Simplismart Submission

(By: Yogesh Dhyani)

As part of this assignment I have submitted to files by the llm_inferencing.ipynb & server.py.

1) **llm_inferencing.ipynb:** This notebook contains the code for model compilation (base model: lmsys/vicuna-7b) & quantization using llama.cpp for optimized inference, medusa head implementation from scratch, integration of medusa head with base model, speculative decoding implementation from scratch and dynamic batching
2) **server.py:** This scripts contains the code to serve llm using fastapi endpoint

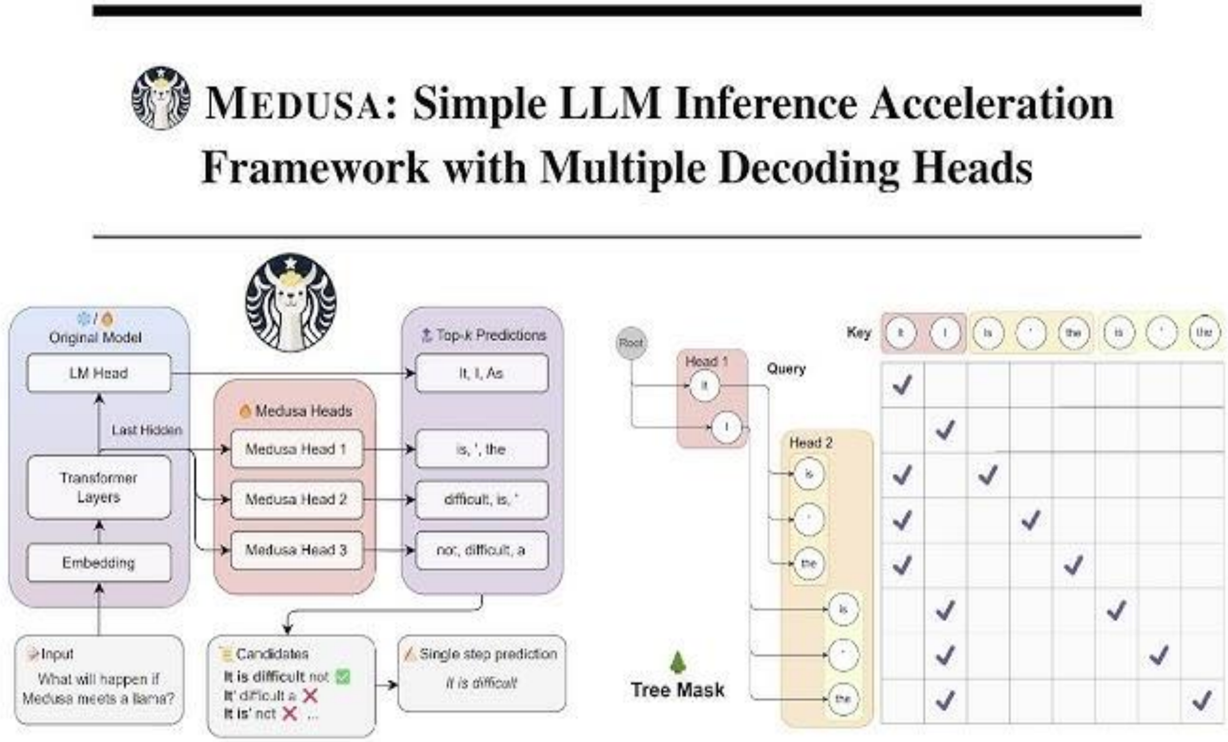
KEYPOINTS:

1. **Model Compilation:** I have used llama.cpp for model compilation because of it's highly optimized nature for cpu-based inference and also has a low memory footprint especially when combined with quantization techniques (e.g., Q4_K_M, Q5_K, etc). For better performance in

addition to converting optimized gguf format, I have quantized the model to 4 bit precision observing (2-2.5x) of performance gain

2. **Medusa Head Implementation & speculative decoding:** I have implemented medusa head from scratch as provided in the research paper. I have integrated it with base model creating 5 medusa head.

   While implementing speculative decoding the idea was medusa head will be used to guess the next set of tokens. For example, in my case I have used 5 medusa head where each medusa head will be responsible for guessing a single token. medusa_head_1 will guess first token, medusa_head_2 will guess second token and so on. Only when the medusa_head_1 token is guessed correctly we will go for verification medusa_head_2 token and so on. In this way, in the best scenario, we could guess 6 tokens in 1 forward pass(1 from base model + 5 from medusa head)



MEDUSA: Simple LLM Inference Acceleration Framework with Multiple Decoding Heads

3. **Dynamic Batching:** The idea while implementing dynamic batching was there will be a fixed time window and how many request we receive during that window will form a batch and also there will be a max cap on the batch size
4. **FastAPI Endpoint:** server.py script contains the code to serve optimized llm model using fastapi endpoint with host= '127.0.0.1' and port= 7860. I have used ‘vicuna-7B_Q4_K_M.gguf’ model for testing