El Colapso del RAG Local: Límites Cognitivos en Modelos Pequeños (8B)

Cuadernillo | Vol. 1

mercedev.es — 2026-05-08 | Fase 3 (Orquestación de Contenidos)

El Desafío (Síntoma)

El objetivo de la Fase 3 era construir un "Agente Bibliotecario" (`merci-librarian.py`) capaz de procesar notas crudas de la autora, cruzar esa información con el registro histórico de la bitácora (RAG Local) y redactar cuadernillos Markdown con formato YAML estricto (Zero-Shot).

Al utilizar modelos locales ligeros (como Llama 3 8B), el agente fallaba sistemáticamente. Inyectaba texto conversacional (`Here is the output:`), destruía el Frontmatter y sufría de *Recency Bias* (Sesgo de Recencia) y *Context Window Stuffing* (Colapso por exceso de contexto), ignorando la nota principal para limitarse a resumir ciegamente la bitácora.

La Maniobra (Lógica)

Se implementaron múltiples escudos arquitectónicos para intentar domar al modelo local: 1. **One-Shot Prompting:** Inyección de plantillas físicas para forzar el formato. 2. **Filtrado Semántico:** RAG optimizado que extraía solo las entradas de bitácora coincidentes con palabras clave de la nota, limitando el contexto a 3000 caracteres. 3. **Aggressive Output Sanitization:** Extracción quirúrgica con Python (`text.find("---\n")`) para amputar las alucinaciones conversacionales previas al YAML.

El Aprendizaje / Deuda Técnica

A pesar de aplicar técnicas avanzadas de contención, se demostró empíricamente que la ingeniería de *prompts* y la sanitización de *outputs* no pueden compensar la falta de capacidad cognitiva bruta. Los modelos locales de menos de 70B parámetros carecen de la atención sostenida necesaria para procesar RAG denso y mantener restricciones de formato estrictas simultáneamente.

Se aplica el principio *Fail-Fast*: la generación de documentación compleja se asume como territorio exclusivo de modelos Cloud de frontera. El script `merci-librarian.py` se retira del núcleo operativo y se relega a *Art de Coté*, a la espera de hardware local más potente o de la llegada de Small Language Models (SLMs) con mejor seguimiento de instrucciones.