Research Task

What LLM stack (OpenAI / Hugging Face / open-source like LLaMA) would you recommend and why?

Recommended Model based on testing: **Mistral-8x7B-Instruct-v0.1**

I recommend using an **open-sourced model** instead of a large proprietary model like those from OpenAI or Anthropic. The task doesn't require deep reasoning or complex understanding of textual relationships, and the context size is expected to be small—likely just a portion of the chat history (important part) rather than the entire conversation. Also, open-source models are cost efficient and secures privacy of company data.

How did I come to this conclusion?

I took a hands-on approach to find a winner.

1. **Built a Custom Test:** Created a custom dataset with 20 real-world scenarios. Each scenario included a mock user-profile, a sample chat transcript (like "I am worried about my career"), and a pre-defined "perfect" recommendation to measure against (ground truth).
2. **We compared each:** We tested various LLMs on these 20 test cases. To grade them fairly, we used a very powerful model (Gemini-2.5-pro) to act as an *judge,* ensuring the evaluation was consistent and unbiased.
3. **What was the Evaluation Criteria?:** As mentioned, we already had the ground truth (top 3 recommendations) for each of these 20 test cases. We scored each model's attempt against this key:
    a. **Top-**1 Accuracy: Did the model recommended the single most important expert? We checked if its #1 pick matched our #1 ground truth.
    b. **Top-**3 Overlap: How many of its three suggestions were correct? We counted how many of its recommended astrologers were in ground truth.
    c. **Reasoning Quality:** Was the "Why" behind its choices logical? We checked if the model's explanation made sense of it was just guessing to justify its picks (This was analysed with Gemini-2.5-pro model)

The Result are clear:
Mistral's 7B instruct model clearly performs best with 90% across the board. It understood the user's problems, even the subtle ones, almost perfectly.
You can find detailed evaluation results at – notebooks/llm-testing.ipynb.

Here's our evaluation metrics results –

Metric	Score	Percentage / Rating
Top-1 Accuracy	19 / 20	95%
Top-3 Overlap	56 out of 60 possible	93.3%
Reasoning Quality	58 out of 60 possible	96.7%

How would you host and scale it (cloud provider, deployment options)?

If we use an open-sourced model, for inferencing we either have to host it in our own servers (in AWS, GCP etc) or use third-party inference providers like Together AI, Cohere, etc (Not recommended).

- **Cloud Provider:** We can use AWS, GCP or Azure. All of these offer powerful GPUs servers. I personally recommend **AWS** due to its ease of use and availability of tools and services. However, the best choice depends on which cloud provider the team is already using? Or Which is more easily managed and cost efficient? Etc.
- **Deployment:** We can use a managed service like AWS SageMaker. It lets us quickly launch a model without handling too many configurations. To scale better, as we grow, we will have to use **Kubernetes** (Amazon EKS). This is an industry standard for large-scale apps and give us more control over long-term costs.
- **Scaling:** We can use **auto-scaling.** This simply means our system will automatically add more server power when lots of users are active and shut it down when they are quiet. This keeps the app running fast for everyone while making sure we only pay for the resources we are using.

Estimate the monthly cost for hosting a production-level LLM inference system for 50,000 monthly active users.

Let's suppose we are using our selected model which is "Mistral's 8x7B instruct".

Calculating the cost:

- **The price is $0.60 per million tokens.** A 'token' is basically a word or piece of word. To figure out the cost of one recommendation, we need to estimate how many tokens we send and receive.
    - **What we send (Inputs):** It include instructions, full list and details of astrologers, the user's profile and their chat history.
        - **Estimated 850 tokens.**
    - **What we receive (outputs):** The list of three astrologer recommendation and reasoning.
        - **Estimated 150 tokens.**

  Thus, a single recommendation might be around 1000 tokens.

  Cost per use = (1000/1000000) * 0.6 = $0.0006

- **Calculating the monthly bill:**
    - **Users**: 50,000
    - **Our Usage Assumptions**: Let's assume each active user get an AI-powered recommendation about 4 times-a-month.
    - **Total Recommendations**: 50000*4 = 2,00,000 recommendations monthly

  Total Monthly Cost = 200000*0.0006 each = $120

This is just a rough estimate and more costs will be incurred on deployment.

What privacy/safety concerns would you address?

- **Concern: Exposure of Sensitive User Chats**
    - **The Problem:** A leak of personal conversations about health or relationships would be devastating and destroy user trust.
    - **The Solution:** We will self-host the AI model, ensuring sensitive user data never leaves our secure environment. We will also automatically scrub all personal details (like names and locations) from the data before the AI sees it.
- **Concern: Biased or Unfair Recommendations**
    - **The Problem:** The AI could unfairly favor certain types of astrologers, leading to a poor and inequitable experience.
    - **The Solution:** We will continuously audit our recommendations for fairness and build a user feedback system. This allows us to catch and correct biases quickly.
- Concern: AI Giving Harmful or Irresponsible Advice
    - **The Problem:** The AI might dangerously answer direct questions like "Should I quit my job?"
    - **The Solution:** We will implement strict guardrails that prevent the AI from ever giving direct advice. Its sole purpose is to analyze the user's needs and suggest a human expert who can help, not to be the expert itself.