**SmartVision: Text Recognition & Language Model Interaction**

**Study on serving a Large Language Model on Cloudlet**

Anthony Chen & Blake Liu

Mentor: Thomas Eiszler

---

### Background

Situational awareness is defined as the perception of environmental elements and comprehending their meaning. OpenScout is a platform designed for automated situational awareness, which sends images to cognitive engines on backend server, and sends back the processed results.

We consider situations when user is around unfamiliar texts. Leveraging Large Language Models, we extend OpenScout capabilities by allowing users to perform proactive situational awareness. Users prompt SmartVision to help with comprehending text, enabling numerous proof-of-concept use cases, including translation, summarization, and context-aware QA. Customizable prompts extend SmartVision's capabilities to also be a versatile work assistant.

### Modules

**OCR (Optical Character Recognition)**

1. PytesseractOCR
   - Open Source solution with poor accuracy
   - Poor non-latin language support
   - Deprecation in project since error propagates to LLM

2. Microsoft OCR
   - Much better accuracy with 73 languages support
   - Able to capture large chunks of text

**LLM (Large Language Models)**

1. Llama 2
   - Open-access model released by Meta
   - Pretrained to predict and generate human-like text
   - Used as a reference solution in our project

2. Llama.cpp
   - Developed in C++ to optimize the execution of pre-trained models like Llama 2
   - Quantized checkpoints available for deployment
   - Deploy on various platforms with little dependency

3. OpenAI GPT API
   - Enterprise LLM solution, send prompt request and receive generated texts from SOTA GPT model
   - Used as a reference solution in our project

### System Architecture

**OCR**

- Text detection and extraction using Pytesseract
- Image processing for OCR

**LLM**

- Instruction Following: Capability to follow natural language instructions. Tested on a random sampled subset of IFElF 2.0 (e.g. Format output in bullet points, etc.)
- Multilingual Support: Given texts of 7 languages (including non-latin ones), check if LLM can generate correct English translation.
- Summarization Task: Ask LLM to summarize dozens of short news in DUC'06 dataset into few sentences, rate the results on 3 dimensions with G-EVAL and human annotation.
- Summarization Factuality Consistency (1-5): Factual alignment between the summary and the summarized source.
- Summarization Relevance (1-5): Selection of important content from the source.
- Summarization Fluency (1-3): Quality of the summary in terms of grammar, punctuation, word choice, and sentence structure.

### Lessons & Findings

- GPU VRAM usage & Memory Bandwidth are key factors determining speed
- When the entire model can fit into GPU, increasing GPU VRAM doesn't help
- Edge-based LLM brings privacy preserving advantage over Cloud/GPT API, considering institutional privacy risks.
- As well as usability in WAN disconnected settings.
- LLM response cache on edge is a promising future step to further reduce response time.

---

**Pipeline & Use Cases**

**Summarize Foreign Text Chunks**

- Tell me the key idea of these foreign texts

**Context-aware Question Answering**

- Which of the dishes are for vegetarians?

**System Evaluation & Lessons Learned**

**LLM Inference Time** takes the majority of total Latency

- Varies greatly on different input/output length, hardware, model size, etc.
- For the foreign text summarization task (input ~250tok, output ~80tok)
  - **Inference Time** = TTFT + TPOT * (number of tokens to be generated)

**OCR Latency**

- ~700ms

**Network + Other Processing (ms)**

-akin 1107

---

**Lessons & Findings**

- GPU VRAM usage & GPU Memory Bandwidth are key factors determining speed
- More VRAM provided by cloudlet to offload LLM layers leads to better performance
- When the entire model can fit into GPU, increasing GPU VRAM doesn't help
- Then Memory Bandwidth becomes inference bottleneck, the increase in Memory Bandwidth almost results in a proportional decrease in the TPOT.
- Edge-based LLM provides text generation on-par with GPT in our use cases, with similar but more stable latency.
- Edge-based LLM brings privacy preserving advantage over Cloud/GPT API, considering institutional privacy risks.

---

References: