SmartReview: Revisión Inteligente de Literatura Científica Asistida Localmente

Colaboración
Pedro Antonio Ibarra Facio; Oliver Sánchez Corona;Walter Alexander Mata López; Mónica Cobián Alvarado; Carlos Alberto Torres Cantero, Universidad de Colima

Fecha de recepción: 25 de agosto de 2025
Fecha de aceptación: 13 de septiembre de 2025

RESUMEN: La creciente sobrecarga informativa y la necesidad de confidencialidad en la revisión de literatura científica exigen soluciones innovadoras. Este artículo presenta SmartReview, una herramienta de código abierto que opera exclusivamente en el entorno local del usuario para garantizar la privacidad total. Su principal innovación es un RAG modificado que procesa PDFs página a página, generando resúmenes estructurados concisos que preservan el núcleo semántico, que a diferencia del chunking convencional. La herramienta integra OCR, clasificación temática y una interfaz de chat interactiva. Para demostrar su viabilidad, se realizó una validación funcional sobre un corpus de 25 artículos científicos. Los resultados cuantitativos iniciales son prometedores, reportando tiempos de procesamiento promedio de 4.1 segundos por página y una notable reducción del contenido textual en un factor aproximado de 28.5x. Esto disminuye significativamente los requerimientos computacionales del LLM local, perfilando a SmartReview como una alternativa segura y robusta a los servicios en la nube.

PALABRAS CLAVE: Revisión de literatura, RAG, Confidencialidad de datos, Inteligencia artificial, LLM.

ABSTRACT: The growing information overload and the need for confidentiality in scientific literature review demand innovative solutions. This paper presents SmartReview, an open-source tool operating exclusively on the user's local environment to ensure total privacy. Its main innovation is a modified Retrieval-Augmented Generation (RAG) approach that processes PDFs page-by-page, generating ultra-concise structured summaries that preserve semantic core content, in contrast to conventional chunking methods. The tool integrates Optical Character Recognition (OCR), thematic classification, and an interactive chat interface. To demonstrate its feasibility, a functional validation was performed on a corpus of 25 scientific articles. Initial quantitative results are promising, showing average processing times of 4.1 seconds per page and a remarkable reduction in textual content by a factor of approximately 28.5x. This significantly decreases the computational demands on the local Large Language Model (LLM), positioning SmartReview as a secure and robust alternative to cloud-based services.

KEYWORDS: Literature review, RAG, Data confidentiality, Artificial intelligence, LLM.

INTRODUCCIÓN

La revisión de literatura es un primer paso crucial en cualquier investigación, ya que ayuda a entender el conocimiento existente sobre un tema, encontrar áreas poco estudiadas y conectar ideas con el propio trabajo [1]. No obstante, el volumen de publicaciones científicas crece a diario, lo que dificulta enormemente a los investigadores leer, resumir y organizar tanta información [2], [3].

Recientemente, la inteligencia artificial (IA), y en particular los grandes modelos de lenguaje (LLM), han demostrado una gran capacidad para analizar, clasificar y generar texto, acelerando significativamente la revisión de literatura [4]. Herramientas en línea como ChatPDF o Paperpal son un ejemplo de ello, aunque presentan un gran inconveniente: al funcionar en la nube, exigen subir los documentos a servidores de terceros. Esta práctica genera un serio problema

de confidencialidad, pues en investigación se manejan a menudo datos sensibles o borradores no publicados. La subida de estos documentos a servicios externos infringe normativas institucionales y de organismos como los NIH, que prohíben el uso de procesadores externos para analizar información confidencial, una limitación que impide a muchos investigadores aprovechar estas útiles herramientas.[5], [6].

Para solucionar este problema, se presenta SmartReview, una plataforma web diseñada para funcionar localmente en la computadora del usuario. El objetivo es resolver dos desafíos a la vez: la gran cantidad de información y la necesidad de mantener los datos privados. Para ayudar con la revisión de literatura, se ofrecen cuatro funciones principales: (1) Obtener texto de archivos PDF, (2) clasificar automáticamente los textos por temas, (3) un sistema propio de búsqueda y respuesta basado en los documentos, (4) una interfaz de chat para hacer preguntas sobre uno o varios documentos.

La principal novedad de SmartReview es el procesamiento de documentos para hacer consultas eficientes localmente mediante su método de Generación Aumentada por Recuperación (RAG, por sus siglas en inglés). En lugar de cortar los documentos en pedazos pequeños (técnica conocida como chunking), que a veces puede separar ideas importantes, SmartReview crea resúmenes cortos y estructurados de cada página (con título, puntos clave y conclusión, usualmente menos de 30 tokens). Esto reduce mucho el tamaño de la información que el sistema necesita manejar (de unos 700-800 “tokens” por página a menos de 30 por resumen), pero conservando las ideas principales, lo cual permite ahorrar recursos manteniendo el sentido del texto original. Gracias a estos resúmenes por página y a una forma específica de dar instrucciones al LLM (lo que se explicará en la sección de metodologia), SmartReview encuentra la información más importante en documentos largos o en varios archivos a la vez, usando una cantidad moderada de recursos computacionales y manteniendo todo privado.

Es importante mencionar que estamos en pruebas para medir exactamente qué tan bien funciona SmartReview (precisión, velocidad). Finalmente, discutimos las limitaciones actuales y posibles mejoras futuras, desarrollando una visión completa de esta herramienta para la revisión bibliográfica confidencial.

Revisión de la literatura

El proceso para elaborar un estado del arte, que incluye el buscar, seleccionar, organizar e interpretar un gran volumen de fuentes es normalmente complejo y consume mucho tiempo [1]. Esto se ha intensificado debido a la creciente disponibilidad de literatura digital y al aumento exponencial de publicaciones científicas, lo que ha impulsado la búsqueda de tecnologías que automatizan y optimizan la revisión bibliográfica [3].

Herramientas existentes y el problema de la confidencialidad

La mayoría de las implementaciones RAG disponibles, al igual que muchos servicios LLM, dependen del procesamiento en la nube, lo que implica transferir documentos normalmente confidenciales a servidores externos. Esta práctica puede incumplir normativas y políticas de privacidad institucionales o de organismos como los NIH [5], [7]. Adicionalmente, la propia base de conocimiento utilizada en RAG introduce una nueva superficie de ataque: estudios recientes como PoisonedRAG demuestran que es factible corromper la base de conocimiento inyectando textos maliciosos para inducir respuestas incorrectas específicas en el LLM [8], un riesgo significativo si se depende de bases de conocimiento externas o colaborativas.

Generación aumentada por recuperación (RAG)

En este contexto, la Generación Aumentada por Recuperación se ha consolidado como una técnica que combina la capacidad generativa de los LLM, con la precisión de la información recuperada en tiempo real desde fuentes de conocimiento externas [9]. Este enfoque mejora las limitaciones de los LLM, como el conocimiento desactualizado y la tendencia a la alucinación, al basar las respuestas del modelo en evidencia verificable [10], [11]. El proceso RAG estándar implica típicamente: 1) un recuperador (retriever) que selecciona documentos relevantes de una base de conocimiento dada una consulta, y 2) un generador (LLM) que utiliza estos documentos, usualmente concatenados a la entrada de texto original, como contexto para producir la respuesta final [12].

El proceso RAG estándar no está exento de limitaciones. Su efectividad depende críticamente de la calidad de la recuperación [13], ya que información irrelevante o de baja calidad puede perjudicar la generación final [11]. Además, existe una desconexión inherente o “brecha semántica” entre los recuperadores y los LLM generativos, lo que dificulta que el LLM comprenda por qué ciertos documentos fueron seleccionados. La simple concatenación de información como contexto puede sobrecargar al LLM, especialmente con documentos largos, llevando a problemas como la pérdida de información en el medio del contexto [14].

Para superar estas limitaciones, han surgido técnicas RAG avanzadas. Algunas buscan mejorar la interacción entre el recuperador y el generador o la robustez ante recuperaciones imperfectas. Por ejemplo, CRAG introduce un evaluador de recuperación y mecanismos de autocorrección [11]. Otros enfoques, como R2AG, proponen integrar información sobre el

proceso de recuperación en la entrada del LLM [14], y Parametric RAG busca inyectar conocimiento directamente en los parámetros del LLM [12].

Mientras que RAG y sus variantes ofrecen un gran potencial para automatizar la revisión de literatura [10], persisten problemas relacionados con la calidad de la recuperación, el manejo de documentos largos y la falta de soluciones que garanticen la confidencialidad mediante el procesamiento estrictamente local. Es en este sentido donde surge la necesidad de una herramienta RAG eficiente, local y segura, es aquí donde SmartReview ofrece un enfoque metodológico para la interacción con la literatura académica.

MATERIALES Y MÉTODOS

Materiales

La arquitectura del sistema se construyó con herramientas de código abierto. El backend se desarrolló en Python, utilizando la biblioteca Streamlit para la interfaz gráfica, PyMuPDF para la extracción de texto y easyocr para el reconocimiento óptico de caracteres (OCR). La inferencia del modelo de lenguaje se gestionó mediante Llama.cpp.

Todo el sistema se ejecutó en un equipo con un CPU AMD Ryzen 7 5700G, 64 GB de RAM y una GPU NVIDIA RTX 3090, bajo el sistema operativo Windows 11. El modelo de lenguaje empleado fue Meta-Llama-3.2-8B-Instruct, configurado con una cuantización Q8_0, una temperatura de 0.8 y una ventana de contexto de 32,768 tokens.

La operación de SmartReview se basa en dos flujos principales desacoplados: un flujo de preprocesamiento para preparar los datos y un flujo de consulta para interactuar con ellos. La Figura 1 ilustra el ciclo operativo general del sistema.

Figura 1. Diagrama a bloques de la aplicación web para exámenes rápidos con IA generativa.
Fuente: Elaboración propia.

Métodos

Se inicia con el flujo de preprocesamiento y construcción de la base de conocimiento, este flujo se ejecuta una sola vez por cada conjunto de documentos para preparar la información de cara a consultas eficientes. Consta de las siguientes etapas (ver Figura 2):

a) Etapa de carga y extracción de Texto: El usuario carga los documentos PDF a través de la interfaz. El sistema extrae el texto plano, aplicando OCR si es necesario.

b) Etapa de clasificación temática: El LLM local analiza una porción inicial (resumen o primera página) para asignar etiquetas temáticas.

c) Etapa de segmentación y generación de resúmenes estructurados por página: En lugar de usar la técnica de chunking, se procesa el documento página por página. Para cada una, genera un resumen estructurado y conciso (título, puntos clave e idea principal) de menos de 30 tokens.

d) Etapa de almacenamiento estructurado: La información procesada (texto original por página, resumen estructurado, etiqueta temática) se guarda localmente en archivos JSON, manteniendo la relación explícita entre resúmenes y texto completo.

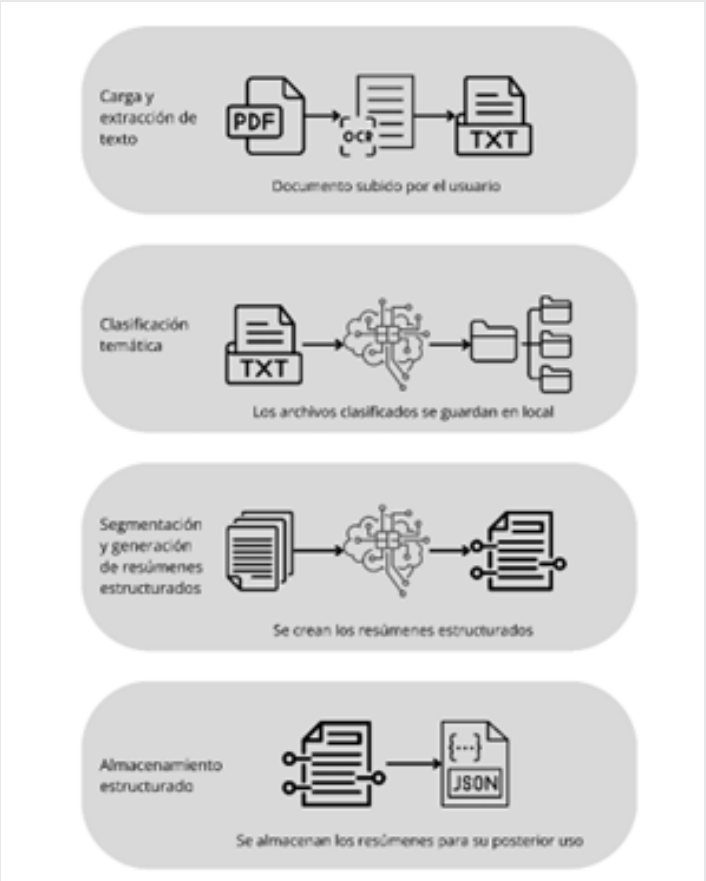

Figura 2: Flujo del proceso de preprocesamiento.
Fuente: Elaboración propia.

Una vez construida la base de conocimiento, el usuario interactúa mediante la interfaz de chat y se activa el flujo de consulta y generación de respuestas, el cual consiste en las siguientes etapas, las cuales se ilustran en la Figura 3.

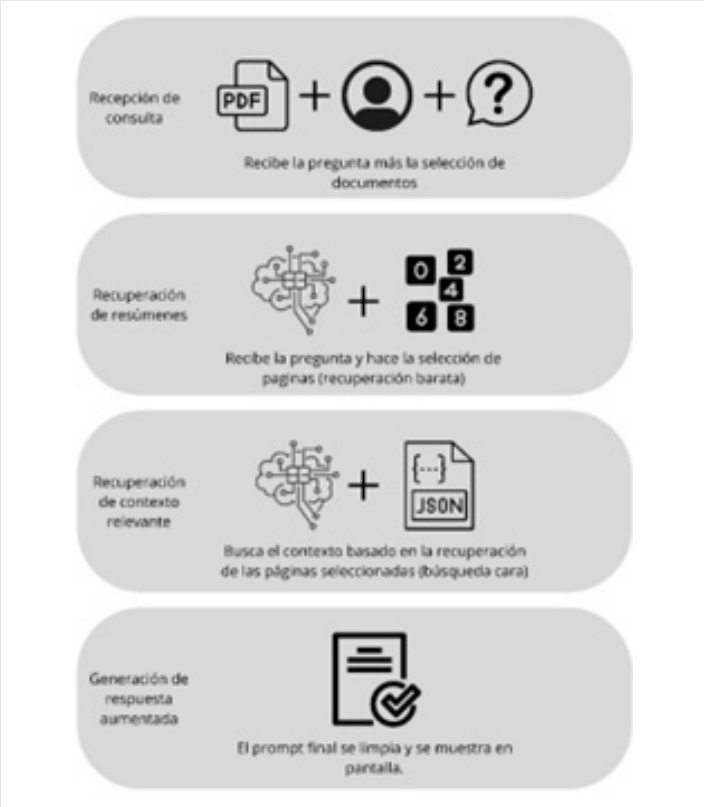

Figura 3: Flujo del proceso de consulta.
Fuente: Elaboración propia.

a) Etapa de recepción de consulta: El usuario formula una pregunta, específica o general, sobre uno o más documentos.

b) Etapa de recuperación basada en resúmenes: El LLM compara semánticamente la consulta con los resúmenes almacenados para identificar las páginas más relevantes.

c) Etapa de recuperación de texto completo relevante: El sistema recupera de la base local únicamente el texto original de las páginas identificadas como relevantes.

d) Etapa de generación de respuesta aumentada: Se construye un prompt para el LLM que incluye la consulta original y el texto recuperado para generar la respuesta, incluyendo citas a los números de página.

Detalles de implementación.
La efectividad del sistema depende en gran medida del diseño de las instrucciones enviadas al LLM por lo que se diseñaron dos prompts principales:
1) Prompt de resumen por página: Para la fase de preprocesamiento, se diseñó una instrucción que le pide al LLM analizar el texto de una página y devolver un resumen estructurado en formato JSON. El prompt instruye al modelo para que identifique un título tentativo, 2-3 puntos clave y una idea principal, con una restricción de longitud total de menos de 30 tokens.

2) Prompt de Consulta Final (RAG): Para generar la respuesta al usuario, se construye un prompt final que incluye: (a) el rol del asistente, (b) la instrucción de basar la respuesta exclusivamente en el contexto proporcionado, (c) el texto completo de las páginas relevantes recuperadas, (d) la pregunta original del usuario, y (e) la orden de citar los números de página de las fuentes utilizadas.

Validación y confidencialidad
La validación funcional se realizó utilizando un corpus de 25 artículos científicos para simular un escenario de revisión de literatura realista. El proceso incluyó una evaluación cualitativa de la eficiencia, donde se observó una aceleración significativa del proceso: el análisis del corpus completo con SmartReview tomó 4 días, en contraste con las 3 semanas que requirió una revisión manual tradicional realizada por dos de los coautores. Adicionalmente, se validó la confidencialidad como pilar fundamental del diseño. Al ejecutar todos sus componentes (extracción, resumen, clasificación y el LLM) de forma estrictamente local, SmartReview garantiza que los documentos y las consultas nunca abandonan el equipo del usuario. Este enfoque no solo elimina los riesgos de privacidad, sino que asegura el cumplimiento de normativas institucionales estrictas como las del NIH, consolidando la herramienta como una alternativa segura y eficiente para la investigación.

RESULTADOS

La validación funcional de SmartReview se realizó en un entorno local con un corpus de 25 artículos científicos. Las pruebas demostraron la operatividad de la herramienta, logrando una extracción completa del texto y generación de resúmenes estructurados para cada página. El principal resultado cuantitativo de este proceso fue una reducción de la representación textual a menos de 30 tokens por resumen, sin pérdida en calidad semántica. Para cuantificar la eficiencia general, se midieron las métricas de rendimiento del preprocesamiento, presentadas en la Tabla 1.

Tabla 1: Métricas de rendimiento del preprocesamiento para el corpus de 25 Artículos.

Métrica	Valor Total	Valor Promedio (por página)
Documentos procesados	25	N/A
Páginas procesadas	443	N/A
Tiempo de procesamiento	1816 s (30.3 min)	4.1 s
Tokens originales	328,936	~740
Tokens resumidos	~11,500	~26
Factor de reducción de contexto	~28.5x	N/A

Fuente: Elaboración propia.

Estos valores confirman la viabilidad del enfoque de SmartReview en hardware de escritorio. Un tiempo de procesamiento promedio de 4.1 segundos por página es un resultado práctico, considerando que incluye la operación de OCR. El hallazgo más significativo es la reducción del volumen de información en un factor aproximado de 28.5x, lo que permite al LLM local gestionar el contexto de múltiples documentos de manera eficiente, optimizando el uso de recursos computacionales. Las comparativas formales con otros métodos se proponen como trabajo futuro.

Además de la eficiencia cuantitativa, se validó la funcionalidad del flujo de consulta interactiva. Como se muestra en la Figura 4, la interfaz de usuario permite a los usuarios seleccionar documentos y formular consultas complejas en lenguaje natural sobre el corpus cargado.

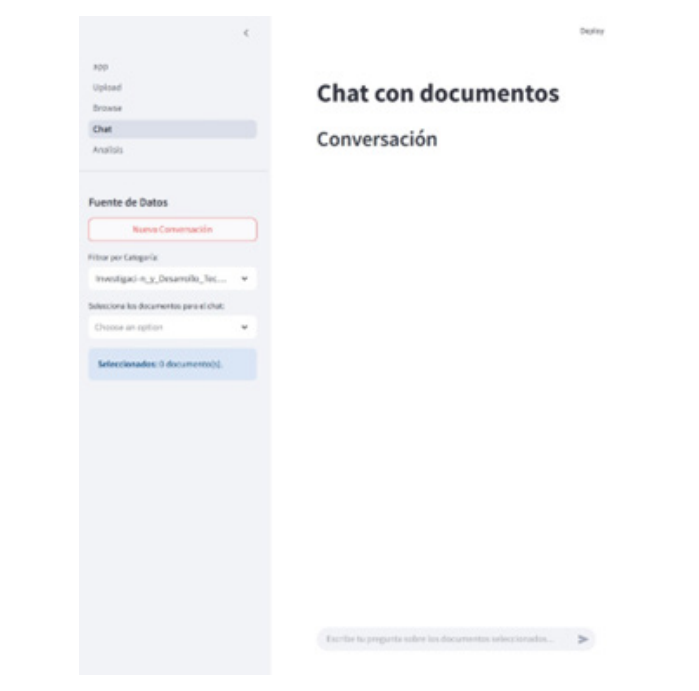

Figura 4: Interfaz de usuario de SmartReview. Se muestra la pantalla principal donde el usuario selecciona los documentos del corpus y formula una consulta en lenguaje natural.
Fuente: Elaboración propia.

Tras recibir la consulta, el sistema identificó correctamente las páginas relevantes mediante la comparación semántica con los resúmenes almacenados y recuperó selectivamente el texto completo original. Basándose exclusivamente en este contexto local, el LLM generó respuestas coherentes e informativas. La Figura 5 ejemplifica una de estas respuestas, donde el sistema fundamenta sus afirmaciones con referencias precisas a las páginas de los documentos fuente. Todo el proceso, desde la carga hasta la respuesta, se ejecutó sin conexión a redes externas, garantizando la confidencialidad.

Figura 5: Ejemplo de respuesta generada por el sistema. Tras una consulta comparativa, SmartReview sintetiza la información de múltiples fuentes y fundamenta sus afirmaciones con referencias explícitas a las páginas de los documentos originales.
Fuente: Elaboración propia.

RESULTADOS

Discusión

SmartReview ha demostrado ser funcional en un entorno local que garantiza la confidencialidad de los datos durante el análisis de literatura. El enfoque RAG modificado, basado en resúmenes por página, ha permitido la recuperación de información relevante y la generación de respuestas contextualizadas en nuestra validación inicial con 25 artículos. Los resultados cuantitativos de la Tabla 1 son un aporte clave especialmente el factor de reducción de contexto de ~28.5x confirma que la carga computacional se reduce significativamente, haciendo viable el uso de LLM en hardware de escritorio. No obstante, el tiempo de procesamiento de 4.1 s/página demuestra que el OCR es un cuello de botella importante a mejorar. La evaluación actual es principalmente funcional; reconocemos como limitación principal la falta de una validación cuantitativa con métricas estándar y un corpus de mayor tamaño, aspectos a los que se otorgará prioridad en trabajos futuros.

CONCLUSIONES

Este trabajo presenta y valida SmartReview, una herramienta que permite el análisis y consulta confidencial de documentos académicos mediante un LLM local. Se demuestra la efectividad de su enfoque RAG modificado, que opera sin conexión para garantizar una recuperación eficiente y sobre todo una generación de respuestas precisa. Su principal contribución es ofrecer una alternativa segura y local frente a herramientas dependientes de la nube, abordando preocupaciones de privacidad y cumplimiento normativo en investigación. El trabajo futuro se centrará en una validación cuantitativa rigurosa con métricas estandarizadas, pruebas de escalabilidad con un corpus ampliado no pasando por alto la optimización del rendimiento para mejorar la accesibilidad de la herramienta.

BIBLIOGRAFÍA

[1] S. T. Molopa and J. Cronje, "Artificial intelligence-based literature review adaptation," SA Journal of Libraries and Information Science, vol. 90, no. 2, 2024. doi:10.7553/90-1-2390.

[2] C. B. Asmussen and C. Møller, "Smart literature review: A practical topic modelling approach to exploratory literature review," Journal of Big Data, vol. 6, no. 1, pp. 1-18, 2019. doi:10.1186/s40537-019-0255-7.

[3] F. Bolaños, A. Salatino, F. Osborne, et al., "Artificial intelligence for literature reviews: Opportunities and challenges," Artificial Intelligence Review, vol. 57, p. 259, 2024. doi:10.1007/s10462-024-10902-3.

[4] D. A. Tovar, "AI literature review suite," arXiv:2308.02443, 2023. doi: 10.48550/arXiv.2308.02443.

[5] M. Lauer, S. Constant, and A. Wernimont, "Using AI in peer review is a breach of confidentiality," NIH Extramural Nexus, Jun. 23, 2023. [Online]. Available: https://nexus.od.nih.gov/all/2023/06/23/using-ai-in-peer-review-is-a-breach-of-confidentiality/.

[6] O. Ngwenyama and F. Rowe, "Should we collaborate with AI to conduct literature reviews? Changing epistemic values in a flattening world," Journal of the Association for Information Systems, vol. 25, no. 1, pp. 122-136, 2024. doi:10.17705/1jais.00869.

[7] J. Evertz, M. Chlosta, L. Schönherr, and T. Eisenhofer, "Whispers in the machine: Confidentiality in LLM-integrated systems," arXiv:2402.06922v3, 2024. doi: 10.48550/arXiv.2402.06922.

[8] W. Zou, R. Geng, B. Wang, and J. Jia, "PoisonedRAG: Knowledge corruption attacks to retrieval-augmented generation of large language models," arXiv:2402.07867v3, 2024. doi:10.48550/arXiv.2402.07867.

[9] P. Lewis, et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in Advances in Neural Information Processing Systems, vol. 33, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds. Curran Associates, Inc., 2020, pp. 9459-9474.

[10] B. Han, T. Susnjak, and A. Mathrani, "Automating systematic literature reviews with retrieval-augmented generation: A comprehensive overview," Applied Sciences, vol. 14, no. 19, p. 9103, 2024. doi: 10.3390/app14199103.

[11] S.-Q. Yan, J.-C. Gu, Y. Zhu, and Z.-H. Ling, "Corrective retrieval augmented generation," arXiv:2401.15884v3, 2024. doi: 10.48550/arXiv.2401.15884.

[12] W. Su, et al., "Parametric retrieval augmented generation," arXiv:2501.15915, 2025. doi: 10.48550/arXiv.2501.15915.

[13] A. Salemi and H. Zamani, "Evaluating retrieval quality in retrieval-augmented generation," in Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24), 2024, pp. 2395-2400. doi:10.1145/3626772.3657957.

[14] F. Ye, S. Li, Y. Zhang, and L. Chen, "R2AG: Incorporating retrieval information into retrieval augmented generation," in Findings of the Association for Computational Linguistics: EMNLP 2024, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds. Association for Computational Linguistics, 2024, pp. 11584-11596. doi: 10.18653/v1/2024.findings-emnlp.678.

[15] J. W. Creswell, Research design: Qualitative, quantitative, and mixed methods approaches, 2nd ed. Sage Publications, 2003.

ACM Transactions on Software Engineering and Methodology, Vol 33, Issue 5. Article No.: 135, pp: 1-50. DOI: https://doi.org/10.1145/3652154.

[9] Kuck, K. (2023). Generative Artificial Intelligence: A Double-Edged Sword. 2023 World Engineering Education Forum - Global Engineering Deans Council (WEEF-GEDC), 1-10. DOI: 10.1109/WEEF-GEDC59520.2023.10343638.

[10] Eager, B., & Brunton, R. (2023). Prompting higher education towards AI-augmented teaching and learning practice. En: Journal of Universi-

ty Teaching & Learning Practice, 20(5), 02. DOI: 10.53761/1.20.5.02.

[11] The Pallet Projects. (2023). Welcome to Flask Flask Documentation (3.0.x). En Palletsprojects. com. Disponible en: https://flask.palletsprojects. com/en/3.0.x/.

[12] PyPI.org. (2023). Google Bard API para Python. Sitio oficial para descarga de la librería para Python en PyPI. Disponible en: https://pypi. org/project/bardapi/.

[13] Yiu, E. et al. (2023). Transmission Versus Truth, Imitation Versus Innovation: What Children Can Do That Large Language and Language-and-Vision Models Cannot (Yet). En: Perspectives on Psychological Science. https://journals.sagepub.com/ doi/full/10.1177/17456916231201401.

[14] Anthropic. (2023). Claude - Asistente de IA Generativa. En línea. Disponible en: http://www. claude.ai.

[15] OpenJS-Foundation (2023) Node.js. Disponible en: https://nodejs.org/en.

[16] ExpressJS.com (2017) Express JS -Node.js web application framework. Sitio official. Disponible en: https://expressjs.com/.