



RETRIEVAL-AUGMENTED
GENERATION (RAG)

La Revolución en Sistemas de IA Confiables

ALUMNO: FRANCO SOLIMANO CURE

Tecnología Emergente 2025



Tabla de Contenido

01

Introducción

02

Arquitectura
RAG

03

Implementación
Práctica

04

Sistema FAQ
Inteligente

05

Ventajas
Comprobadas

06

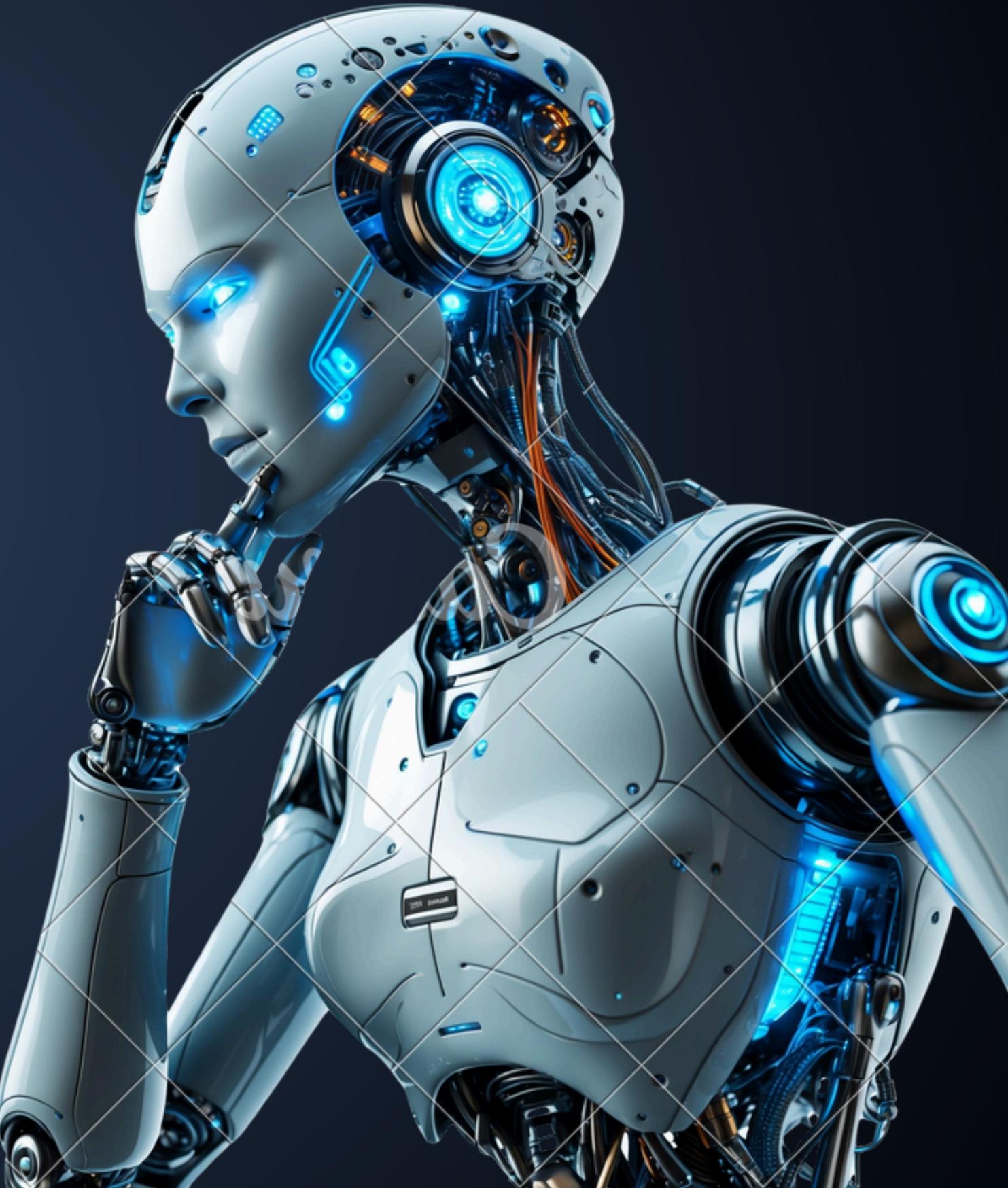
Desafíos
Identificados

07

Tendencias y
Mejoras

08

Conclusiones y
Futuro



Introducción

EL PROBLEMA ACTUAL:

¿Sabía que el 79% de sus empleados están insatisfechos con los sistemas de búsqueda actuales?

Las organizaciones enfrentan una crisis de acceso a la información. A pesar de contar con datos extensos, el 79% de los empleados reporta insatisfacción con los sistemas de búsqueda empresarial existentes.

Los modelos de lenguaje tradicionales no pueden acceder a información organizacional específica como contratos o documentos internos, lo que lleva a generar respuestas sin base factual conocidas como alucinaciones.



LA REVOLUCIÓN RAG

Retrieval-Augmented Generation emerge como un framework transformador que combina la capacidad generativa de los LLMs con bases de datos organizacionales internas.

Esta arquitectura permite acceder de manera confiable a información verificada de la empresa, reduciendo significativamente el riesgo de alucinaciones y proporcionando respuestas contextualizadas basadas en datos reales.



Arquitectura RAG

La arquitectura RAG se basa en tres componentes principales que trabajan en secuencia. El proceso inicia con una consulta del usuario y finaliza con una respuesta contextualizada, pasando por fases de recuperación, aumento y generación que garantizan precisión y relevancia.

Retrieval (Recuperación)

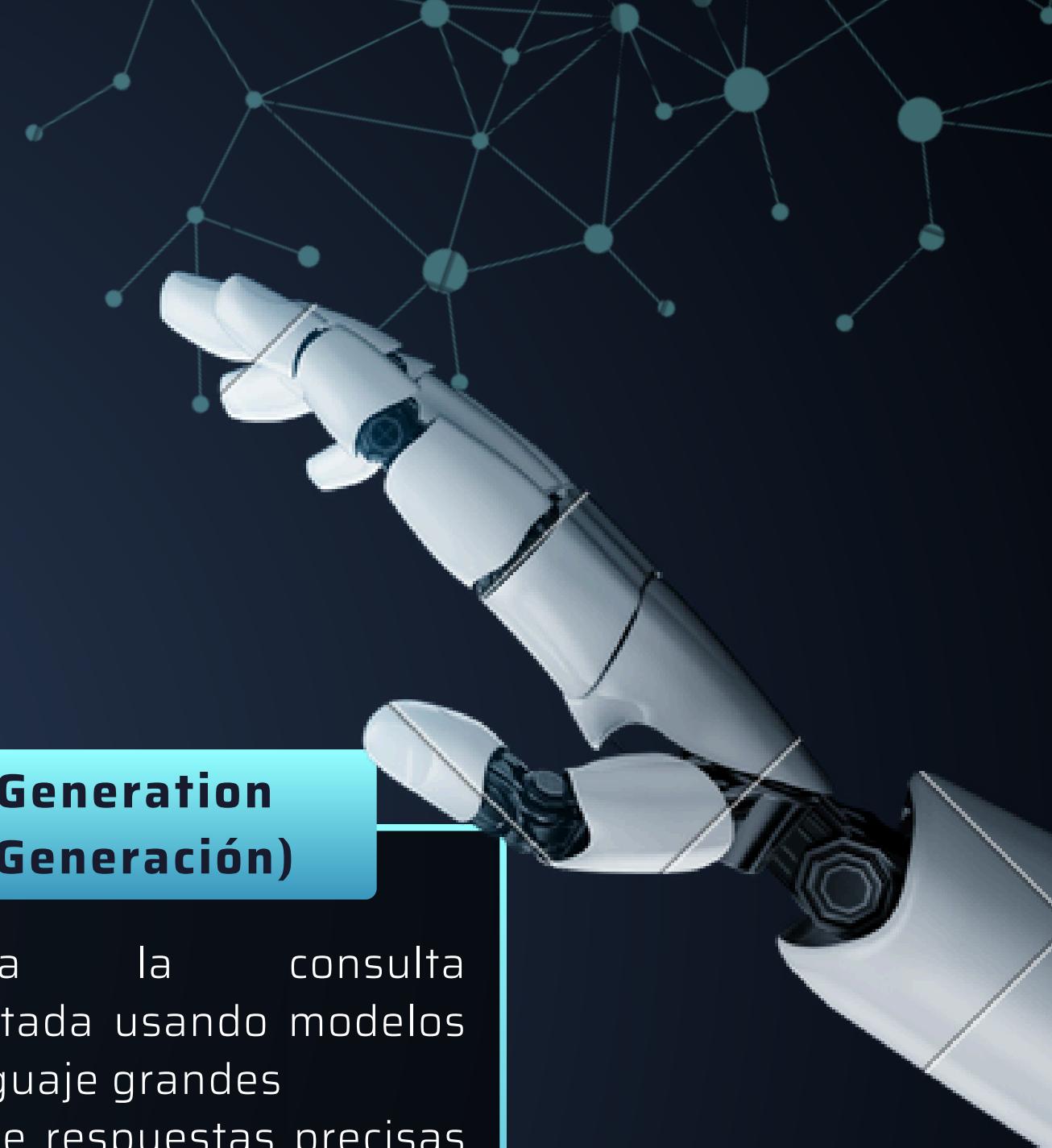
- Traduce consultas y documentos a vectores mediante modelos de embedding
- Busca en bases de datos vectoriales los fragmentos más relevantes
- Utiliza principios de ranking por similitud para identificar información contextu

Augmentation (Aumentación)

- Combina la consulta original con el contexto recuperado
- Utiliza plantillas de aumento como: "Use el siguiente contexto: [contexto]. Dada esta información, responda: [consulta]"
- Enriquece la consulta con información específica y verificada

Generation (Generación)

- Procesa la consulta aumentada usando modelos de lenguaje grandes
- Produce respuestas precisas basadas en el contexto proporcionado
- Puede incluir referencias a los documentos fuente originales



Arquitectura RAG

La arquitectura RAG se basa en tres componentes principales que trabajan en secuencia. El proceso inicia con una consulta del usuario y finaliza con una respuesta contextualizada, pasando por fases de recuperación, aumento y generación que garantizan precisión y relevancia.

Retrieval (Recuperación)

- Traduce consultas y documentos a vectores mediante modelos de embedding
- Busca en bases de datos vectoriales los fragmentos más relevantes
- Utiliza principios de ranking por similitud para identificar información contextu

Augmentation (Aumentación)

- Combina la consulta original con el contexto recuperado
- Utiliza plantillas de aumento como: "Use el siguiente contexto: [contexto]. Dada esta información, responda: [consulta]"
- Enriquece la consulta con información específica y verificada

Generation (Generación)

- Procesa la consulta aumentada usando modelos de lenguaje grandes
- Produce respuestas precisas basadas en el contexto proporcionado
- Puede incluir referencias a los documentos fuente originales



MEMORIA PARAMÉTRICA VS NO PARAMÉTRICA

MEMORIA PARAMÉTRICA (LLMs)

La memoria paramétrica representa el conocimiento adquirido durante el entrenamiento del modelo, almacenado directamente en sus parámetros internos. Esta memoria contiene información general proveniente de corpus masivos de texto como Wikipedia y Common Crawl, pero queda fijada en el momento del entrenamiento.

Los modelos actuales manejan escalas extraordinarias - desde los 7 billones de parámetros de Mistral 7B hasta los billones de parámetros en modelos más avanzados. Cada parámetro contribuye a representar patrones lingüísticos y conocimiento general, aunque sin capacidad de acceder a información específica de organizaciones o datos actualizados posteriores al entrenamiento.

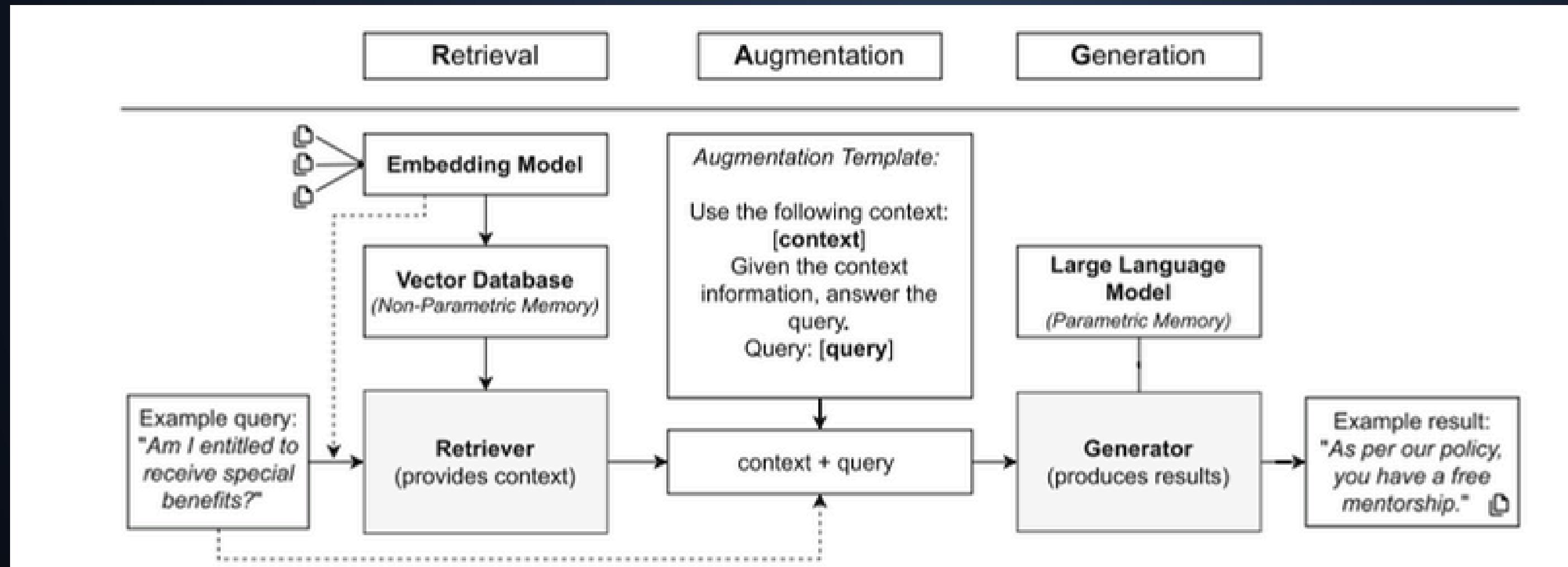
MEMORIA NO PARAMÉTRICA (BASES DE DATOS)

La memoria no paramétrica funciona como un sistema de almacenamiento externo independiente del modelo. Esta arquitectura permite integrar conocimiento específico que nunca fue parte del entrenamiento original, incluyendo documentos organizacionales, políticas internas, datos de contratos e información operativa actualizada.

Las bases de datos vectoriales permiten búsqueda semántica donde tanto las consultas como los documentos se representan como vectores, calculando similitudes para recuperar el contexto más relevante. Esta aproximación ofrece flexibilidad total para actualizar y expandir el conocimiento sin necesidad de reentrenar el modelo completo.

DIAGRAMA DE FLUJO RAG

Consulta → Embedding → Vector DB → LLM → Respuesta



Implementación Práctica

MODELOS DE EMBEDDING

Los modelos de embedding son el componente fundamental que permite la magia de RAG. Estas arquitecturas transforman texto, imágenes o audio en representaciones vectoriales que capturan el significado semántico. El mismo modelo debe aplicarse consistentemente tanto para crear la base de datos como para procesar las consultas de usuarios, garantizando que las búsquedas por similitud funcionen con precisión.

BASES DE DATOS VECTORIALES

Las bases de datos vectoriales almacenan estos embeddings organizados para permitir búsquedas ultra-rápidas. Utilizan métricas como la similitud coseno para identificar los documentos más relevantes, devolviendo típicamente los "top-k" resultados mejor alineados con la consulta del usuario. Esta aproximación sigue el principio de ranking probabilístico que prioriza la relevancia sobre otros factores.

Flujo plain vanilla vs enhanced RAG

Flujo Plain Vanilla:

- Retrieval: Busca información relevante en base de datos vectorial
- Augmentation: Genera consulta extendida con plantilla de aumento
- Generation: LLM produce resultados usando contexto aumentado

Enhanced RAG:

- RAPTOR: Incrusta, agrupa y resume recursivamente texto
- GraphRAG: Extrae grafos de conocimiento jerárquicos
- RAFT: Combina RAG con fine-tuning para dominios especializados
- RAT: Mejora aumento mediante Cadena de Pensamiento (CoT)

Sistema FAQ Inteligente

Problema Tradicional:

Los sistemas tradicionales de preguntas frecuentes (FAQ) utilizan un conjunto de preguntas y respuestas estáticas "predefinidas". Estas soluciones convencionales están limitadas a respuestas preestablecidas y no pueden manejar consultas específicas formuladas en lenguaje natural por los usuarios.

Solución con RAG:

La Generación Aumentada por Recuperación permite desarrollar un agente sofisticado de respuesta a preguntas que incorpora conocimiento específico del dominio organizacional. Este sistema basado en RAG responde con precisión a preguntas específicas formuladas en lenguaje natural por el usuario, reemplazando el método tradicional y mejorando así la experiencia general del usuario.

Ejemplo de Implementación:

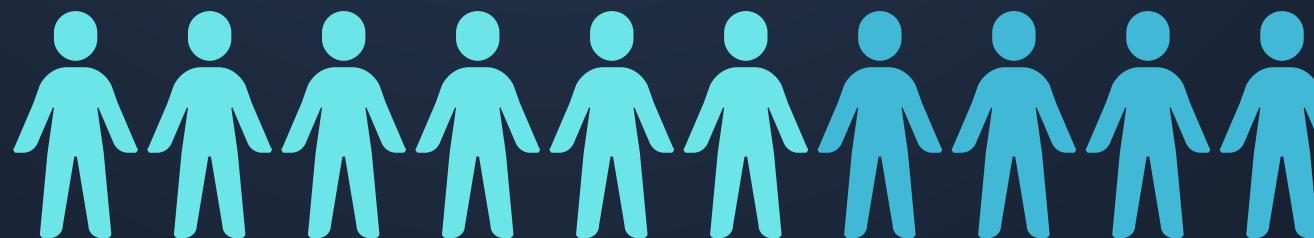
- Consulta del usuario: "¿Tengo derecho a recibir beneficios especiales?"
- Proceso RAG:
 - a. Recupera documentos relevantes de políticas internas
 - b. Aumenta la consulta con el contexto encontrado
 - c. Genera respuesta precisa basada en los documentos
- Resultado: "Según nuestra política, usted tiene derecho a una mentoría gratuita" + referencia al documento de políticas

VENTAJAS COMPROBADAS

CARACTERÍSTICA	RAG	FINE-TUNING	MODELOS BASE
Reducción de Alucinaciones	70-80% menos alucinaciones	30-40% menos alucinaciones	Alto riesgo de alucinaciones
Costo de Implementación	50-70% más económico	Alto costo computacional	Costo moderado (solo inferencia)
Actualización de Conocimiento	Tiempo real	Requiere reentrenamiento	Conocimiento estático
Capacidad de Referenciación	Referencias verificables a documentos	Sin referencias específicas	Sin capacidad de referencia
Flexibilidad de Dominio	Múltiples dominios simultáneos	Especializado en un dominio	Dominio general limitado
Tiempo de Implementación	Días/semanas	Semanas/meses	Inmediato
Recursos Computacionales	Bajos-Medios	Muy Altos	Medios
Gestión de Sesgos	Control activo de sesgos	Sesgos incorporados al modelo	Sesgos del entrenamiento original

Desafíos Identificados

- Blinkered Chunk Effect (BCE): Limitación para comprender el contexto global cuando se usan fragmentos de texto.
- Gestión de Datos y MLOps: Necesidad de nuevas capacidades para fusionar fuentes heterogéneas y garantizar calidad.
- Nuevos Sesgos: Riesgo de introducir sesgos presentes en los datos corporativos.
- Efectividad del Retrieval: Depende de la calidad del sistema de clasificación y ranking.



Tendencias y Mejoras (Enhanced RAG)

RAPTOR

Recuperación recursiva y resumen en múltiples niveles de abstracción

GraphRAG

Utiliza grafos de conocimiento para una mejor estructuración de la información

RAFT

Combina RAG con Fine-Tuning para dominios altamente especializados (ej. medicina)

RAT

Mejora el razonamiento mediante Cadenas de Pensamiento (CoT)

Conclusiones y futuro

1

Impacto en Organizaciones: RAG permite crear Sistemas de Información basados en IA más confiables y contextualizados.

2

- Direcciones de Investigación:
 - Organizacional: Cómo gestionar capacidades de datos y arquitectura TI.
 - Individual: Cómo el "grounding" afecta la confianza y satisfacción del usuario (XAI).
 - Económico: Cómo RAG puede proporcionar ventaja competitiva y ayudar al cumplimiento normativo (ej. Ley de IA de la UE).

