DailyDoseofDS Technical Writer Task

Pick any one task of your choice from the three tasks below and share your work with us. **Deadline: 14th March 2025.**

Read the full document for more information.

Task 1) RAG and Text-to-SQL

Objective: Your task is to write a **Twitter/X thread** on combining RAG and Text-to-SQL in a single query interface. You don't have to implement the application yourself. We have given you a reference code notebook below.

Reference notebook:
https://github.com/run-llama/llamacloud-demo/blob/main/examples/advanced_rag/llamacloud_sql_router.ipynb

Task 2) Patient Case Summary Workflow

Objective: Your task is to write a **Twitter/X thread** on building a patient case summary workflow. You don't have to implement the application yourself. We have given you a reference code notebook below.

Reference code:
https://github.com/run-llama/llamacloud-demo/blob/main/examples/document_workflows/patient_case_summary/patient_case_summary.ipynb

Task 3) Insurance claim workflow

Objective: Your task is to write a **Twitter/X thread** on building an auto insurance claim processing workflow. You don't have to implement the application yourself. We have given you a reference code notebook below.

Reference:
https://github.com/run-llama/llamacloud-demo/blob/main/examples/document_workflows/auto_insurance_claims/auto_insurance_claims.ipynb

Important instructions to complete the task:

How to complete the task:

- Download the reference notebook shared above.
- Gather any API keys (if needed) to make sure it is running on your local machine.
- Build a Streamlit interface (a sample is available here) to have a working application of the code. Use any AI tools/ChatGPT/Cursor to integrate streamlit.
- Write a Twitter thread about it detailing the implementation just like we have described above.

Examples of how your thread should look like:

- Example 1: https://x.com/akshay_pachaar/status/1895814021418451027
- Example 1: https://x.com/_avichawla/status/1896810340991803443
- Example 2: https://x.com/akshay_pachaar/status/1883497754649047192

Let's break down the above thread so that you can create it too:

- **Tweet 1**: The thread starts with what we shall build.
- **Tweet 2**: It then presents a quick demonstration of what is the final outcome [we don't expect you to create a video recording like we have above but if you can do that, great!]
- **Tweet 3**: Next, we have a tweet detailing the implementation overview with a diagram. **You must create a diagram**. It does not have to be high quality but you should create a diagram. We have shared the tool below in the document.
- **Tweet 4 onwards**: Moving on, we have a few tweets with code screenshots. **You must create code screenshots**. We have shared the tool to create it below in the document.
- **Last and 2nd last tweet**: Finally, we conclude with a call to action (like check the code here, follow us for more, etc.)
- **Important notes**:
    - In the thread, we never detail any streamlit code. You can skip that too. Only focus on core technologies used like LlamaIndex, Qdrant, Ollama, etc.
    - Each tweet in the thread must be <= 280 characters long.
    - Use AI/ChatGPT to write the content, we don't mind but make sure it looks natural and human.
    - Don't overuse emojis—AI tools overly produce them so if you use AI to write content, ensure you remove unnecessary emojis.

Tools to use:

- Use Snappify for code snippets: https://snappify.com/ [free plan]
- Use excalidraw to create diagrams and visuals: https://excalidraw.com/ [free to use]
- Create a free Typefully account, create a draft of your Twitter thread, and share it with us: typefully.com

How to share code:

- Make a PR to the following repository: https://github.com/patchy631/ai-engineering-hub
- Make sure that you have proper README with installation and running instructions. You can check existing examples in this repo for reference. Check this **sample README file**.

We can't wait to see what you produce. And trust us, it's not as hard as you think.

- **You know exactly what to do.**
- **You almost have the entire implementation with you.**
- **You know exactly what the output should look like.**
- **You know exactly which tools to use.**
- **You are allowed to use any AI tool to assist you with writing.**
- **All we expect is a thread that's as close to our work as possible—we don't mind whichever way you do it.**

Tech stack:

- LlamaIndex workflows for orchestration
- Local LLM using Ollama/vLLM
- Streamlit to build a UI (a must-have)
- Qdrant as vector database (optional, use when required)
- CometML's Opik for evaluation, tracing, and observability (whenever applicable)

Here's why this is important and why we expect you to care about it.

With just a team of 2 people, we reach more than 600,000 readers every day. These include ML engineers, CTOs, founders, and many more profiles. With that audience size, we take our content quality very seriously.

That is why the take-home assignment is a bit challenging. Realistically, we only expect a handful of the 600 shortlisted candidates (maybe 5-10) will complete it.

But if you are one of those exceptional few, we'd love to see what you produce.

Of course, we don't expect 100% perfection.

So even if you complete this assignment while being directionally correct, you will have a much higher likelihood of getting hired.

We would love to see you among the 5-10 candidates who can complete this task. The task description, tools to use, expected output, and every piece of important information is available to you.

P.S. We're prepared to offer USD 40k–120k per year to our technical writers. If you have a more important project that will yield greater benefits for you, it might make sense to focus on that. But if your schedule is open and you're likely to spend that time on less productive tasks, devoting a couple of days to this can give you an incredible return on your time. Who knows...You may get the chance to share your writing with an influential audience—and be well-rewarded for it.