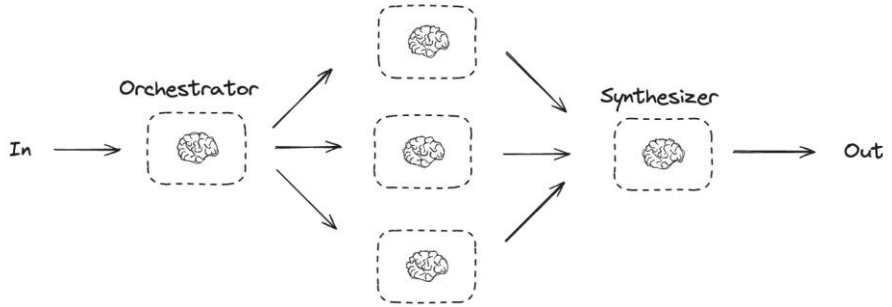
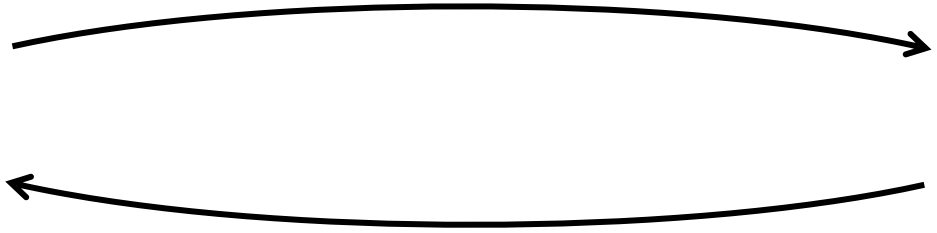
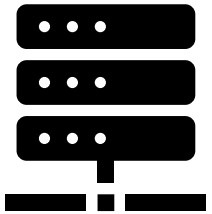
CS4262/5462 Project 1 – LLM Serving

LLM Serving Engine for Workflows or Chatbot



I have an inquiry about …





vLLM

Thanks for your inquiry….

Containerized LLM serving engine

- Track A. Agentic workflows with branches and bounded loops
  - Challenges: Data flow handling, batching strategies
- Track B. Customer service chatbot
  - Challenges: High concurrency, cold start

Track A. Dynamic Agentic Workflows

- Control flows: conditional branches & bounded loops
- Potential optimizations: Branch prediction & Precomputation
- Evaluation
  - Latency (Avg/P50/P95); throughput (workflows/second)
  - Average perplexity across all outputs; average trace length

Track B. Customer Service Chatbot

- High concurrency and short context length
- Repetitive patterns: different users might have same questions
- Potential optimizations: Improve caching strategies
- Evaluation
  - Latency (Avg/P50/P95); throughput (requests/second)
  - Average perplexity across all outputs

Starter Kit

- A codebase that implements all required APIs and logics
- A benchmark client that evaluates the performance and accuracy of an engine hosted at any URL
- Some sample requests data (Training set) for local testing
  - 1k workflows for track A
  - 13k chat requests for track B

Local Benchmark

- Local Deployment
  - Start the engine via Docker Compose
  - Call the engine at localhost
- Vast AI Deployment
  - Push the image to your GitHub Container Registry (GHCR)
  - Start a Vast AI instance using your image

Submission

- PDF Report
- Archived Source Code
- Full image reference on GHCR with SHA256 digest
- Read-only access token with the associated GitHub user name

Evaluation

- The evaluation data has three equal-size and non-overlapping split
  - Training set, Validation set, Test set
- We will host a leaderboard using the result on validation set
- Final grades will be based on the test set

Thanks for Listening!