Current Problems Aziz Is Facing in LLM Integration

1. **Model Incompatibility in Colab (Phi-3)**
2. Issue: Running Phi-3-mini from Google Drive results in errors like:
    - `DynamicCache object has no attribute 'get_max_length'`
    - `flash-attention package not found`
3. Cause: Incompatible `transformers` version or missing optional dependencies like `flash-attn`.
4. Fix in Progress: Updating `transformers` resolves some errors, but causes dependency issues with `Colab`, `TensorFlow`, and other packages.
5. **Dependency Conflicts After Updating Packages**
6. After upgrading `transformers`, `numpy`, etc., dependency resolver throws errors:
    - `tensorflow 2.18.0 requires numpy<2.1.0,>=1.26.0`
    - `gcsfs 2025.3.2 requires fsspec==2025.3.2`
7. These conflicts persist even after runtime reset.
8. **Factory Reset Runtime Not Visible**
9. Confusion around where to find the option in Colab.
10. Fix: It should be under `Runtime > Factory reset runtime`, but may be labeled differently in some versions or themes.
11. **LLM Prompt Pipeline Working But LLM Fails**
12. Code works fine locally but fails in Colab when model is loaded from Drive.
13. Suspected Issue: Either model or tokenizer is corrupted/incompatible.
14. **Need for a Truly Standalone, Deployable LLM**
15. Requirement: A model that can be downloaded once and run entirely without an internet connection or external dependencies.
16. Rejected Options:
    - Phi-3 (not fully standalone, transformer-version sensitive)
    - LLaMA-3 (requires license and setup via Ollama or Meta's terms)
17. Shortlisted:
    - Mistral-7B-Instruct v0.2 (GGUF, local-friendly, minimal deps)
18. **Clarification Needed:**
19. Whether GGUF models (like Mistral) are compatible with `transformers` pipeline.

20. Whether to use `llama-cpp-python`, `ctransformers`, or other runtime.

21. **Next Immediate Tasks:**

22. Validate Mistral-7B GGUF quant version (Q4_K_M) on Colab.
23. Replace Phi-3 with Mistral in current pipeline.
24. Store finalized flowchart and parser logic in version-controlled repo.
25. Optionally: Add `llama.cpp` interface to simulate production deployment.

Let me know when you'd like me to generate shell commands, setup scripts, or Docker configurations to make your LLM truly portable.