	133.25	**23.77**	372.58 KB	**78.24**	**78.32**
Edge-only inference with lossy data compression	33.86 KB	82.83%	19.16 KB	38.84	129.30	23.32	20.51 KB	77.22	77.38
Client-edge co-inference	**32.83 KB**	84.02%	**18.77 KB**	39.58	**133.29**	23.75	**20.39 KB**	78.22	78.22

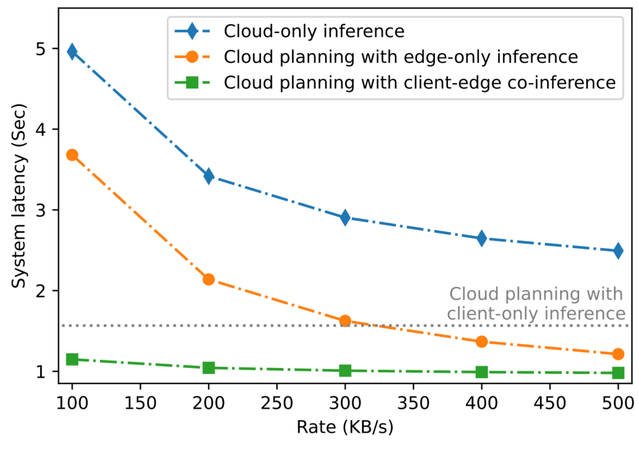
Evaluation for task planning

- Correctly understand users' requests.
- A quick response is preferred, but GPT models have a vast number of parameters.

Model	Acc↑	F1↑	Latency↓
GPT-3 350M	32.93%	33.95%	0.37 sec
GPT-3 6.7B	40.24%	42.31%	0.55 sec
GPT-3 175B	68.89%	74.70%	0.45 sec
GPT-3 175B IT	**84.44%**	**85.39%**	0.58 sec
Zero-shot Classification	36.59%	36.59%	**0.28 sec**

System end-to-end Latency

- System latency comprises two main components: task planning time and edge inference time.
- Our client-edge co-inference scheme achieves lower latency compared to baselines when the data rate between the device and edge server varies from 100 to 500 KB/s.



Full Paper