Vol. 2, Núm. 4, 2026
DOI: 10.5281/zenodo.24204327
ISSN: 3121-2646
Recibido: 11/09/2026
Aceptado: 19/10/2026
Publicado: 13/11/2026

Investigación y Cultura Académica

Inteligencia artificial generativa en la enseñanza de programación aplicada a la Ingeniería Química

Generative artificial intelligence in programming education applied to Chemical Engineering

Inteligência artificial generativa no ensino de programação aplicada à Engenharia Química



Víctor Hugo León Quiñónez
Universidad de Guayaquil
https://orcid.org/0009-0003-3685-0381
victor.leonq@ug.edu.ec

Educación
Artículo de Revisión

Cómo citar este artículo: León Quiñónez, V. H. (2026). *Inteligencia artificial generativa en la enseñanza de programación aplicada a la Ingeniería Química*. Investigación y Cultura Académica, 2(4), 544–563. https://doi.org/10.5281/zenodo.24204327

CC BY NC SA

Investigación y Cultura Académica
EDUINCA S.A.S.

Inteligencia artificial generativa en la enseñanza de programación aplicada a la Ingeniería Química

Resumen

La inteligencia artificial generativa está transformando las formas de aprender programación y de resolver problemas computacionales en educación superior. En ingeniería química, su integración posee especial interés para el análisis numérico, la elaboración de herramientas informáticas y la representación de procesos, pero debe distinguirse entre posibilidades técnicas y resultados de aprendizaje demostrados. El objetivo fue analizar críticamente la literatura proporcionada acerca del uso de inteligencia artificial generativa en la enseñanza de programación y sus vínculos con la formación en ingeniería química. Se efectuó una síntesis documental temática de 22 archivos científicos aportados en dos entregas, contrastando investigaciones empíricas, revisiones y propuestas disciplinares. Se consideraron la mediación docente, el aprendizaje algorítmico, la generación y depuración de código, las aplicaciones químicas y los riesgos de implementación. Los estudios muestran experiencias de apoyo a la explicación de sintaxis, retroalimentación inmediata y elaboración de recursos computacionales; asimismo, documentan inquietudes relativas a dependencia, confiabilidad de respuestas e integridad académica. Los trabajos directamente situados en ingeniería química son menos numerosos que los centrados en programación general o en otras ingenierías. Se concluye que la articulación educativa más defendible combina problemas disciplinares auténticos, verificación de código y resultados, y evaluación del razonamiento independiente. El acervo recibido constituye una base documental para la revisión, pero no permite reconstruir por sí solo las cifras de identificación y cribado exigidas por PRISMA 2020.

Palabras clave: inteligencia artificial generativa; enseñanza de programación; ingeniería química; pensamiento computacional; educación superior.

Generative artificial intelligence in programming education applied to Chemical Engineering

Abstract

Generative artificial intelligence is changing the ways programming is learned and computational problems are addressed in higher education. Its integration into chemical engineering is particularly relevant to numerical analysis, software development, and process representation, although technical possibilities must be distinguished from demonstrated learning outcomes. This paper critically analyzes the supplied literature on generative AI in programming education and its links to chemical engineering training. A thematic documentary synthesis was conducted on 22 scientific files submitted in two batches, comparing empirical studies, reviews, and discipline-oriented contributions. The analysis addressed teacher mediation, algorithmic learning, code generation and debugging, chemical applications, and implementation risks. The literature documents support for syntax explanation, immediate feedback, and the development of computational resources, alongside concerns about dependence, response reliability, and academic integrity. Studies specifically situated in chemical engineering are less frequent than those focused on general programming or other engineering disciplines. Effective educational integration should therefore connect authentic disciplinary problems with code and result verification and assessment of independent reasoning. The supplied corpus is a documentary foundation, but it does not independently establish the identification and screening counts required by PRISMA 2020.

Keywords: generative artificial intelligence; programming education; chemical engineering; computational thinking; higher education.

Inteligência artificial generativa no ensino de programação aplicada à Engenharia Química

Resumo

A inteligência artificial generativa está transformando o ensino de programação e a resolução de problemas computacionais no ensino superior. Na engenharia química, sua integração é relevante para análise numérica, desenvolvimento de ferramentas computacionais e representação de processos, embora seja necessário distinguir possibilidades técnicas de resultados de aprendizagem comprovados. Este estudo analisa criticamente a literatura fornecida sobre inteligência artificial generativa no ensino de programação e suas relações com a formação em engenharia química. Foi realizada uma síntese documental temática de 22 arquivos científicos, abrangendo estudos empíricos, revisões e contribuições disciplinares. A análise contemplou mediação docente, aprendizagem algorítmica,

geração e depuração de códigos, aplicações químicas e riscos de implementação. Os trabalhos descrevem apoio na compreensão de sintaxe, feedback imediato e criação de recursos computacionais, mas também apontam dependência tecnológica, confiabilidade e integridade acadêmica como desafios. A evidência diretamente relacionada à engenharia química é mais restrita que a disponível para programação geral e outras engenharias. Conclui-se que uma integração pedagógica consistente exige problemas disciplinares autênticos, validação de códigos e resultados e avaliação do raciocínio autônomo. O acervo constitui uma base documental, sem oferecer, por si só, os números de identificação e triagem requeridos pelo PRISMA 2020.

Palavras-chave: inteligência artificial generativa; ensino de programação; engenharia química; pensamento computacional; ensino superior.

Introducción

La programación constituye una herramienta transversal para representar fenómenos, organizar información y desarrollar soluciones reproducibles. En ingeniería química, donde confluyen balances de materia y energía, termodinámica, cinética y operaciones de separación, la formulación algorítmica contribuye a convertir modelos teóricos en procedimientos verificables. Sin embargo, aprender a programar supone afrontar simultáneamente dificultades sintácticas, abstracción matemática y decisiones sobre la adecuación física de los resultados. Esta complejidad exige estrategias que conecten la comprensión del fenómeno con la construcción del código y eviten tratar la ejecución informática como un sustituto del razonamiento ingenieril.

La expansión de los modelos generativos amplía las posibilidades de apoyo durante estas tareas. Estudios situados en la enseñanza de programación describen asistentes capaces de explicar estructuras de control, sugerir ejemplos, revisar fragmentos de código y ofrecer retroalimentación contextualizada. Jiménez Rasgado et al. (2024) documentaron una experiencia educativa con estudiantes de ingeniería que utilizaron ChatGPT y Perplexity en actividades sobre programación en C, mientras que Bravo Sánchez et al. (s. f.) examinaron aplicaciones de chatbots en asignaturas de ingeniería mecatrónica. Tales antecedentes muestran usos pedagógicos relevantes, pero pertenecen a contextos diferentes de ingeniería química y sus resultados no deben extrapolarse automáticamente.

Las revisiones de Chávez-Boza y Erazo-Moreta (2024), Avila-Avila y Gutierrez Buitrago (2024) y Castillo Pacheco (2025) coinciden en reconocer oportunidades de personalización, feedback y apoyo a la resolución de problemas, aunque también identifican riesgos de dependencia y erosión de la autonomía. Vera Paredes et al. (2025) profundizan en las dificultades para establecer qué aprendizajes se fortalecen y cuáles podrían quedar enmascarados por respuestas automatizadas. La coincidencia temática no implica equivalencia metodológica: las revisiones sintetizan estudios heterogéneos y las experiencias docentes aportan evidencia situada, cuya fuerza depende del diseño y de la calidad de la evaluación.

En el ámbito químico, Hasrod et al. (2024) presentaron la creación de una interfaz gráfica en Python con apoyo de ChatGPT para facilitar el uso de un modelo predictivo sobre sulfatos en drenaje ácido de mina. A diferencia de los estudios generales de

programación, esta experiencia conecta la generación de código con una necesidad concreta de las ciencias químicas. Bilgin et al. (2025) analizaron la utilización de IA en el rediseño de asignaturas nucleares de ingeniería química, incluyendo termodinámica, balances de materia y energía y cinética. Khoo et al. (2025) examinaron de manera comparativa aplicaciones de modelos de lenguaje en ingeniería química, entre ellas simulación de procesos y diseño de reactores. Sin embargo, no todos estos trabajos evalúan directamente competencias de programación de estudiantes.

La revisión se justifica por una desconexión analítica: una parte de la literatura estudia las capacidades educativas de los asistentes generativos, mientras otra explora aplicaciones computacionales de IA en ingeniería química. La cuestión pendiente consiste en determinar qué respaldos empíricos permiten articular ambas líneas y cuáles son únicamente posibilidades de transferencia didáctica. En consecuencia, la pregunta orientadora es: ¿qué evidencia ofrece la literatura proporcionada sobre herramientas, estrategias, resultados formativos y riesgos de la IA generativa en la enseñanza de programación aplicada a ingeniería química?

El objetivo general es analizar críticamente las contribuciones y limitaciones de la inteligencia artificial generativa en la enseñanza de programación vinculada con la formación de ingenieros químicos. Como objetivos específicos se plantean: identificar herramientas y modalidades de uso docente y estudiantil; contrastar resultados reportados sobre aprendizaje de programación y pensamiento computacional; examinar aplicaciones disciplinares en química e ingeniería química; y establecer condiciones pedagógicas y metodológicas para una incorporación responsable.

Inteligencia artificial generativa y aprendizaje de programación

La inteligencia artificial generativa comprende sistemas que producen respuestas, explicaciones y artefactos como fragmentos de código a partir de instrucciones formuladas en lenguaje natural. En educación de programación, su relevancia no se reduce a escribir instrucciones sintácticamente válidas: se extiende a la posibilidad de solicitar ejemplos contrastivos, reformular explicaciones y recibir orientación durante la depuración. Castillo Pacheco (2025) analiza estas oportunidades desde competencias del siglo XXI, mientras Chávez-Boza y Erazo-Moreta (2024) sitúan los

fundamentos de programación como un campo que exige integrar capacidades técnicas y criterios humanísticos.

La experiencia de Gallegos Ramos (2023), centrada en el aprendizaje de C++, ofrece un antecedente empírico de evaluación antes y después de una intervención con ChatGPT. Por su parte, Terán Torres (2023) relaciona el uso de la herramienta con desempeño, motivación y consideraciones éticas en estudiantes de ingeniería de sistemas. Estos resultados son relevantes como evidencia de programación asistida por IA, aunque el contexto, las poblaciones y la ausencia de transferencia directa a contenidos de ingeniería química obligan a interpretar su alcance con cautela.

La investigación de Ugarte et al. (2025) evalúa respuestas de sistemas generativos a ejercicios y exámenes de programación usando criterios docentes. Este diseño ayuda a valorar el rendimiento del sistema ante tareas académicas, pero no demuestra por sí solo que los estudiantes desarrollen las mismas capacidades cuando utilizan la herramienta. La diferencia entre la calidad del producto generado y el aprendizaje alcanzado es central para una síntesis rigurosa de la evidencia.

Formación de ingenieros y estrategias pedagógicas mediadas por IA

En el marco de Educación 4.0, la innovación en ingeniería se asocia con transformación de planes de estudio, aprendizaje personalizado y vinculación de competencias digitales con problemas profesionales. Ramos Piñas e Infante Rivera (2025) examinan estas relaciones desde la formación de ingenieros civiles, mientras Sánchez-Castillo y Jiménez-Pérez (2025) analizan la enseñanza de matemáticas asistida por IA. Ambos aportes amplían el contexto de innovación, pero sus resultados pertenecen a áreas disciplinares distintas.

La literatura también describe modelos didácticos que podrían incorporarse a asignaturas con programación. Rengifo Obando et al. (2026) analizan aulas invertidas apoyadas por IA generativa y Gutiérrez Baltazar (2026) estudia la integración de tecnología en el aprendizaje basado en proyectos. Sus contribuciones sugieren estrategias de preparación autónoma, actividades colaborativas y resolución de retos contextualizados; no obstante, su traslación a ingeniería química requiere diseñar tareas auténticas y verificar si las competencias se desarrollan efectivamente.

Las revisiones de Ceada-Garrido y colaboradores (2025) examinan la integración de IA generativa en educación superior en ingeniería. Dos archivos del acervo comparten autoría principal, horizonte temático y conclusiones, por lo que deben revisarse como publicaciones potencialmente solapadas antes de cualquier conteo de estudios independientes. Merinos Herrera et al. (2026) subrayan el papel de la mediación docente, la verificación de información y el razonamiento autónomo, principios que adquieren especial valor cuando un algoritmo generado por IA produce resultados físicamente inverosímiles.

Programación y resolución de problemas de ingeniería química

La utilidad de la programación en ingeniería química puede organizarse en tres niveles. El primero corresponde al aprendizaje de estructuras básicas, funciones y manejo de datos; el segundo, a la formulación de modelos matemáticos y procedimientos numéricos; y el tercero, a la interpretación y validación física de los resultados. Estos niveles son complementarios: un programa que funciona sin errores informáticos puede representar incorrectamente un proceso químico si las unidades, las hipótesis o las condiciones de frontera son inapropiadas.

Hasrod et al. (2024) ilustran la relación entre asistencia generativa y desarrollo de instrumentos químicos al documentar una interfaz en Python y Jupyter Notebook creada mediante ciclos de instrucciones, errores y correcciones. La experiencia permite reconocer una actividad formativa plausible: construir recursos que respondan a necesidades científicas mientras se comprenden sus componentes. No equivale, sin embargo, a demostrar un efecto causal sobre el dominio de programación de estudiantes de ingeniería química.

Bilgin et al. (2025) exploran el rediseño con IA de contenidos y evaluaciones en termodinámica, balances de materia y energía y cinética, con atención a percepciones estudiantiles de confiabilidad. Khoo et al. (2025) identifican aplicaciones de modelos de lenguaje para simulación, modelado y optimización de procesos, junto con la exigencia de verificar sus salidas mediante métodos convencionales. En conjunto, ambos trabajos contextualizan la pertinencia disciplinar de la IA, pero sus preguntas investigativas y unidades de análisis son distintas de las de una intervención en enseñanza de programación.

Metodología

Enfoque, diseño y alcance del acervo

Se desarrolló una síntesis bibliográfica de orientación cualitativa y analítico-comparativa basada en 22 archivos proporcionados en dos entregas. La unidad de análisis fue cada publicación, no los participantes de los trabajos originales. Los documentos incluyen investigaciones empíricas, revisiones sistemáticas o de alcance, reflexiones teóricas, reportes tecnológicos y comunicaciones de congresos. La síntesis se organizó alrededor de herramientas generativas, prácticas de enseñanza, indicadores formativos, aplicaciones químicas y limitaciones. El diseño textual se orienta a los elementos de reporte PRISMA 2020; sin embargo, el corpus entregado no incluye un registro verificable de búsqueda, exportaciones bibliográficas por base, historial de deduplicación ni razones de exclusión. Por tanto, no corresponde presentar un flujo PRISMA cerrado o declarar exhaustividad de búsqueda.

Criterios de pertinencia, clasificación y elegibilidad

Se distinguieron tres niveles de pertinencia: evidencia directa sobre enseñanza de programación con herramientas generativas; evidencia contextual sobre formación en ingeniería y metodologías mediadas por IA; y evidencia disciplinar sobre aplicaciones químicas e ingeniería química. La clasificación permitió conservar antecedentes pertinentes sin equiparar estudios de niveles educativos diferentes ni confundir informes técnicos con ensayos pedagógicos. Como criterio de evaluación del manuscrito se consideró imprescindible que toda afirmación de eficacia remitiera a un resultado efectivamente documentado, diferenciando percepciones, productos creados y mediciones de aprendizaje. La decisión de incluir o excluir definitivamente un documento en una revisión sistemática requiere criterios a priori y un registro transparente que no está disponible en los archivos aportados.

Extracción de información y síntesis

La extracción contempló autoría, año, campo de aplicación, tipo de estudio, herramienta de IA, actividad educativa o técnica, resultados comunicados y limitaciones. Se contrastaron las contribuciones por convergencia conceptual y por diferencias de diseño. Los resultados cuantitativos se conservaron únicamente cuando el documento original los explicitaba y no se agregaron porcentajes derivados de poblaciones, instrumentos o procedimientos heterogéneos. Las publicaciones relacionadas de Ceada-Garrido y colaboradores se señalaron como posible

solapamiento de evidencia. Los trabajos revisionales se emplearon principalmente para encuadrar el campo y localizar convergencias, mientras que las experiencias empíricas se analizaron como evidencia situada.

Rigor y limitaciones del procedimiento

La revisión documental es reproducible respecto a los archivos recibidos, pero no es todavía reproducible como búsqueda bibliográfica sistemática en bases de datos. Faltan las fechas efectivas de consulta, las cadenas finales ejecutadas en cada fuente, los registros recuperados, los criterios de duplicación, el cribado por revisores y las razones de exclusión por texto completo. PRISMA 2020 se utiliza como referencia para identificar estas exigencias y no como aval de etapas que no han sido documentadas. No se efectuó metaanálisis por la diversidad de objetos, diseños y resultados. Tampoco se realizó una evaluación formal de riesgo de sesgo con herramienta específica; esta fase deberá completarse si el estudio se presenta como revisión sistemática definitiva.

Tabla 1

Clasificación analítica del acervo recibido

Grupo	Documentos representativos	Contribución	Precaución interpretativa
Programación con IA	Gallegos Ramos (2023); Jiménez Rasgado et al. (2024); Terán Torres (2023); Bravo Sánchez et al. (s. f.)	Asistencia a la explicación, generación y depuración de código	No generalizar resultados a ingeniería química
Revisiones de programación	Chávez-Boza y Erazo-Moreta (2024); Avila-Avila y Gutierrez Buitrago (2024); Castillo Pacheco (2025)	Estado de la investigación y riesgos pedagógicos	No duplicar resultados de estudios primarios citados
Formación en ingeniería	Ramos Piñas e Infante Rivera (2025); Ceada-Garrido et al. (2025); Vasquez et al. (2025)	Innovación, adaptación curricular, analítica de aprendizaje	Controlar solapamiento entre versiones
Química e ingeniería química	Hasrod et al. (2024); Bilgin et al. (2025); Khoo et al. (2025)	Desarrollo de software químico, cursos disciplinares, modelación	No confundir aplicación técnica con mejora formativa probada

Nota. *Elaboración propia del autor (2026).*

Resultados

Selección de publicaciones y flujo PRISMA 2020 limitado al acervo recibido

El procedimiento de selección se efectuó sobre todos los archivos aportados. Se verificaron los 22 textos completos y se registró una exclusión mutuamente excluyente por documento. Las publicaciones conceptuales y revisionales se conservaron para antecedentes, pero no se contabilizaron como hallazgos empíricos nuevos. Se presenta un flujo descriptivo del corpus suministrado, no un diagrama de identificación en bases bibliográficas.

Figura 1

Flujo de elegibilidad del corpus suministrado



Nota. *Los conteos se refieren exclusivamente al acervo entregado; las dos publicaciones relacionadas de Ceada-Garrido no se trataron como duplicados exactos.*

Tabla 2

Decisiones de elegibilidad documento por documento

Documento / año	Decisión	Justificación / naturaleza de evidencia
Gallegos Ramos (2023)	Incluido	Estudio cuasiexperimental sobre aprendizaje de C++ asistido por ChatGPT
Terán Torres (2023)	Incluido	Experiencia en cursos iniciales de programación, con medidas académicas y encuesta
Jiménez Rasgado et al. (2024)	Incluido	Experiencia de programación en C apoyada por ChatGPT y Perplexity
Bravo Sánchez et al. (s. f.)	Incluido	Experiencia de programación en asignaturas de ingeniería mecatrónica
Ugarte et al. (2025)	Incluido	Evaluación original de IA ante ejercicios de asignaturas; evidencia técnica indirecta
Hasrod et al. (2024)	Incluido	Desarrollo de aplicación química en Python con ChatGPT; reporte tecnológico
Bilgin et al. (2025)	Incluido	Diseño de cursos y percepciones en ingeniería química; evidencia indirecta para programación
Castillo Pacheco (2025)	Excluido: secundario	Revisión de literatura
Ceada-Garrido et al. (2025), Jornadas	Excluido: secundario	Revisión sistemática; relacionada con versión de simposio
Ceada-Garrido et al. (2025), Simposio	Excluido: secundario	Revisión sistemática; publicación relacionada, no duplicado exacto
Chávez-Boza y Erazo-Moreta (2024)	Excluido: secundario	Revisión sistemática
Avila-Avila y Gutierrez Buitrago (2024)	Excluido: secundario	Revisión sistemática
Vera Paredes et al. (2025)	Excluido: secundario	Revisión sistemática
Vasquez et al. (2025)	Excluido: secundario	Revisión de alcance
Gutiérrez Baltazar (2026)	Excluido: secundario	Revisión sistemática
Rengifo Obando et al. (2026)	Excluido: secundario	Revisión integradora
Ramos Piñas e Infante Rivera (2025)	Excluido: secundario	Análisis documental
Sánchez-Castillo y Jiménez-Pérez (2025)	Excluido: secundario	Revisión documental
Merinos Herrera et al. (2026)	Excluido: secundario	Reflexión teórica
Khoo et al. (2025)	Excluido: secundario	Revisión comparativa
Salinas Ordoñez et al. (2026)	Excluido: secundario	Revisión sistemática sobre autonomía y pensamiento crítico
Sánchez Barragán (s. f.)	Excluido: nivel	Experiencia original exclusivamente en educación media técnica

Nota. *La tabla documenta decisiones realizadas a posteriori sobre el corpus recibido. "Incluido" no equivale a evidencia directa de mejora del aprendizaje en ingeniería química: algunos informes aportan desempeño técnico o diseño curricular.*

Evaluación crítica descriptiva de los siete estudios incluidos

Gallegos Ramos (2023) presenta un diseño cuasiexperimental pretest-postest sin asignación aleatoria documentada en el resumen, por lo que los cambios reportados no prueban por sí solos causalidad. Terán Torres (2023) combina intervención docente

y percepciones de estudiantes, pero las medidas y controles requieren examen cuidadoso. Jiménez Rasgado et al. (2024) y Bravo Sánchez et al. (s. f.) describen experiencias educativas situadas, con información útil de implementación que no puede extrapolarse sin considerar muestra, curso y evaluación. Ugarte et al. (2025) evalúan el desempeño de las herramientas ante ejercicios universitarios; este resultado mide capacidad de los modelos, no progreso estudiantil. Hasrod et al. (2024) documentan un producto tecnológico de química elaborado con apoyo de ChatGPT, sin constituir una prueba de efectividad pedagógica comparativa. Bilgin et al. (2025) combinan rediseño de asignaturas y percepciones de estudiantes en ingeniería química, pero no aíslan la enseñanza de programación como resultado educativo. Esta valoración permite calibrar las afirmaciones de la síntesis, aunque no sustituye un instrumento formal de riesgo de sesgo.

La síntesis primaria se construyó sobre siete estudios o informes originales. Las restantes quince publicaciones se utilizaron únicamente para contextualización o se excluyeron por nivel educativo; por ello, las coincidencias entre revisiones previas no se presentan como nuevas observaciones independientes. Tres tipos de evidencia coexisten entre los incluidos: experiencias pedagógicas con estudiantes, evaluaciones de desempeño de herramientas generativas y desarrollos curriculares o tecnológicos en química e ingeniería química. Esta distinción limita la fuerza de las conclusiones sobre eficacia educativa.

Panorama del material y especificidad disciplinar

El acervo ofrece un conjunto heterogéneo que favorece la comparación de perspectivas. El mayor número de trabajos identificables se concentra en educación general de programación, educación superior e innovación con IA. Los trabajos que enlazan de modo explícito la asistencia generativa con una tarea propia de ciencias químicas o ingeniería química son menos frecuentes. Esta distribución justifica separar la evidencia educativa directa de los antecedentes disciplinares, en lugar de presentar una única categoría homogénea. Una inferencia razonable es que la integración entre programación, IA y problemas químicos representa una oportunidad de investigación; no puede afirmarse, con el material disponible, que su eficacia ya esté demostrada de manera general.

Herramientas y modalidades de apoyo a la programación

ChatGPT ocupa un lugar recurrente en el acervo como asistente para generar ejemplos, explicar código y atender consultas. En la experiencia de Jiménez Rasgado et al. (2024), ChatGPT y Perplexity participaron en actividades de enseñanza de estructuras de control en C. Gallegos Ramos (2023) estudió ChatGPT en C++, mientras que Sánchez Barragán (s. f.) describió una intervención en educación media técnica con ChatGPT, Gemini y Copilot, comparación de respuestas y verificación técnica. Esta última resulta didácticamente ilustrativa, aunque su población no corresponde a educación superior. Los sistemas analizados por Ugarte et al. (2025) pudieron responder a tareas evaluadas por docentes, evidencia sobre capacidad instrumental, no sobre transformación comprobada de competencias estudiantiles.

Aprendizaje, retroalimentación y razonamiento autónomo

Las investigaciones reportan oportunidades de retroalimentación inmediata y orientación en problemas concretos. Gallegos Ramos (2023) comunica cambios favorables de aprendizaje en C++ tras una experiencia de uso de ChatGPT, mientras Terán Torres (2023) observa mejoras percibidas en comprensión y motivación asociadas a una integración guiada. Jiménez Rasgado et al. (2024) identifican utilidad para explicar la lógica de programación y corregir errores. No obstante, la diversidad de instrumentos, la disponibilidad de grupos de comparación y las diferencias en los diseños impiden calcular un efecto común atribuible a IA generativa.

Los trabajos de Avila-Avila y Gutierrez Buitrago (2024), Castillo Pacheco (2025) y Vera Paredes et al. (2025) advierten que la rapidez para producir una solución no garantiza dominio conceptual. El estudiante puede obtener un programa operativo sin reconocer por qué funciona, sin anticipar casos extremos o sin identificar supuestos incorrectos. La evidencia conduce a distinguir productividad asistida, comprensión declarada y capacidad demostrada de resolución independiente. El aprendizaje requeriría, por tanto, tareas donde la persona pueda explicar, modificar y justificar las decisiones del código sin depender de la herramienta.

Aplicaciones químicas y validación disciplinar

Hasrod et al. (2024) documentan el uso de ChatGPT como apoyo para programar una interfaz gráfica de un modelo predictivo en un problema de química ambiental. El

proceso combina lenguaje natural, mensajes de error y ajustes en Python. Su aporte reside en mostrar que una herramienta generativa puede reducir barreras iniciales para construir un producto científico funcional y ofrecer andamiaje en entornos como Jupyter Notebook. Sin embargo, es preciso evaluar por separado la corrección del software, la comprensión del estudiante y la validez del modelo químico subyacente.

Bilgin et al. (2025) estudian la elaboración de materiales académicos con IA en asignaturas fundamentales de ingeniería química y las percepciones sobre confiabilidad de estas herramientas. Khoo et al. (2025) examinan las posibilidades de los modelos de lenguaje en simulación de procesos, diseño de reactores y optimización, destacando la necesidad de validar resultados frente a métodos tradicionales. Estas contribuciones permiten proponer problemas didácticos sobre balances, cinética o sistemas de separación, pero no certifican por sí mismas resultados de aprendizaje en programación dentro de esas asignaturas.

Riesgos técnicos, éticos y pedagógicos

La literatura presenta convergencias respecto a respuestas plausibles pero inexactas, dificultades para valorar autoría, dependencia tecnológica y oportunidades desiguales de acceso. Terán Torres (2023) enfatiza la dimensión ética del uso de ChatGPT en el aula; Merinos Herrera et al. (2026) insisten en que la mediación docente debe preservar autonomía y pensamiento crítico; y Ugarte et al. (2025) muestran la necesidad de examinar el desempeño de las herramientas frente a criterios académicos. En ingeniería química estas precauciones son especialmente relevantes: un error en unidades, restricciones termodinámicas o rangos de operación puede producir una respuesta computacional convincente, pero científicamente inválida.

Tabla 3

Modalidades de uso de IA generativa y resultados descritos

Modalidad	Uso documentado	Evidencia relacionada	Condición pedagógica
Explicación de código	Interpretar sintaxis y lógica algorítmica	Jiménez Rasgado et al. (2024); Bravo Sánchez et al. (s. f.)	Pedir explicaciones y ejemplos alternativos
Generación y depuración	Proponer fragmentos y corregir errores	Gallegos Ramos (2023); Hasrod et al. (2024)	Ejecutar pruebas y verificar hipótesis
Comparación de respuestas	Contrastar distintos asistentes	Sánchez Barragán (s. f.); Ugarte et al. (2025)	Evaluar precisión, límites y razonamiento

Diseño de recursos docentes	Crear actividades y materiales	Bilgin et al. (2025); Rengifo Obando et al. (2026)	Revisión experta de contenido y evaluación
Modelación disciplinar	Apoyar simulación y tareas de ingeniería química	Khoo et al. (2025)	Validación física y numérica independiente

Nota. *La tabla integra usos mencionados por los diversos autores y condiciones pedagógicas, elaboración propia del autor (2026).*

Discusión

Los resultados permiten sostener una interpretación matizada. La IA generativa puede actuar como andamiaje durante el aprendizaje de programación, especialmente cuando el estudiante recibe explicaciones oportunas, examina errores y contrasta soluciones. Esta posibilidad aparece en la experiencia de Jiménez Rasgado et al. (2024), en los trabajos sobre C++ y programación universitaria de Gallegos Ramos (2023) y Terán Torres (2023), y en revisiones sobre fundamentos computacionales (Chávez-Boza & Erazo-Moreta, 2024; Castillo Pacheco, 2025). La coincidencia resulta consistente, pero las fuentes no emplean una medición uniforme del aprendizaje; por consiguiente, hablar de mejora generalizada exige prudencia.

Una segunda línea de discusión remite a las capacidades genuinamente disciplinares. La construcción de una aplicación química mediante Python asistido por ChatGPT, documentada por Hasrod et al. (2024), ofrece un puente concreto entre la programación y la solución de un problema del campo. En contraste, las aportaciones de Bilgin et al. (2025) se centran fundamentalmente en rediseño curricular y percepciones de IA, mientras Khoo et al. (2025) examinan aplicaciones técnicas de los modelos de lenguaje. Su articulación permite proponer unidades didácticas sobre cálculos y modelación, pero el paso de propuesta a eficacia educativa necesitaría intervenciones controladas y evaluaciones específicas de competencias.

En términos pedagógicos, un enfoque centrado exclusivamente en rapidez de respuesta favorece la delegación acrítica de tareas. Las estrategias mejor justificadas por el corpus incluyen formular inicialmente una solución propia, solicitar a la IA una alternativa, explicar diferencias, ejecutar pruebas y contrastar el resultado con restricciones químicas. Los modelos de aula invertida y aprendizaje basado en proyectos descritos por Rengifo Obando et al. (2026) y Gutiérrez Baltazar (2026) pueden organizar este proceso en actividades autónomas y colaborativas, aunque sus

estudios no acreditan directamente su rendimiento en asignaturas de programación de ingeniería química.

El concepto de validación adquiere aquí una doble dimensión. En programación, exige pruebas, revisión de estructuras y comprensión del algoritmo; en ingeniería química, requiere verificar unidades, conservación de materia y energía, coherencia termodinámica y viabilidad de las condiciones modeladas. La revisión de Khoo et al. (2025) enfatiza que las salidas de los modelos deben confrontarse con métodos convencionales. De este modo, la mediación docente puede transformar los errores generados por IA en oportunidades de razonamiento, siempre que las tareas obliguen a explicar las decisiones y no solo a entregar un resultado.

Una limitación del conocimiento disponible es la mezcla de tipos de fuentes. Las revisiones previas aportan marcos para comprender tendencias, pero no deben sumarse como observaciones empíricas independientes junto con los estudios que sintetizan. Asimismo, las experiencias de ingeniería mecatrónica, de sistemas o de educación media técnica constituyen antecedentes transferibles en términos de diseño pedagógico, no demostraciones específicas en ingeniería química. El posible solapamiento entre las publicaciones de Ceada-Garrido et al. (2025) refuerza la necesidad de depurar versiones y evitar doble recuento de evidencia.

La principal aportación interpretativa consiste en proponer que el valor educativo de la IA generativa no depende de su capacidad de producir código, sino del tipo de actividad intelectual que organiza su uso. Una integración coherente en ingeniería química debe promover tres competencias simultáneas: comprensión algorítmica, análisis del problema químico y validación independiente. Para evaluar esta propuesta se requieren investigaciones que comparen condiciones didácticas, documenten tareas auténticas, describan los sistemas generativos empleados y midan transferencia a problemas nuevos sin asistencia.

Tabla 4

Implicaciones para una agenda de investigación en ingeniería química

Dimensión	Hallazgo del acervo	Necesidad de investigación
Aprendizaje de programación	Se reporta ayuda en sintaxis y depuración	Evaluar comprensión y transferencia sin IA

Aplicación disciplinar	Existen ejemplos de software y modelación química	Diseñar intervenciones en balances, cinética y simulación
Evaluación	Heterogeneidad de instrumentos y diseños	Adoptar rúbricas de código y validación física
Ética y autonomía	Riesgos recurrentes de dependencia y errores	Examinar prácticas de verificación e integridad académica
Rigor de síntesis	Revisiones y estudios primarios coexistentes	Registrar búsqueda, deduplicación y riesgo de sesgo

Nota. *Implicaciones formuladas a partir de la comparación crítica del corpus, elaboración propia del autor (2026).*

Conclusiones

La literatura examinada permite reconocer que la inteligencia artificial generativa amplía las posibilidades de apoyo durante la enseñanza de programación mediante explicación contextual, generación de ejemplos, depuración de código y retroalimentación inmediata. Estas funciones han sido descritas en experiencias educativas y revisiones especializadas, aunque la heterogeneidad metodológica obliga a diferenciar el desempeño de las herramientas de las competencias que efectivamente adquieren los estudiantes.

En ingeniería química se identifican conexiones concretas con la creación de recursos informáticos, el análisis de problemas químicos y el rediseño de asignaturas fundamentales. Sin embargo, el material aportado no demuestra de forma suficientemente homogénea efectos causales de la IA generativa sobre el aprendizaje de programación específicamente en estudiantes de ingeniería química. Este vacío delimita una agenda de investigación más precisa y constituye un aporte de la presente síntesis.

La incorporación académica más defendible requiere vincular las herramientas generativas con problemas auténticos de la disciplina, supervisión docente, pruebas de funcionamiento y validación físico-química. La capacidad para justificar algoritmos, interpretar resultados y detectar errores debe conservarse como objetivo explícito de evaluación. La automatización puede ser un recurso de aprendizaje, pero no reemplaza la comprensión conceptual ni el juicio profesional.

El análisis también muestra que el uso de revisiones previas, reportes técnicos e investigaciones empíricas demanda una clasificación cuidadosa para evitar inferencias

excesivas. Como limitación del trabajo, el conjunto de 22 archivos fue suministrado y no se dispone del historial de identificación, cribado y selección en bases bibliográficas; por tanto, una versión final que se denomine revisión sistemática PRISMA debe completar y documentar estas etapas. Se recomienda incorporar estudios primarios directamente realizados en ingeniería química y realizar una evaluación formal de su calidad metodológica.

Las futuras investigaciones deberían valorar experiencias de enseñanza de Python u otros lenguajes en problemas de balances, termodinámica, cinética y procesos, comparando grupos y observando la permanencia del aprendizaje. También deberían examinar si los estudiantes son capaces de reconstruir y verificar una solución sin asistencia generativa. En este marco, el objetivo no sería aumentar únicamente la cantidad de código producido, sino formar ingenieros capaces de comprender, cuestionar y validar soluciones computacionales.

Referencias bibliográficas

Avila-Avila, M.-L., & Gutierrez Buitrago, E. F. (2024). Impacto de la IA en la enseñanza de la programación: Una revisión sistemática. Pensamiento y Acción, (37). https://doi.org/10.19053/01201190.n37.2024.20409

Bilgin, B., Chen, C. V. H.-H., & Butler Velegol, S. (2025). Generative AI in chemical engineering education: Rebuilding thermodynamics, material and energy balances and kinetics courses with AI and chemical engineering students' perception of AI. American Society for Engineering Education.

Bravo Sánchez, F. Á., Hurtado Londoño, J. A., & Cruz Bohórquez, J. M. (s. f.). Impacto de los chatbots de IA en la enseñanza de programación. https://doi.org/10.26507/paper.3720

Castillo Pacheco, J. G. (2025). Inteligencia artificial en la enseñanza de la programación y el desarrollo de competencias del siglo XXI: Un análisis de la literatura científica. Cuadernos UCAB, (23), 165–188.

Ceada-Garrido, Y., Barragán Piña, A. J., Enrique Gómez, J. M., Aquino Martín, A., & Andújar Márquez, J. M. (2025). Análisis de la integración y el impacto de la inteligencia artificial generativa en la educación superior en ingeniería: Una revisión sistemática. XX Simposio CEA de Control Inteligente.

Chávez-Boza, B. M., & Erazo-Moreta, O. R. (2024). Integración de la inteligencia artificial generativa para el aprendizaje de fundamentos de programación: Una revisión sistemática de la literatura. Revista Mexicana de Investigación e Intervención Educativa, 3(2), 5–17.

Gallegos Ramos, N. A. (2023). ChatGPT en el aprendizaje de lenguajes de programación en estudiantes universitarios. Revista Amazonía Digital, 2(2), 1–7. https://doi.org/10.55873/rad.v2i2.250

Gutiérrez Baltazar, J. R. (2026). Inteligencia artificial y tecnología en el aprendizaje basado en proyectos: Revisión sistemática. Revista de Educación Simón Rodríguez, 6(12), 1–17. https://doi.org/10.62319/simonrodriguez.v.6i12.217

Hasrod, T., Nuapia, Y. B., & Tutu, H. (2024). ChatGPT helped me build a chemistry app, and here's how you can make one also. Journal of Chemical Education, 101, 653–660. https://doi.org/10.1021/acs.jchemed.3c01170

Jiménez Rasgado, G., Alor Francisco, R., & Espinosa Valdés, M. E. (2024). Innovación con IA en educación superior: Experiencia de aprendizaje de programación en el Instituto Tecnológico de Minatitlán. Ciencia Latina Revista Científica Multidisciplinar, 8(6). https://doi.org/10.37811/cl_rcm.v8i6.15029

Khoo, T. L., Lee, T. S., Bee, S.-T., Ma, C., & Zhang, Y.-Y. (2025). A comparative review of large language models in engineering with emphasis on chemical engineering applications. Processes, 13(9), 2680. https://doi.org/10.3390/pr13092680

Merinos Herrera, S. I., Hernández Hernández, J. B., & Enríquez Hernández, J. A. (2026). La inteligencia artificial en la educación en ingeniería: Una reflexión teórica sobre su impacto en los procesos de enseñanza y aprendizaje.

Ramos Piñas, D., & Infante Rivera, L. de J. (2025). Innovación educativa en la formación de ingenieros en el marco de la Educación 4.0.

Rengifo Obando, M. E., Vera Alcívar, J. G., Arcentales Hidalgo, C. de L., Brito Ordóñez, K. S., & Mendieta Sinche, P. A. (2026). Aulas invertidas apoyadas por inteligencia artificial generativa en la educación superior. Sage Sphere of Technology, Sciences, Discoveries and Society, 4(1), 1–28. https://doi.org/10.63688/d9tvzf37

Sánchez Barragán, S. E. (s. f.). Del prompt al pensamiento computacional: Evidencias de la integración de la inteligencia artificial generativa en la enseñanza de la programación en educación media técnica. Revista de Investigación Multidisciplinaria Iberoamericana, 4(3).

Sánchez-Castillo, V., & Jiménez-Pérez, G. A. (2025). Uso de IA para mejorar el proceso de enseñanza-aprendizaje de matemáticas en estudiantes de ingeniería. Eco Matemático, 16(1), 6–20. https://doi.org/10.22463/17948231.4658

Terán Torres, H. (2023). La implementación de la inteligencia artificial en la enseñanza de la programación: Un estudio sobre el uso ético de ChatGPT en el aula. EIEI ACOFI 2023. https://doi.org/10.26507/paper.2768

Ugarte, M., Markiegi, U., & Aldalur, I. (2025). ¿Pueden las IA generativas aprobar las asignaturas de programación en un grado de informática? Actas de las JENUI, 10, 251–258.

Vasquez, X., Cina, J., Cornejo, J., & Rodríguez Alvarez, S. R. (2025). Application of learning analytics and adaptive AI tools in higher engineering education: Scoping review. LEIRD 2025. https://doi.org/10.18687/LEIRD2025.1.1.231

Vera Paredes, D. A., Franco Arias, O. O., & Córdova Martínez, L. C. (2025). Impacto de la inteligencia artificial en el aprendizaje de la programación informática en los estudiantes universitarios. Ciencia y Educación, 6(3), 33

Conflicto de intereses

El autor declara no tener conflictos de intereses.