IA más allá del prompt

RAG, automatización y memoria

2026-01-11

Enlaces

- segana@fen.uchile.cl
- https://segana.netlify.app
- https://www.linkedin.com/in/sebastian-egana-santibanez/
- https://github.com/sebaegana

¿Por qué ir más allá?

Cuando ya debemos pasar de la interacción un poco más simple e intentar estabilizar lo que obtenemos desde los LLM's.

Mensaje central

- Para que los LLM generen valor real en organizaciones, debemos estabilizar sus salidas mediante contexto controlado, flujos reproducibles y persistencia de conocimiento.

El problema: la inestabilidad de los LLM

Los LLM's son probabilísticos, no deterministas.

El mismo prompt → resultados distintos

Sensibles a:

- Orden del texto
- Contexto previo
- Temperatura
- Estado de la conversación

¿Que implica esto?

- Buen desempeño en demos
- Frágiles en producción
- Difíciles de auditar
- Resultados no reproducibles

¿Por qué el prompt no escala?

Aquí ocurre el quiebre conceptual

El prompt:

- Vive en la cabeza del usuario
- No es versionable
- No es auditable
- No es reproducible

Analogía potente

> Usar solo prompts es como entrenar un modelo "de memoria" cada vez.

Objetivo real: estabilizar la salida del modelo

Cambio de foco

No buscamos:

- "respuestas creativas"

Buscamos:

- consistencia
- trazabilidad
- control
- alineamiento con negocio

Introduce el concepto

LLM como componente, no como oráculo

Primer pilar: RAG (Retrieval-Augmented Generation)

Qué problema resuelve

- Alucinaciones
- Conocimiento obsoleto
- Dependencia del prompt

Qué hace RAG

Separa:

- conocimiento (documentos)
- razonamiento (modelo)

El modelo no inventa, recupera

Frase clave

RAG transforma un LLM genérico en un sistema experto contextual.

En detalle

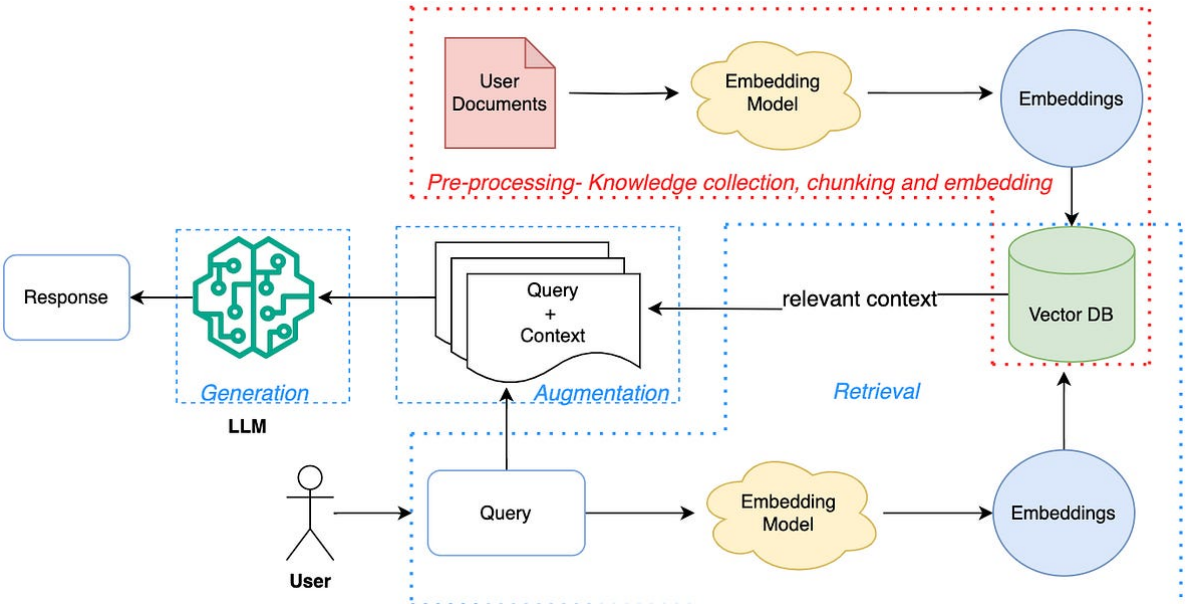

Figura 1: Recuperada en: https://mindfulmatrix.substack.com/p/build-a-simple-llm-application-with?utm_source=chatgpt.com

---

Segundo pilar: Automatización (orquestación)

Problema:

- Uso manual
- Resultados no repetibles
- Dependencia humana

Automatizar implica

- Flujos definidos
- Entradas controladas
- Salidas esperadas
- Evaluación automática

Ejemplo de un flujo

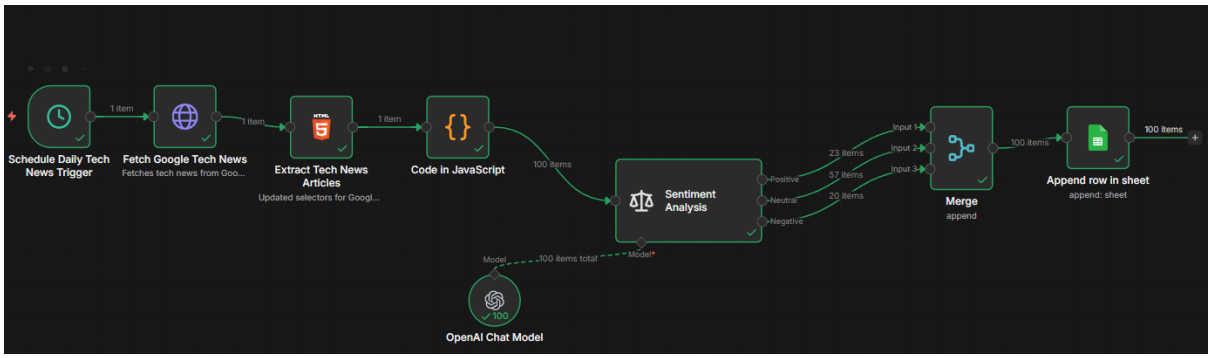

Figura 2: Ejemplo generado dentro del curso

¿Dónde entra el LLM?

System Instructions

Role: You are a highly intelligent and accurate sentiment analyzer.

Task: Analyze the sentiment of the provided text and categorize it into one of the following classes:

Positive

Neutral

Negative

Output Rules

Output only JSON

Follow the provided formatting instructions strictly

Do not include any additional text outside the JSON

Output Format Constraint

JSON Schema Compliance

The output must conform exactly to a given JSON Schema instance.

All required fields must be present.

No extra fields are allowed.

No trailing commas.

Schema:

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "type": "object",
  "properties": {
    "sentiment": {
      "type": "string",
      "enum": ["Positive", "Neutral", "Negative"]
    },
    "strength": {
      "type": "number",
      "minimum": 0,
      "maximum": 1,
      "description": "Strength score for sentiment in relation to the category"
    },
    "confidence": {
      "type": "number",
      "minimum": 0,
      "maximum": 1
    }
  },
  "required": ["sentiment", "strength", "confidence"],
  "additionalProperties": false
}
```

Frase clave:

> Sin automatización, la IA es solo una herramienta; con automatización, es un sistema.

El paso posterior a esto es la construcción de agentes en donde el LLM opera como un orquestador de distintas herramnientas, que pueden ser tan simple como una calculadora o más compleja como otro LLM, entre otras.

Tercer pilar: Memoria (estado y aprendizaje)

Qué NO es memoria

Historial infinito del chat

Qué SÍ es memoria

Persistencia selectiva

Estado del proceso

Preferencias

Decisiones pasadas

Tipos

Memoria de corto plazo (ventana de contexto): se mantiene dentro de la covnersación y está limitada por tokens

Memoria de largo plazo (vectores / DB): permite "recuerdar" en el largo plazo y habilita la posibilidad de personalizar al modificarla

Memoria operacional (estado del flujo): relacionada con agentes y orientada a determinar los pasos del proceso y el estado en el que se encuentra

---

La integración: Prompt + RAG + Automatización + Memoria

Aquí está la tesis completa

Componente Rol Prompt Interfaz humana RAG Contexto confiable Automatización Repetibilidad Memoria Continuidad

Mensaje central:

> La estabilidad no viene del prompt, sino del sistema.

Implicancias para organizaciones

Aterrizaje ejecutivo

Menos riesgo Menos alucinaciones Más confianza Escalabilidad real Auditoría y control

Cambio de mentalidad:

> No implementar IA → diseñar sistemas con IA.

Cierre

> El futuro de la IA no es escribir mejores prompts, es diseñar mejores sistemas.

Referencias

Gandhi, S. (2024, April 7). Building LLM application using RAG: How to query your document using LLM. Mindful Matrix. Retrieved from https://mindfulmatrix.substack.com/p/build-a-simple-llm-application-with