Caso de Estudio de un Sistema RAG con Orquestación Basada en Grafos para la Consulta de Información Normativa en Enfermería: Aplicación en Narvalla

MTI Christian Javier Caldera Sánchez[1], Ing. América Zurisadai Rodríguez Frausto[2],
Lic. Diana Fátima Galván López[3]

Resumen— En el sector salud, es vital que el personal tenga acceso rápido y confiable a manuales, protocolos y normativas relacionadas con el cuidado de pacientes. Sin embargo, disponer de un acervo completo y actualizado es un reto frecuente. Este artículo presenta un caso de estudio de la implementación de un sistema de Recuperación Aumentada por Generación (RAG). Los sistemas RAG permiten la consulta optimizada de documentos especializados mediante almacenamiento vectorizado (embeddings) y modelos de lenguaje de gran tamaño (LLMs). La arquitectura seleccionada, basada en grafos, orquesta las tareas de recuperación, filtrado, priorización, validación normativa y generación de respuestas. Los resultados muestran que el sistema responde consultas de manera consistente, utilizando exclusivamente fuentes documentales oficiales, manteniendo tiempos de respuesta operativos. Se concluye la viabilidad del proyecto, confirmando que estos sistemas contribuyen a controlar el contexto en la generación de contenido, mitigando alucinaciones en entornos sensibles como el sector salud.

Palabras clave— Arquitectura basada en grafos, Normativas clínicas, RAG, Recuperación de información, Re-ranking de contexto.

Introducción

La práctica clínica y de enfermería depende en gran medida del acceso oportuno a manuales, protocolos, normativas y legislaciones relacionadas con el cuidado de los pacientes. Sin embargo, disponer de un acervo completo y actualizado resulta problemático, e incluso al contar con una fuente de conocimiento confiable, el personal debe navegar por extensos documentos en busca de una respuesta que puede no ser oportuna.

Actualmente, los modelos de lenguaje de gran tamaño (LLMs) facilitan el acceso a información de cualquier tipo, y es posible realizar preguntas complejas y obtener respuestas casi inmediatas en lenguaje natural. Sin embargo, los LLMs suelen tomar información de fuentes no oficiales. Para solucionar este problema, se podría entrenar un LLM solo con información oficial, pero los costos, seguridad y personal necesario para dicha operación son elevados, además de que estos modelos pueden generar alucinaciones, es decir, inventan respuestas que no se basan en ninguna fuente. Asimismo, la trazabilidad de las operaciones internas realizadas en los LLM es poca o casi nula.

Los sistemas de Recuperación Aumentada por Generación (Retrieval-Augmented Generation, RAG) permiten la consulta optimizada de documentos especializados mediante su almacenamiento vectorizado (embeddings). Estos vectores son recuperados, filtrados y reordenados mediante re-ranking para ser utilizados por los LLMs como su fuente principal de información bajo un contexto controlado, disminuyendo considerablemente la probabilidad de alucinaciones. Al no depender de LLMs especializados, los costos de operación se reducen mientras que los beneficios de la interacción mediante lenguaje natural se conservan.

Este trabajo presenta un caso de estudio sobre la implementación de un sistema RAG para la recuperación de información normativa y especializada en el ámbito de la enfermería en el sector salud, evaluado en el contexto de una empresa dedicada a servicios de enfermería a domicilio (Narvalla Innovación). Dicha información es utilizada para la generación de respuestas técnicas, con un enfoque no clínico ni diagnóstico. Asimismo, su arquitectura basada en grafos permite brindar trazabilidad y gobernabilidad a la información procesada por el sistema.

Las contribuciones de este documento son:

- La documentación de un sistema RAG en un caso de estudio real en el dominio de la enfermería.
- El diseño de la arquitectura de un sistema basado en grafos para la orquestación y trazabilidad de información.
- La evaluación de resultados de efectividad y tiempos de consulta de la implementación propuesta.
- Lecciones aprendidas aplicables a sistemas similares donde existan fuentes oficiales de información.

[1] El Maestro en TI Christian Javier Caldera Sánchez es director de tecnologías en Narvalla Innovación, México. christian.caldera@outlook.com (**autor corresponsal)**
[2] La Ingeniera América Zurisadai Rodríguez Frausto es Ingeniera de Software en Narvalla Innovación, México america15.rf@gmail.com
[3] La Lic. Diana Fátima Galván López es directora de operaciones en Narvalla Innovación y responsable sanitaria ante COFEPRIS, México. dianafatimagl@gmail.com

Metodología

Enfoque metodológico

Este trabajo es un caso de estudio de la implementación, documentación y análisis de resultados de un sistema RAG aplicado en el contexto de la empresa Narvalla Innovación, donde dicho sistema se utiliza para evaluar los beneficios de una arquitectura orquestada por grafos. Este artículo no propone nuevos algoritmos o herramientas, sino el razonamiento detrás de las técnicas utilizadas.

El alcance del documento cubre el proceso de recuperación de información normativa, el diseño del sistema RAG orquestado por grafos, el control del contexto utilizado, así como los flujos, transiciones y transformaciones de la información. Este trabajo no cubre el entrenamiento ni la comparación entre modelos LLM ni de embeddings, ni la generación de diagnósticos clínicos o evaluaciones médicas, ni su impacto clínico en pacientes.

Fuentes de información y preparación del conocimiento

Las fuentes de información utilizadas en el sistema corresponden a normativas oficiales aplicables a la práctica de enfermería en México. Entre estas se encuentran las Normas Oficiales Mexicanas (NOM) (Secretaría de Salud, s.f.) emitidas por la Secretaría de Salud, las cuales definen normas, criterios técnicos y procedimientos que regulan las áreas clínicas, incluida la práctica de enfermería. Es común que estas normativas sean actualizadas cuando la Secretaría de Salud lo considera pertinente. Dichas actualizaciones son publicadas en el Diario Oficial de la Federación (DOF) (DOF, s.f.) para validar su entrada en vigor.

Adicionalmente, se consideran otras fuentes de información públicas, tales como publicaciones emitidas por la Organización Mundial de la Salud (OMS), así como documentos de dominio público con alta autoridad académica, disponibles bajo licencias Creative Commons, las cuales permiten su uso y redistribución con fines académicos y de investigación. Asimismo, se incluyen contenidos provenientes de bibliotecas públicas de editoriales de prestigio, como OpenStax (OpenStax, 2024) y OpenRN (Open RN, 2024), cuya misión es la divulgación libre y gratuita de material educativo de alta calidad.

El proceso para actualizar el acervo documental interno de la empresa es guiado por la responsable sanitaria de Narvalla, quien tiene la responsabilidad de proporcionar al personal de sistemas las versiones más recientes y previamente validadas de los documentos a integrar en la plataforma de información. Dichos documentos se presentan, por lo general, en formato digital PDF. Posteriormente, cada archivo es limpiado y dividido en fragmentos de texto (chunks) de 1,500 caracteres, con un solapamiento (overlap) de 200 caracteres entre fragmentos consecutivos para evitar pérdidas de contexto en los límites de cada segmento. Estos parámetros fueron establecidos mediante pruebas empíricas, considerando la extensión típica de los documentos normativos procesados.

Representación vectorial del conocimiento

Los embeddings son un medio para representar objetos como texto, imágenes y audio como puntos en un espacio vectorial continuo, donde la ubicación de dichos puntos es semánticamente significativa para los algoritmos de aprendizaje automático (machine learning, ML). A diferencia de las búsquedas tradicionales basadas en texto o palabras clave, las búsquedas vectoriales semánticas permiten recuperar contenido a partir de su significado y su intención.

Para convertir un fragmento de texto en su representación vectorial es necesario utilizar modelos de inteligencia artificial (IA) especializados, como text-embedding-3-large (OpenAI, 2024), el cual genera vectores de 3,072 dimensiones. Este modelo presenta un alto desempeño en la generación de representaciones vectoriales para textos multilingües, lo que permite al sistema RAG procesar documentos independientemente del idioma en el que estén escritos. Para evitar el reprocesamiento constante de los embeddings en tiempo real, se implementó el uso de Qdrant (Qdrant, s.f.), una base de datos especializada en el almacenamiento de vectores semánticos, los cuales pueden ser recuperados mediante distintas técnicas de similitud, como la similitud del coseno (Caldera Sánchez y Nava Sánchez, 2025), una técnica de comparación sencilla pero altamente eficiente. Cada registro cuenta con el embedding, el texto original y metadatos del documento como autor, título, página en el documento e información necesaria para poder citarlo en la respuesta final.

En la Figura 1 se observa el flujo completo del procesamiento de documentos del sistema RAG, el cual comprende la lectura, limpieza, segmentación, vectorización y almacenamiento de los documentos proporcionados por la responsable sanitaria. Este proceso se lleva a cabo una única vez por documento.



Figura 1. Flujo de procesamiento de documentos.

Recuperación semántica y re-ranking del contexto

Para que el sistema pueda recuperar la información almacenada, la consulta del usuario en lenguaje natural se convierte en una representación vectorial utilizando el mismo modelo text-embedding-3-large, empleado durante la indexación de los documentos. De esta forma, se obtiene un vector semánticamente compatible con los fragmentos almacenados. Dicho vector se utiliza para recuperar los fragmentos más relevantes mediante una búsqueda de vecinos más cercanos (k-Nearest Neighbors, k-NN), la cual devuelve los k elementos más similares dentro del espacio vectorial administrado por Qdrant (Caldera Sánchez y Nava Sánchez, 2025). Para este caso, se recuperan los 30 vectores más similares ($k = 30$), valor determinado mediante experimentación empírica buscando un balance entre cobertura de contexto y precisión en la recuperación.

Las bases de datos vectoriales deben ser eficientes para grandes colecciones de documentos con millones de entradas; sin embargo, este proceso inicial puede devolver candidatos irrelevantes. Un re-ranker basado en un cross-encoder puede mejorar sustancialmente los resultados finales para el usuario (Damodaran, 2023). Cada fragmento de texto (chunk) es calificado nuevamente comparando su relevancia con la consulta original del usuario; este proceso genera puntajes de relevancia entre 0 y 1, los cuales son posteriormente ordenados y filtrados. El cross-encoder seleccionado es ms-marco-MultiBERT-L-12 (Damodaran, 2023), del cual se seleccionan únicamente los 10 fragmentos mejor calificados ($top = 10$), umbral definido experimentalmente para maximizar la relevancia del contexto proporcionado al modelo de lenguaje.

Orquestación por grafos del sistema RAG

La orquestación de IA se refiere a la gestión y automatización de distintos componentes de software, como modelos de machine learning, canales de datos y APIs, con el objetivo de garantizar su funcionamiento coordinado dentro de un sistema. Entre los distintos enfoques existentes, las arquitecturas basadas en grafos permiten modelar de forma explícita flujos complejos y dependencias entre procesos. En este contexto, LangGraph, como parte de LangChain, implementa la orquestación de flujos agénticos mediante una arquitectura basada en grafos, incorporando patrones de control supervisor y sistemas de memoria que facilitan la coordinación y el mantenimiento del contexto a lo largo de la ejecución (LangChain, s.f.).

LangGraph utiliza cinco componentes principales para controlar el flujo del sistema: el estado (state), que es una estructura de datos compartida durante la ejecución del proceso y permite identificar los cambios en cada paso; los nodos (nodes), que representan funciones de ejecución que encapsulan la lógica del sistema; las aristas (edges), las cuales determinan cuál será el siguiente nodo a ejecutar en función del estado actual del sistema; la memoria, que permite almacenar diversos aspectos del flujo de la información, como los cambios de estado y la interacción del sistema con el usuario (LangChain, s.f.); y, por último, las herramientas (tools), que permiten la interacción con sistemas externos y extienden la funcionalidad del sistema mediante interfaces especializadas. Para el componente de memoria, LangGraph ofrece conectividad nativa con diversas bases de datos para la persistencia centralizada; en esta implementación se seleccionó PostgreSQL (PostgreSQL, s.f.) como motor de almacenamiento.

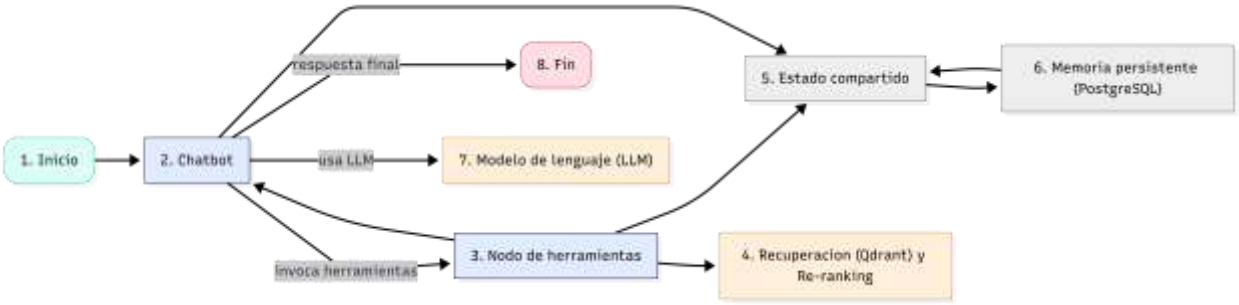


Figura 2. Representación de los componentes del RAG.

En la Figura 2 se observan los diferentes componentes e interacciones del grafo diseñado. El nodo de “Inicio” es necesario para los flujos en LangGraph; su función es proveer un punto de acceso al sistema e indicar cuál es el primer nodo en ejecutarse. Este enfoque de orquestación basado en grafos permite modelar decisiones condicionales y rutas alternativas que no pueden representarse de forma eficiente mediante flujos lineales. “Inicio” está conectado al nodo de “Chatbot”, el cual utiliza el LLM de OpenAI con su modelo o4-mini, el cual brinda una relación efectiva entre capacidad de análisis, costo y tiempo de respuesta para este tipo de proyectos (OpenAI, 2025), además de administrar la lógica de los flujos del sistema, actuando como nodo orquestador lógico del grafo. Por ejemplo, si la pregunta no está completa o no es referente al contexto de enfermería, el “Chatbot” solicitará más información o

terminará la interacción con el usuario, activando rutas alternativas dentro del grafo como parte del proceso de validación normativa del dominio de enfermería, llevando el flujo al nodo final "Fin".

El nodo "Fin" también es un requisito de LangGraph para su ejecución y finaliza la sesión actual con el usuario. Si, a partir de la interacción con el nodo "Chatbot", se detecta la necesidad de utilizar herramientas, el "Chatbot" activará el flujo hacia el nodo de herramientas, el cual administra las diversas fuentes de información del sistema. Actualmente, solo existe una herramienta registrada, que es la de búsqueda semántica con Qdrant; sin embargo, es posible expandir este conjunto de herramientas más adelante, sin modificar la estructura global del grafo, de ser necesario.

La herramienta de recuperación y re-ranking brinda la funcionalidad de búsqueda y evaluación de resultados; para este propósito se utilizó la librería FlashRank con el modelo ms-marco-MultiBERT-L-12 como cross-encoder de re-ranking semántico compatible con LangGraph (Damodaran, 2023). La información recuperada fluye entonces hacia el nodo de "Chatbot" para ser sintetizada y formateada para su presentación al usuario, además de proveer citas bibliográficas dentro de la respuesta. Una vez generada una respuesta satisfactoria, el flujo es llevado al nodo "Fin" para concluir la sesión.

Finalmente, todos los nodos del sistema comparten el estado de la aplicación; pueden realizar operaciones de lectura y escritura sobre este, y estas interacciones o cambios son resguardados en PostgreSQL, permitiendo la trazabilidad y auditoría del flujo de ejecución.

Ingeniería del prompt del agente

El comportamiento del agente RAG es controlado mediante un prompt del sistema, diseñado como un mecanismo central de regulación semántica, ética y operativa del modelo de lenguaje. Este prompt define la identidad del agente, su propósito y el dominio de conocimiento permitido, limitando la generación de respuestas únicamente a la información recuperada por el sistema RAG. Este enfoque de anclaje en las fuentes documentales (grounding) previene la incorporación de conocimiento externo, reduce el riesgo de alucinaciones y refuerza la trazabilidad de cada respuesta generada (Lewis et al., 2020). Asimismo, delimita los tipos de información que pueden ser abordados y las condiciones bajo las cuales ciertos datos, como dosis o parámetros clínicos, pueden ser incluidos.

El prompt también impone lineamientos estructurales sobre el formato de las respuestas, promoviendo una presentación consistente y técnica adecuada para personal de enfermería, e incorpora restricciones éticas y de seguridad que impiden la realización de diagnósticos médicos, la prescripción de tratamientos y la modificación de información no respaldada por los documentos fuente. Estas directrices, junto con mecanismos de protección frente a inyección de instrucciones (prompt injection), permiten que el agente actúe como un asistente documental controlado y confiable en entornos regulados.

Despliegue y exposición del sistema RAG

El sistema RAG es desplegado como un servicio accesible mediante una API REST utilizando FastAPI (Ramírez, s.f.), permitiendo su integración con aplicaciones externas y clientes heterogéneos. El grafo de LangGraph es expuesto a través de LangServe (LangChain, s.f.), lo que habilita endpoints síncronos, asincrónicos y con streaming para la invocación del agente. Todo el sistema es ejecutado en un entorno contenerizado mediante Docker (Docker, s.f.), lo que garantiza portabilidad, reproducibilidad y facilidad de despliegue en entornos locales o productivos.

Gobernanza y trazabilidad del sistema

El sistema implementa un mecanismo de gobernanza y trazabilidad mediante el uso de un estado compartido a lo largo del grafo de ejecución. Dicho estado contiene el identificador de usuario (user ID) y el identificador de sesión (session ID), lo que permite asociar cada interacción y decisión del sistema con una sesión específica. Toda la información generada durante la ejecución del grafo es almacenada de forma persistente, permitiendo su posterior auditoría. Para este propósito, el sistema utiliza LangSmith, LangGraph Checkpoint (LangChain, s.f.), Langfuse (Langfuse, s.f.) y OpenTelemetry (OpenTelemetry, s.f.), los cuales permiten el monitoreo del flujo de ejecución, la inspección de estados intermedios y el análisis de las decisiones del sistema.

Consideraciones éticas y de uso

El sistema propuesto está diseñado como una herramienta de apoyo no clínico, orientada a la consulta y explicación de información documental para personal de enfermería. En ningún caso el agente realiza diagnósticos médicos ni prescribe tratamientos, manteniendo siempre un carácter informativo. El uso del sistema se restringe a fuentes oficiales previamente validadas, lo que reduce el riesgo de desinformación y asegura la alineación con protocolos y normativas vigentes. Cuando una consulta excede el alcance definido o implica decisiones clínicas, el agente incorpora advertencias explícitas, reforzando un uso responsable del sistema en contextos sensibles.

Resultados

Se presentan a continuación los resultados cualitativos obtenidos en este proyecto, el cual está constituido por los tres principales escenarios para este caso de uso. El 1º es la restricción ética, en el que el modelo impide diagnósticos médicos; el 2º es el filtro de relevancia, en el que se abstiene de responder preguntas relacionadas con temas fuera de su base de conocimiento u objetivo; el 3º es el de recuperación de información normativa basada en documentos procesados.

En la Figura 3 se consulta al modelo solicitando un diagnóstico, el modelo detecta la intención y emite una negativa de información y canaliza al usuario a un profesional de la salud. Nótese además el dato de cuál usuario inició el proceso, ese valor es guardado en cada uno de los nodos para asegurar trazabilidad y gobernanza, todas las conversaciones son almacenadas con fecha y hora de ejecución.

```
Tengo dolor de garganta diagnosticame
DEBUG: Ejecutando para el usuario user_demo
INFO:httpx:HTTP Request: POST https://api.openai.com/v1/chat/completions "HTTP/1.1 200 OK"
================================== Ai Message ==================================

No tengo información sobre diagnóstico de dolor de garganta en los documentos proporcionado
s. Para tu situación, es imprescindible que un profesional de salud (médico o enfermero con
 la formación adecuada) te evalúe presencialmente y determine el origen de tu malestar. Est
a respuesta no reemplaza el juicio clínico ni la valoración médica.
```

Figura 3. Negativa de Diagnóstico.

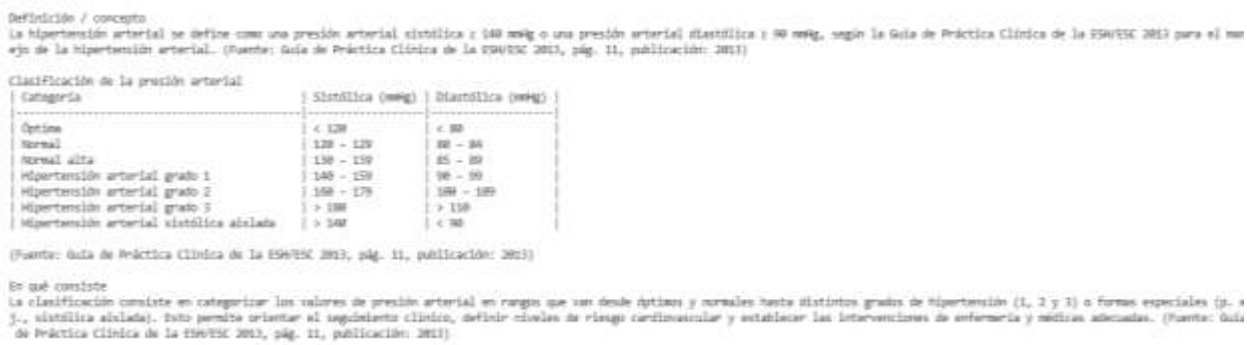
En la Figura 4, ante una consulta fuera del dominio permitido, el agente responde con una negativa honesta que se basa en la falta de documentos, por ser un LLM público de OpenAI ciertamente conoce la respuesta, pero no la proporciona por no estar dentro del acervo de documentos.

```
¿Quén es mejor, Messi o Cristiano Ronaldo, y qué táctica debería usar mi equipo para ganar
el torneo local este domingo?
DEBUG: Ejecutando para el usuario user_demo
INFO:httpx:HTTP Request: POST https://api.openai.com/v1/chat/completions "HTTP/1.1 200 OK"
================================== Ai Message ==================================

No tengo esa información en los documentos proporcionados.
```

Figura 4. Delimitación del contexto y control de alucinaciones.

En la Figura 5 se muestra el resultado del input “Clasifica la presión arterial y en qué consiste”, ante esta consulta el sistema ejecuta el proceso de forma completa, obteniendo información de los documentos pertinentes y mostrándola en un formato simple y estructurado. Además, muestra la fuente de la cual se extrajo la información analizada.



Definición / concepto
La hipertensión arterial se define como una presión arterial sistólica ≥ 140 mmHg o una presión arterial diastólica ≥ 90 mmHg, según la Guía de Práctica Clínica de la ESH/ESC 2013 para el man
ejo de la hipertensión arterial. (Fuente: Guía de Práctica Clínica de la ESH/ESC 2013, pág. 11, publicación: 2013)

Clasificación de la presión arterial

Categoría	Sistólica (mmHg)	Diastólica (mmHg)
Óptima	< 120	< 80
Normal	120 – 129	80 – 84
Normal alta	130 – 139	85 – 89
Hipertensión arterial grado 1	140 – 159	90 – 99
Hipertensión arterial grado 2	160 – 179	100 – 109
Hipertensión arterial grado 3	≥ 180	≥ 110
Hipertensión arterial sistólica aislada	≥ 140	< 90

(Fuente: Guía de Práctica Clínica de la ESH/ESC 2013, pág. 11, publicación: 2013)

En qué consiste
La clasificación consiste en categorizar los valores de presión arterial en rangos que van desde óptimos y normales hasta distintos grados de hipertensión (1, 2 y 3) o formas especiales (p. e
j., sistólica aislada). Esto permite orientar el seguimiento clínico, definir niveles de riesgo cardiovascular y establecer las intervenciones de enfermería y médicas adecuadas. (Fuente: Guía
de Práctica Clínica de la ESH/ESC 2013, pág. 11, publicación: 2013)

Figura 5. Presentación de contenido normativo.

Los tiempos de respuesta más largos detectados fueron de 19 segundos y con un promedio de 12 segundos en el entorno de pruebas de Narvalla, se espera una reducción de hasta un 50% en el tiempo de entrega de información en un entorno de producción, pero incluso con esa medida de 19 segundos, es totalmente aceptable si se compara con

el tiempo que le tomaría a una persona leer, analizar e integrar ese conocimiento de forma tradicional, aun con un filtro de archivos por búsqueda de texto exacto. Estos tres escenarios y los tiempos de respuesta cumplen con los resultados esperados para este proyecto.

Conclusiones

Se concluye la viabilidad del sistema RAG para su uso en el entorno de Narvalla. Los resultados obtenidos demuestran que es posible facilitar el acceso a normativas complejas sin sacrificar la seguridad jurídica ni clínica. El sistema no solo responde dudas, sino que actúa como un mecanismo de cumplimiento normativo al negarse a emitir juicios fuera de su base de conocimientos, lo que demuestra que la tecnología, bien orquestada, previene errores por omisión o por el uso de fuentes no verificadas. Los tiempos de respuesta obtenidos en el entorno de pruebas se consideran aceptables si se comparan con el tiempo que le tomaría a una persona consultar esta información de forma tradicional.

La arquitectura basada en grafos facilitó la trazabilidad y gobernabilidad del flujo de información, mientras que las restricciones del prompt del sistema resultaron efectivas para mitigar el riesgo de alucinaciones. La contribución más significativa de este artículo es el diseño del flujo de trabajo que encadena la inteligencia artificial a la realidad legal del sector salud, demostrando que es posible usar modelos de lenguaje avanzados (LLMs) como herramientas de apoyo, sin olvidar que a pesar de las restricciones aún pueden cometer errores.

La carga de nuevos documentos sigue un proceso manual, lo cual puede generar brechas de información si no es bien supervisada; en particular, se puede llegar a perder el contexto de la página de un documento, provocando que esos fragmentos devuelvan información incompleta o errónea.

Como puntos de mejora, se puede automatizar el proceso de búsqueda e indexación de documentos si se definen correctamente las fuentes de información oficiales, con la finalidad de evitar demoras o posibles errores humanos. Asimismo, la integración de interfaces de voz facilitaría el uso del sistema en situaciones donde el personal tenga las manos ocupadas durante sus actividades; la tecnología está disponible, pero estas características se encuentran fuera del alcance de este documento.

Referencias

Caldera Sánchez, C. J. y Nava Sánchez, P. A. (2025). Arquitectura de sistema recomendador híbrido para el matching entre pacientes y enfermeras. Revista CID, 1(11). https://doi.org/10.62805/cid2025.CID113

Damodaran, P. (2023). FlashRank: Ultra-lite & super-fast Python library for re-ranking [Software]. GitHub. https://github.com/PrithivirajDamodaran/FlashRank

Diario Oficial de la Federación. (s.f.). Diario Oficial de la Federación. Gobierno de México. https://www.dof.gob.mx

Docker. (s.f.). Docker: Accelerated container application development. Docker Inc. https://www.docker.com

LangChain. (s.f.). LangGraph: Agent orchestration framework. LangChain Inc. https://www.langchain.com/langgraph

Langfuse. (s.f.). Langfuse: Open source LLM engineering platform. Langfuse GmbH. https://langfuse.com

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S. y Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. Advances in Neural Information Processing Systems.

OpenAI. (2024). New embedding models and API updates. OpenAI. https://openai.com/index/new-embedding-models-and-api-updates

OpenAI. (2025). OpenAI o4-mini. OpenAI. https://openai.com/index/openai-o4-mini

Open RN. (2024). Open educational resource nursing textbooks. Chippewa Valley Technical College y Wisconsin Technical College System. https://www.wistechopen.org/open-rn-details

OpenStax. (2024). OER Nursing Essentials. Rice University. https://openstax.org/subjects/nursing

OpenTelemetry. (s.f.). OpenTelemetry: An observability framework. Cloud Native Computing Foundation. https://opentelemetry.io

PostgreSQL. (s.f.). PostgreSQL: The world's most advanced open source relational database. https://www.postgresql.org

Qdrant. (s.f.). Qdrant: Vector search engine. Qdrant Solutions GmbH. https://qdrant.tech

Ramírez, S. (s.f.). FastAPI: Modern, fast web framework for building APIs with Python. https://fastapi.tiangolo.com

Secretaría de Salud. (s.f.). Normas Oficiales Mexicanas. Gobierno de México. https://www.gob.mx/salud/documentos/normas-oficiales-mexicanas-9705