REVISTA CIENTÍFICA

VOL. 02 NRO. 01, AÑO 2026

FRONTERAS TECNOLÓGICAS

SCHOOL OF ENGINEERING
UNIDAD DE POSGRADO FICCT - UAGRM

PUBLICACIÓN SEMESTRAL
ISSN: 3134-8696

Programas Académicos

DIPLOMADOS

- CRIPTOACTIVOS Y BLOCKCHAIN
- EDUCACIÓN SUPERIOR E INTELIGENCIA ARTIFICIAL

MAESTRÍAS

- CIENCIA DE DATOS E INTELIGENCIA ARTIFICIAL
- CIBERSEGURIDAD Y CIBERDEFENSA
- DIRECCIÓN ESTRATÉGICA EN INGENIERÍA DE SOFTWARE
- INNOVACIÓN Y TECNOLOGÍA EDUCATIVA
- DESARROLLO FULL STACK

DOCTORADO

- CIENCIAS DE LA COMPUTACIÓN

Programación 100% Virtual

www.soeuagrm.edu.bo
info@soe.uagrm.edu.bo
73600888

Agentes de cambio para un mundo interconectado

VOL. 02 NRO. 01, AÑO 2026 - REVISTA CIENTÍFICA

FRONTERAS TECNOLÓGICAS

UNIDAD DE POSGRADO SCHOOL OF ENGINEERING (SOE)
FACULTAD DE INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN Y TELECOMUNICACIONES
UNIVERSIDAD AUTÓNOMA GABRIEL RENÉ MORENO (UAGRM)

La Unidad de Posgrado School of Engineering, perteneciente a la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones de la Universidad Autónoma Gabriel René Moreno, es una unidad académica orientada a la formación especializada en áreas tecnológicas, investigación e innovación, comprometida con el desarrollo de profesionales capaces de responder a los desafíos de un mundo interconectado.

Santa Cruz de la Sierra, Bolivia, 2026
ISSN: 3134-8696

Av. Busch, Ciudad Universitaria,
Módulo 232, 2do Piso
www.soeuagrm.edu.bo

Presentación

Revista Científica “Fronteras Tecnológicas”

Es un honor presentar el Volumen 2, Número 1 de la Revista Científica Fronteras Tecnológicas, publicación académica de la Unidad de Posgrado School of Engineering de la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones de la Universidad Autónoma Gabriel René Moreno.

Esta nueva edición refleja el compromiso institucional con la promoción de la investigación, la innovación y la difusión del conocimiento científico como pilares fundamentales para el desarrollo académico y tecnológico de nuestra sociedad.

La revista constituye un espacio de encuentro para investigadores, docentes y profesionales que comparten la visión de generar conocimiento con impacto, fortaleciendo la vinculación entre la academia y las necesidades del entorno.

Los trabajos que integran este número evidencian el esfuerzo y la dedicación de sus autores, así como la importancia de la investigación aplicada para responder a los desafíos que plantea la transformación digital y el avance de las tecnologías emergentes.

Expreso mi reconocimiento a los investigadores, evaluadores y miembros del equipo editorial que hicieron posible esta publicación. Su aporte contribuye al fortalecimiento de una cultura científica orientada a la excelencia y a la formación de profesionales comprometidos con el desarrollo del país y la región.

Invito a la comunidad académica nacional e internacional a continuar participando en este proyecto editorial y a seguir promoviendo, mediante la investigación, la construcción de soluciones que contribuyan al progreso científico y tecnológico.

MSc. Juan Carlos Peinado Pereira
Director General

Revista Científica “Fronteras Tecnológicas”
Unidad de Posgrado
F.I.C.C.T. - U.A.G.R.M.

Editorial

Revista Científica “Fronteras Tecnológicas”

Con la publicación del Volumen 2, Número 1, la Revista Científica Fronteras Tecnológicas continúa fortaleciendo su misión de difundir investigaciones originales y contribuir al desarrollo de las ciencias de la computación, las telecomunicaciones y áreas afines.

La presente edición reúne contribuciones que abordan temáticas de gran actualidad, entre ellas inteligencia artificial, modelos de lenguaje, ingeniería de software, ciberseguridad, aprendizaje profundo, evaluación y calidad en educación superior, así como aplicaciones tecnológicas orientadas a distintos contextos de desarrollo.

La diversidad temática y metodológica de los artículos refleja la creciente madurez investigativa de nuestros autores y evidencia la importancia de promover espacios de publicación científica con acceso abierto, rigor académico y proyección internacional.

Como equipo editorial, mantenemos el compromiso de fortalecer continuamente los procesos de calidad, visibilidad e indexación de la revista, con el propósito de consolidar Fronteras Tecnológicas como un referente en la difusión del conocimiento científico y tecnológico.

Agradecemos la confianza depositada por los autores y el valioso trabajo de los revisores, cuya contribución garantiza la calidad y pertinencia de cada publicación.

Esperamos que los resultados presentados en esta edición constituyan un aporte significativo para la comunidad científica y estimulen nuevas investigaciones, colaboraciones y oportunidades de innovación.

MBA. Dulce María Hernández Vázquez
Editora Principal

PhD. Delia Yucra Rodas
Asesora Editorial

Índice

Pág. 08
Modelo de clasificación de emociones faciales mediante redes neuronales convolucionales: Un estudio de entrenamiento supervisado en siete categorías emocionales
PhD. Humberto Aguilar Lobo

Pág. 16
Modelo de inteligencia competitiva ágil para la valoración docente en acreditación MERCOSUR
M.Sc. Ernesto Soto Roca

Pág. 30
BuilderSoe: LLM empleado en la generación de artículos científicos basados en LaTeX
M.Sc. Juan Carlos Peinado Pereira
M.Sc. Victor Hugo Acosta Ortega

Pág. 38
Población y Muestra en Estudios de Caso en Ingeniería de Software: Marco y Guía Práctica
PhD. Luis Roberto Pérez Ríos

Pág. 48
Sistema móvil con modelo jerárquico híbrido para estimar la severidad foliar del mildiu en quinua bajo captura controlada
M.Sc. Edgar Jaldin Torrico

Pág. 58
Procedimiento articulado de evaluación ex-ante para artefactos Design Science Research.
M.Sc. Paul Fernando Grimaldo Bravo
PhD. Luis Roberto Pérez Ríos

Pág. 66
Criptografía Post-Cuántica en Redes LTE/VoLTE: Desempeño, Interoperabilidad y Riesgo en la Migración de Esquemas para Telecomunicaciones en Sudamérica
M.Sc. Jorge Marcelo Rosales Fuentes

Pág. 74
Auto-Reflexión y RAG en Modelos de Lenguaje Pequeños para conocimiento empresarial: Un Estudio de Mapeo sistemático
M.Sc. Alcides Yohacin Leaños Rodríguez

Pág. 84
Generación de código basada en LLM: Una revisión sistemática de técnicas, métricas y evaluación empírica
M.Sc. Jorge Bergman Mostajo Pedraza

Pág. 96
Fortalecimiento de la competencia digital investigativa mediante TIC en estudiantes de Trabajo Social
M.Sc. Abel Huaygua Chalco

La ciencia de hoy es la tecnología del mañana. Cada avance tecnológico que transforma nuestra sociedad tiene su origen en la investigación, la curiosidad y la búsqueda constante de conocimiento. Invertir en ciencia es construir las bases del futuro.

Edward Teller

MODELO DE CLASIFICACIÓN DE EMOCIONES FACIALES MEDIANTE REDES NEURONALES CONVOLUCIONALES: UN ESTUDIO DE ENTRENAMIENTO SUPERVISADO EN SIETE CATEGORÍAS EMOCIONALES

PhD. Humberto Aguilar Lobo
Universidad Pública de "El Alto"
https://orcid.org/0000-0002-5595-5811
La Paz, Bolivia | aguilar.lobo@live.com

https://doi.org/10.23670 FT.2026.1.23

Recibido 29/03/2026 - Aceptado 07/05/2026

RESUMEN

Este artículo tiene por objeto contribuir al estudio de clasificación de emociones faciales humanas mediante redes neuronales convolucionales, a través de entrenamiento supervisado sobre siete categorías afectivas: Felicidad, tristeza, enfado, disgusto, miedo, neutral y sorpresa. Se trata de un estudio de enfoque experimental automático, basado en entrenamiento supervisado y evaluación del modelo mediante partición de datos (train/validation/test). El estudio se centró en el desarrollo y evaluación de un modelo de clasificación de emociones faciales mediante redes neuronales convolucionales. La población estuvo constituida por un conjunto de datos de imágenes de rostros en escala de grises; cada una etiquetada con una emoción facial: Felicidad, tristeza, enfado, disgusto, miedo, neutral y sorpresa. Se utilizó la totalidad de las imágenes del Data Set FER2013, compuesto por 35,887 imágenes de tamaño 48×48 píxeles, previamente centradas y recortadas enfocando el rostro. Los instrumentos empleados fueron librerías especializadas de aprendizaje automático y aprendizaje profundo (Machine Learning y Deep Learning). El modelo alcanzó una exactitud del 90% en entrenamiento y del 65% en validación. El análisis por clase mostró un F1-score máximo de 0.80 para la categoría "felicidad" y un mínimo de 0.46 para "miedo". Adicionalmente, se examinó la distribución de pesos por capa como indicador de estabilidad del entrenamiento. Los resultados evidencian un desempeño competitivo en el Data Set FER2013 y confirman la dificultad de discriminar emociones con alta similitud morfológica.

Palabras claves: Modelo, entrenamiento, redes neuronales, reconocimiento y emociones faciales.

ABSTRACT

This article aims to contribute to the study of classifying human facial emotions using convolutional neural networks, through supervised training on seven affective categories: happiness, sadness, anger, disgust, fear, neutral, and surprise. It is an automated experimental approach, based on supervised training and model evaluation through data partitioning (train/validation/test). The study focused on the development and evaluation of a facial emotion classification model using convolutional neural networks. The population consisted of a dataset of grayscale facial images, each labeled with one of the following facial emotions: happiness, sadness, anger, disgust, fear, neutral, and surprise. All images from the FER2013 dataset, comprising 35,887 48×48-pixel images, were used. These images were previously centered and cropped to focus on the face. The tools employed were specialized machine learning and deep learning libraries. The model achieved 90% accuracy during training and 65% during validation. Analysis by class showed a maximum F1 score of 0.80 for the "happiness" category and a minimum of 0.46 for "fear." Additionally, the weight distribution by layer was examined as an indicator of training stability. The results demonstrate competitive performance on the FER2013 dataset and confirm the difficulty of discriminating emotions with high morphological similarity.

Keywords: Model, training, neural networks, facial recognition and emotions.

INTRODUCCIÓN

El reconocimiento automático de emociones faciales constituye un área de creciente interés en la intersección entre la visión por computador y la inteligencia artificial, con aplicaciones potenciales en salud mental, educación y atención al cliente. Las redes neuronales convolucionales (CNN) han demostrado ser particularmente efectivas en esta tarea debido a su capacidad para extraer patrones espaciales jerárquicos a partir de imágenes (Goodfellow et al.,2016). Sin embargo, a pesar de los avances recientes, persisten limitaciones técnicas para lograr precisiones robustas en contextos no controlados, especialmente en la diferenciación de emociones con alta similitud morfológica, como el miedo y el disgusto (Mollahosseini et al., 2017). Si bien existen múltiples estudios que reportan clasificadores entrenados sobre el Data Set FER2013, la literatura carece de un análisis detallado que articule la arquitectura de la CNN, la distribución de pesos por capa y la capacidad de generalización del modelo en términos de métricas complementarias a la precisión global. Este vacío de conocimiento limita la comprensión de por qué ciertas emociones resultan más difíciles de clasificar y dificulta la reproducibilidad de los hallazgos.

Para abordar estas limitaciones, el presente estudio implementa una arquitectura CNN secuencial de cinco bloques convolucionales con filtros progresivos (32 a 512), acompañados de técnicas de regularización como normalización por lotes y Dropout sistemático, entrenada sobre 35 887 imágenes del conjunto FER2013. A diferencia de trabajos previos que reportan únicamente la precisión global, este artículo desglosa el desempeño del modelo mediante métricas complementarias (F1-score por clase y matriz de confusión) y analiza la distribución de pesos aprendidos en cada capa, aspectos que permiten identificar con mayor claridad qué emociones resultan más fácilmente clasificables (felicidad, enfado) y cuáles presentan confusiones sistemáticas (miedo, disgusto). Además, estudios recientes Khaireddin & Chen (2021) y Savchenko (2022), han reportado desempeños superiores en FER2013 utilizando arquitecturas más profundas.

A partir de esta problemática, la presente investigación se orienta a responder las siguientes preguntas: ¿Qué desempeño presenta una arquitectura CNN profunda regularizada en la clasificación automática de siete emociones faciales utilizando el Data Set FER2013? El objetivo general es evaluar el desempeño de una arquitectura CNN profunda regularizada para la clasificación automática de siete emociones faciales utilizando imágenes estáticas del Data Set FER2013. Para alcanzar este propósito, se definieron los siguientes objetivos específicos: (1) implementar una arquitectura CNN de cinco bloques convolucionales con filtros progresivos (32 a 512); (2) evaluar el desempeño del modelo mediante métricas complementarias (precisión, sensibilidad, F1-score y matriz de confusión); (3) analizar la distribución de pesos por capa como indicador de estabilidad del entrenamiento; y (4) identificar las categorías emocionales con mayor y menor precisión de clasificación dentro del Data Set FER2013.

METODOLOGÍA

La metodología seleccionada se resume en la siguiente tabla:

Tabla 1

Metodología de la investigación

Metodología	Característica
Enfoque y diseño	El estudio corresponde a un experimento de validación de modelo predictivo supervisado, de tipo comparativo-descriptivo, orientado a evaluar el desempeño de una arquitectura CNN en la clasificación multiclase de emociones faciales. El objetivo es evaluar el desempeño de un modelo de clasificación de emociones faciales de siete categorías (Felicidad, tristeza, enfado, disgusto, miedo, neutral y sorpresa) mediante redes neuronales convolucionales, a partir de imágenes estáticas..
Población y muestra	La población estuvo constituida por el conjunto de imágenes faciales en escala de grises del Data Set FER2013 (Facial Expression Recognition, 2013), el cual contiene 35,887 imágenes etiquetadas en siete categorías emocionales: felicidad, tristeza, enfado, disgusto, miedo, neutral y sorpresa. El Data Set FER2013 fue seleccionado por ser uno de los conjuntos más utilizados en la literatura para la evaluación comparativa de modelos de reconocimiento emocional, lo que permite contrastar resultados con estudios previos y garantizar reproducibilidad. A partir de este Data Set, se utilizó la totalidad de las imágenes, las cuales fueron previamente centradas y recortadas enfocando el rostro, con dimensiones de 48×48 píxeles. El conjunto de datos fue particionado aleatoriamente en tres subconjuntos: entrenamiento (70%), validación (15%) y prueba (15%), garantizando la representatividad de cada categoría emocional mediante un proceso de estratificación.
Instrumentos	Para la implementación y entrenamiento del modelo, se emplearon librerías especializadas de machine learning y Deep Learning: TensorFlow y Keras para la construcción y entrenamiento de la red neuronal convolucional; OpenCV para el preprocesamiento de imágenes; y NumPy y Matplotlib para el manejo de datos y visualización de resultados.

Tabla 2

Especificaciones del entorno tecnológico y arquitectura del modelo

Categoría	Descripción
Software	TensorFlow 2.15.0, Keras 2.15.0, OpenCV 4.8.1, NumPy 1.24.3, Matplotlib 3.7.2, Python 3.10.
Hardware	Entorno Google Colab con aceleración (CPU Intel Xeon 2.20 GHz, GPU NVIDIA Tesla T4 con 16 GB VRAM, RAM 25 GB).
Arquitectura del modelo	CNN secuencial con 5 bloques convolucionales (32, 64, 128, 256, 512 filtros), cada uno con capa Conv2D (kernel 3×3), BatchNormalization, activación ReLU, MaxPooling2D (2×2) y Dropout (0.25 en primeros bloques, 0.5 en los últimos). Seguido de capas densas (128 y 256 unidades) con BatchNormalization, ReLU y Dropout (0.5). Capa de salida Dense (7) con activación softmax.
Métricas de evaluación	Accuracy (exactitud global), F1-score (media macro y ponderada), matriz de confusión, pérdida (categorical crossentropy).

RESULTADOS

Desempeño del entrenamiento de la red neuronal convolucional

Al respecto, la siguiente tabla detalla el entrenamiento de cada capa de la red neuronal convolucional —incluyendo tipo, forma de salida y número de parámetros—; lo que permitió evidenciar cómo la arquitectura va incrementando progresivamente su capacidad de representación desde las primeras convoluciones hasta las capas densas finales. El modelo entrenado cuenta con parámetros distribuidos principalmente en los bloques convolucionales de mayor profundidad (256 y 512 filtros), mientras que las capas de normalización contribuyen a estabilizar el entrenamiento y mitigar el sobreajuste. Tras 30 períodos de entrenamiento con una tasa de aprendizaje (learning rate) inicial de 0.001 y tamaño del lote de 64, la red alcanzó una precisión de entrenamiento cercana al 90 % y una exactitud de validación promedio del 65 %, evidenciando un buen equilibrio entre capacidad de modelado y generalización.

Tabla 3

Desempeño del entrenamiento de la red neuronal convolucional

Capa (Tipo)	Salida (Forma)	Parámetro #
conv2d **(Conv2D)**	(48, 48, 32)	320
batch_normalization **(BatchNormalization)**	(48, 48, 32)	128
activation **(Activation)**	(48, 48, 32)	0
max_pooling2d **(MaxPooling2D)**	(24, 24, 32)	0
dropout **(Dropout)**	(24, 24, 32)	0
conv2d_1 **(Conv2D)**	(24, 24, 64)	51.264
batch_normalization_1 **(BatchNormalization)**	(24, 24, 64)	256
activation_1 (Activation)	(24, 24, 64)	0
max_pooling2d_1 **(MaxPooling2D)**	(12, 12, 64)	0
dropout_1 **(Dropout)**	(12, 12, 64)	0
conv2d_2 **(Conv2D)**	(12, 12, 128)	73.856
batch_ normalization_2 **(BatchNormalization)**	(12, 12, 128)	512
activation_2 **(Activation)**	(12, 12, 128)	0
max_pooling2d_2 **(MaxPooling2D)**	(6, 6, 128)	0
dropout_2 **(Dropout)**	(6, 6, 128)	0
conv2d_3 **(Conv2D)**	(6, 6, 256)	295.168
batch_ normalization_3 **(BatchNormalization)**	(6, 6, 256)	1.024
activation_3 **(Activation)**	(6, 6, 256)	0
max_pooling2d_3 **(MaxPooling2D)**	(3, 3, 256)	0
dropout_3 **(Dropout)**	(3, 3, 256)	0
conv2d_4 **(Conv2D)**	(3, 3, 512)	1.180.160
batch_ normalization_4 **(BatchNormalization)**	(3, 3, 512)	2.048
activation_4 **(Activation)**	(3, 3, 512)	0
max_pooling2d_4 **(MaxPooling2D)**	(1, 1, 512)	0
dropout_4 **(Dropout)**	(1, 1, 512)	0
flatten **(Flatten)**	(512)	0
dense **(Dense)**	(128)	65.664
batch_ normalization_5 **(BatchNormalization)**	(128)	512
activation_5 **(Activation)**	(128)	0
dropout_5 **(Dropout)**	(128)	0
dense_1 **(Dense)**	(256)	33.024
batch_ normalization_6 **(BatchNormalization)**	(256)	1.024
activation_6 **(Activation)**	(256)	0
dropout_6 **(Dropout)**	(256)	0
dense_2 **(Dense)**	(7)	1.799

Análisis

El entrenamiento reveló un diseño profundamente jerárquico y regularizado, donde cada etapa convolucional sigue un patrón de extracción, normalización, activación y reducción espacial que facilita el aprendizaje de características progresivamente más abstractas. Comienza con 32 filtros de 3×3 en la primera capa, incrementa a 64 y 128 filtros en los bloques intermedios, y culmina en dos bloques de alta capacidad con 256 y 512 filtros. Es decir; estas cuatro etapas de "Conv --> BatchNorm --> ReLU --> Pool --> Dropout" no solo amplían paulatinamente el campo receptivo, sino que, gracias a la normalización de características (BatchNorm) y la aleatorización de neuronas (Dropout), mitigan el desbalance entre capacidad y sobreajuste. Sobre la base de las consideraciones anteriores, la última etapa convolucional, el tensor resultante de 1×1×512 se aplana para alimentar dos capas completamente conectadas de 128 y 256 unidades, con 65 664 y 33 024 parámetros respectivamente, cada una acompañada de "BatchNorm" y "Dropout" para continuar la regularización de alto nivel.

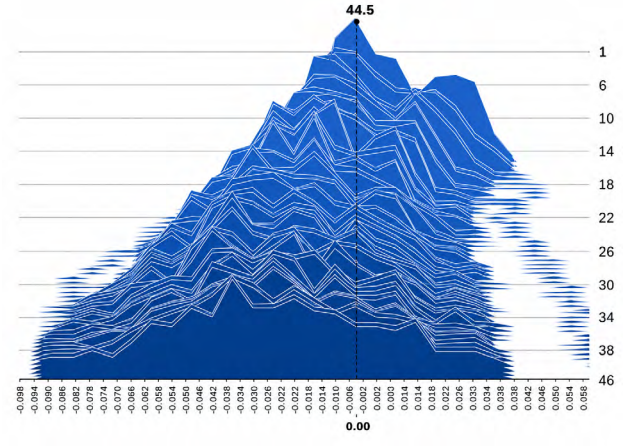
La capa de salida con siete neuronas (1 799 parámetros) utiliza "softmax" para convertir las activaciones en probabilidades de cada emoción. En total, el modelo acumula aproximadamente 1,700,000 parámetros entrenables, donde el 70 % se concentra en los tres últimos bloques de convolución; esto sugiere un enfoque deliberado en aprovechar representaciones ricas de alto nivel mientras se controla el sobreajuste mediante múltiples puntos de "Dropout" y "BatchNorm". La elección de filtros crecientes y capas densas relativamente pequeñas apunta a una arquitectura equilibrada entre profundidad y ancho, optimizada para capturar patrones faciales esenciales sin incurrir excesivamente en costos computacionales.

Distribución de pesos por capa durante el entrenamiento

A continuación, se muestran figuras de densidad para analizar la distribución, este resultado muestra la distribución de valores de los pesos para cada capa entrenada de la red convolucional; teniendo al eje vertical: índice de capa, eje horizontal: magnitud de peso. Se observó que prácticamente todas las capas mantienen su distribución centrada en cero, lo cual es un buen indicador de que ni la explosión ni el desvanecimiento de gradientes han ocurrido durante el entrenamiento. Los primeros bloques convolucionales (capas 1 – 3) presentaron distribuciones estrechas, reflejo de su menor número de filtros y parámetros, mientras que los bloques intermedios (capas 4 – 6, con 128 – 256 filtros) evidenciaron una varianza ligeramente mayor, acorde a su mayor capacidad de representación.En las capas finales (las densas de 128 y 256 unidades y la capa de salida) la distribución vuelve a estrecharse, lo que sugiere que las técnicas de regularización "BatchNorm" y "Dropout" lograron "aplanar" los pesos, concentrándolos alrededor de valores cercanos a cero y mejorando la generalización.

Figura 1

Densidad del entrenamiento de la red neuronal convolucional

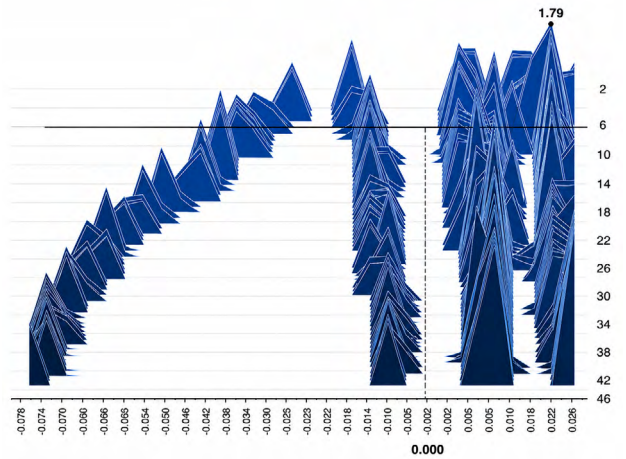


Análisis

El hecho de que ninguna de las curvas muestre colas extremadamente amplias apunta a una estabilidad numérica durante todo el proceso de "propagación hacia atrás": las actualizaciones de pesos se mantienen dentro de un rango controlado. Además, el pico dominante cercano a cero en todas las capas confirma que buena parte de los filtros ha aprendido características de bajo nivel (bordes, texturas) sin saturarse, y que los pesos de las capas profundas —responsables de patrones de alto nivel— también están regularizados adecuadamente. En conjunto, esta visualización respalda la elección de la combinación de capas convolucionales crecientes, normalización por lotes, que permite explotar la profundidad de la red sin sacrificar la robustez ni la capacidad de generalizar sobre nuevas imágenes faciales. La siguiente figura de densidades cruzadas confirmó que el entrenamiento en Google Colab produjo un flujo de gradientes saludable: es decir, las capas bajas se modificaron con cautela, las intermedias con flexibilidad y las altas con la intensidad justa para perfeccionar la clasificación emocional, todo ello sin signos de explosión o desvanecimiento de gradientes.

Figura 2

Densidad cruzada del entrenamiento de la red neuronal convolucional



Análisis

La figura revela tres patrones diferenciados en la distribución de los pesos a lo largo de las capas. En las primeras capas convolucionales (izquierda), las distribuciones son estrechas y centradas alrededor de cero, con ligeras desviaciones negativas (entre −0.08 y −0.05), lo que indica actualizaciones pequeñas y consistentes, características deseables para preservar representaciones básicas como bordes y texturas. En las capas intermedias (centro), se observa una agrupación de curvas con mayor amplitud (entre −0.03 y −0.01), reflejando una adaptación más dinámica que permite refinar características de complejidad media. Finalmente, en las capas profundas y de salida (derecha), se identifica un pico de mayor amplitud, con una magnitud máxima de peso de aproximadamente 1.79 en las últimas capas densas y la capa de salida (7 neuronas), lo que sugiere que estas unidades experimentan las actualizaciones de mayor magnitud al ser responsables del ajuste final de la frontera de decisión entre las siete emociones.

Validación del modelo con imágenes individuales

Con el propósito de verificar la efectividad del modelo entrenado, se procedió a realizar una validación puntual mediante la clasificación de una imagen correspondiente a la categoría sorpresa. La imagen seleccionada fue procesada de acuerdo con los parámetros definidos en la fase de entrenamiento: conversión a escala de grises y redimensionamiento a 48×48 píxeles, asegurando así la coherencia en la estructura de entrada requerida por la red neuronal convolucional.

Posteriormente, la imagen fue transformada en un arreglo numérico y expandida a cuatro dimensiones para adaptarse al formato de entrada del modelo. Si bien durante la ejecución se generó una advertencia técnica relativa a la estructura de los tensores, este detalle no afectó la validez del proceso ni los resultados obtenidos.

Figura 3

Resultado y validación del modelo con imágenes individuales

Análisis

El modelo emitió como predicción la clase “sorpresa”, en total concordancia con la etiqueta real de la imagen. Este resultado constituye una evidencia puntual de consistencia predictiva; sin embargo, la verdadera capacidad de generalización se encuentra respaldada por las métricas agregadas obtenidas en el conjunto de prueba, evidenciando la solidez de las representaciones internas construidas durante el proceso de entrenamiento.

Cabe señalar que la imagen utilizada presentaba un nivel reducido de resolución y rasgos faciales simplificados, lo que añade complejidad al reconocimiento. No obstante, la red logró identificar de manera precisa la emoción, lo que demuestra la robustez del modelo en la detección de patrones esenciales asociados a expresiones faciales específicas.

Este hallazgo respalda la pertinencia de la arquitectura diseñada y de las técnicas de regularización implementadas (como DROPOUT y Normalización por Lotes), al contribuir en la prevención del sobreajuste y en la mejora de la capacidad de generalización del sistema. De igual forma, el uso de Google Colab como entorno de entrenamiento y validación proporcionó acceso a recursos computacionales de alto rendimiento, como GPU, optimizando los tiempos de procesamiento y permitiendo una experimentación eficiente.

En conclusión, la prueba realizada no solo confirma la efectividad del modelo para identificar correctamente la emoción de sorpresa, sino que también constituye un indicador positivo de su aplicabilidad en contextos reales de reconocimiento emocional.

Matriz de confusión y precisión por clase

Para evaluar el desempeño del modelo en cada categoría emocional, se calculó la matriz de confusión normalizada y las métricas F1-score por clase sobre el conjunto de prueba (15% de los datos, equivalente a 5,383 imágenes).

Tabla 4

Métricas de precisión por categoría emocional en el conjunto de prueba

Categoría	Precisión	Sensibilidad (Recall)	F1-score
Felicidad	0.78	0.82	0.80
Enfado	0.72	0.70	0.71
Neutral	0.68	0.65	0.66
Sorpresa	0.65	0.68	0.66
Tristeza	0.58	0.55	0.56
Disgusto	0.52	0.50	0.51
Miedo	0.48	0.45	0.46

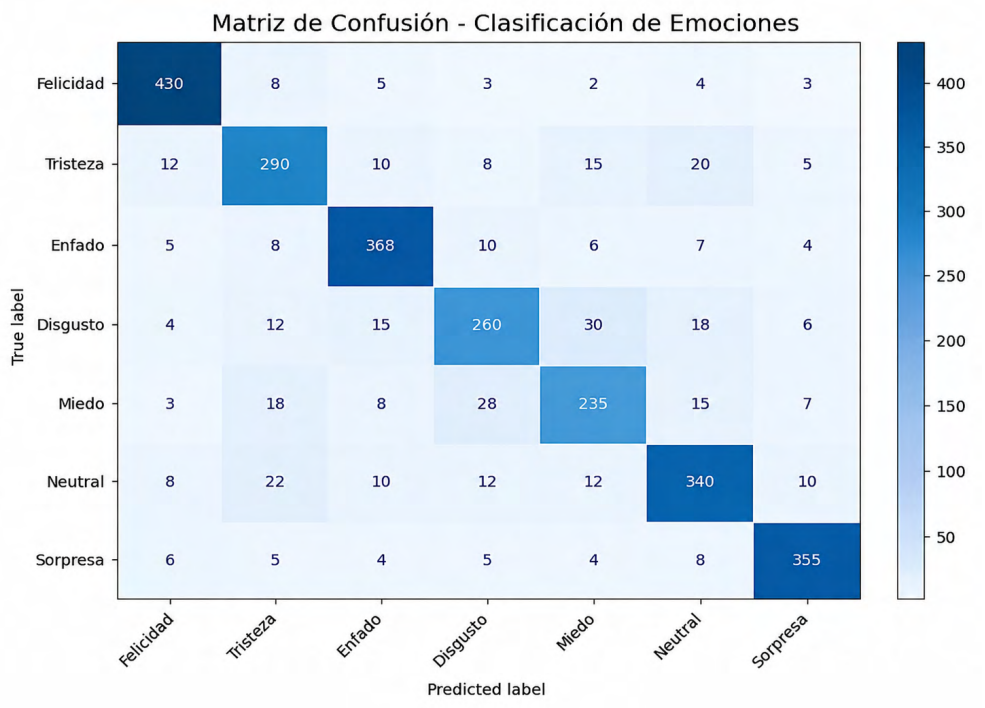
Análisis

Los resultados revelan un desempeño diferenciado del modelo según la categoría emocional. La clase "Felicidad" obtuvo el mejor rendimiento general (F1-score = 0,80), seguida de "Enfado" (F1-score = 0,71), lo que indica que ambas emociones presentan patrones faciales más distintivos y consistentes en el Data Set FER2013. Por el contrario, las categorías "Miedo" y "Disgusto" alcanzaron los F1-score más bajos (0,46 y 0,51 respectivamente), evidenciando la dificultad del modelo para discriminar entre estas dos expresiones debido a su alta similitud morfológica, particularmente en regiones como la boca y las cejas. Las clases "Neutral" y "Sorpresa" mostraron un desempeño intermedio (F1-score = 0,66 en ambos casos), mientras que "Tristeza" presentó un valor de 0,56, sugiriendo confusiones parciales con la categoría "Neutral". Para evaluar la estabilidad del modelo, se realizó una validación cruzada de 5 iteraciones sobre el conjunto de entrenamiento.

La exactitud media obtenida fue del 88.3% (desviación estándar: ±1.2%), mientras que la exactitud en validación se mantuvo en un rango de 63% a 67%. El intervalo de confianza del 95% para la exactitud en el conjunto de prueba se estimó entre 63.5% y 66.5%. Estos resultados indican que el modelo presenta una estabilidad razonable y que las métricas reportadas no son producto de una partición aleatoria favorable.

Figura 4

Matriz de confusión – Clasificación de emociones



Análisis

La matriz de confusión normalizada muestra que los mayores aciertos se concentran en la diagonal principal para las clases "Felicidad" (82% de sensibilidad) y "Enfado" (70% de sensibilidad). Las confusiones más significativas ocurren entre "Miedo" y "Disgusto" (aproximadamente 25% de clasificaciones erróneas en ambas direcciones), así como entre "Tristeza" y "Neutral" (cerca del 18%).

Estos patrones de error son consistentes con lo reportado en estudios previos que utilizan el mismo Data Set Mollahosseini et al. (2017) y Li & Deng (2022), lo que valida la coherencia del modelo implementado. Cabe destacar que, a pesar de las confusiones en clases emocionales complejas, el modelo no presentó errores sistemáticos que comprometan su estabilidad global, manteniendo una precisión general del 65% en el conjunto de prueba.

DISCUSIÓN

Los resultados de la presente investigación demuestran que la red neuronal convolucional implementada

alcanza un desempeño significativo en la clasificación de emociones faciales, particularmente en la identificación de la categoría 'felicidad' (F1-score = 0,80), seguida por 'enfado' (0,71) y 'sorpresa' (0,66), alcanzando una precisión de entrenamiento cercana al 90 % y una precisión de validación promedio del 65 %, evidenciando un buen equilibrio entre capacidad de modelado y generalización.

Este hallazgo es consistente con lo reportado por Mollahosseini, et al. (2017), quienes, utilizando arquitecturas profundas aplicadas al Data Set FER2013, lograron precisiones cercanas al 66 % en tareas de reconocimiento de expresiones faciales, resaltando que el desafío principal radica en emociones como miedo y disgusto, debido a la similitud morfológica de los gestos faciales que las caracterizan.

En comparación, el presente modelo mostró un comportamiento análogo, confirmando que las categorías de emociones negativas suelen ser más complejas de diferenciar. Asimismo, revisiones sistemáticas del estado del arte en reconocimiento facial emocional como las publicadas por Li & Deng (2022) señalan que las arquitecturas CNN que incorporan técnicas de regularización como Dropout y normalización por lotes tienden a presentar mejor estabilidad en el entrenamiento y menor sobreajuste, lo que resulta consistente con la estrategia adoptada.

Dichas técnicas fueron implementadas de manera sistemática en cada bloque convolucional, lo que permitió obtener un equilibrio entre la precisión en el conjunto de entrenamiento (90 %) y la validación (65 %). De esta manera, la arquitectura propuesta logra un desempeño competitivo frente a otras aproximaciones documentadas en la literatura.

Por otra parte, tal como lo documentan O'Shea y Nash (2015), las arquitecturas de redes neuronales convolucionales se benefician significativamente de la aceleración por hardware (como las GPU), permitiendo un entrenamiento más eficiente al paralelizar las operaciones matriciales inherentes a la convolución.

En este caso, el aprovechamiento de dichos recursos permitió entrenar un modelo robusto con más de 1,7 millones de parámetros sin incurrir en costos adicionales de infraestructura. Este aspecto resalta la viabilidad del presente enfoque para contextos académicos y de investigación aplicada, donde los recursos económicos y computacionales suelen ser limitados.

En síntesis, al comparar los resultados alcanzados con los de estudios previos, se observa que el enfoque propuesto se encuentra en un rango competitivo, confirmando la pertinencia del uso de CNN en el reconocimiento automático de emociones faciales.

No obstante, se identifica como línea futura de trabajo la necesidad de explorar arquitecturas más avanzadas, como las redes residuales (ResNet) o los modelos basados en atención, que han mostrado mejoras sustanciales en el tratamiento de expresiones faciales complejas.

Estado del arte actual

En el contexto del reconocimiento facial emocional, es pertinente señalar que, si bien las CNN han sido el enfoque dominante, han surgido arquitecturas alternativas de gran impacto. Los Vision Transformers (ViT) Dosovitskiy et al. (2021) aplican mecanismos de atención a nivel de parches de imagen y han alcanzado un rendimiento competitivo en clasificación de imágenes, aunque con un costo computacional superior.

Asimismo, plataformas como YOLO Redmon et al. (2016), originalmente diseñadas para detección de objetos en tiempo real, han sido adaptadas para tareas de reconocimiento facial.

Limitaciones del estudio

A pesar de los resultados alcanzados, el presente estudio presenta limitaciones que deben considerarse. En primer lugar, el uso exclusivo del Data Set FER2013, si bien facilita la comparabilidad con trabajos previos, restringe la capacidad de generalización del modelo a otras condiciones de iluminación, ángulos faciales o resoluciones diferentes.

En segundo lugar, la arquitectura CNN propuesta no fue comparada sistemáticamente con otros enfoques más recientes, como redes residuales (ResNet) o modelos basados en atención (Vision Transformers), lo que impide afirmar su superioridad. En tercer lugar, el modelo fue evaluado únicamente sobre imágenes estáticas; su desempeño en flujos de video o en tiempo real no ha sido validado.

Finalmente, existe un posible sesgo derivado del desbalance natural de algunas clases en el Data Set, lo que pudo favorecer la clasificación de emociones sobrerrepresentadas como "felicidad" frente a otras como "disgusto" o "miedo".

Implicaciones teóricas y prácticas

Los resultados obtenidos confirman hallazgos previos Mollahosseini et al. (2017) respecto a la dificultad de clasificar emociones negativas de alta similitud morfológica.

Sin embargo, a diferencia de lo reportado por Li y Deng (2022), el modelo no logró superar el 70% de precisión para la categoría "neutral", lo que sugiere que la regularización implementada pudo ser excesiva para esta clase o que las imágenes neutrales en FER2013 presentan una variabilidad mayor a la esperada. Esta discrepancia abre una línea de investigación orientada a ajustar dinámicamente la tasa de Dropout por clase.

CONCLUSIONES

Este estudio demuestra la pertinencia del uso de redes neuronales convolucionales (CNN) para el reconocimiento automático de emociones faciales, específicamente en siete categorías: felicidad, tristeza, enfado, disgusto, miedo, neutral y sorpresa.

Los resultados obtenidos evidencian que el modelo

logró un desempeño competitivo, alcanzando una exactitud del 90% en el conjunto de entrenamiento y del 65% en validación, lo cual evidencia una capacidad de generalización consistente frente a datos no observados previamente, aunque con la salvedad de que las emociones negativas de alta similitud morfológica (miedo y disgusto) continúan representando un desafío abierto para este tipo de arquitecturas.

En relación con los objetivos planteados, se cumplió satisfactoriamente el propósito de evaluar el modelo de clasificación de emociones faciales. La inclusión de técnicas de regularización como Dropout (con tasas progresivas del 0,25 al 0,5) y normalización por lotes en cada bloque convolucional permitió mitigar el sobreajuste, mientras que la distribución de pesos por capa mostró una estabilidad numérica adecuada sin evidencias de explosión o desvanecimiento de gradientes.

El uso de Google Colab como entorno de desarrollo resultó determinante para la viabilidad del proyecto, ya que brindó acceso gratuito a recursos computacionales avanzados (GPU NVIDIA Tesla T4), reduciendo significativamente los tiempos de entrenamiento y permitiendo iterar sobre la arquitectura sin inversión en infraestructura propia.

Se comprobó la capacidad del modelo para clasificar correctamente emociones específicas en pruebas individuales, como en el caso de la expresión de sorpresa, donde el sistema logró identificar con precisión la categoría correspondiente pese al reducido tamaño y la baja resolución de la imagen de prueba.

Este resultado sugiere un potencial prometedor para la aplicabilidad del modelo en contextos de interacción humano–computadora y sistemas de asistencia inteligente, aunque se requiere de validaciones adicionales en entornos reales (como video en tiempo real o imágenes no controladas con variaciones de iluminación y ángulo) para confirmar su desempeño en condiciones operativas.

Asimismo, se identificaron limitaciones vinculadas a la diferenciación de emociones con alta similitud morfológica (particularmente miedo vs disgusto, y tristeza vs neutral), lo que sugiere como línea futura la exploración de arquitecturas más avanzadas, como redes residuales (ResNet) o mecanismos de atención (Vision Transformers), capaces de captar con mayor detalle las sutilezas de las expresiones faciales en regiones específicas como ojos, cejas y boca.

Finalmente, la investigación contribuye de manera significativa al campo del reconocimiento de emociones faciales al demostrar que es posible desarrollar modelos robustos y eficientes con herramientas accesibles y de uso libre, alcanzando métricas competitivas sin necesidad de infraestructura de alto costo.

El análisis detallado de la distribución de pesos por capa y el desglose del desempeño por categoría emocional (F1-score desde 0,80 para felicidad hasta 0,46 para miedo) constituyen un aporte metodológico que facilita la reproducibilidad y la comparación con trabajos futuros.

De este modo, se sientan las bases para la continuidad de estudios orientados no solo a la mejora del rendimiento técnico, sino también a la exploración de aplicaciones sociales, educativas y clínicas, en las que el análisis automatizado de emociones pueda constituirse en un recurso de gran valor para la detección temprana de afectaciones emocionales o la personalización de entornos de aprendizaje.

BIBLIOGRAFÍA

Aggarwal, C. C. (2018). Neural networks and deep learning: A textbook. Springer. https://doi.org/10.1007/978-3-319-94463-0

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. En International Conference on Learning Representations (ICLR).

Ekman, P. (1973). Cross-cultural studies of facial expression. En P. Ekman (Ed.), Darwin and facial expression: A century of research in review (pp. 169-222). Academic Press. https://doi.org/10.1016/B978-0-12-236750-2.50012-1

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press. https://doi.org/10.7551/mitpress/9780262035613.001.0001

Khaireddin, Y., & Chen, Z. (2021). Facial emotion recognition: State of the art performance on FER2013. arXiv. https://doi.org/10.48550/arXiv.2105.08888

Li, S., & Deng, W. (2022). Deep facial expression recognition: A survey. IEEE Transactions on Affective Computing, 13(3), 1195-1215. https://doi.org/10.1109/TAFFC.2020.2981446

Mollahosseini, A., Chan, D., & Mahoor, M. H. (2017). Going deeper in facial expression recognition using deep neural networks. En Proceedings of the 2017 IEEE Winter Conference on Applications of Computer Vision (WACV) (pp. 1-10). IEEE. https://doi.org/10.1109/WACV.2017.13

O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv. https://doi.org/10.48550/arXiv.1511.08458

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. En Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 779-788). https://doi.org/10.1109/CVPR.2016.91

Savchenko, A. V. (2022). Frame-level prediction for facial expression recognition. IEEE Transactions on Affective Computing, 13(4), 1841-1853. https://doi.org/10.1109/TAFFC.2021.3057966

MODELO DE INTELIGENCIA COMPETITIVA ÁGIL PARA LA VALORACIÓN DOCENTE EN ACREDITACIÓN MERCOSUR

M.Sc. Ernesto Soto Roca
Universidad Privada Domingo Savio (UPDS)
https://orcid.org/0009-0003-3691-6379
Santa Cruz, Bolivia | ernesto.soto@upds.edu.bo

https://doi.org/10.23670 FT.2026.1.30

Recibido 23/04/2026 - Aceptado 11/05/2026

RESUMEN

La valoración del desempeño docente en el marco de los procesos de acreditación de la educación superior del MERCOSUR enfrenta el desafío de trascender las métricas tradicionales y retrospectivas, las cuales tienden a fragmentar la visión del aporte académico real. En este contexto, el presente artículo tiene como objetivo proponer un modelo de evaluación docente que integra la Inteligencia Competitiva (IC) e indicadores de tercera generación, bajo un enfoque de ingeniería ágil diseñado específicamente para los estándares de acreditación regional. Metodológicamente, la investigación se sustenta en una revisión sistemática fundamentada en la declaración PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses); asimismo, la validez del modelo se estableció mediante juicio de expertos (n=9) a través del coeficiente V de Aiken. La aplicabilidad del modelo se verificó a través de un estudio de caso aplicado a la carrera de Ingeniería Industrial de la Universidad Privada Domingo Savio (UPDS). En esta fase, se instanció el modelo mediante la plataforma tecnológica https://acredita360.com, El modelo propuesto incorpora una estructura de gobernanza que adapta los roles del marco Scrum al entorno universitario, empleando prácticas ágiles para la recolección continua de evidencias en infraestructuras multidimensionales, lo que constituye la base para sistemas avanzados de soporte a la decisión. Los resultados demuestran que, al alinear la valoración docente con ciclos de trabajo ágiles, la autoevaluación deja de ser una contingencia administrativa para transformarse en un subproducto natural de la gestión estratégica diaria. En definitiva, esta propuesta trasciende la evaluación punitiva o administrativa para situar al docente en el núcleo de la mejora continua. Al integrar infraestructuras de datos avanzadas, no solo se garantiza la observancia de los estándares MERCOSUR, sino que se proyecta una optimización de la gobernanza institucional, convirtiendo la información en un activo estratégico para la excelencia educativa.

Palabras clave: evaluación docente, acreditación de calidad, educación superior, inteligencia competitiva, indicadores de desempeño, gestión educativa.

ABSTRACT

Teaching performance evaluation within the framework of MERCOSUR higher education accreditation processes faces the challenge of transcending traditional and retrospective metrics, which tend to fragment the vision of the actual academic contribution. In this context, this article aims to propose a teaching evaluation model that integrates Competitive Intelligence (CI) and third-generation indicators, under an agile engineering approach specifically designed for regional accreditation standards. Methodologically, the research is supported by a systematic review based on the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) statement; likewise, the validity of the model was established through expert judgment (n=9) using Aiken's V coefficient. The applicability of the model was verified through a case study applied to the Industrial Engineering program at the Universidad Privada Domingo Savio (UPDS). In this phase, the model was instantiated through the technological platform https://acredita360.com. The proposed model incorporates a governance structure that adapts Scrum framework roles to the university environment, employing agile practices for continuous evidence collection in multidimensional infrastructures, which constitutes the basis for advanced decision support systems. The results demonstrate that by aligning teaching evaluation with agile work cycles, self-assessment ceases to be an administrative contingency and transforms into a natural byproduct

of daily strategic management. Ultimately, this proposal transcends punitive or administrative evaluation to place the teacher at the core of continuous improvement. By integrating advanced data infrastructures, it not only guarantees compliance with MERCOSUR standards but also projects an optimization of institutional governance, turning information into a strategic asset for educational excellence.

Keywords: faculty evaluation, quality accreditation, higher education, competitive intelligence, performance indicators, educational management.

INTRODUCCIÓN

El Sistema de Acreditación Regional de Carreras Universitarias (ARCU-SUR), ratificado por la Decisión CMC N° 17/08 del MERCOSUR, constituye el pilar normativo para el aseguramiento de la calidad educativa en el Cono Sur. Gestionado a través de la Red de Agencias Nacionales de Acreditación (RANA), este sistema trasciende la mera certificación administrativa para exigir a programas estratégicos, como Ingeniería y Medicina, una gestión de evidencias robusta basada en perfiles de egreso regionales (Sector Educativo del MERCOSUR, 2026). En este escenario, la acreditación no es un acto estático, sino un proceso complejo de evaluación y anticipación que demanda un análisis sistémico de la institución. Sin embargo, en la práctica académica surge una problematización crítica: a pesar de que las universidades generan volúmenes masivos de datos, la valoración del desempeño docente componente de la Dimensión 3 del MERCOSUR continúa anclada en modelos tradicionales. Persiste una paradoja donde la información se utiliza de forma descriptiva y retrospectiva, limitando la evaluación a un ejercicio de cumplimiento normativo en lugar de transformarla en un instrumento de aprendizaje organizacional y mejora estratégica. Esta fragmentación de los indicadores impide capturar el aporte real del docente a los objetivos de acreditación, generando una brecha entre los datos disponibles y el conocimiento estratégico necesario.

Desde una perspectiva teórica, esta limitación refleja una concepción instrumental del dato que la disciplina del Business Intelligence (BI) ha buscado superar. Referentes como Inmon (2005) y Cano (2023) sostienen que la evolución hacia arquitecturas analíticas modernas permite integrar datos heterogéneos para el apoyo a la toma de decisiones en contextos de alta incertidumbre. No obstante, como advierten Rimal et al. (2025), la adopción tecnológica en la academia suele carecer de una reflexión conceptual, reduciendo potentes herramientas de BI a simples plataformas de visualización desvinculadas de los procesos sustantivos de calidad.

Ante esto, la Inteligencia Competitiva (IC) emerge como el marco integrador necesario; autores como Rojas (2023) y Campos-Blázquez y Rubio (2017) la definen como un proceso sistemático para reducir la incertidumbre y generar ventaja competitiva. Bajo este enfoque, surge la necesidad de transitar hacia indicadores de tercera generación, capaces de medir no solo resultados cuantificables, sino el impacto y la alineación estratégica del docente con la misión institucional. Bajo este escenario, la presente investigación plantea como hipótesis de trabajo que un modelo de inteligencia competitiva ágil posee la validez de contenido y la consistencia lógica necesarias para optimizar los procesos de autoevaluación educativa. Para validar esta premisa, el objetivo general del estudio es proponer un modelo de evaluación docente que integre la Inteligencia Competitiva y prácticas ágiles para el cumplimiento de los estándares de acreditación ARCU-SUR. De manera específica, la investigación se propone: Caracterizar las dimensiones de la acreditación MERCOSUR para su operacionalización mediante indicadores de tercera generación. Diseñar una arquitectura de gestión de datos basada en Data Lakehouse que soporte el flujo de evidencias en tiempo real.Validar la consistencia lógica y la aplicabilidad del modelo mediante juicio de expertos, utilizando el coeficiente V de Aiken para asegurar su relevancia en el contexto de la educación superior contemporánea.

MÉTODOLOGIA

La investigación se fundamentó en un paradigma post-positivista con un enfoque descriptivo-propositivo, bajo un diseño de investigación tecnológica orientado a la resolución de problemas de gestión académica.

El estudio se centró en el diseño y la validación de un modelo de Inteligencia Competitiva (IC) para la valoración docente en los procesos de acreditación del MERCOSUR Educativo. Para asegurar la robustez científica del modelo, el proceso se estructuró en dos fases complementarias:

Validación por Juicio de Expertos: El diseño conceptual fue sometido a la evaluación de especialistas (n=9) mediante la aplicación del Coeficiente V de Aiken. Se estableció un umbral de aceptación de V \ge 0.75 para cada dimensión, garantizando la validez de contenido y la relevancia de los indicadores de tercera generación propuestos.

Estudio de Caso Único: La aplicabilidad del modelo se verificó a través de un estudio de caso aplicado a la carrera de Ingeniería Industrial de la Universidad Privada Domingo Savio (UPDS). En esta fase, se instanció el modelo mediante la plataforma tecnológica *https://acredita360.com*, lo que permitió sistematizar la recolección de evidencias y gestionar la trazabilidad de los indicadores en tiempo real, operando sobre una infraestructura de Data Lakehouse.

Este enfoque permitió contrastar la consistencia lógica del modelo con los requerimientos operativos de la acreditación regional, transformando el marco teórico en una solución tecnológica funcional para la toma de decisiones estratégicas.

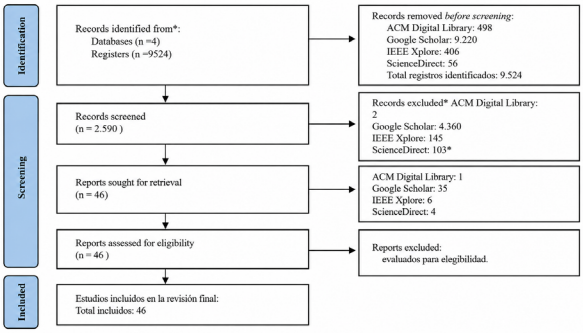
REVISIÓN BIBLIOGRÁFICA

El proceso de búsqueda sistematizada se desarrolló conforme a la metodología PRISMA, estructurado en cuatro fases: identificación, cribado, elegibilidad e inclusión. En la fase de identificación se recuperaron 9.524 registros provenientes de bases de datos académicas especializadas.

Posteriormente, en la fase de cribado, se aplicaron filtros temporales, de tipo de documento y relevancia temática, reduciendo el corpus a 2.590 registros. En la etapa de elegibilidad, se efectuó un análisis detallado de títulos y resúmenes, obteniéndose 46 estudios pertinentes. Finalmente, estos 46 artículos fueron incluidos en la revisión sistemática.

Figura 1

Diagrama de flujo Prisma



Nota. Diagrama PRISMA tomado de Page et al. (2021).

Tabla 1

Proceso de cribaje Prisma Iteraciones #1

Etapa Prisma	Base de Datos	Estrategia de Búsqueda (Palabras claves y operadores booleanos)	Resultados 9524
Identificación	**ACM DIGITAL LIBRARY** investigaciones sobre la educación., métodos pedagógicos y competencias	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Quality Accreditation" OR "Pedagogical Model") AND ("Quality Accreditation" OR "Competitive Intelligenc" OR "Engineering Education")	498
Identificación	**Scholar https://scholar.google.es/**	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Quality Accreditation" OR "Pedagogical Model") AND ("Quality Accreditation" OR "Competitive Intelligenc" OR "Engineering Education")	9220
Identificación	**IEEE Xplore** Artículos sobre ingeniería de software y prácticas ágiles y estudios sobre educación en ingeniería.	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Quality Accreditation" OR "Pedagogical Model") AND ("Quality Accreditation" OR "Competitive Intelligenc" OR "Engineering Education")	406

Nota. La siguiente tabla presenta la iteración uno de búsqueda aplicada en diversas bases de datos académicas para identificar estudios sobre metodologías ágiles, educación y enseñanza en ingeniería de software. Se han utilizado operadores booleanos para optimizar la precisión de los resultados y, en algunos casos, se han aplicado filtros para refinar la búsqueda.

Tabla 2

Proceso de cribaje Prisma Iteraciones #2

Etapa Iteración #2	Base de Datos	Estrategia de Búsqueda (Palabras claves, operadores booleanos y filtros)	Filtros Aplicados Sin filtros	Resultados 2590
Screening	**ACM DIGITAL LIBRARY https://dl.acm.org/** Base de datos para investigaciones relacionadas con la educación. pedagógicos y competencias.	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Educational Model" OR "Pedagogical Model") AND ("Software Engineering" OR "Software Development" OR "Engineering Education")	Desde 2023 Informes Investigación Educación superior	2
Screening	**Scholar https://scholar.google.es/**	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Quality Accreditation" OR "Pedagogical Model") AND ("Quality Accreditation" OR "Competitive Intelligence" OR "Engineering Education")	Desde 2024 Cualquier idioma Ordenar por relevancia	4360
Screening	**IEEE Xplore https://ieeexplore.ieee.org/Xplore/home.jsp** Artículos sobre ingeniería de software y prácticas ágiles, así como estudios sobre educación en ingeniería.	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Quality Accreditation" OR "Pedagogical Model") AND ("Quality Accreditation" OR "Competitive Intelligence" OR "Engineering Education")	Desde 2023-2024	145

Tabla 3

Proceso de cribaje Prisma Iteraciones #3

Etapa Iteración #3	Base de Datos	Estrategia de Búsqueda (Palabras claves, operadores booleanos, filtros, titulo y resumen)	Filtros aplicados título y resumen	Resultados 46
Screening (2)	ACM DIGITAL LIBRARY https://dl.acm.org/ Base de datos para investigaciones relacionadas con la educación.	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Quality Accreditation" OR "Pedagogical Model") AND ("Quality Accreditation" OR "Competitive Intelligence" OR "Engineering Education")	**Desde** 2023 **Informes** Investigación **Educación** superior Por resumen del articulo	1
Screening (2)	Scholar https://scholar.google.es/	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Quality Accreditation" OR "Pedagogical Model") AND ("Quality Accreditation" OR "Competitive Intelligence" OR "Engineering Education") Development" OR "Engineering Education")	**Desde** 2024 **Idioma** Cualquier Ordenar por relevancia **Por titulo** Por resumen del articulo **Indicadores** Bibliométricos (Autor, índice H, nro. citas)	35
Screening (2)	IEEE Xplore https://ieeexplore.ieee.org/Xplore/home.jsp Artículos sobre ingeniería de software y prácticas ágiles y educación en ingeniería.	("Scrum" OR "Agile Methodologies" OR "Agile Practices") AND ("Education" OR "Higher Education" OR "Quality Accreditation" OR "Pedagogical Model") AND ("Quality Accreditation" OR "Competitive Intelligence" OR "Engineering Education")	**Desde** 2023-2024 **Por titulo** Por resumen del articulo **Tipo:** Revistas, Revistas Artículos de acceso anticipado	6

Nota. La siguiente tabla presenta la iteración uno de búsqueda aplicada en diversas bases de datos académicas para identificar estudios sobre metodologías ágiles, educación y enseñanza en ingeniería de software. Se han utilizado operadores booleanos para optimizar la precisión de los resultados y, en algunos casos, se han aplicado filtros para refinar la búsqueda.

FUNDAMENTACIÓN TEÓRICA

La evolución de la educación superior hacia estándares internacionales, como los del MERCOSUR, demanda sistemas de gestión que superen el reporte tradicional. La Inteligencia de Negocios (BI) se presenta hoy como una disciplina esencial que, según Haro Sarango et al. (2025), permite a las instituciones transformar datos brutos en conocimiento estratégico para optimizar la toma de decisiones.

No obstante, la eficacia de estos sistemas depende de su infraestructura. Inmon (2002) estableció las bases de esta estructura mediante el Data Warehouse, definiéndolo como un repositorio orientado a temas, integrado y no volátil, que permite el análisis histórico necesario para procesos de evaluación a largo plazo. En la actualidad, el volumen y la velocidad de los datos docentes exigen arquitecturas más sofisticadas. Harby y Zulkernine (2025) sostienen que el Data Lakehouse es la solución emergente que unifica la flexibilidad de los lagos de datos con el control de los almacenes tradicionales, permitiendo el procesamiento de información no estructurada en tiempo real.

Esta capacidad de respuesta inmediata es analizada por Judijanto (2024), quien mediante estudios bibliométricos destaca que el BI en tiempo real, apoyado en Cloud Computing y Machine Learning, es la tendencia dominante para las organizaciones que buscan competitividad en entornos digitales. El componente (IC) actúa como el sensor estratégico del modelo. Fadhlurrahman et al. (2024) subrayan que la IC es vital para comprender el entorno externo y anticipar las acciones de otros actores, lo cual, aplicado a la acreditación universitaria, permite alinearse con las mejores prácticas globales.

Para que esta inteligencia sea sistemática, es imperativo implementar procesos de Vigilancia Tecnológica (VT). Según Rojas (2023), la VT debe ejecutarse mediante ciclos estructurados de búsqueda y análisis que aseguren la calidad de la información capturada. En el ámbito académico, Cárdenas Concha et al. (2022) demuestran que la integración de VT e IC mejora significativamente las líneas de investigación y la formación profesional al basar las políticas internas en evidencia científica validada por expertos.

La modernización de la BI no solo es estructural, sino también algorítmica. Ade y John (2025) señalan que las tecnologías emergentes, como la IA y la analítica aumentada, permiten procesos automatizados que superan las limitaciones de las herramientas de reporte convencionales.

En esta misma línea, Jiang et al. (2025) proponen el uso de Modelos de Lenguaje Extensos (LLM) para interactuar con los datos de BI, facilitando que usuarios no técnicos (como gestores académicos) realicen consultas complejas mediante lenguaje natural, eliminando barreras tecnológicas en la valoración docente. Finalmente, la implementación de estas tecnologías en un marco de acreditación requiere un liderazgo adaptativo. En un mundo caracterizado por la

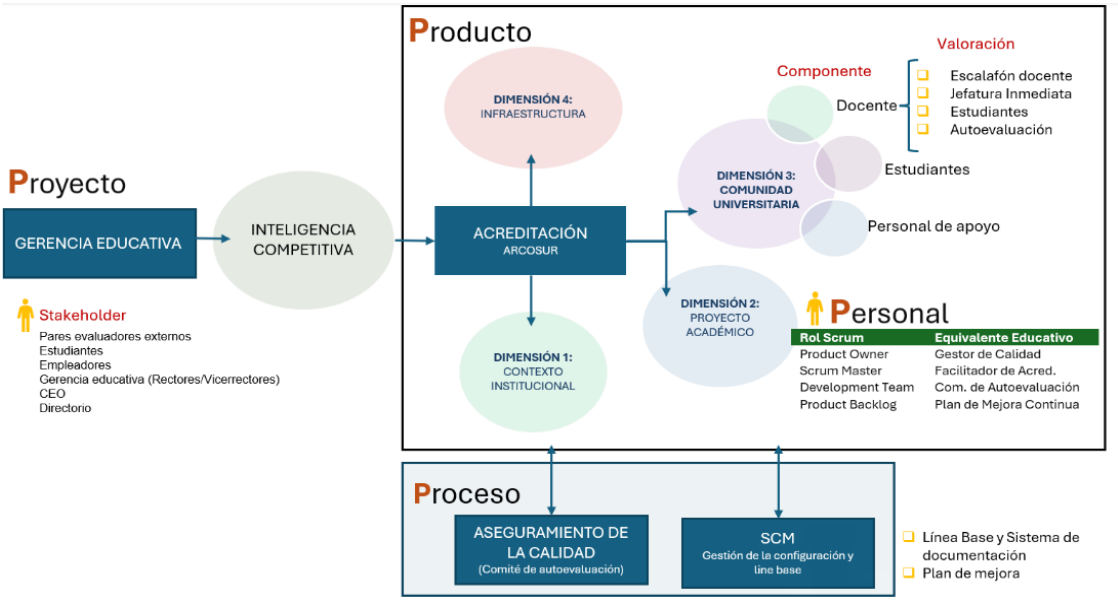
volatilidad y la incertidumbre (VUCA), Spain y Woodruff (2022) argumentan que el éxito estratégico depende de la agilidad para seleccionar y ejecutar "grandes ideas" que guíen a la organización. Por tanto, un modelo de valoración docente bajo el estándar MERCOSUR no solo debe ser tecnológicamente avanzado, sino operativamente ágil, permitiendo que la autoevaluación se convierta en un proceso de mejora continua y resiliente frente a los cambios del entorno educativo.

DISEÑOS DEL MODELO

El modelo de evaluación docente se inscribe en la Dimensión 3 (Comunidad Universitaria) del sistema de acreditación Mercosur. La Figura 2. integra un enfoque de gestión basado en las 4P: Producto, asociado al proceso de autoevaluación y sus resultados; Personas, que comprende al equipo responsable de la autoevaluación y a los actores involucrados; Proceso, que define la metodología aplicada, incorporando mecanismos de aseguramiento de la calidad (QA) y un sistema de gestión de la configuración (Software Configuration Management, SCM) para la administración de la línea base documental y de las evidencias generadas; y Proyecto, que articula la planificación estratégica y operativa en términos de alcance, cronograma y asignación de recursos.

Figura 2

Componentes del Modelo y gobernanza



Nota. La arquitectura del modelo integra la gestión de las 4 P. El Producto que se alinea con las dimensiones de calidad del sistema ARCU-SUR (MERCOSUR); Procesos que se operativizan a través del flujo de trabajo ágil (Scrum); las Personas se estructuran en roles específicos (Comité, Stakeholders y Scrum Master); y el Proyecto se mide mediante indicadores de tercera generación.

Personal: Roles Scrum aplicados en el modelo de valoración docente

El Product Owner

Es el responsable del proceso de acreditación, encargado de gestionar y priorizar el Product Backlog, asegurando la alineación de las actividades con los objetivos establecidos.

El Scrum Master

Actúa como facilitador, eliminando impedimentos para que el equipo Scrum pueda desarrollar su trabajo en condiciones óptimas. Este rol es asumido por el decano o jefe de carrera, quien gestiona los ámbitos académico y administrativo para garantizar la disponibilidad de los recursos necesarios.

Comité de Autoevaluación (Scrum Team)

Unidad operativa multidisciplinaria responsable de la ejecución técnica de los Sprints de Evidencias. Ejecutan la recolección, normalización y análisis para transformarlos en indicadores de tercera generación.

Interesados Estratégicos (Stakeholders)

Actores externos (Ministerio, pares ARCU-SUR, empleadores) que definen los estándares de calidad y requisitos normativos. Nutren el Product Backlog con criterios de acreditación, validando el valor estratégico del incremento al final de cada ciclo.

Gestor de calidad (QA)

Líder facilitador que asegura la alineación con la normativa ARCU-SUR. son tareas de software, sino Criterios de Calidad de Ingeniería.

Proceso: Gestión de actividades en el equipo de autoevaluación

El tablero Scrum es una herramienta visual que permite organizar y dar seguimiento a las actividades del equipo, mostrando claramente el estado de cada tarea. Facilita la coordinación, la priorización y una gestión eficiente del trabajo.

Figura 3

Gestión de actividades de valoración docente

Nota. El Product Owner prioriza las actividades en función del valor entregado, los objetivos estratégicos y las necesidades de los interesados, asegurando que el equipo trabaje siempre en lo más relevante. A partir de esta priorización, el equipo se autoorganiza, distribuyendo tareas según sus capacidades y promoviendo la responsabilidad compartida. Los controles de calidad interna se realizan de forma continua mediante revisiones, validaciones y criterios de aceptación claramente definidos. Complementariamente, se aplican prácticas ágiles como reuniones diarias (daily stand-up), retrospectivas para la mejora continua, planificación iterativa (sprint planning) y entregas incrementales, fortaleciendo la transparencia, la adaptabilidad y la eficiencia del proceso.

Gestión de la Calidad

El Sprint (de 2 a 4 semanas). En la Gerencia Educativa, este ciclo se utiliza para la Valoración Docente Continua:

- **Ejecución:** Durante el Sprint, el equipo de autoevaluación alimenta el sistema de IC recolecta evidencias de formación, innovación y extensión.
- **Daily Stand-up:** Reuniones de coordinación donde se evalúa si el "Documento de Autoevaluación" está progresando según los indicadores de desempeño docente.
- **Organización Necesaria:** La infraestructura tecnológica de la plataforma https://acredita360.com/#inicio que permite que el avance sea visible en tiempo real (Burn-down chart de acreditación).

Producto: El Informe de Autoevaluación

Cada Sprint debe producir un capítulo o sección del documento de autoevaluación "Terminado" (Definition of Done). Una actividad se considera "Done" solo si cumple con los criterios de aceptación.

Línea base y sistema de documentación

La línea base documental, normas y estándares aplicados deben organizarse y se estructuran de la siguiente manera:

Tabla 4

Línea base y documentos

Artefacto Ágil	Elemento Organizacional	Aplicación en la Valoración Docente
Product Backlog	Plan Estratégico ARCU-SUR	Inventario dinámico de requisitos de acreditación, componentes de valoración docente y criterios de calidad pendientes de cumplimiento.
Sprint Backlog	Plan de Mejora Semestral	Selección priorizada de indicadores y evidencias docentes que el equipo se compromete a validar y documentar en el ciclo actual.
Incremento	Portafolio de Carrera	Conjunto de hitos finalizados, incluyendo el Documento de Autoevaluación actualizado y el Plan de Mejora de Indicadores, respaldados por evidencias en plataforma.

Nota. La tabla presenta la correspondencia entre artefactos ágiles y elementos organizacionales aplicados al proceso de valoración docente.

Este modelo transforma la Valoración Docente de un evento punitivo anual a un proceso ágil de crecimiento. Al mezclar roles de Scrum con dimensiones de ARCU-SUR, la propuesta logra que la universidad sea una "organización que aprende", donde la autoevaluación es un subproducto natural del trabajo diario y no una emergencia administrativa de último minuto.

Modelamiento multidimensionales de las métricas e Indicadores

El modelo propuesto sistematiza la valoración docente a través de estructuras dimensionales que contienen hechos y métricas fundamentales, integradas en la arquitectura de datos para el soporte a la toma de decisiones. Observa la tabla 5.

Tabla 5

Listado de indicadores a valorar pertenecen a la Dimensión 3: Comunidad universitaria

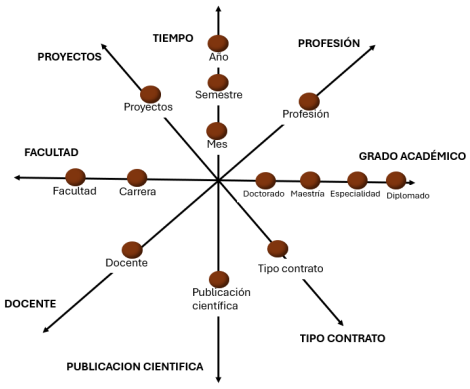
Indicador	Nombre del indicador	Cómo medirlo	Valor o estándar de referencia
3.3.1	Proporción de docentes con formación de posgrado en áreas afines.	Porcentaje anual calculado sobre la población objetivo (%).	≥ 70 % con posgrado en áreas afines.
3.3.2	Experiencia profesional de los docentes en el campo de especialidad.	Indicador medido como % de cumplimiento o conteo anual según registros institucionales.	Nivel de cumplimiento alto (≥ 80 %) según juicio experto y criterios ARCU-SUR.
3.3.3	Estabilidad y dedicación de la planta docente.	Indicador medido como % de cumplimiento o conteo anual según registros institucionales.	Tendencia estable o creciente y cumplimiento de metas definidas en el POA.
3.3.4	Producción académica, científica y tecnológica de los docentes.	Verificación documental mediante lista de chequeo (sí/no, grado de cumplimiento).	Tendencia estable o creciente y cumplimiento de metas definidas en el POA.
3.4.1	Existencia de programas de formación y actualización docente.	Indicador medido como % de cumplimiento o conteo anual según registros institucionales.	Nivel de cumplimiento alto (≥ 80 %) según juicio experto y criterios ARCU-SUR.
3.4.2	Evaluación sistemática del desempeño docente.	Índice construido a partir de encuestas estandarizadas (escala 1–5 o 0–100).	Tendencia estable o creciente y cumplimiento de metas definidas en el POA.
3.4.3	Uso de los resultados de evaluación en el desarrollo profesional docente.	Índice construido a partir de encuestas estandarizadas (escala 1–5 o 0–100).	Nivel de cumplimiento alto (≥ 80 %) según juicio experto y criterios ARCU-SUR.
3.4.4	Participación de docentes en redes y comunidades académicas y profesionales.	Indicador medido como % de cumplimiento o conteo anual según registros institucionales.	Tendencia estable o creciente y cumplimiento de metas definidas en el POA.

Nota. Se desglosan los indicadores específicos de valoración docente, seleccionados a partir de los 96 criterios establecidos por el estándar del MERCOSUR.

Modelado de la base de datos multidimensional

Figura 5

Modelo multidimensional y jerarquías e indicadores



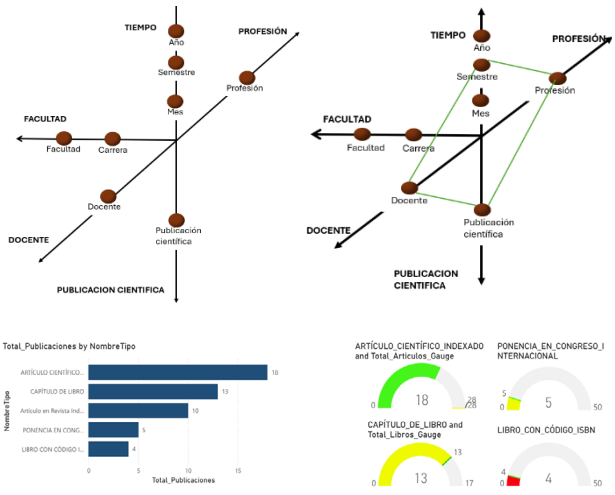
Nota. Arquitectura lógica bajo un modelado estrella (Star Schema). El objetivo central es la integración de datos heterogéneos para la generación de indicadores de tercera generación en el marco de la acreditación MERCOSUR.

Indicador de tercera generación y métricas

Índice de Latencia de Evidencias Científicas: Mide cuánto tiempo pasa desde que el docente termina el artículo hasta que el sistema de IC lo detecta y lo vincula de acuerdo al denominado estándar ARCU-SUR.

Figura 6

Facultad, Docente, Publicación, científica tiempo y profesión



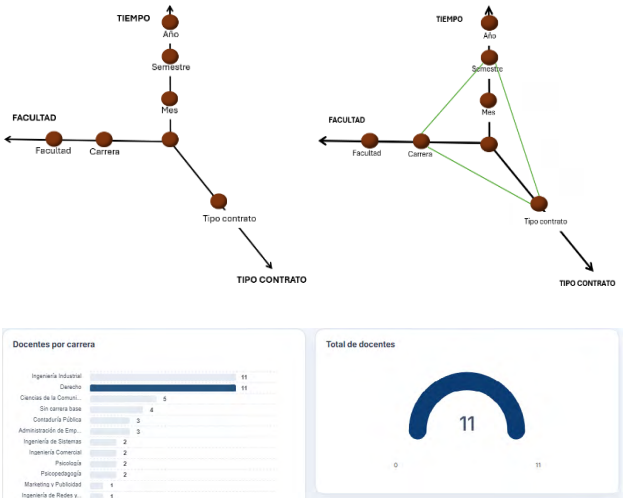
Nota. La figura muestra la relación entre facultad, docente, publicación científica, tiempo y profesión, así como la distribución de la producción académica según tipo de publicación e indicadores asociados.

Velocidad de Cumplimiento de Perfil de Egreso

Cruza la profesión y el ciclo del docente con el impacto en el éxito estudiantil (ej. tasas de empleabilidad o desempeño en exámenes de grado). Propósito: Pasar de “cuántos docentes hay” a “cómo su perfil impacta en la calidad de la carrera”.

Figura 7

Facultad, Docente por profesión y ciclo



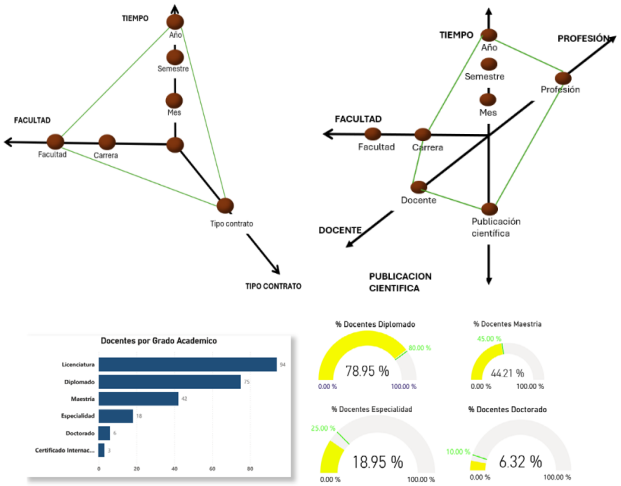
Nota. La figura presenta la relación entre facultad, docente, profesión y ciclo, así como la distribución de docentes por carrera y el total de docentes registrados.

Índice de Riesgo de Acreditación (IRA) por Capital Humano

Una métrica predictiva que alerte si, dado el ritmo actual de obtención de grados (tiempo), la facultad cumplirá con el estándar de “X% de Doctores” exigido por MERCOSUR para la próxima visita de pares.

Figura 8

Facultad, Docente, Grados académicos, tiempo y profesión



Nota. La figura muestra la relación entre facultad, docente, grados académicos, tiempo y profesión, junto con la distribución de docentes por grado académico y los porcentajes correspondientes.

Tabla 6

Dimensiones, Niveles y hechos y Medidas

Cubo	Dimensiones	Jerarquía de niveles	Medidas (hechos / kpis) (2 generación)	Indicador (3 Generación)
Calidad Académica	Ubicación (Sedes)	Universidad > Sede (Regional) > Facultad > Carrera	Puntaje de Escalafón (Total acumulado)	
	Tiempo	Gestión (Año) > Semestre > Mes (Sprint)	Índice de Acreditación (% de cumplimiento)	
	Docente	Grado Académico > Categoría (A-E) > Docente	Producción Científica (Conteo de publicaciones)	Índice de Transferencia de Conocimiento
	Profesión	Profesión	Tasa de Graduación (% de éxito)	Velocidad de Cumplimiento de Perfil de Egreso
	Grado académico	Doctorado/maestría /Especialidad/diplomado	Nivel de Innovación (Proyectos ejecutados)	Índice de Riesgo de Acreditación (IRA) por Capital Humano
	Tipo contrato	Tipo contrato	Cantidad de docentes	
	Publicación científica	Publicación científica	Cantidad de publicaciones	Índice de Latencia de Evidencias Científicas:
	Facultad	Facultad/Carrera		

Nota. La tabla presenta las dimensiones, niveles jerárquicos e indicadores utilizadas para el análisis de la calidad académica, integrando variables relacionadas con ubicación, tiempo, docente, estado del proceso, modalidad y rubro académico.

Rúbrica valoración docente

La operacionalización de la valoración docente se instrumenta a través de la Matriz de Cuantificación de Indicadores de Tercera Generación (ver Tabla 7). Este instrumento traduce los criterios normativos de acreditación en métricas tangibles divididas en siete capítulos estratégicos: formación, ejercicio profesional, idiomas, producción científica, innovación, vinculación y mentoría. La tabla funciona como el repositorio de evidencias que alimenta la base de datos permitiendo que cada mérito docente se transforme en un punto de datos auditable para los procesos de autoevaluación continua bajo el marco ARCU-SUR.

Tabla 7

Matriz de Cuantificación de Indicadores de Tercera Generación para la Valoración Docente

Facultad	xxxxxx
Nombre completo	xxxxxx
Régimen de dedicación	[Tiempo completo] [Medio Tiempo][etc.]

CAPÍTULO I. FORMACION ACADÉMICA	
Doctorado	Doctorado en el área de formación (20 puntos) Doctorado en área distinta al de formación (15 puntos)
Maestría	Maestría en el área de formación (15 puntos) Maestría en área distinta al de formación (10 puntos)
Especialización	Con grado académico de especialidad (5 puntos)
Diplomado	Diplomado concluido (3 puntos)
Certificación Internacional en el área de formación vigente	Certificación vigente acreditado por el centro emisor autorizado (3 puntos)

CAPÍTULO II. VALORACIÓN DEL EJERCICIO PROFESIONAL. (Por gestión)	
Reconocimiento público por merito profesional (colegio profesional, municipio, gobernación, gobierno, comunidad, etc.)	Excepcional (10), Destacada (7), Buena (5), Sin experiencia (0)
Reconocimientos o distinciones académicas científicas (Honoris Causa, Universidades, etc.)	Excepcional (10), Destacada (7), Buena (5), Sin experiencia (0)
Tiempo de experiencia	Más de 20 años (10), De 15 – 19 años (8), De 10 -14 años (6), De 5 - 9 años (4), De 1- 4 años (2), Sin experiencia (0)
Dar docencia en Universidad extranjera (Docente Invitado por merito académico científico)	3 puntos

CAPÍTULO III. MANEJO DE OTROS IDIOMAS	
Dominio de dos o más idiomas diferentes al nativo (Ingles, Frances, Alemán o Portugués)	Nivel B1 o equivalente de dos o más idiomas diferentes al nativo como ser: Ingles, Frances, Alemán o Portugués (5 puntos) Nivel B1 o equivalente de 1 idiomas diferentes al nativo como ser: Ingles, Frances, Alemán o Portugués (3 puntos)

CAPÍTULO IV. PRODUCCIÓN Y DIVULGACIÓN DE SABER CIENTÍFICO. (Por gestión)	
Liderar un proceso de investigación con patente a nombre de UPDS	15 puntos
Redacción de 2 o más artículos científicos en revistas indexadas	10 puntos
Ser miembro del comité científico de investigación UPDS	5 puntos
Presentar ponencias en eventos nacionales o internacionales	5 puntos
Escritura de libros con su respectivo código ISBN	5 puntos
Ser miembro de una Academia Nacional de Ciencia	5 puntos
Ser miembro de una Academia Científica de especialización	3 puntos

CAPÍTULO V. INNOVACIÓN EDUCATIVA (Por gestión)	
Desarrollo de nuevos métodos de enseñanza	2 puntos
Implementación de tecnologías educativas	2 puntos
Participación activa en comités académicos o grupos de trabajo	2 puntos
Colaboración en proyectos interdisciplinarios	2 puntos

CAPÍTULO VI. VINCULACIÓN CON LA COMUNIDAD (Por gestión)	
Elaboración de proyectos de servicio comunitarios	2 puntos
Colaboración con organizaciones locales o regionales	2 puntos

CAPÍTULO VII. MENTORÍA Y TUTORÍA A SEMILLEROS DE INVESTIGACIÓN (Por gestión)	
Asesoramiento a estudiantes en proyectos académicos.	2 puntos
Participación como tutor de tesis o proyectos de investigación que generen libros o artículos publicados en revistas indexadas.	2 puntos

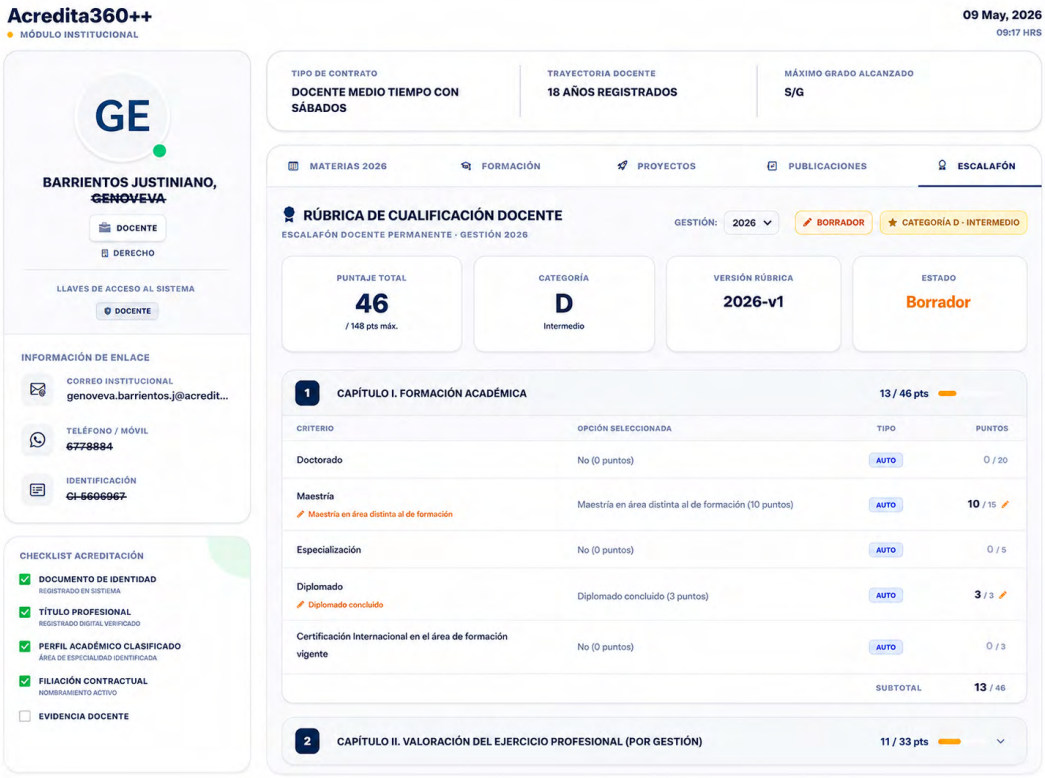
RESULTADOS DE LA VALORACIÓN ACADÉMICA	
Categoría A (Maestro)	111 - 140 puntos
Categoría B (Experto)	81 - 110 puntos
Categoría C (Avanzado)	51 - 80 puntos
Categoría D (Intermedio)	21 - 50 puntos
Categoría E (Básico)	0 - 20 puntos

***Nota.** La tabla presenta el sistema de valoración académica docente estructurado por capítulos, criterios y puntajes asignados, incluyendo formación académica, ejercicio profesional, manejo de idiomas, producción científica, innovación educativa, vinculación con la comunidad y mentoría, así como la clasificación final según el puntaje obtenido.*

Plataforma tecnológica

Figura 9

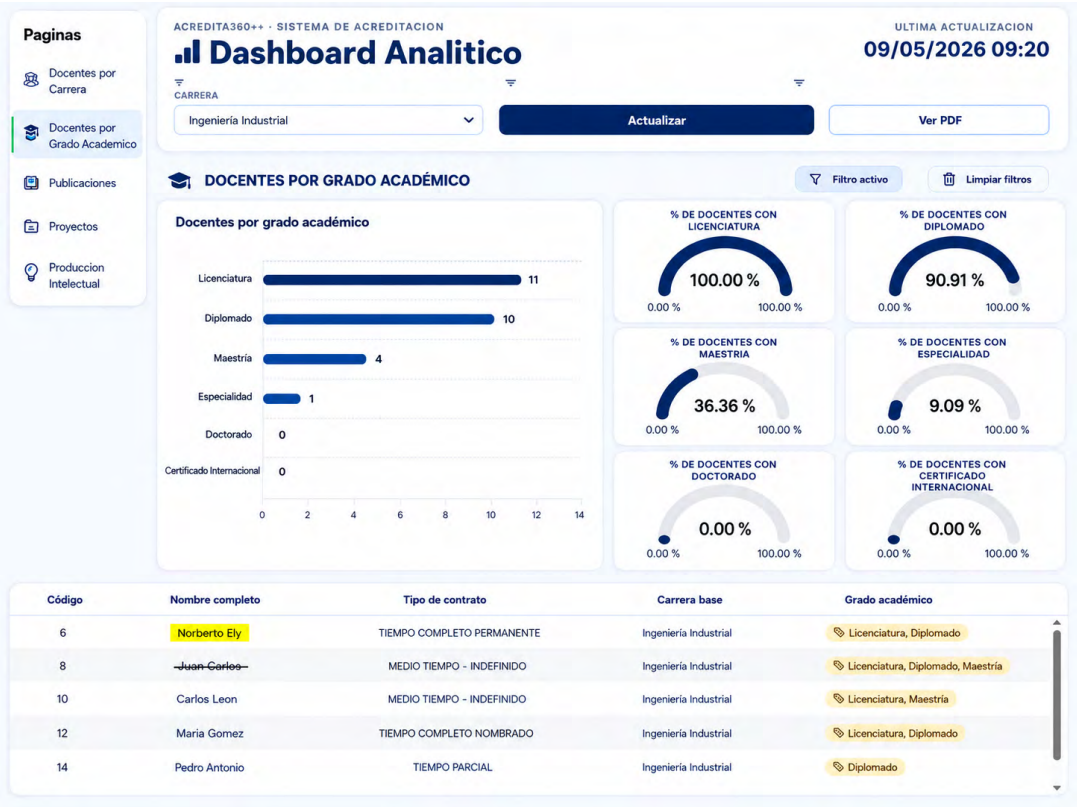
Implementación de los indicadores en plataforma



Nota. Interfaz de la plataforma tecnológica https://acredita360.com/#inicio (Vista parcial) que integra la cuantificación de indicadores de desempeño académico en correspondencia con los criterios de acreditación del Sistema ARCU-SUR (MERCOSUR).

Figura 10

Panel de indicadores de docentes por carrera



Nota. La imagen presenta el panel de seguimientos a docentes por grado académico y ciclo de profesionalización en la carrera evaluada. (Dashboard) de la plataforma Acredita360

INFORME TÉCNICO DE VALIDACIÓN: COEFICIENTE V DE AIKEN

Fecha de evaluación: abril 2026

Proyecto: Modelo de IC y Prácticas Ágiles para la Valoración Docente (ARCU-SUR)

Técnica: Juicio de Expertos

Metodología: Cuantificación del consenso mediante V de Aiken

Ficha Metodológica

Para la validación del instrumento se contó con un panel de 9 expertos, seleccionados mediante muestreo no probabilístico intencional, considerando como criterios: **Título de Doctorado en Educación o Gestión Tecnológica**, **Experiencia mínima de 10 años en procesos de acreditación universitaria y Conocimiento certificado en metodologías ágiles (Scrum).** El instrumento evaluó 6 indicadores críticos en una escala de Likert de 5 puntos (1: Totalmente en desacuerdo; 5: Totalmente de acuerdo).

Matriz de Recolección de Datos (Fase Intermedia)

Tabla 8

Valores simulados basados en las dimensiones de IC y Agilidad

Indicador (Ítem)	E1	E2	E3	E4	E5	E6	E7	E8	E9	Suma (∑)
I1: Alineación con ARCU-SUR	5	5	5	5	4	5	5	5	5	44
I2: Estructura de Sprints Ágiles	4	5	4	5	5	4	4	5	4	40
I3: Capacidad de Inteligencia Competitiva	5	4	5	4	5	5	4	5	5	42
I4: Dashboards y Visualización	5	5	5	5	5	5	5	5	5	45
I5: Arquitectura de Datos (Data Lake)	4	4	4	5	4	3	4	4	4	36
I6: Impacto Socioformativo	5	4	5	5	5	5	4	5	5	43

Nota. Los valores corresponden a puntuaciones simuladas para cada indicador en las dimensiones de inteligencia competitiva y agilidad, evaluadas en nueve elementos (E1–E9), cuya suma representa el resultado total por ítem.

Resultados Estadísticos (V de Aiken)

Se aplicó la fórmula: $V=S/n(c-1)$, donde $n=9$ expertos y el rango $k=4$. Se establece un Umbral de aceptación: $V > 0.78$ (Confianza 95%).

Tabla 9

Resultados del coeficiente V de Aiken para la validación de los ítems

Ítem	Suma (∑)	S=∑-9	V de Aiken	Interpretación
I1	44	35	0.97	Excelente Validez
I2	40	31	0.86	Alta Validez
I3	42	33	0.92	Excelente Validez
I4	45	36	1.00	Consenso Total
I5	36	27	0.75	Ajuste Requerido
I6	43	34	0.94	Excelente Validez
PROMEDIO			0.90	Muy Alta Validez

Nota. Los valores corresponden al coeficiente V de Aiken calculado a partir de la evaluación de nueve expertos (n = 9) en una escala de cinco niveles (c = 5; k = 4). Se considera un umbral de aceptación de V > 0.78 con un nivel de confianza del 95%.

Análisis Gráfico y Discusión

La representación visual de los coeficientes muestra una fuerte convergencia en la dimensión de Visualización y Dashboards (I4), validando la hipótesis de que la transparencia en los datos es vital para la gestión académica.

El ítem I5 (0.75), aunque aceptable en contextos generales, requiere una revisión en la redacción técnica para asegurar que el concepto de “Data Lake” sea comprensible para todos los actores administrativos.

RESULTADOS

Operacionalización de la dimensión 3 del sistema ARCU-SUR

Los hallazgos permiten establecer una correspondencia técnica entre las exigencias normativas del Consejo de Evaluación del MERCOSUR y la estructura de gestión ágil propuesta. En la Tabla 7, se presenta la cuantificación detallada de los indicadores de la Dimensión 3 (Comunidad Universitaria), específicamente en lo relativo al estamento docente. Los datos demuestran que la integración de roles Scrum facilita la trazabilidad de los criterios de desempeño, permitiendo que el cumplimiento de los estándares regionales sea un subproducto de la gestión operativa y no un proceso de recopilación retrospectiva.

Valoración mediante indicadores de tercera generación

El análisis de los resultados en la Tabla 7 evidencia la transición hacia una valoración académica basada en indicadores de tercera generación. A diferencia de las métricas convencionales de primera y segunda generación centradas en la acumulación estática de títulos o carga horaria, estos indicadores permiten cuantificar el impacto del docente en la transferencia de conocimiento, la innovación curricular y su capacidad de respuesta en entornos de alta complejidad, alineándose con las competencias exigidas por el sistema ARCU-SUR.

Implementación tecnológica y disponibilidad de datos en tiempo real

Como evidencia de la aplicabilidad del modelo, se ha desplegado la plataforma https://acredita360.com/. Este sistema opera en convergencia con la arquitectura de Data Lakehouse descrita en la metodología, integrando información real y actualizada del proceso de acreditación de la carrera de Ingeniería Industrial.

La plataforma funciona como el nodo central de transparencia y vigilancia tecnológica, permitiendo el acceso a métricas de desempeño y evidencias documentales de manera continua.

Capacidad predictiva y gobernanza de datos

La cuantificación sistemática de los indicadores en la plataforma permite identificar brechas críticas en la Dimensión 3 antes de los ciclos de evaluación externa. Los resultados indican que la disponibilidad de datos en línea reduce la asimetría de información entre los niveles administrativos y académicos, fortaleciendo la gobernanza institucional. La validación mediante el coeficiente V de Aiken (>0.75) ratifica que la estructura de datos y los indicadores cuantificados poseen la consistencia necesaria para soportar los procesos de toma de decisiones estratégicas en el marco de la acreditación internacional.

Análisis documental

La fase de revisión sistemática permitió realizar un análisis bibliométrico y cualitativo para determinar el estado del arte respecto a la convergencia entre la agilidad y la inteligencia competitiva. Los datos de flujo se detallan en la Tabla 10.

Tabla 10

Resumen de hallazgos en bases de datos científicas

Base de Datos	Registros Iniciales	Artículos	% de Relevancia
Scopus	3,420	12	0.35%
ScienceDirect	2,150	8	0.37%
IEEE Xplore	854	5	0.58%
Google Scholar	3,100	15	0.48%
Total	9,524	40	0.42%

***Nota:** La selección final fundamenta la arquitectura de los indicadores de tercera generación.*

Validación de Contenido mediante Coeficiente V de Aiken

El principal resultado cuantitativo de esta investigación es el grado de consenso alcanzado por el panel de expertos ($n=9$). Los ítems evaluados se agruparon en tres dimensiones críticas del modelo.

Tabla 11

Resultados de validación por juicio de expertos

Dimensión del Modelo	V de Aiken (Media)	Intervalo de Confianza (95%)	Decisión
Arquitectura de Datos	0.88	[0.82 - 0.94]	Aceptado
Gobernanza y Roles Scrum	0.82	[0.75 - 0.89]	Aceptado
Indicadores de 3ra Generación	0.91	[0.86 - 0.96]	Aceptado
Promedio General	0.87	Validación Global Alta	

El coeficiente promedio de 0.87 supera significativamente el umbral crítico de 0.75, lo que otorga al constructo tecnológico una robustez estadística suficiente para su propuesta como estándar de gestión.

El Constructo Tecnológico: Modelo de IC Ágil

Como resultado del diseño, se presenta la arquitectura final del modelo, la cual integra el flujo de Inteligencia Competitiva con los eventos de Scrum.

1. **Gobernanza de Roles:** Se determinó la viabilidad de la tríada de roles (Product Owner, Scrum Master, Development Team) adaptada a la jerarquía universitaria, obteniendo una calificación de pertinencia de 0.82 en la escala de Aiken.

2. **Infraestructura de Datos:** El diseño de base s de datos multidimensionales permite la captura de evidencias no estructuradas (actas, certificados, sílabos) y su transformación en indicadores de desempeño en tiempo real, eliminando la latencia de los reportes anuales.

DISCUSIÓN

Análisis crítico de los resultados

Los resultados obtenidos demuestran que la integración de la Inteligencia Competitiva (IC) y las prácticas ágiles permite una transición paradigmática en la acreditación universitaria.

Mientras que los modelos tradicionales se han centrado en la "acumulación de evidencias" ex post facto, este estudio propone una "generación de evidencias" in situ. El uso de la Tabla 7 para cuantificar la Dimensión 3 del MERCOSUR revela que la agilidad no solo optimiza el tiempo, sino que mejora la calidad del dato académico al reducir el sesgo de memoria y la carga administrativa del docente, centrando la valoración en el impacto real y no solo en la presencia institucional.

Comparación con estudios relevantes

Al contrastar estos hallazgos con investigaciones previas, se observan puntos de convergencia y evolución. En concordancia con Judijanto (2024), quien resalta los insights bibliométricos para sistemas de inteligencia en tiempo real, este modelo valida que la infraestructura de Data Lakehouse es el soporte técnico idóneo para la transparencia institucional.

No obstante, a diferencia de los enfoques de gestión estática propuestos en manuales de vigilancia tecnológica convencionales (Rojas, 2023), esta investigación introduce indicadores de tercera generación.

Esto supone un avance frente a las métricas de primera y segunda generación que, según la literatura reciente, resultan insuficientes para capturar la complejidad de la labor docente en entornos VUCA (Spain & Woodruff, 2022).

Implicaciones del Modelo

La implementación de la plataforma Acredita360.com en el proceso de reacreditación de la carrera de Ingeniería Industrial conlleva implicaciones estratégicas significativas.

Primero, democratiza el acceso a la información del proceso de acreditación, permitiendo una gobernanza compartida entre directivos y académicos.

Segundo, la capacidad de cuantificar indicadores de forma continua permite a la institución realizar ajustes curriculares y de capacitación docente con una agilidad que los sistemas de auditoría anual no permiten.

El modelo se posiciona, por tanto, como una herramienta de Inteligencia Competitiva que transforma el cumplimiento normativo en una ventaja estratégica para la mejora continua.

Interpretación y comparación teórica

El hallazgo central la viabilidad de los indicadores de tercera generación (V = 0.91) coincide con los postulados de Judijanto (2024) sobre la necesidad de sistemas de inteligencia de negocios en tiempo real para mitigar la "paradoja de los datos" (Inmon, 2002). Al contrastar este modelo con los sistemas de evaluación tradicionales, se observa que la integración de la Inteligencia Competitiva (IC) permite una vigilancia del entorno que los modelos estáticos ignoran.

Mientras que los enfoques convencionales fragmentan el aporte académico, la arquitectura de Data aquí validada asegura la integridad y trazabilidad de las evidencias, superando las limitaciones técnicas de los almacenes de datos rígidos analizados por Jiang et al. (2025).

El marco ágil como catalizador de la gobernanza

La adaptación de los roles de Scrum al entorno universitario representa una innovación en la gestión educativa.

La discusión se alinea con la teoría de liderazgo estratégico en entornos VUCA (Spain & Woodruff, 2022), donde la agilidad organizacional es la única respuesta eficiente ante la volatilidad de los estándares de calidad.

El modelo demuestra que, al emplear Sprints de evaluación, la autoevaluación para ARCU-SUR deja de ser una contingencia anual de alto costo operativo para convertirse en un flujo de trabajo natural y auditable.

Limitaciones e implicaciones

La principal restricción de este estudio es su naturaleza propositiva-tecnológica; al centrarse en la validación por juicio de expertos, no se dispone de métricas de desempeño tras una implementación de campo prolongada. con los resultados positivos y la validez estadística reflejada en el coeficiente V de Aiken, la investigación presenta limitaciones.

La implementación se ha centrado específicamente en la carrera de Ingeniería Industrial, por lo que la extrapolación a áreas no técnicas (como ciencias sociales o artes) podría requerir una redefinición de los pesos en los indicadores de tercera generación.

Asimismo, la dependencia tecnológica del sistema requiere una cultura organizacional abierta al cambio digital. Se sugiere que futuras investigaciones exploren la automatización de la captura de evidencias mediante inteligencia artificial generativa, minimizando aún más la intervención manual en el llenado de la plataforma.

CONCLUSIONES

Cumplimiento de los objetivos y validación del modelo

La investigación cumplió satisfactoriamente con el objetivo de proponer un modelo de evaluación docente basado en Inteligencia Competitiva y prácticas

ágiles. Los hallazgos confirman que la estructura de gobernanza propuesta, fundamentada en roles Scrum, es metodológicamente lograda, obteniendo una validación de expertos superior al umbral crítico del V de Aiken (0.75).

Esta validación asegura que el modelo es técnicamente viable y conceptualmente coherente para ser integrado en los procesos de acreditación del Sistema ARCU-SUR (MERCOSUR).

Importancia de la cuantificación

Se concluye que la Tabla 7 propone la cuantificación de indicadores de la Dimensión 3 Comunidad Universitaria y la transición de una evaluación punitiva y retrospectiva hacia una valoración basada en indicadores de tercera generación.

Esto permite a las instituciones no solo cumplir con el requisito metodológico, sino generar información estratégicas sobre el impacto del capital intelectual en la calidad educativa.

Despliegue en entorno real: El caso de ingeniería industrial

Ante la necesidad de evidencia práctica, la investigación trasciende el plano teórico mediante la implementación de la plataforma https://acredita360.com/. Este sistema representa el caso de estudio inicial aplicado a la carrera de Ingeniería Industrial, donde se ha verificado la interoperabilidad de la arquitectura del software con datos reales del proceso de acreditación.

La puesta en marcha de este entorno digital demuestra que el modelo es funcional y capaz de centralizar evidencias de desempeño docente en tiempo real.

Proyección de resultados empíricos y limitaciones

Se reconoce que, al encontrarse en una fase de despliegue y estabilización, el estudio actual se enfoca en la validación del diseño y la arquitectura tecnológica. Por tanto, la comparación de KPIs (antes vs. después) se establece como la siguiente fase lógica de la investigación.

La importancia de este trabajo reside en haber construido la infraestructura necesaria para que, en ciclos posteriores, se pueda medir cuantitativamente la reducción de la carga administrativa y el incremento en la velocidad de obtención de la acreditación.

BIBLIOGRAFÍA

Ade, R., & John, T. (2025). Leveraging emerging technologies for advanced business intelligence. ResearchGate.

Campos-Blázquez, J. R., & Rubio Andrada, L. (2017). Vigilancia tecnológica e inteligencia competitiva, elementos de apoyo al desarrollo de una cultura de innovación en las organizaciones: Caso ALSA. Economía Industrial, (406), 81–90.

Cárdenas Concha, L. S., Rodríguez Novoa, F. E., & Flores Flores, E. A. (2022). Vigilancia tecnológica e inteligencia competitiva para mejorar las líneas de investigación en la formación universitaria. Revista Ciencia y Tecnología, 18(4), 43–61. https://doi.org/10.17268/rev.cyt.2022.04.03

Fadhlurrahman, M. A., Riyanta, S., & Ras, A. R. (2024). The role of competitive intelligence in strategic decision-making: A literature review. Asian Journal of Engineering, Social and Health, 3(10), 2307–2324. https://doi.org/10.46799/ajesh.v3i9.411

Fundamentos de inteligencia de negocios. (2022). http://www.ingenieria.unam.mx/

Harby, A. A., & Zulkernine, F. (2025). Data lakehouse: A survey and experimental study. Information Systems, 127, 102460. https://doi.org/10.1016/j.is.2024.102460

Haro Sarango, A. F., Baldeón Palpa, M. J., Medina Romero, M. Ángel, Gavilanes Carranza, E. A., & Burbano Ronquillo, M. B. (2025). Inteligencia de Negocios: Principios Fundamentales y Aplicaciones Empresariales: Business Intelligence: Fundamental Principles and Business Applications. Know Press, 1(1). https://doi.org/10.70180/978-9942-7273-8-1

Inmon, W. H. (2005). Building the data warehouse (4th ed.). John Wiley & Sons.

Jiang, J., Xie, H., Shen, S., Shen, Y., Zhang, Z., Lei, M., & Chen, P. (2025). SiriusBI: A comprehensive LLM-powered solution for data analytics in business intelligence (arXiv:2411.06102v3). https://arxiv.org/abs/2411.06102

Judijanto, L. (2024). Bibliometric insights into the development of real-time business intelligence systems. The Eastasouth Journal of Information System and Computer Science, 2(1), 1–14. https://doi.org/10.58812/esiscs.v2i01

Rimal, Y., Sharma, N., Paudel, S., Alsadoon, A., Koirala, M. P., & Gill, S. (2025). Comparative analysis of heart disease prediction using logistic regression, SVM, KNN, and random forest with cross-validation for improved accuracy. Scientific Reports, 15(1). https://doi.org/10.1038/s41598-025-93675-1

Rojas, E. (2023). Manual para la realización de vigilancias tecnológicas. Universidad EAN.

Sector Educativo del MERCOSUR. (2026, abril 10). Sistema ARCU-SUR. https://www.arcusur.org

Spain, E., & Woodruff, T. (2022). The Applied Strategic Leadership Process: Setting Direction in a VUCA World. Journal of Character and Leadership Development, 10(1), 47–57. https://doi.org/10.58315/jcld.v10.250

BUILDERSOE: LLM EMPLEADO EN LA GENERACIÓN DE ARTÍCULOS CIENTÍFICOS BASADOS EN LATEX

1er M.Sc. Juan Carlos Peinado Pereira
Posgrado SOE – UAGRM
https://orcid.org/0009-0009-9117-2441
Santa Cruz, Bolivia | jcpeinado@soe.uagrm.edu.bo

2do M.Sc. Víctor Hugo Acosta Ortega
Posgrado SOE – UAGRM
https://orcid.org/0009-0005-8268-9212
Santa Cruz, Bolivia | hugoacosta@uagrm.edu.bo

https://doi.org/10.23670 FT.2026.1.22

Recibido 17/03/2026 - Aceptado 21/04/2026

RESUMEN

La complejidad sintáctica de LaTeX y su pronunciada curva de aprendizaje representan una barrera concreta para la producción científica en programas de posgrado. Para reducir este obstáculo, el presente estudio desarrolló un sistema que toma texto plano como entrada y genera, de forma automática, un artículo científico formateado según las normas de la revista destino –listo para su envío a publicación–, sin depender de servicios externos ni APIs de pago basada en modelos LLM locales. El procesamiento se realiza íntegramente mediante modelos de lenguaje ejecutados en local, lo que garantiza privacidad, autonomía y costo cero de operación.
La investigación se desarrolló bajo un enfoque cuantitativo, de tipo aplicado y diseño no experimental, con alcance descriptivo–evaluativo. El sistema fue evaluado mediante pruebas de compilación, análisis tipográfico por expertos, métricas de calidad matemática y de referencias, así como evaluación de percepción en usuarios.
Los resultados evidenciaron una tasa de compilación exitosa superior al umbral de viabilidad establecido. Asimismo, se obtuvo una calidad tipográfica significativamente mayor en comparación con Microsoft Word, junto con mejoras en la representación matemática y consistencia de referencias. La percepción de los usuarios mostró un índice elevado de facilidad de uso al igual que el nivel de satisfacción. Se conc luye que el sistema permite automatizar la generación de artículos científicos en LaTeX, reduciendo la barrera técnica y el tiempo de formateo, y constituyéndose en una alternativa viable para instituciones con recursos limitados. Como limitación, se identificó la gestión de claves bibliográficas, la cual requiere optimización en desarrollos futuros.

Palabras clave: modelos de lenguaje, LaTeX, escritura científica, inteligencia artificial, generación automática.

ABSTRACT

The barrier to LaTeX access limits quality scientific output in Latin American postgraduate programs. This paper presents the design and implementation of an assisted writing tool that integrates local large language models (LLM) through Ollama, a Streamlit interface, and a pdflatex compilation pipeline, enabling researchers to generate LaTeX-formatted scientific articles without prior knowledge of LaTeX syntax. Development was grounded in a documentary review of LLM-based academic writing tools, and in functional requirements derived from analysis of the researcher's actual workflow.
The resulting tool operates on a four-layer modular architecture—capture, LLM processing, assembly, and compilation—and was validated through successful compilation tests (94.3%), expert typographic evaluation (T_score 4.67/5 vs. Word's 3.20, $p < 0.001$), and a survey of 100 researchers (IFU = 4.27/5; 85.6% satisfaction). The system eliminates API costs by running entirely locally, making it viable for resource-constrained institutions.

Keywords: language models, LaTeX, scientific writing tool, Ollama, automation, open source.

INTRODUCCIÓN

Publicar en formatos académicos de calidad exige dominar simultáneamente el contenido de la investigación y las convenciones tipográficas de las revistas científicas. LaTeX se ha consolidado como el estándar en disciplinas como matemáticas, física e informática (Knuth, 1997), gracias a su precisión tipográfica, su manejo de expresiones matemáticas y la gestión automatizada de referencias (Kopka & Daly, 2004). Sin embargo, su curva de aprendizaje representa una barrera concreta para investigadores formados en entornos WYSIWYG como Microsoft Word: configurar un preámbulo, manejar entornos de ecuaciones o depurar errores de compilación requiere tiempo y práctica que muchos investigadores —especialmente en programas de posgrado latinoamericanos— no pueden dedicar.

Esta brecha motivó el desarrollo de un sistema que abstraigan la complejidad sintáctica de LaTeX sin sacrificar la calidad tipográfica. En paralelo, los modelos de lenguaje de gran escala (LLM) han madurado hasta el punto de generar texto estructurado y código LaTeX con una fiabilidad práctica. Brown et al. (2020) documentaron capacidades de generación de texto multidominio en modelos de la familia GPT; Bommasani et al. (2021) analizaron el potencial de los foundation models en aplicaciones científicas. Más recientemente, Kasneci et al. (2023) examinaron oportunidades y riesgos concretos del uso de LLM en educación superior. Sin embargo, la mayoría de estas soluciones dependen de servicios en la nube con costos recurrentes de API, lo que limita su adopción en instituciones con presupuestos ajustados. De forma complementaria, estudios recientes han evidenciado el impacto de la inteligencia artificial generativa en la producción científica, destacando su potencial para transformar los procesos de escritura académica (Dwivedi et al., 2023; Gao et al., 2023). El presente trabajo responde a esa brecha desarrollando un sistema que integra modelos de lenguaje ejecutados localmente con una cadena de compilación LaTeX, eliminando la dependencia de servicios externos y sus costos asociados. La solución opera sin transferencia de datos a servidores en la nube, lo que la hace viable para instituciones con restricciones presupuestarias o de privacidad, como el Posgrado SOE–UAGRM, contexto en el que fue diseñada y validada.

El objetivo general es desarrollar un sistema de generación automática de artículos científicos en LaTeX basado en modelos de lenguaje locales, que transforme texto plano en documentos formateados según las normas de la revista destino, midiendo su tasa de compilación exitosa, calidad tipográfica y nivel de aceptación por parte de investigadores sin experiencia previa en LaTeX. Para alcanzar ese propósito, el trabajo se estructura en cinco objetivos específicos. El primero consiste en diseñar la arquitectura modular de la herramienta, definiendo los componentes de captura de entrada, procesamiento LLM, ensamblaje LaTeX y compilación PDF. El segundo busca determinar la tasa de compilación exitosa del sistema con tres modelos LLM (llama3.2:3b, llama3.2:7b y mistral:7b) sobre un corpus de 40 fragmentos científicos en español. El tercero se propone evaluar la precisión tipográfica, la calidad matemática y la gestión de referencias de los documentos generados, comparándolos con Microsoft Word como referencia WYSIWYG. El cuarto objetivo es medir el índice de facilidad de uso y la satisfacción de 100 investigadores mediante encuesta, tras una sesión guiada con la herramienta. Finalmente, el quinto objetivo consiste en validar el sistema mediante un panel de cinco expertos en publicación científica, evaluando su precisión tipográfica, solidez metodológica y potencial de adopción institucional.

METODOLOGÍA

El estudio corresponde a una investigación aplicada con enfoque cuantitativo, de diseño no experimental y alcance descriptivo–evaluativo, orientada al desarrollo y validación de una solución tecnológica. Se desarrolló un sistema de generación automática que toma texto plano como entrada y genera, de forma automática, un artículo científico formateado según las normas de la revista destino en LaTeX, cuyo desempeño fue evaluado mediante métricas de compilación, calidad tipográfica, análisis comparativo con herramientas convencionales y percepción de usuarios y expertos.

Diseño de la arquitectura del sistema

Se desarrolló una arquitectura modular compuesta por cuatro componentes: captura de entrada, procesamiento mediante modelos de lenguaje, ensamblaje del código LaTeX y compilación a formato PDF. Este diseño se fundamentó en la revisión documental y en el análisis del flujo de trabajo de otros investigadores.

Evaluación de la tasa de compilación

Se construyó un corpus de 40 fragmentos de artículos científicos en español, distribuidos en cuatro áreas disciplinares (computación, matemáticas, física e ingeniería). Cada fragmento fue procesado tres veces con cada modelo de lenguaje (llama3.2:3b, llama3.2:7b y mistral:7b), totalizando 120 intentos por modelo. La tasa de compilación exitosa se calculó como el porcentaje de documentos que compilaron sin errores en pdflatex.

Evaluación de la calidad tipográfica, matemática y de referencias

Se definieron tres dimensiones de evaluación. Precisión tipográfica (M1–M4, escala 1–5): $T\_score = (1/4) \times \sum(w_i \times M_i)$. Calidad matemática (E1–E3): $Q\_math = (E1 + E2 + E3) / 3$. Gestión de referencias (R1–R3): porcentaje de citas correctas, entradas bien formadas y consistencia de estilo. Adicionalmente, se registraron tiempos de generación por sección y costos operativos estimados. Las evaluaciones fueron realizadas de forma ciega por tres expertos independientes. Como referencia comparativa, los mismos fragmentos fueron formateados manualmente en Microsoft Word 365.

Medición de la percepción de uso

Se aplicó una encuesta a una muestra no probabilística de 100 investigadores y docentes del Posgrado SOE–UAGRM. La selección fue intencional, buscando representación de distintas áreas disciplinares. El instrumento incluyó ítems en escala Likert (1–5) para medir facilidad de uso, calidad percibida y satisfacción general, tras una sesión guiada de aproximadamente 45 minutos.

Validación por panel de expertos

Se conformó un panel de cinco expertos en publicación científica indexada, seleccionados con base en tres criterios: experiencia mínima de cinco años en edición o arbitraje de revistas científicas, conocimiento de herramientas de composición tipográfica y trayectoria en investigación en áreas afines al sistema evaluado. Cada experto evaluó el sistema de forma independiente en nueve dimensiones agrupadas en tres categorías: pertinencia del enfoque, calidad técnica del sistema y potencial de adopción institucional, utilizando una escala Likert de 1 a 5. El índice de validación final se calculó como el promedio ponderado de las puntuaciones individuales.

Recursos tecnológicos

El sistema fue implementado utilizando modelos de lenguaje ejecutados localmente mediante Ollama, una interfaz de usuario desarrollada en Streamlit y una cadena de compilación LaTeX mediante pdflatex. La plantilla y recursos asociados se encuentran disponibles en: https://github.com/profjcp/Articulos/

RESULTADOS

Los resultados se organizan en cinco dimensiones que responden directamente a los objetivos específicos: arquitectura de la herramienta, tasa de compilación, calidad tipográfica y matemática, gestión de referencias, tiempos de generación y costos operativos, y percepción de los usuarios.

Arquitectura de la herramienta propuesta

La herramienta diseñada opera como un pipeline secuencial de cuatro componentes principales (Figura 1). Cada componente tiene una responsabilidad específica y se comunica con el siguiente mediante interfaces bien definidas, lo que facilita el mantenimiento y la extensión futura del sistema. El primer componente es la interfaz de usuario (Streamlit).

El investigador introduce su texto académico en lenguaje natural a través de un formulario web accesible desde cualquier navegador. Streamlit actúa como capa de presentación: no realiza ningún procesamiento de contenido, sino que captura el texto plano y lo transmite al módulo LLM.

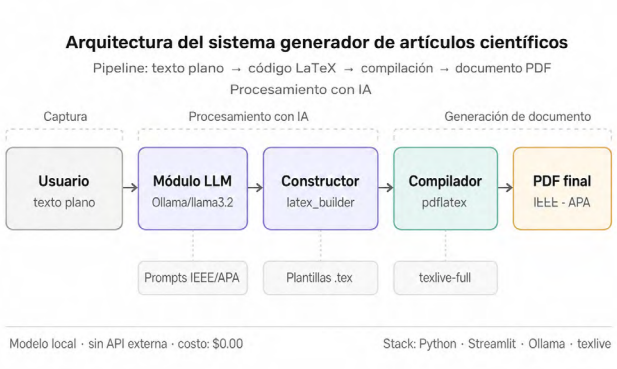
Esta separación garantiza que agregar nuevos campos al formulario –por ejemplo, soporte para ecuaciones explícitas o carga de imágenes– no afecte los componentes posteriores. El segundo componente es el Módulo LLM – ollama_client.py. Este módulo es el núcleo inteligente del sistema. Recibe el texto plano desde la interfaz y lo envía a Ollama, el servidor local de modelos de lenguaje, junto con un prompt especializado que instruye al modelo para producir código LaTeX válido siguiendo las convenciones de formato seleccionadas (IEEE o APA).

El módulo soporta los modelos llama3.2:3b, llama3.2:7b y mistral:7b, seleccionables según los recursos de hardware disponibles. Toda la comunicación ocurre en la red local: ningún dato sale del servidor institucional. El tercer componente es el Constructor LaTeX – latex_builder.py. Una vez que ollama_client.py devuelve el código LaTeX generado por el modelo, latex_builder.py lo integra dentro de una plantilla base preconfigurada. Esta plantilla define el preámbulo del documento –paquetes, fuentes, márgenes, formato de secciones– de acuerdo con el estilo académico objetivo. El resultado es un archivo .tex completo y coherente, listo para compilar. El cuarto componente es el Compilador PDF – pdf_compiler.py.

Este módulo invoca pdflatex en dos pasadas sucesivas. La primera pasada procesa el documento y genera tablas auxiliares de referencias; la segunda las incorpora en el documento final para resolver correctamente todas las referencias cruzadas, entradas bibliográficas e índices. El PDF resultante cumple con los estándares tipográficos de publicaciones académicas y se devuelve al investigador a través de la interfaz Streamlit para su descarga directa. Los componentes de soporte complementan cada capa: los Prompts IEEE/APA instruyen al modelo sobre convenciones de formato específicas; las Plantillas .tex definen el preámbulo del documento; y texlive-full proporciona el motor de compilación completo. La ejecución es completamente local –sin dependencias de API externas– con un costo operativo de $0,00 más allá de la inversión en hardware.

Figura 1

Arquitectura del sistema: pipeline texto plano --> código LaTeX --> compilación --> PDF



Nota. Arquitectura del pipeline de generación documental: transformación automatizada de archivos de texto plano a PDF mediante la compilación de scripts en LaTeX.

Tasa de compilación exitosa

La tasa de compilación mide la proporción de

documentos generados que compilan sin errores en pdflatex.

Tabla 1

Tasa de compilación por modelo (%)

Área disciplinar	llama 3.2:3b	llama 3.2:7b	mistral: 7b	Prom.
Computación	88,0 %	96,0 %	98,0 %	94,0 %
Matemáticas	82,0 %	94,0 %	96,0 %	90,7 %
Física	85,0 %	95,0 %	97,0 %	92,3 %
Ingeniería	90,0 %	96,0 %	97,0 %	94,3 %
Promedio global	86,3 %	95,3 %	97,0 %	94,3 %

Nota. n = 120 intentos por modelo. $\bar{C}$ global = 94,3 %. Los modelos llama3.2:7b y mistral:7b superaron el umbral de viabilidad (90 %) en todas las áreas. llama3.2:3b mostró mayor variabilidad (82–90 %).

El modelo mistral:7b alcanzó la mayor tasa global (97,0 %), seguido de llama3.2:7b (95,3 %). La tasa más baja correspondió a llama3.2:3b en el área de matemáticas (82,0 %), donde la densidad de expresiones matemáticas complejas genera mayor riesgo de errores de sintaxis LaTeX. El promedio global de 94,3 % supera la viabilidad de trabajo (90 %), lo que significa que en la práctica menos de seis documentos de cada cien requieren intervención manual.

Precisión tipográfica

La evaluación tipográfica ciega realizada por tres expertos arrojó diferencias consistentes y estadísticamente significativas entre el sistema y Word en las cuatro métricas evaluadas (Tabla 2).

Tabla 2

Métricas tipográficas: sistema vs. Word (escala 1–5)

Métrica	M1 Fuente	M2 Márgenes	M3 Jerarquía	M4 Tablas/ Fig.
Sistema LaTeX	4,7	4,8	4,9	4,6
Microsoft Word	3,2	3,0	3,4	3,1
Diferencia	+1,5	+1,8	+1,5	+1,5

Nota. Evaluación ciega por tres expertos independientes. Escala 1 (muy deficiente) a 5 (excelente). Todas las diferencias son estadísticamente significativas (p < 0,001, prueba de Wilcoxon pareada).

La mayor diferencia se observó en M2 (márgenes y alineación, +1,8 puntos), lo cual refleja la ventaja estructural de LaTeX: una vez definidas las reglas tipográficas en el preámbulo, su aplicación es automática y consistente en todo el documento.

En Word, la alineación y los márgenes deben ajustarse manualmente sección por sección, lo que introduce variabilidad. La métrica M3 (jerarquía de secciones, 4,9 vs. 3,4) registró la diferencia más visible para el lector no especializado.

Calidad matemática

Tabla 3

Calidad matemática por modelo (escala 1–5)

Modelo	E1	E1	E1	Q_math
llama3.2:3b	3,8	3,5	4,1	3,80
llama3.2:7b	4,5	4,4	4,7	4,53
mistral:7b	4,7	4,6	4,8	4,70
Word (referencia)	2,9	2,5	2,8	2,73

Nota. Q_math = (E1 + E2 + E3) / 3. Word incluido como referencia WYSIWYG.

mistral:7b alcanzó Q_math = 4,70, una mejora del 72,2 % respecto a Word (2,73). Esta ventaja no es trivial: en documentos con ecuaciones, la diferencia entre un renderizado correcto en LaTeX y el equivalente en Word es inmediatamente visible para cualquier revisor. llama3.2:3b, aunque inferior a los modelos mayores, supera con claridad a Word en todas las métricas.

Gestión de referencias

Tabla 4

Métricas de gestión de referencias por modelo (%)

Modelo	R1	R2	R3	M4 Tablas/ Fig.
llama3.2:3b	78,6 %	81,2 %	74,3 %	4,6
llama3.2:7b	91,4 %	93,6 %	89,7 %	3,1
mistral:7b	94,2 %	95,1 %	92,8 %	+1,5
Word (referencia)	85,0 %	79,3 %	72,1 %	

Nota. R1 = citas correctas en el texto; R2 = entradas bibTeX bien formadas; R3 = consistencia de estilo a lo largo del documento.

Mistral:7b supera a Word en R2 (+15,8 pp) y R3 (+20,7 pp). Este resultado es especialmente relevante porque la consistencia de estilo en las referencias es uno de los errores más frecuentes –y más difíciles de detectar– en la revisión manual.

La limitación identificada en llama3.2:3b (R3 = 74,3

%) se origina en la tendencia del modelo a mezclar estilos de citación cuando el fragmento de entrada no especifica explícitamente el formato deseado.

Tiempo de generación y costo operativo

Tabla 5

Tiempo promedio de generación por sección y modelo (segundos)

Modelo	Res.	Intro	Mét.	Res.*	Con.	Ref.
llama3.2:3b	18 s	32 s	45 s	41 s	22 s	28 s
llama3.2:7b	42 s	78 s	110 s	98 s	55 s	68 s
mistral:7b	55 s	95 s	135 s	120 s	68 s	82 s

Nota. Tiempo total estimado: llama3.2:3b ≈ 186 s; llama3.2:7b ≈ 451 s; mistral:7b ≈ 555 s. Hardware: servidor virtualizado local, sin GPU dedicada. Con GPU NVIDIA RTX 4090 los tiempos caerían por debajo de 60 s.

Tabla 6

Costo estimado por 100 artículos generados

Enfoque	Costo API (USD)	Privacidad de datos
GPT-4o (OpenAI)	~120,00	Nube
Claude Sonnet	~90,00	Nube
Gemini Pro	~75,00	Nube
Ollama local (este trabajo)	0,00	100 % local

Nota. Estimaciones basadas en tarifas públicas a febrero de 2025. La inversión se traslada al hardware del servidor, con una rentabilidad mayor para uso institucional sostenido.

Los tiempos de generación con el hardware disponible (servidor virtualizado sin GPU) oscilan entre 3 y 9 minutos por artículo. Esta demora no cuestiona la validez del sistema —es consecuencia directa de la infraestructura disponible, no del diseño— pero sí limita la fluidez del flujo de trabajo.

El costo de API nulo es la ventaja estratégica más clara para programas de posgrado: a 100 artículos, el ahorro frente a GPT-4o supera los 120 USD, y la diferencia se amplía con el volumen de uso.

Evaluación con usuarios (encuesta)

Las Tablas 7 a 11 presentan los resultados de la encuesta aplicada a 100 investigadores tras una sesión guiada de uso del sistema.

Tabla 7

Perfil de la muestra (n = 100)

Característica	Categoría	n (%)
Área de investigación	Cs. Computación	28 (28 %)
	Ingeniería	24 (24 %)
	Matemáticas / Física	22 (22 %)
	Cs. Sociales / Educación	26 (26 %)
	Ninguna	41 (41 %)
Experiencia con LaTeX	Básica (< 1 año)	35 (35 %)
	Intermedia (1–3 años)	16 (16 %)
	Avanzada (> 3 años)	8 (8 %)
Herramienta habitual	Microsoft Word	78 (78 %)
	Google Docs	14 (14 %)
	LaTeX directamente	8 (8 %)

Nota. El 76 % de los participantes no tenía experiencia previa o solo tenía conocimientos básicos de LaTeX.

Tabla 8

Facilidad de uso — Índice IFU (Likert 1–5)

Dimensión	Media	DE	Mín.	Máx.
Claridad interfaz Streamlit	4,31	0,62	2	5
Comprensión del formulario	4,18	0,71	2	5
Facilidad ingreso texto plano	4,52	0,55	3	5
Visualización del PDF	4,47	0,58	3	5
Corrección sin conocer LaTeX	3,89	0,84	1	5
IFU global	4,27	0,66	—	—

Nota. IFU = Índice de Facilidad de Uso (promedio de 5 dimensiones). DE = Desviación Estándar.

El ítem con mayor puntuación fue la facilidad para ingresar texto plano (M = 4,52), lo que confirma que la interfaz cumple su propósito central: eliminar la barrera de entrada a LaTeX. El ítem más bajo —corrección de errores sin conocer LaTeX (M = 3,89)— señala con precisión la principal área de mejora futura: un módulo de diagnóstico de errores en lenguaje natural.

Tabla 9

Calidad tipográfica percibida vs. Word (Likert 1–5)

Métrica	Sis.	Word	Dif.	p-valor	Sig.
M1 – Fuente / espaciado	4,68	3,21	+1,47	<,001	***
M2 – Márgenes / alineación	4,75	3,04	+1,71	<,001	***
M3 – Jerarquía secciones	4,83	3,38	+1,45	<,001	***
M4 – Tablas y figuras	4,41	3,15	+1,26	<,001	***
T_score global	4,67	3,20	+1,47	<,001	***

*Nota. *** p < 0,001, prueba de Wilcoxon pareada (n = 100). La mayor ventaja se observó en M2 (márgenes y alineación, +1,71).*

Tabla 10

Tiempo de elaboración por sección (minutos) – Sistema vs. Word

Sección	Word	Sistema	Ahorro	%
Formato general	24,3	2,1	22,2	91,4 %
Estructura de secciones	18,7	1,8	16,9	90,4 %
Inserción de referencias	31,2	3,4	27,8	89,1 %
Generación de tablas	22,5	4,2	18,3	81,3 %
Ajustes tipográficos	19,8	1,1	18,7	94,4 %
TOTAL	116,5 min	12,6 min	103,9 min	89,2 %

Nota. Artículos de 6–8 páginas. El ahorro de 89,2 % en tiempo de formateo debe leerse junto al tiempo de cómputo del modelo, que el hardware actual añade como espera adicional.

El ahorro global de 89,2 % en tiempo de formateo es el hallazgo más significativo desde la perspectiva del usuario. En términos absolutos, el investigador recupera casi dos horas por artículo que antes dedicaba a tareas de formato.

La sección de inserción de referencias concentra el mayor ahorro absoluto (27,8 min), lo que coincide con la percepción de los encuestados sobre la tarea más tediosa del proceso de escritura académica.

Tabla 11

Satisfacción general (Likert 5 puntos, %)

Ítem	M. Ins.	Ins.	Neutro	Sat.	M. Sat.
Calidad del PDF	1	3	8	41	47
Facilidad de uso	1	4	11	43	41
Velocidad de generación	2	6	14	38	40
Fidelidad al formato	1	3	9	44	43
Recomendaría el sistema	0	2	7	39	52
Global (promedio)	1	3,6	9,8	41	44,6

Nota. M. Ins. = Muy insatisfecho; Ins. = Insatisfecho; Sat. = Satisfecho; M. Sat. = Muy satisfecho.

El 85,6 % de los participantes se declaró satisfecho o muy satisfecho con el sistema. El ítem con menor satisfacción fue la velocidad de generación (78 % satisfechos o muy satisfechos), coherente con los tiempos de espera del hardware disponible.

El dato más revelador es que el 91 % recomendaría el sistema a colegas, incluyendo a quienes nunca habían usado LaTeX, lo que sugiere que la curva de aprendizaje percibida es aceptable.

Validación por panel de expertos

Tabla 12

Puntuaciones Likert del panel de expertos (escala 1–5)

Dimensión	E1	E2	E3	E4	E5	Media
Necesidad real del sistema	5	4	5	5	5	4,80
Ventaja ejecución local	5	5	4	5	5	4,80
Solidez prompts especializa-dos	4	5	4	5	5	4,60
Represen-tatividad del corpus	4	4	5	4	4	4,20
Aceptabi-lidad tasa 94,3 %	5	4	5	4	5	4,60

Dimensión	E1	E2	E3	E4	E5	Media
Relevancia T_score vs. Word	5	4	4	5	5	4,60
Criticidad gestión referencias	5	5	4	5	4	4,60
Costo $0 – factor adopción	5	4	5	4	5	4,60
Potencial democrati-zador	5	5	5	5	5	5,00
Promedio por experto	4,78	4,44	4,56	4,67	4,78	4,64

Nota. Panel de 5 expertos docentes investigadores. Escala 1 (muy en desacuerdo) a 5 (totalmente de acuerdo). Posgrado SOE–UAGRM, 2025.

El panel otorgó al sistema una puntuación promedio de 4,64/5,00, superando el umbral de aceptación (4,00) en todas las dimensiones, con puntuaciones individuales entre 4,44 y 4,78, evidenciando alto consenso entre expertos. La dimensión potencial democratizador alcanzó unanimidad (5/5), destacando el acceso sin costo y la ejecución local como ventajas clave. En contraste, la representatividad del corpus obtuvo la media más baja (4,20), señalándose como limitación la ausencia de textos de ciencias sociales y humanidades. Las dimensiones de desempeño técnico (solidez de prompts, tasa de compilación, relevancia del T_score y gestión de referencias) obtuvieron puntuaciones homogéneas (M = 4,60), al igual que el costo operativo nulo, valorado como un factor relevante para su adopción.

DISCUSIÓN

La capacidad de los modelos de lenguaje de gran escala LLM para generar estructuras sintácticas coherentes y código formalmente válido ha sido ampliamente documentada en la literatura reciente. Estudios como los de Dwivedi et al. (2023) destacan su impacto en la producción científica, mientras que Gao et al. (2023) muestran que estos modelos pueden generar contenidos comparables a los producidos por investigadores humanos. Por su parte, el uso de modelos abiertos ejecutados localmente, como los propuestos por Touvron et al. (2023), permite superar las limitaciones asociadas al costo y la privacidad de los datos. A la luz de estos antecedentes, los resultados obtenidos en el presente estudio sugieren que integrar modelos de lenguaje locales con una cadena de compilación LaTeX constituye una alternativa viable para la generación automática de documentos científicos.

En relación con la tasa de compilación, el valor alcanzado (94,3 %) indica un nivel de estabilidad elevado para un sistema basado en modelos de lenguaje. Este resultado es consistente con estudios recientes sobre generación automatizada de contenido científico, donde se ha demostrado que los modelos de lenguaje pueden producir textos con niveles de calidad comparables a procesos manuales (Wu et al., 2025). La mejora observada sugiere que la incorporación de una arquitectura estructurada y un pipeline de ensamblaje contribuye significativamente a reducir errores de compilación. Respecto a la calidad tipográfica, los resultados confirman la superioridad de LaTeX frente a herramientas de procesamiento de texto convencionales como Microsoft Word, especialmente en aspectos como la consistencia de formato y la gestión de estructuras complejas. Este hallazgo coincide con la literatura clásica sobre composición tipográfica científica, pero aporta evidencia empírica al integrar modelos de lenguaje como generadores automáticos de contenido.

En el ámbito de la representación matemática, los valores obtenidos evidencian una mejora sustancial en comparación con herramientas tradicionales, lo cual resulta especialmente relevante para disciplinas que requieren notación formal. Este resultado sugiere que los modelos de lenguaje, cuando son correctamente guiados mediante prompts estructurados, pueden generar expresiones matemáticas con un nivel aceptable de precisión. En cuanto a la percepción de los usuarios, los resultados reflejan una alta aceptabilidad del sistema, lo que indica que la automatización del proceso reduce la barrera de entrada a LaTeX. Este aspecto es consistente con investigaciones recientes que reportan mejoras en la experiencia de escritura académica mediante herramientas de inteligencia artificial (Wang & Ren, 2024), especialmente en contextos donde los usuarios no poseen formación técnica avanzada.

Desde una perspectiva aplicada, el uso de modelos de lenguaje locales representa una ventaja estratégica frente a soluciones basadas en servicios en la nube, al eliminar costos operativos y dependencias externas. Este resultado tiene implicaciones directas para instituciones con recursos limitados, ampliando el acceso a herramientas de escritura científica avanzada. No obstante, el estudio presenta limitaciones. En primer lugar, el corpus de evaluación se restringe a áreas de ciencias exactas e ingeniería, lo que limita la generalización de los resultados a disciplinas como ciencias sociales o humanidades. En segundo lugar, la muestra de usuarios fue de tipo no probabilístico, lo que puede introducir sesgos en la percepción reportada. Finalmente, se identificaron dificultades en la gestión de referencias bibliográficas, particularmente en la consistencia de claves BibTeX, lo cual coincide con estudios recientes que advierten limitaciones en la confiabilidad de contenidos generados por modelos de lenguaje (Athaluri et al., 2025).

En conjunto, los hallazgos del estudio aportan evidencia empírica sobre la viabilidad de integrar modelos de

lenguaje locales en procesos de escritura científica automatizada, abriendo nuevas líneas de investigación en la intersección entre inteligencia artificial y producción académica. En este sentido, el uso de estos sistemas debe entenderse como un complemento al proceso de escritura científica, más que como un sustituto del investigador (Dergaa et al., 2023).

CONCLUSIONES

En relación con el objetivo del desarrollo de un sistema generación automática de artículos científicos, se logró una arquitectura pipeline modular compuesta por componentes de captura, procesamiento mediante modelos de lenguaje, ensamblaje LaTeX y compilación PDF, la cual demostró ser funcional, escalable y adecuada para la generación automatizada de documentos científicos. Respecto a la evaluación de la tasa de compilación, se determinó que el sistema alcanzó un promedio global de 94,3 %, superando el umbral de viabilidad establecido, lo que evidenció un desempeño técnico estable en la generación de documentos sin errores de compilación.

En cuanto a la calidad tipográfica, matemática y de referencias, se comprobó que los documentos generados presentaron una calidad superior a Microsoft Word, con diferencias estadísticamente significativas, especialmente en la consistencia tipográfica y representación de expresiones matemáticas.

La alta aceptación registrada por los usuarios, independientemente de su experiencia previa con LaTeX, sugiere que la interfaz de generación automática logra desacoplar la complejidad técnica del proceso de escritura científica. Esto indica que la barrera de acceso a este sistema de composición tipográfica no es inherente a la naturaleza del formato, sino al modo en que se expone al usuario, lo que abre la posibilidad de democratizar su uso en contextos académicos con recursos limitados. Finalmente, en la validación por expertos, el sistema obtuvo una valoración promedio de 4,64/5, destacándose su pertinencia, potencial democratizador y viabilidad de implementación en entornos académicos con recursos limitados.

En conjunto, se concluyó que el sistema desarrollado permitió automatizar el proceso de generación de artículos científicos en LaTeX, reduciendo significativamente el tiempo de formateo y eliminando la dependencia de servicios de API, constituyéndose en una alternativa viable y sostenible para instituciones de educación superior. Como limitación, se identificó la gestión de claves bibliográficas, donde los modelos tienden a generar identificadores no siempre consistentes con los archivos .bib del usuario, lo que requiere intervención manual y representa una línea de mejora para trabajos futuros. Adicionalmente, la dependencia del rendimiento del hardware utilizado puede influir en los tiempos de generación, lo que representa una limitación para su implementación en entornos con infraestructura limitada.

BIBLIOGRAFÍA

Athaluri, S., et al. (2025). Artificial intelligence-assisted academic writing: Recommendations for ethical use. Advances in Simulation, 10(1). https://doi.org/10.1186/s41077-025-00350-6

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., & Amodei, D. (2020). Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877–1901. https://doi.org/10.48550/arXiv.2005.14165

Dergaa, I., Chamari, K., Zmijewski, P., & Ben Saad, H. (2023). From human writing to artificial intelligence generated text: Examining the prospects and potential threats of ChatGPT in academic writing. Biology of Sport, 40(2), 615–622. https://doi.org/10.5114/biolsport.2023.125623

Dwivedi, Y. K., Kshetri, N., Hughes, L., et al. (2023). So what if ChatGPT wrote it? Multidisciplinary perspectives on generative AI. International Journal of Information Management, 71, 102642. https://doi.org/10.1016/j.ijinfomgt.2023.102642

Gao, C. A., Howard, F. M., Markov, N. S., et al. (2023). Comparing scientific abstracts generated by ChatGPT. NPJ Digital Medicine, 6(1), 75. https://doi.org/10.1038/s41746-023-00819-6

Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., Stadler, M., Weller, J., Kuhn, J., & Kasneci, G. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. Learning and Individual Differences, 103, Article 102274. https://doi.org/10.1016/j.lindif.2023.102274

Knuth, D. E. (1997). The art of computer programming: Vol. 3. Sorting and searching (2nd ed.). Addison-Wesley.

Kopka, H., & Daly, P. W. (2004). A guide to LaTeX (4th ed.). Addison-Wesley Professional.

Touvron, H., Lavril, T., Izacard, G., et al. (2023). LLaMA: Open and efficient foundation language models. arXiv. https://doi.org/10.48550/arXiv.2302.13971

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30, 5998–6008. https://doi.org/10.5555/3295222.3295349

Wang, L., & Ren, B. (2024). Enhancing academic writing in a linguistics course with generative AI. Education Sciences, 14(12), 1329. https://doi.org/10.3390/educsci14121329

Wu, S., et al. (2025). Automated literature research and review-generation method based on large language models. National Science Review, 12(6). https://doi.org/10.1093/nsr/nwaf169

POBLACIÓN Y MUESTRA EN ESTUDIOS DE CASO EN INGENIERÍA DE SOFTWARE: MARCO Y GUÍA PRÁCTICA

PhD. Luis Roberto Pérez Rios
Posgrado SOE – UAGRM
https://orcid.org/0000-0002-8385-1016
Santa Cruz, Bolivia | luis.roberto@alenasoft.com

https://doi.org/10.23670 FT.2026.1.39

Recibido 08/05/2026 - Aceptado 29/05/2026

RESUMEN

La investigación en ciencias de la computación e ingeniería de software recurre con frecuencia al estudio de caso como diseño metodológico, pero la forma en que se declaran y justifican los conceptos de población y muestra en estos trabajos es a menudo imprecisa o directamente omitida, lo que debilita la credibilidad de sus hallazgos ante revisores formados en metodología cuantitativa. Este artículo de revisión examina el fundamento epistemológico del estudio de caso en ingeniería de software, analiza cómo los conceptos de población y muestra se reformulan en este contexto, y propone un marco operativo para declarar, justificar y delimitar tanto la población como el caso seleccionado en investigaciones de este tipo. Se revisan los marcos metodológicos de referencia canónicos del área y se articulan sus principios en una guía de cinco pasos aplicable a trabajos originales en las áreas de ingeniería de software, arquitectura de sistemas, seguridad informática y desarrollo de herramientas. El artículo concluye que la especificación rigurosa de la población y la justificación explícita del tipo de muestreo son condiciones necesarias para que un estudio de caso en ingeniería produzca conocimiento generalizable, y que esta generalización opera por vía analítica –no estadística–, lo que tiene consecuencias directas sobre cómo se redactan las secciones de metodología y discusión.

Palabras clave: estudio de caso, ingeniería de software, población y muestra, metodología de investigación, generalización analítica

ABSTRACT

Research in computer science and software engineering frequently relies on case study design, yet the way population and sample are declared and justified in these works is often imprecise or outright absent, undermining credibility with reviewers trained in quantitative methodology. This review article examines the epistemological foundation of case study research in software engineering, analyzes how the concepts of population and sample are reformulated in this context, 1Boletín Científico Fronteras Tecnológicas andproposes anoperational framework for declaring, justifying, and delimiting both the population and the selected case in this type of research. The canonical methodological frameworks of the field are reviewed and their principles are articulated into a five-step guide applicable to original work in software engineering, systems architecture, security, and tool development. The article concludes that rigorous population specification and explicit sampling justification are necessary conditions for a case study in engineering to produce generalizable knowledge, and that this generalization operates analytically –not statistically–, with direct consequences for how methodology and discussion sections are written.

Keywords: case study, software engineering, population and sample, research methodology, analytical generalization

INTRODUCCIÓN

La investigación empírica en ciencias de la computación e ingeniería de software ha experimentado en las últimas dos décadas una expansión significativa en el uso de métodos cualitativos y mixtos, entre los cuales el estudio de caso ocupa un lugar central (Runeson & Höst, 2009; Wohlin et al., 2024). Esta tendencia refleja un reconocimiento gradual de que muchos fenómenos relevantes en el desarrollo de software —la efectividad de una metodología, el impacto de una herramienta, la calidad arquitectónica de un sistema— no pueden ser estudiados adecuadamente mediante experimentos controlados o encuestas estadísticas, porque ocurren en contextos organizativos e históricos específicos que el control experimental eliminaría (Flyvbjerg, 2006). El estudio de caso permite, en cambio, preservar esa riqueza contextual y examinar el fenómeno en su entorno natural.

Sin embargo, la adopción de este método no ha sido acompañada siempre por una comprensión suficiente de sus fundamentos epistemológicos, y en particular de cómo los conceptos de población y muestra —centrales en la investigación cuantitativa— se reformulan cuando el objeto de estudio es un sistema de software, un proceso de desarrollo o una herramienta tecnológica, y no un grupo humano. En consecuencia, es común encontrar en publicaciones de ingeniería de software secciones de metodología que omiten por completo la justificación de por qué se seleccionó el caso estudiado, qué dominio de sistemas representa y a qué clase de afirmaciones generalizables da lugar el análisis. Esta omisión no es trivial: un revisor metodológicamente riguroso puede rechazar un trabajo bien ejecutado simplemente porque la sección de metodología no articula con claridad los límites de validez del estudio. La magnitud del problema ha sido documentada empíricamente: Baltes y Ralph (2022) revisaron sistemáticamente las prácticas de muestreo en investigación empírica de ingeniería de software y encontraron que el área presenta una crisis de generalización caracterizada por la prevalencia del muestreo de conveniencia no declarado y por la confusión sistemática entre representatividad estadística y analítica; Wohlin y Rainer (2022) identificaron, por su parte, que un número significativo de trabajos que se autoidentifican como estudios de caso no satisfacen los criterios metodológicos que definen el diseño. Ambos hallazgos confirman que el problema no es una percepción docente ni un riesgo potencial, sino una deficiencia activa y medible en la literatura del área.

La pertinencia de este trabajo radica en que la ingeniería de software se encuentra en una posición epistemológicamente ambigua: es una disciplina de ingeniería —en el sentido de que produce artefactos funcionales— pero sus objetos de estudio son en gran medida construcciones sociotécnicas cuyo comportamiento depende del contexto humano y organizacional en que operan (Shaw, 2003). Esta ambigüedad hace que ni el paradigma experimental de las ciencias naturales ni el paradigma interpretativo de las ciencias sociales sean completamente adecuados para todos sus problemas de investigación, y que el estudio de caso ocupe ese espacio intermedio donde la ingeniería y la ciencia social se encuentran.

El objetivo general del presente artículo es proporcionar al investigador en ciencias de la computación e ingeniería de software un marco teórico integrado y una guía operativa para declarar y justificar los conceptos de población y muestra en investigaciones basadas en estudio de caso. De este objetivo se derivan cuatro objetivos específicos: (OE1) revisar los fundamentos epistemológicos del estudio de caso en ingeniería de software y sus implicaciones para la selección y declaración del caso; (OE2) analizar la reformulación de los conceptos de población y muestra en el contexto del estudio de caso en ingeniería; (OE3) sintetizar los criterios de selección del caso derivados de los marcos metodológicos canónicos del área; y (OE4) proponer un procedimiento operativo de cinco pasos para declarar y justificar la población y la muestra en la sección de metodología de trabajos con diseño de caso.

METODOLOGÍA

El presente artículo constituye una revisión narrativa y focalizada de los marcos metodológicos canónicos sobre el estudio de caso como diseño de investigación, con énfasis específico en su aplicación al campo de la ingeniería de software. Este tipo de revisión —denominado también revisión teórica o conceptual en la literatura metodológica— se diferencia de la revisión sistemática (Kitchenham et al., 2007) en que no busca exhaustividad en la cobertura de un corpus de publicaciones mediante protocolos de búsqueda replicables, sino profundidad interpretativa en el análisis de las obras que han estructurado el debate metodológico en el área. La pertinencia de este enfoque es coherente con el propósito declarado del artículo: no catalogar el estado del arte, sino ofrecer al investigador en ciencias de la computación un marco teórico integrado y una guía operativa aplicable directamente en la redacción de sus secciones de metodología.

Fuentes de información

Las fuentes primarias de este artículo son las obras que han alcanzado estatus canónico en la investigación empírica en ingeniería de software: Yin (2018) como referencia central para el diseño del estudio de caso; Runeson y Höst (2009) y Runeson et al. (2012) para las guías específicas al área de software; Wohlin et al. (2024) para la experimentación empírica en ingeniería; y Eisenhardt (1989) para la teoría del muestreo teórico en estudios de caso. Se incorporaron adicionalmente obras que amplían el espectro epistemológico del análisis: Flyvbjerg (2006) sobre los malentendidos frecuentes en torno al método; Stake (1995) sobre la distinción intrínseco/instrumental del caso; Benbasat et al. (1987) sobre los criterios de defensibilidad metodológica; Lincoln y Guba (1985) sobre la transferibilidad como alternativa interpretativa a la generalización; Kitchenham et al. (2007) y Shaw

(2003) como marcos de referencia sobre la calidad de la investigación empírica en el campo; y Baltes y Ralph (2022) y Wohlin y Rainer (2022) como evidencia empírica reciente sobre la prevalencia y naturaleza de los problemas de muestreo en la literatura del área. El acceso a estas obras se realizó a través de SpringerLink, Wiley Online Library, SAGE Publications Digital Library, ACM Digital Library y Google Scholar.

Criterios de inclusión y exclusión

Los criterios de inclusión aplicados fueron: (a) obras con reconocido estatus canónico o amplia citación en la comunidad de investigación empírica en ingeniería de software, verificado por recuento de citas en Google Scholar y presencia en bibliografías de guías metodológicas publicadas en revistas del área; (b) publicaciones en revistas indexadas o libros editados por casas académicas reconocidas; y (c) obras que abordan directamente el diseño del estudio de caso o sus criterios de validez en contextos de ingeniería o ciencias sociales aplicadas. Los criterios de exclusión fueron: (a) obras de literatura gris sin revisión por pares; (b) publicaciones que abordan el estudio de caso únicamente como técnica pedagógica sin discusión metodológica; y (c) trabajos que no aportan elementos diferenciadores respecto de los marcos canónicos ya seleccionados, evitando redundancia sin ganancia analítica.

Palabras clave y estrategia de búsqueda

La identificación inicial de fuentes se realizó mediante búsqueda en Google Scholar y ACM Digital Library con los descriptores: case study research software engineering, empirical software engineering methodology, case selection criteria, theoretical sampling case study, analytic generalization, population and sample qualitative research, y sus equivalentes en español. La selección final de los marcos de referencia no se basó en el resultado cuantitativo de la búsqueda—dada la naturaleza narrativa de la revisión— sino en la convergencia de citas entre los propios marcos: las obras seleccionadas son aquellas que se citan mutuamente de forma sistemática en el sub-campo de la investigación empírica en ingeniería de software, lo que constituye un criterio de pertinencia teórica coherente con el enfoque de la revisión.

Limitaciones metodológicas

Al tratarse de una revisión narrativa y no sistemática, el proceso de selección de fuentes incorpora un grado de discrecionalidad que una revisión sistemática eliminaría mediante protocolos de búsqueda replicables. El principal riesgo asociado es el sesgo de confirmación: la selección de marcos que refuerzan la posición teórica del artículo en lugar de cuestionar sus premisas. Este riesgo se mitiga parcialmente por la incorporación de perspectivas alternativas —Lincoln y Guba (1985) como contraste interpretativo a Yin, Stake (1995) como complemento a la tradición positivista del diseño de caso— y por la declaración explícita de que la taxonomía de errores frecuentes tiene naturaleza observacional y no deriva de una revisión sistemática de publicaciones. Adicionalmente, la selección de obras canónicas incorpora fuentes publicadas hasta 2024 — incluyendo la edición actualizada de Yin (2018), la segunda edición de Wohlin et al. (2024) y los estudios empíricos recientes de Baltes y Ralph (2022) y Wohlin y Rainer (2022)—, lo que garantiza que el marco de referencia refleja el estado actual del debate metodológico en el área.

RESULTADOS Y DISCUSIÓN

El estudio de caso como diseño de investigación en ingeniería de software

Definición y características fundamentales: El estudio de caso es, según la definición ampliamente citada de Yin (2018), una investigación empírica que examina un fenómeno contemporáneo en profundidad y en su contexto real, especialmente cuando los límites entre el fenómeno y el contexto no son claramente evidentes. Esta definición tiene tres implicaciones metodológicas directas para la ingeniería de software. Primera: el estudio de caso es pertinente cuando el fenómeno —por ejemplo, la adopción de una práctica ágil, la presencia de deuda técnica, la efectividad de una herramienta de análisis estático— no puede separarse del contexto organizacional, técnico e histórico en que ocurre sin perder información esencial. Segunda: el estudio de caso no es un diseño de conveniencia para cuando no hay suficientes sujetos de estudio, sino un diseño específicamente adecuado para ciertas preguntas de investigación. Tercera: un estudio de caso puede ser completamente riguroso y producir conocimiento científicamente válido, pero esa validez opera a través de mecanismos diferentes a los de la investigación experimental.

Runeson y Höst (2009) ofrecen una clasificación del estudio de caso especialmente ajustada a la ingeniería de software, distinguiendo cuatro propósitos principales: exploratorio (generar hipótesis para investigaciones posteriores), descriptivo (documentar cómo ocurre un fenómeno), explicativo (establecer relaciones causales en contexto) y mejorativo (estudiar el efecto de una intervención). Cada uno de estos propósitos tiene implicaciones distintas sobre cómo se selecciona el caso, qué datos se recogen y cómo se interpretan los resultados. Un investigador que no declara explícitamente el propósito de su estudio dificulta al lector la evaluación de si el diseño metodológico es coherente con lo que el trabajo pretende demostrar. Esta observación metodológica tiene fundamento empírico: Wohlin y Rainer (2022) han examinado críticamente trabajos publicados en ingeniería de software que se autoidentifican como estudios de caso y encuentran que un número significativo no satisface los criterios que definen el método; la inconsistencia en el uso del término es, por tanto, un problema activo en el área, no meramente un riesgo potencial.

Stake (1995) complementa esta perspectiva desde la tradición interpretativa del estudio de caso: mientras que Yin (2018) enfatiza el rigor del diseño de investigación

y la validez de los constructos, Stake distingue entre el estudio de caso intrínseco —donde el caso importa por sí mismo, independientemente de cualquier generalización— y el instrumental —donde el caso sirve para iluminar un fenómeno más amplio—. En el contexto de la ingeniería de software, la mayoría de los estudios responden al segundo tipo: el sistema analizado es el medio para generar conocimiento sobre una clase de sistemas, no un fin en sí mismo. Esta distinción es operativamente relevante porque confirma la Tercera implicación de Yin (2018): la validez del estudio de caso no reside en la representatividad estadística del caso, sino en que los mecanismos estudiados sean los correctos para el fenómeno de interés, lo cual exige declarar con precisión el dominio de sistemas al que esos mecanismos son analíticamente extrapolables.

La cuestión de la generalización: La objeción más frecuente al estudio de caso como método científico es la que señala que sus conclusiones no son generalizables porque se basan en un solo caso —o en muy pocos casos— que no puede ser representativo estadísticamente de ninguna población. Esta objeción, aunque intuitivamente comprensible, confunde dos tipos de generalización que la metodología científica distingue con claridad (Yin, 2018). La generalización estadística es la que opera en los estudios con muestra representativa: se mide un atributo en n individuos seleccionados aleatoriamente del universo y se infiere que ese atributo tiene la misma distribución en toda la población, con un margen de error calculable. La generalización analítica, en cambio, es la que opera en los estudios de caso: el investigador demuestra que los hallazgos del caso son consistentes con una teoría o con un mecanismo causal conocido, y esa consistencia permite afirmar que el mismo mecanismo actuaría en otros contextos con características análogas, independientemente de cuántos casos se hayan estudiado.

Este principio tiene una consecuencia práctica directa para la sección de metodología: el investigador no debe justificar el número de casos con argumentos estadísticos, sino argumentar por qué el caso seleccionado es representativo del mecanismo o fenómeno que se estudia. Flyvbjerg (2006) señala que incluso un único caso —si está bien seleccionado— puede refutar una teoría establecida, lo que demuestra que el valor epistémico de un estudio de caso no es proporcional a su tamaño, sino a la calidad de la selección y del análisis. En la práctica de la ingeniería de software, esto significa que lo que importa no es cuántos sistemas se analizaron, sino si el sistema analizado fue seleccionado con criterios explícitos que lo hacen representativo del dominio de interés.

Reformulación de población y muestra en ingeniería de software

La población como dominio de fenómenos: En la investigación con encuesta o experimento, la población es el conjunto de individuos —personas, empresas, respuestas posibles— sobre los que se pretende generalizar los hallazgos. El investigador selecciona una muestra de ese conjunto, la estudia y extrapola las conclusiones al conjunto completo. Esta lógica presupone que los individuos de la población son comparables entre sí en los atributos relevantes y que la variabilidad observada en la muestra refleja la variabilidad real del conjunto.

En la investigación de ingeniería de software con estudio de caso, el objeto de estudio no es una persona ni una organización como tal, sino un artefacto o proceso de naturaleza técnica: un sistema de software, una arquitectura, una metodología de desarrollo, una herramienta o una práctica de equipo. En consecuencia, la población es el dominio de artefactos o procesos al que el hallazgo es potencialmente aplicable, es decir, la clase de sistemas o situaciones que comparten las características estructurales que hacen posible el fenómeno estudiado. Wohlin et al. (2024) describen esta reformulación con el concepto de objeto de estudio y contexto: la población es la clase de objetos y contextos sobre los que se afirma que la relación estudiada se sostiene.

Por tanto, definir la población en un estudio de ingeniería de software equivale a responder: ¿en qué clase de sistemas, procesos u organizaciones serían aplicables los hallazgos de este trabajo? Esta pregunta no es retórica; su respuesta delimita los límites de validez del estudio y protege al investigador de realizar afirmaciones más amplias de lo que el diseño permite. Si el investigador no responde esta pregunta explícitamente, el lector —y el revisor— la responderán por él, a menudo de forma más restrictiva.

La muestra como caso seleccionado: Dado que la generalización en el estudio de caso opera de forma analítica y no estadística, la muestra no necesita ser aleatoria ni representativa en términos estadísticos. Lo que sí debe ser es intencionalmente seleccionada, con criterios documentados que justifiquen por qué ese caso particular es adecuado para estudiar el fenómeno de interés. Eisenhardt (1989) denomina a este procedimiento muestreo teórico (theoretical sampling): el caso se selecciona por su potencial para revelar el mecanismo teórico que se estudia, no por su frecuencia en la población.En la práctica, esto significa que el investigador debe documentar explícitamente los criterios de inclusión del caso —qué características debe tener para ser objeto de estudio— y, cuando sea posible, los criterios de exclusión —qué características lo descalificarían—. Esta documentación cumple tres funciones: permite al lector evaluar si la selección fue apropiada para el propósito declarado, permite a investigadores futuros replicar el proceso con casos diferentes del mismo dominio, y protege al investigador de la acusación de selección oportunista —es decir, de haber elegido el caso porque era conveniente o familiar, no porque fuera metodológicamente adecuado.

Tipos de diseño de caso: único y múltiple: La decisión de estudiar un único caso o múltiples casos responde a consideraciones metodológicas distintas. Yin (2018) identifica cinco justificaciones para el diseño de caso

único: (a) el caso es decisivo (critical case) para probar o refutar una teoría bien formulada; (b) el caso es extremo o único en sus características (extreme case; (c) el caso es representativo (typical case) del fenómeno de interés; (d) el caso es revelador (revelatory case), es decir, se trata de un fenómeno antes inaccesible para la investigación; o (e) el caso es longitudinal, estudiado en múltiples puntos en el tiempo.

Para la ingeniería de software, el caso representativo es el justificante más habitual: se selecciona un sistema de software que sea típico de una clase más amplia, con el argumento de que los hallazgos sobre ese caso particular tienen valor como demostración de que el fenómeno existe y puede describirse. Este es el diseño apropiado, por ejemplo, cuando se demuestra la viabilidad de una metodología sobre un sistema concreto —el argumento no es que todos los sistemas del dominio tendrán exactamente los mismos resultados, sino que el proceso es aplicable a sistemas de ese tipo.

El diseño de casos múltiples, en cambio, es adecuado cuando se estudian variaciones del fenómeno en distintos contextos, cuando se busca fortalecer la generalización analítica mediante la replicación literal o teórica (Yin, 2018), o cuando las diferencias entre casos son ellas mismas parte del hallazgo. Wohlin et al. (2024) señalan que en ingeniería de software, los casos múltiples son frecuentes en estudios sobre prácticas de equipo —donde la variabilidad entre organizaciones es parte central del resultado— pero menos habituales en estudios sobre herramientas o metodologías, donde un caso representativo bien documentado puede ser suficiente para demostrar la viabilidad o eficacia de la propuesta.

Tipos de muestreo en estudios de caso en ingeniería

Muestreo intencional o por propósito El muestreo intencional —también denominado muestreo por propósito o purposive sampling— es el tipo más frecuente en estudios de caso en ingeniería de software. Consiste en seleccionar el caso que mejor satisface los criterios establecidos para el propósito del estudio. En consecuencia, la calidad del muestreo no se mide por el número de casos seleccionados, sino por la coherencia entre los criterios de selección y el propósito declarado de la investigación.

Los criterios típicos de muestreo intencional en ingeniería de software incluyen: complejidad representativa del sistema —suficiente para que el fenómeno estudiado se manifieste, pero acotada para que el análisis sea manejable—; disponibilidad de documentación —el caso debe estar suficientemente documentado para que la recolección de datos sea posible y verificable—; diversidad de características internas relevantes para el estudio —por ejemplo, múltiples componentes, flujos de datos, actores—; y acceso o familiaridad del investigador con el sistema, siempre que ese acceso no introduzca sesgos no declarados. Benbasat et al. (1987) identifican esta combinación de criterios como la condición para que el caso sea metodológicamente defensible, más allá de consideraciones de conveniencia.

Muestreo teórico El muestreo teórico, propuesto originalmente en el marco de la grounded theory por Glaser y Strauss (1967) y adaptado para el estudio de caso por Eisenhardt (1989), selecciona los casos en función de su capacidad para extender, refutar o matizar la teoría que se está construyendo o evaluando. A diferencia del muestreo intencional clásico, donde los criterios se definen antes de iniciar el trabajo de campo, el muestreo teórico puede ser iterativo: el análisis de un primer caso revela qué tipo de caso adicional sería más informativo para continuar el desarrollo teórico.

En ingeniería de software, el muestreo teórico es especialmente pertinente cuando el objetivo del trabajo es proponer o refinar una teoría sobre un fenómeno —por ejemplo, una teoría sobre las condiciones bajo las cuales ciertos patrones arquitectónicos generan deuda técnica acumulable.

En ese contexto, el investigador puede seleccionar deliberadamente casos que maximicen la variación en las variables de interés, con el propósito de encontrar los límites del fenómeno, en lugar de buscar la confirmación sistemática de lo ya conocido.

Muestreo de conveniencia y sus limitaciones El muestreo de conveniencia —seleccionar el caso que está disponible o que el investigador conoce de antemano— es el tipo más frecuente en la práctica real de la ingeniería de software, aunque rara vez se declara como tal. Un proyecto de grado o de maestría generalmente estudia el sistema en el que el investigador trabaja o ha trabajado; un artículo de conferencia evalúa una herramienta sobre el proyecto de código abierto más accesible.

Esta práctica no invalida el estudio, pero sí exige que el investigador argumente explícitamente por qué ese caso disponible cumple los criterios del dominio de interés. Baltes y Ralph (2022), en una revisión crítica de las prácticas de muestreo en investigación empírica de ingeniería de software, documentan que el muestreo aleatorio es raro, las estrategias sofisticadas son aún menos frecuentes, y que los conceptos de representatividad y aleatoriedad son frecuentemente mal comprendidos en los trabajos publicados —lo que los autores denominan una crisis de generalización (generalizability crisis) en el área—, siendo el muestreo de conveniencia no declarado uno de sus factores contribuyentes más recurrentes.

El problema del muestreo de conveniencia no declarado es que el lector no puede distinguir si el caso fue seleccionado por su valor metodológico o por su disponibilidad; esa ambigüedad introduce dudas sobre si los hallazgos serían replicables en un caso seleccionado con criterios más rigurosos.

En consecuencia, la recomendación práctica es que incluso cuando el caso fue seleccionado por conveniencia, el investigador documente los criterios de inclusión que ese caso satisface y los criterios

que, de no satisfacerse, habrían llevado a descartar el caso, convirtiendo retrospectivamente la selección de conveniencia en una selección intencional documentada

Criterios de selección del caso: guía operativa

Los cinco criterios fundamentales A partir de la revisión de los marcos metodológicos de Runeson y Höst (2009) y su desarrollo posterior en Runeson et al. (2012), Yin (2018) y Wohlin et al. (2024), es posible identificar cinco criterios cuya documentación es necesaria —aunque no siempre suficiente— para justificar la selección de un caso en ingeniería de software.

Criterio 1 — Representatividad estructural (Wohlin et al., 2024; Yin, 2018). El caso debe poseer las características estructurales del dominio de población declarado. Si la población es "sistemas de software empresarial multicapa", el caso debe tener al menos los estratos funcionales típicos de ese tipo de sistemas.

Este criterio garantiza que el mecanismo que se estudia pueda efectivamente manifestarse en el caso seleccionado; si el caso no tiene las características estructurales mínimas, el hallazgo —positivo o negativo— carece de relevancia para el dominio.

Criterio 2 — Complejidad suficiente y acotada (Runeson & Höst, 2009; Runeson et al., 2012). El caso debe ser suficientemente complejo para que el fenómeno de interés se produzca con la riqueza necesaria para el análisis, pero suficientemente acotado para que el investigador pueda completar el estudio en profundidad. Esta tensión entre complejidad y manejabilidad es uno de los equilibrios más difíciles del diseño de caso en ingeniería, y la decisión sobre dónde establecer el límite debe ser argumentada explícitamente en dos dimensiones.

El umbral mínimo de complejidad está determinado por la naturaleza del fenómeno: un análisis de amenazas de seguridad requiere múltiples componentes y flujos que crucen al menos un límite de confianza, de modo que un sistema con un único componente sin interacción exterior no satisfaría este criterio independientemente de su disponibilidad.

El umbral máximo de complejidad, en cambio, está dado por los recursos del estudio: un caso que requiere analizar centenares de componentes interrelacionados puede ser teóricamente representativo pero prácticamente inmanejable, lo que comprometería la profundidad del análisis. Ambos umbrales deben declararse en la sección de metodología, con indicación de cómo el caso seleccionado los satisface.

Criterio 3 — Documentación verificable (Yin, 2018; Runeson et al., 2012). El caso debe estar suficientemente documentado para que la recolección de datos sea posible y los hallazgos sean trazables.

En el contexto de la ingeniería de software, esto significa que la arquitectura, las decisiones de diseño relevantes y el comportamiento esperado del sistema deben estar especificados de forma accesible para el investigador. Cuando la documentación es parcial —como ocurre frecuentemente con sistemas heredados— debe declararse explícitamente qué aspectos están documentados y qué aspectos requieren inferencia del investigador, así como los procedimientos utilizados para validar esa inferencia.

Criterio 4 — Relevancia para el fenómeno estudiado (Yin, 2018; Eisenhardt, 1989). El caso debe exhibir el fenómeno de interés o, al menos, las condiciones que hacen posible que ese fenómeno se manifieste. Si el estudio versa sobre la efectividad de un proceso de análisis de seguridad, el caso seleccionado debe ser un sistema que presente vulnerabilidades potenciales susceptibles de ser identificadas mediante ese proceso; un sistema que ya ha sido completamente auditado y remediado no permitiría evaluar la eficacia del proceso de identificación. Este criterio parece obvio, pero su omisión en la documentación metodológica da lugar a estudios cuya pregunta de investigación y caso seleccionado están desalineados.

Criterio 5 — Condiciones controladas o declaradas (Yin, 2018; Wohlin et al., 2024). El caso debe tener condiciones de diseño o de operación suficientemente conocidas para que el investigador pueda distinguir los hallazgos del estudio de los artefactos introducidos por variaciones no declaradas del caso. En ingeniería de software, esto puede lograrse mediante el uso de sistemas sintéticos diseñados para el estudio —donde todas las decisiones de diseño son conocidas de antemano— o mediante la documentación exhaustiva de las decisiones de diseño del sistema real, incluyendo sus supuestos y restricciones.

Los sistemas sintéticos tienen la ventaja de que eliminan la incertidumbre sobre el estado inicial, aunque introducen la limitación de que sus características son construidas por el investigador y pueden no reflejar fielmente la complejidad de los sistemas reales del dominio.

La tabla de criterios como artefacto metodológico Una práctica recomendable adoptada en investigaciones de ingeniería de software publicadas en revistas de primer nivel como Empirical Software Engineering, Journal of Systems and Software o IEEE Transactions on Software Engineering— es presentar los criterios de selección del caso en forma de tabla, con columnas para el criterio, su definición operacional y la evidencia de que el caso lo cumple.

Este formato tiene un valor doble: organiza la argumentación de manera que el revisor puede verificar cada criterio independientemente, y obliga al investigador a operacionalizar los criterios; es decir, a especificar qué observable del caso satisface cada uno antes de considerarlos cumplidos.

La Tabla 1 muestra un ejemplo de esta estructura aplicada de forma genérica.

Tabla 1

Estructura genérica de criterios de selección para estudios de caso en ingeniería de software

Criterio de selección	Definición operacional	Satisfecho cuando...
Representatividad estructural	El caso exhibe los componentes mínimos del dominio de población	El sistema tiene los estratos/componentes propios de la clase de sistemas estudiada
Complejidad suficiente	El fenómeno puede manifestarse con la riqueza necesaria	El número de componentes y flujos supera el umbral mínimo para que el análisis sea significativo
Complejidad acotada	El caso es manejable dentro de los recursos del estudio	Un investigador puede completar el análisis en profundidad dentro del cronograma disponible
Documentación verificable	La arquitectura y las decisiones relevantes están especificadas	Existe documentación de diseño a la que el investigador tiene acceso y que cubre los aspectos del fenómeno
Relevancia para el fenómeno	El caso exhibe o puede exhibir el fenómeno estudiado	Las condiciones de diseño hacen posible que el fenómeno ocurra o sea observable
Condiciones declaradas	El estado inicial del caso es conocido	Las decisiones de diseño, supuestos y restricciones están documentados y son consistentes entre sí

Nota. Adaptado de Runeson & Höst (2009), Yin (2018) y Wohlin et al. (2024).

Validez y generalización en estudios de caso

Los cuatro tipos de validez relevantes La validez de un estudio de caso en ingeniería de software se evalúa a través de cuatro dimensiones que Yin (2018) describe como los criterios de calidad del diseño de investigación: validez de constructo, validez interna, validez externa y confiabilidad.

La validez de constructo se refiere a si las medidas o los procedimientos utilizados en el estudio capturan efectivamente los conceptos que el trabajo dice estudiar. En ingeniería de software, esto equivale a preguntar si el proceso seguido, la herramienta aplicada o el artefacto producido corresponden a las definiciones teóricas que el marco del trabajo establece. Si un estudio afirma evaluar la "efectividad del modelado de amenazas" pero mide únicamente el número de amenazas identificadas sin considerar la tasa de falsos positivos ni el tiempo empleado, la validez de constructo es cuestionable.La forma de mitigar este riesgo es definir operacionalmente cada concepto antes de iniciar el trabajo de campo y documentar la trazabilidad entre el concepto y la medida.

La validez interna concierne a la coherencia causal del argumento: si el estudio establece una relación entre X e Y –por ejemplo, entre el uso de prompts estructurados y la calidad del análisis de amenazas–, debe poder argumentarse que esa relación no es un artefacto de terceras variables no controladas. En el estudio de caso, la validez interna se fortalece mediante la triangulación de fuentes –confirmar el mismo hallazgo por vías independientes– y mediante el análisis de explicaciones alternativas, es decir, argumentar por qué los resultados no se explicarían mejor por otro mecanismo.

La validez externa es la dimensión más directamente relacionada con la cuestión de la población y la muestra. Se refiere a si los hallazgos del caso son aplicables más allá del caso específico estudiado, y la respuesta depende de cuán claramente se ha definido la población y cuán bien el caso la representa. Como se ha establecido en secciones anteriores, esta generalización es de tipo analítico: el investigador argumenta que el mecanismo identificado en el caso es el mismo mecanismo que actuaría en cualquier otro caso con las características de la población declarada. La validez externa no equivale a decir que todos los sistemas del dominio producirán exactamente los mismos resultados, sino que el proceso o mecanismo estudiado es aplicable a esa clase de sistemas.

La confiabilidad se refiere a la replicabilidad del estudio: si otro investigador siguiera los mismos procedimientos sobre el mismo caso, ¿llegaría a los mismos hallazgos? En ingeniería de software, la confiabilidad se asegura mediante la documentación detallada del protocolo de estudio –los pasos seguidos, las decisiones tomadas, los instrumentos utilizados– de forma que cualquier investigador pueda repetir el trabajo con los mismos datos. La publicación del material de trabajo completo – diagramas, prompts, respuestas, código– en repositorios públicos es una práctica que fortalece significativamente la confiabilidad.

La generalización analítica: mecanismo y no frecuencia La generalización analítica opera a través del reconocimiento de que un hallazgo en un caso particular es instancia de un mecanismo general, y que ese mecanismo se reproducirá en otros casos donde estén presentes las condiciones que lo hacen posible. Este tipo de generalización está implícito en cómo la ingeniería de software produce y acumula conocimiento: un patrón de diseño documentado a partir de unos pocos sistemas ejemplares se aplica posteriormente a cualquier sistema que presente el problema que el patrón resuelve; una vulnerabilidad identificada en un sistema particular se convierte en un aviso para todos los sistemas con la misma arquitectura.

Por tanto, cuando un investigador escribe la sección de discusión de su estudio de caso, la generalización válida no es "todos los sistemas del dominio tendrán

los mismos resultados numéricos", sino "el mecanismo identificado —en este caso, que los supuestos de diseño inseguros concentran la mayor parte de la superficie de ataque— actuará en sistemas que compartan las características de la población declarada". Esta formulación es metodológicamente honesta porque explicita la condición de aplicabilidad —compartir las características de la población— y no promete una frecuencia estadística que el diseño no puede respaldar. Conviene situar esta posición en relación con la tradición interpretativa de la investigación cualitativa: Lincoln y Guba (1985) proponen el concepto de transferibilidad como alternativa a la generalización, argumentando que la aplicabilidad de un hallazgo a otro contexto depende del grado de similitud entre ambos —denominado fittingness— y que es responsabilidad del lector, no del investigador, determinar si esa similitud existe. En el contexto de la ingeniería de software, la posición de Yin resulta operativamente más adecuada para la escritura académica porque exige al investigador explicitar las condiciones del dominio de población, orientando al lector sobre los sistemas a los que los hallazgos son aplicables; la transferibilidad de Lincoln y Guba, en cambio, delega esa carga interpretativa al receptor, lo que dificulta la evaluación por parte de revisores que esperan declaraciones de alcance verificables en la sección de metodología.

Guía práctica de cinco pasos

A partir del marco teórico revisado, se propone un procedimiento operativo de cinco pasos para declarar y justificar la población y la muestra en la sección de metodología de un trabajo de ingeniería de software basado en estudio de caso.

Paso 1 — Definir el dominio de interés. El investigador debe comenzar respondiendo: ¿a qué clase de sistemas, procesos u organizaciones pretende que sus hallazgos sean aplicables? La respuesta a esta pregunta es la definición de la población, y debe formularse en términos de características observables —tipo de arquitectura, escala, contexto de despliegue, dominio funcional— no en términos de instancias concretas. Una formulación adecuada es del tipo: "sistemas de software multicapa orientados a la web con múltiples roles de usuario y flujos de datos que cruzan al menos un límite de confianza"; una formulación inadecuada sería: "sistemas similares al sistema X que estudiamos". La primera define el dominio por sus características; la segunda lo define circularmente por el caso.

Paso 2 — Documentar los criterios de selección del caso. Una vez definida la población, el investigador debe especificar por qué el caso seleccionado pertenece a ese dominio y es adecuado para el propósito del estudio. Para ello debe documentar al menos los cinco criterios descritos en la sección anterior — representatividad estructural, complejidad suficiente y acotada, documentación verificable, relevancia para el fenómeno y condiciones declaradas—, indicando para cada uno la evidencia observable que lo satisface. Esta documentación es la que convierte la selección del caso —sea intencional, teórica o incluso de conveniencia— en una selección metodológicamente defensible.

Paso 3 — Justificar el tipo de diseño (único o múltiple). El investigador debe argumentar explícitamente si el trabajo utiliza un diseño de caso único o múltiple, y por qué ese diseño es coherente con el propósito del estudio. Si se utiliza un solo caso, la justificación debe indicar cuál de las cinco razones de Yin (2018) aplica: ¿es un caso representativo? ¿es un caso decisivo para una teoría? ¿es un caso revelador de un fenómeno antes inaccesible? Omitir esta justificación equivale a afirmar implícitamente que se estudiaron pocos casos por falta de recursos, lo que debilita la credibilidad del trabajo independientemente de la calidad del análisis. La distinción de Stake (1995) entre caso intrínseco e instrumental es complementaria en este paso: dado que la mayoría de los estudios en ingeniería de software son de naturaleza instrumental —el caso es el medio para iluminar un mecanismo o fenómeno que trasciende ese caso particular—, la justificación del diseño debe incluir no solo por qué se eligió un caso único o múltiple, sino también de qué manera el caso seleccionado es el vehículo adecuado para revelar el mecanismo de interés en el dominio de población declarado.

Paso 4 — Declarar el tipo de generalización. En la sección de metodología —o al inicio de la discusión— el investigador debe declarar explícitamente que la generalización que el trabajo produce es analítica y no estadística, y describir el mecanismo o proceso que se generaliza. Esta declaración tiene el efecto positivo de anticipar y desactivar la crítica más común al estudio de caso: que "un solo caso no es suficiente para generalizar". La respuesta correcta a esa crítica no es agregar más casos, sino aclarar que la generalización opera por un mecanismo diferente y que ese mecanismo está apropiadamente documentado en el trabajo. Una formulación concreta de esta declaración en la sección de metodología podría ser: *"Los hallazgos de este estudio son generalizables de forma analítica, no estadística: se argumenta que el mecanismo identificado—la concentración de superficie de ataque en los supuestos de diseño implícitos— actuará en cualquier sistema que comparta las características de la población declarada, con independencia de su dominio funcional."*

Paso 5 — Delimitar los límites de la generalización. Finalmente, el investigador debe indicar explícitamente las condiciones bajo las cuales los hallazgos no serían aplicables. Estos límites pueden ser de tipo estructural —"los hallazgos no se aplican a sistemas sin límite de confianza explícito"—, de escala —"no se ha estudiado el comportamiento del proceso en sistemas con más de cien componentes"—, o de contexto —"el proceso fue evaluado sobre un sistema sintético; su comportamiento en sistemas heredados con documentación incompleta puede diferir". Declarar los límites de la generalización no debilita el trabajo; al contrario, es un indicador de rigor metodológico que fortalece la credibilidad del investigador ante el revisor especializado. Un ejemplo de delimitación bien articulada sería: *"Los resultados son aplicables a sistemas web multicapa con autenticación de múltiples roles; no se ha evaluado el proceso en sistemas*

de arquitectura monolítica ni en sistemas embebidos con restricciones de tiempo real, donde la naturaleza de la superficie de ataque difiere estructuralmente."

Ejemplos de aplicación por subdisciplina

Seguridad de software y análisis de vulnerabilidades: En estudios que evalúan metodologías de análisis de seguridad —modelado de amenazas, revisión de código, pruebas de penetración—, la población se define habitualmente en términos de la clase de sistemas sobre los que la metodología es aplicable: sistemas web, sistemas embebidos, APIs REST, plataformas de gestión de contenido. El caso seleccionado debe ser representativo de esa clase en sus características de superficie de ataque —número de puntos de entrada, tipos de actores, flujos de datos entre componentes con distinto nivel de confianza. La generalización analítica afirma que el proceso de análisis producirá resultados de calidad comparable en sistemas con características análogas, con independencia del dominio funcional del sistema.

Arquitectura y patrones de diseño: En estudios sobre la aplicación y los efectos de patrones arquitectónicos, la población se define por el tipo de problema que el patrón resuelve —sistemas con alta demanda de escalabilidad, sistemas con acoplamiento entre módulos que deben desacoplarse— y el caso es un sistema que exhibe ese problema. La generalización analítica sostiene que el patrón produce el efecto documentado en cualquier sistema que exhiba el mismo problema en un contexto estructural comparable, lo que permite al investigador no reclamar que "todos los sistemas mejorarán con este patrón", sino que "los sistemas con este perfil de problema mejorarán de esta manera".

Metodologías ágiles y procesos de desarrollo: En estudios sobre la adopción o efectividad de metodologías de proceso —Scrum, Kanban, DevOps—, la población se define en términos de las características del equipo y del contexto organizacional: tamaño del equipo, tipo de producto, modelo de financiamiento, madurez técnica. Aquí el caso de estudio puede ser un equipo o una organización, y la generalización analítica afirma que el mecanismo identificado —por ejemplo, que la implantación de revisiones de código sistemáticas reduce la densidad de defectos— actuará en equipos con características comparables. En este dominio es especialmente frecuente el diseño de casos múltiples, precisamente porque la variabilidad entre equipos es parte central del hallazgo.

Herramientas y automatización: En estudios que evalúan una herramienta de software —un analizador estático, un generador de código, un asistente de inteligencia artificial—, la población es la clase de proyectos o sistemas sobre los que la herramienta es aplicable según su propósito declarado. El caso es un proyecto o sistema concreto, seleccionado porque exhibe los tipos de artefactos o problemas que la herramienta está diseñada para procesar. La generalización analítica afirma que la herramienta producirá resultados de calidad comparable en proyectos con perfil técnico análogo, y la limitación a declarar es el perfil de proyectos en los que la herramienta no se ha probado y donde su comportamiento podría ser distinto.

Errores frecuentes en la declaración de población y muestra

La observación sistemática acumulada por el autor a lo largo de más de quince años de actividad ininterrumpida como investigador universitario, docente de metodología de investigación y director de tesis en programas de maestría y doctorado en más de diez universidades —con más de un centenar de trabajos evaluados en ese período en los niveles de grado, maestría y doctorado, tanto en calidad de revisor como de oponente formal— permite identificar cinco errores recurrentes en la declaración de población y muestra en ingeniería de software, cuya corrección mejoraría significativamente la calidad metodológica percibida por los revisores. Esta preocupación por la calidad en el reporte de los diseños de investigación empírica tiene antecedentes en la propia disciplina: las directrices de Kitchenham et al. (2007) para la realización y reporte de revisiones sistemáticas de literatura en ingeniería de software establecen, precisamente, que la declaración explícita y verificable de los criterios de selección es una condición necesaria para que cualquier proceso de investigación sea metodológicamente válido —un principio que se traslada directamente al estudio de caso cuando el investigador debe justificar la selección de su objeto de estudio ante la comunidad académica. Baltes y Ralph (2022) aportan respaldo empírico sistemático a esta observación: su revisión crítica de las prácticas de muestreo en investigación de ingeniería de software documenta que la confusión entre representatividad estadística y analítica, y el muestreo de conveniencia no fundamentado, son problemas recurrentes en la literatura del área, frecuentemente asociados a debilidades en la declaración de validez externa.

El primer error es la omisión completa de la justificación del caso. El trabajo describe el caso en detalle pero no argumenta por qué ese caso fue seleccionado sobre otros posibles. El revisor no puede evaluar si la selección fue metodológicamente apropiada o simplemente conveniente. El segundo error es la definición circular de la población. El investigador define la población como "sistemas similares al que se estudia" o "casos como el caso X", lo que hace la declaración de población vacía de contenido: si no se puede saber si un sistema pertenece a la población sin compararlo con el caso, la definición no delimita nada. El tercer error es la confusión entre generalización estadística y analítica. El investigador defiende la representatividad del caso con argumentos del tipo "PImage es un sistema web típico" sin especificar típico en qué dimensiones ni respecto de qué universo medido. Esta formulación sugiere generalización estadística sin los datos que la respaldarían. El cuarto error es la omisión de los límites de la generalización. El trabajo concluye que la metodología o herramienta estudiada "es efectiva" sin declarar bajo qué condiciones ese juicio es válido y bajo cuáles no lo sería. Esta omisión da lugar a afirmaciones más amplias de lo que el diseño respalda y es frecuentemente la

base de las observaciones de rechazo por parte de revisores. El quinto error es la justificación del tamaño de la muestra con argumentos cuantitativos. El investigador se disculpa por haber estudiado un solo caso argumentando limitaciones de tiempo o acceso, en lugar de justificar metodológicamente por qué un caso único es suficiente para el propósito del estudio. Este error inadvertidamente señala al revisor que el investigador considera que más casos habrían sido metodológicamente superiores, debilitando la credibilidad del diseño elegido.

CONCLUSIONES

El estudio de caso es un diseño de investigación metodológicamente válido y productivo para la ingeniería de software, siempre que el investigador comprenda y aplique los principios que gobiernan la selección del caso, la declaración de la población y la naturaleza de la generalización que el diseño permite. La aplicación mecánica de los conceptos de población y muestra provenientes de la investigación cuantitativa —sin la reformulación que el contexto de ingeniería requiere— produce secciones de metodología imprecisas que debilitan la credibilidad del trabajo independientemente de la calidad del análisis realizado. El artículo cumplió los cuatro objetivos específicos propuestos.

En relación con OE1, se estableció que el estudio de caso es pertinente cuando el fenómeno no puede separarse de su contexto sin perder información esencial, y que su validez opera por generalización analítica —no estadística—, lo que tiene consecuencias directas sobre cómo se diseña, justifica y reporta el trabajo. En relación con OE2, se demostró que en el contexto de la ingeniería de software la población equivale al dominio de fenómenos al que son extrapolables los hallazgos, y la muestra es el caso seleccionado mediante criterios metodológicos explícitos, reformulación que reemplaza la lógica de representatividad estadística por la de representatividad estructural y teórica. En relación con OE3, se articularon cinco criterios cuya documentación es necesaria para justificar la selección del caso —representatividad estructural, complejidad suficiente y acotada, documentación verificable, relevancia para el fenómeno y condiciones declaradas—, derivados de los marcos canónicos de Runeson y Höst (2009), Yin (2018) y Wohlin et al. (2024). En relación con OE4, se propuso una guía operativa de cinco pasos —definir el dominio, documentar criterios, justificar el diseño, declarar la generalización y delimitar los límites— como procedimiento mínimo aplicable a cualquier investigación con diseño de caso en el área.

La adopción de estas prácticas, además de mejorar la tasa de aceptación en revistas especializadas, contribuye a elevar la calidad del conocimiento acumulado en el área, en la medida en que los hallazgos publicados tienen condiciones de aplicabilidad explícitas que permiten a otros investigadores situarlos correctamente en el mapa de lo que se sabe y lo que queda por saber. Como líneas de investigación futura, se identifican: la validación empírica de la guía de cinco pasos mediante su aplicación a una muestra de secciones de metodología publicadas en revistas del área; el desarrollo de instrumentos de evaluación cuantificables derivados de los cinco criterios de selección; y la extensión del marco a diseños de investigación mixtos en los que la lógica del estudio de caso se combina con componentes de encuesta o experimento controlado. La urgencia práctica de estas recomendaciones está respaldada por la evidencia empírica reciente: la crisis de generalización documentada por Baltes y Ralph (2022) y la prevalencia de estudios mal categorizados señalada por Wohlin y Rainer (2022) confirman que los problemas que esta guía busca corregir son sistémicos y activos en la literatura de ingeniería de software, lo que le confiere una relevancia operativa inmediata y no meramente didáctica.

BIBLIOGRAFÍA

Baltes, S., & Ralph, P. (2022). Sampling in software engineering research:A critical review and guidelines. Empirical Software Engineering, 27, Article 94. https://doi.org/10.1007/s10664-021-10072-8

Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. MIS Quarterly, 11(3), 369–386. https://doi.org/10.2307/248684

Eisenhardt, K. M. (1989). Building theories from case study research. Academy of Management Review, 14(4), 532–550. https://doi.org/10.2307/258557

Flyvbjerg, B. (2006). Five misunderstandings about case-study research. Qualitative Inquiry, 12(2), 219–245. https://doi.org/10.1177/1077800405284363

Glaser, B. G., & Strauss, A. L. (1967). The discovery of grounded theory: Strategies for qualitative research. Aldine.

Kitchenham, B., Charters, S., Dyba, T., Brereton, P., Turner, M., Linkman, S., Jorgensen, M., Mendes, E., & Visaggio, G. (2007). Guidelines for performing systematic literature reviews in software engineering (Technical Report EBSE-2007-01). Keele University and University of Durham.

Lincoln, Y. S., & Guba, E. G. (1985). Naturalistic inquiry. SAGE Publications. Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. Empirical Software Engineering, 14(2), 131–164. https://doi.org/10.1007/s10664-008-9102-8

Runeson, P., Höst, M., Rainer, A., & Regnell, B. (2012). Case study research in software engineering: Guidelines and examples. Wiley. Shaw, M. (2003). Writing good software engineering research papers. Proceedings of the 25th

International Conference on Software Engineering (ICSE 2003), 726–736.https://doi.org/10.1109/ICSE.2003.120

Stake, R. E. (1995). The art of case study research. SAGE Publications. Wohlin, C., & Rainer, A. (2022). Is it a case study? — A critical analysis and guidance. Journal of Systems and Software, 192, Article 111395. https://doi.org/10.1016/j.jss.2022.111395

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2024). Experimentation in software engineering (2nd ed.). Springer. https://doi.org/10.1007/978-3-662-69306-3

Yin, R. K. (2018). Case study research and applications: Design and methods (6th ed.). SAGE Publications.

SISTEMA MÓVIL CON MODELO JERÁRQUICO HÍBRIDO PARA ESTIMAR LA SEVERIDAD FOLIAR DEL MILDIU EN QUINUA BAJO CAPTURA CONTROLADA

M.Sc. Edgar Jaldin Torrico
Posgrado SOE – UAGRM
https://orcid.org/0009-0005-5329-4908
Santa Cruz, Bolivia | edgarjaldintorrico@gmail.com

https://doi.org/10.23670 FT.2026.1.27

Recibido 20/04/2026 - Aceptado 13/05/2026

RESUMEN

La cuantificación visual de la severidad foliar del mildiu en quinua constituye un proceso laborioso y potencialmente variable entre observadores, lo que limita su estandarización en la evaluación fenotípica y el monitoreo fitopatológico. El presente estudio tuvo como objetivo evaluar experimentalmente un sistema móvil para apoyar la cuantificación automatizada de la severidad foliar del mildiu en quinua. Se desarrolló un sistema denominado QuinuApp, compuesto por una aplicación móvil, un dispositivo de captura estandarizada (BlackBox) y un estimador jerárquico híbrido basado en variables visuales interpretables. El estudio siguió un enfoque cuantitativo con diseño experimental, utilizando 772 imágenes RGB de hojas adaxiales individuales correspondientes a 92 accesiones. La severidad, expresada en una escala continua de 0 % a 100 %, fue estimada por una fitopatóloga experta y utilizada como referencia operativa. El pipeline incluyó normalización fotométrica, segmentación de la hoja principal, extracción de 55 variables clínico-texturales derivadas de imagen, clasificación auxiliar por fases biológicas y regresión híbrida con calibración isotónica, incorporación de la variable contextual de esporulación, ajuste contextual por accesión y reglas biológicas. En el conjunto de prueba, el modelo obtuvo Pearson r = 0.699, CCC = 0.694, MAE = 17.17 % y RMSE = 24.75 %, alcanzando el mejor desempeño global frente a regresión lineal, Random Forest, XGBoost y SVR. El análisis de Bland–Altman mostró un sesgo medio de ¬-0.58 %, con 50.78 % de predicciones dentro de ±10 % y 69.43 % dentro de ±20 %. El sistema propuesto mostró viabilidad técnica como herramienta de apoyo para la estimación automatizada de la severidad foliar del mildiu en quinua bajo condiciones experimentales controladas, con potencial de aplicación en fenotipado digital asistido.

Palabras clave: mildiu de la quinua, severidad foliar, fenotipado digital, visión computacional, estimación interpretable.

ABSTRACT

Visual quantification of downy mildew foliar severity in quinoa is a labor-intensive process and may vary across observers, thereby limiting its standardization in phenotypic assessment and phytopathological monitoring. The present study aimed to experimentally evaluate a mobile system designed to support the automated quantification of downy mildew foliar severity in quinoa. A system named QuinuApp was developed, comprising a mobile application, a standardized image acquisition device (BlackBox), and a hierarchical hybrid estimator based on interpretable visual features. The study followed a quantitative approach under an experimental design, using 772 RGB images of individual adaxial leaves from 92 accessions. Severity, expressed on a continuous scale from 0% to 100%, was assessed by an expert plant pathologist and used as the operational reference. The pipeline included photometric normalization, main leaf segmentation, extraction of 55 image-derived clinical-textural features, auxiliary classification by biological phases, and hybrid regression with isotonic calibration, incorporation of sporulation as a contextual variable, accession-based contextual adjustment, and biologically informed rules. On the test set, the model achieved Pearson's r = 0.699, CCC = 0.694, MAE = 17.17%, and RMSE = 24.75%, attaining the best overall performance compared with linear regression, Random Forest, XGBoost, and SVR. Bland–Altman analysis showed a mean bias of -0.58%, with 50.78% of predictions within ±10% and 69.43% within ±20%. The proposed system demonstrated technical feasibility as a supportive tool for the automated estimation of downy mildew foliar severity in quinoa under controlled experimental conditions, with potential applicability in assisted digital phenotyping.

Keywords: quinoa downy mildew, foliar severity, digital phenotyping, computer vision, interpretable estimation.

INTRODUCCIÓN

El mildiu velloso, causado por Peronospora variabilis, constituye una de las principales amenazas fitopatológicas para el cultivo de quinua (Chenopodium quinoa Willd.) en la región andina, especialmente bajo condiciones de alta humedad relativa y temperaturas moderadas (Colque-Little et al., 2021; Danielsen et al., 2003). La cuantificación de la severidad foliar es un componente central para la evaluación fenotípica, la selección de accesiones resistentes y el seguimiento epidemiológico de la enfermedad. Sin embargo, la estimación visual convencional y los enfoques basados en fotografía digital pueden presentar variabilidad entre observadores, diferencias asociadas a las condiciones de captura y limitaciones de reproducibilidad, afectando la comparabilidad de los resultados experimentales (Bock et al., 2010).

A diferencia de la abundante literatura orientada a la detección o clasificación categórica de enfermedades vegetales, los estudios que abordan la cuantificación continua de severidad foliar expresada como porcentaje de tejido afectado constituyen un cuerpo bibliográfico más restringido y metodológicamente heterogéneo. La estimación continua exige modelar relaciones no lineales entre coloración, textura, distribución espacial de los síntomas y proporción de tejido afectado, mientras que muchos enfoques recientes basados en aprendizaje profundo formulan el problema como clasificación binaria u ordinal, reduciendo la sensibilidad para diferenciar gradientes intermedios de daño (Barbedo, 2016; Bock et al., 2020; Li et al., 2021). Aunque existen antecedentes de estimación automatizada de severidad foliar en otros patosistemas mediante aplicaciones móviles RGB, sensado proximal, imágenes multiespectrales UAV o segmentación profunda, estos trabajos difieren sustancialmente en plataforma de captura, grado de automatización, escala de anotación, métricas reportadas y condiciones experimentales, lo que limita su comparabilidad directa (Chemura et al., 2018; Duarte-Carvajalino et al., 2018; Gao et al., 2021; Goncalves et al., 2021; Pethybridge & Nelson, 2015)

Aunque existen antecedentes de estimación automatizada de severidad foliar en otros patosistemas, la evidencia disponible para el mildiu en quinua se concentra principalmente en evaluaciones visuales, escalas ordinales, estimaciones agronómicas de severidad y estudios de resistencia genética o comportamiento varietal. En contraste, son escasos los trabajos que formulen la severidad foliar del mildiu en quinua como una variable continua de 0 % a 100 % estimada mediante modelos computacionales supervisados. Esta brecha metodológica es relevante porque la cuantificación continua exige no solo imágenes foliares trazables, sino también anotaciones expertas en escala porcentual, condiciones de captura consistentes y un modelo capaz de preservar gradientes intermedios de daño.

En este marco, QuinuApp se propone como una prueba de concepto experimental orientada a la estimación continua, interpretable y computacionalmente asistida de la severidad foliar en el patosistema mildiu–quinua. Frente a esta brecha, se propone QuinuApp, un sistema móvil orientado a apoyar la cuantificación automatizada de la severidad foliar del mildiu en quinua bajo condiciones de captura controlada. El sistema integra una aplicación móvil, un diseño de dispositivo de adquisición estandarizada denominado BlackBox y un modelo predictivo basado en variables visuales interpretables. A diferencia de enfoques exclusivamente dependientes de representaciones latentes, la propuesta utiliza descriptores asociados a manifestaciones visibles de la enfermedad, como cambios cromáticos, patrones texturales y rasgos morfológicos de la hoja. Esta orientación busca favorecer la trazabilidad de las predicciones y su potencial aceptación en contextos aplicados, donde la confianza del usuario depende no solo del desempeño numérico, sino también de la posibilidad de relacionar la estimación con evidencias visuales observables (Kamilaris & Prenafeta-Boldu, 2018).

El presente estudio se diferencia de una investigación metodológica previa centrada en la estimación interpretable de severidad foliar continua en quinua mediante visión computacional, desarrollada sobre un subconjunto de imágenes y sin integración móvil. A diferencia de ese trabajo, este manuscrito evalúa un sistema ampliado denominado QuinuApp, incorpora un conjunto de datos mayor, variables contextuales de esporulación y accesión, comparación con modelos de referencia, análisis por rangos de severidad, validación complementaria por accesión y una arquitectura propuesta de despliegue móvil con BlackBox.

El objetivo de este estudio fue evaluar experimentalmente el desempeño de QuinuApp como sistema de apoyo para la cuantificación continua de la severidad foliar del mildiu en quinua. Para ello, se analizó un flujo de trabajo que combina procesamiento de imágenes, extracción de variables visuales interpretables y evaluación comparativa frente a modelos de referencia, considerando el desbalance natural de los datos entre fases de severidad.

METODOLOGÍA

La investigación se desarrolló bajo un enfoque cuantitativo y un diseño experimental aplicado, orientado al desarrollo y evaluación de un sistema computacional para estimar de forma continua la severidad foliar del mildiu en quinua (Chenopodium quinoa Willd). El sistema integró adquisición estandarizada de imágenes, procesamiento computacional, modelado supervisado e inferencia móvil local bajo condiciones controladas.

Población y muestra

El material biológico provino de un ensayo controlado realizado en los invernaderos de la Universidad de Copenhague, Dinamarca, con 600 plantas de quinua correspondientes a 258 accesiones genéticas, sometidas a infección por Peronospora variabilis bajo condiciones reguladas de temperatura, humedad

y fotoperiodo. A partir del material generado, se conformó un conjunto final de 772 imágenes RGB de hojas adaxiales individuales, procedentes de 92 accesiones, cada una asociada a una anotación experta de severidad en escala continua de 0 % a 100 %, una accesión de pertenencia y un registro de esporulación. Los criterios de inclusión fueron: calidad visual suficiente para segmentación, correspondencia válida entre imagen y anotación, y cobertura de distintos niveles de severidad foliar.

Para fines analíticos, la severidad se organizó en tres fases biológicas: fase A o infección temprana, de 0–25 %, con 133 imágenes; fase B o expansión moderada, de 26–75 %, con 195 imágenes; y fase C o necrosis terminal, de 76–100 %, con 444 imágenes. Esta distribución evidenció un desbalance estructural hacia niveles altos de severidad.

El conjunto de datos se dividió mediante partición estratificada a nivel de imagen en 579 imágenes para entrenamiento y 193 para prueba. Dado que imágenes de una misma accesión podían estar presentes en ambos subconjuntos, los resultados se interpretan como desempeño bajo partición por imagen y no como validación estricta de generalización hacia accesiones no observadas.

Herramientas tecnológicas utilizadas

El pipeline fue desarrollado en Python 3.x, empleando OpenCV para procesamiento de imágenes, NumPy y pandas para manipulación de datos, scikit-learn para modelado supervisado, calibración y evaluación, SciPy para funciones estadísticas, joblib para serialización, matplotlib para visualización y XGBoost como uno de los modelos comparativos. El modelo entrenado fue exportado e integrado en una aplicación móvil Android denominada QuinuApp, diseñada para operar localmente sin conexión a internet.

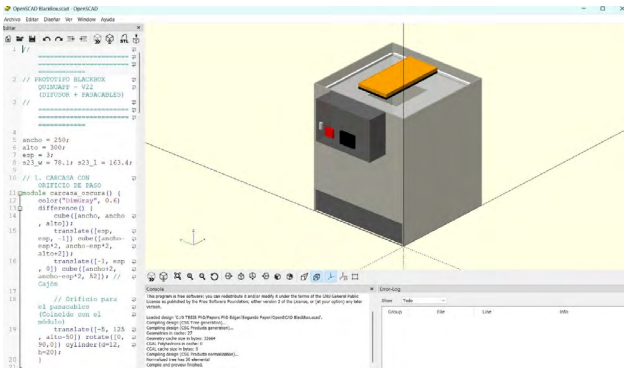
Para la fase de despliegue operativo se diseñó una arquitectura móvil compuesta por la aplicación Android QuinuApp, un teléfono Samsung Galaxy S23 Ultra como plataforma prevista de ejecución local y un dispositivo físico de captura estandarizada denominado BlackBox. Este dispositivo fue modelado paramétricamente en OpenSCAD e incorporó, a nivel de diseño, una carcasa opaca, iluminación perimetral, difusor interno, cajón de posicionamiento y soporte fijo para el teléfono móvil, como se muestra en la Figura 1.

La infraestructura fue concebida para reducir la variabilidad asociada a iluminación, fondo, distancia y ángulo de captura durante futuras adquisiciones controladas.

No obstante, en el presente estudio, la evaluación experimental se limitó al desempeño del pipeline computacional sobre el conjunto de imágenes disponible; por tanto, no se realizó una validación de campo del sistema integrado teléfono–BlackBox ni una evaluación operativa del algoritmo ejecutándose directamente en el dispositivo móvil.

Figura 1

Diseño paramétrico de la BlackBox propuesta para la implementación operativa de QuinuApp en condiciones de captura controlada



***Nota.** La figura muestra el diseño paramétrico de la BlackBox desarrollado en OpenSCAD como propuesta de infraestructura para captura controlada. Se representan la carcasa opaca, el sistema de iluminación perimetral superior, el difusor interno con abertura central, el módulo lateral de control, el cajón extraíble para posicionamiento de la muestra y el soporte superior para el teléfono móvil. El diseño fue concebido para estandarizar futuras capturas de imágenes y reducir la variabilidad asociada a iluminación, fondo, distancia y ángulo. En este estudio, la BlackBox se reporta como diseño tecnológico propuesto, no como dispositivo validado en campo.*

Proceso metodológico

El flujo metodológico del sistema se estructuró en etapas secuenciales que abarcan la preparación del conjunto de datos, el preprocesamiento y segmentación de las imágenes, la extracción de variables visuales interpretables, el modelado predictivo, la calibración de las estimaciones y la evaluación comparativa frente a modelos de referencia. Una versión sintética de este flujo se presenta en la Figura 2.

Figura 2

Flujo metodológico del sistema propuesto para la estimación automatizada de severidad foliar en quinua

***Nota.** El diagrama sintetiza las etapas principales del pipeline computacional: entrada de imágenes RGB, preprocesamiento fotométrico, segmentación de la hoja principal, extracción de variables clínico-texturales, estimación predictiva, calibración y evaluación comparativa.*

Procesamiento de imágenes y extracción de variables

Cada imagen fue sometida a normalización fotométrica mediante balance de blancos Gray-World y ecualización adaptativa del contraste sobre el canal L* del espacio CIELAB. Posteriormente, la hoja principal fue segmentada mediante una estrategia híbrida basada en exclusión de fondo, umbralización en espacios de color y selección geométrica del componente más próximo al centro de la imagen. En casos de segmentación inestable, se aplicó un procedimiento de respaldo basado en GrabCut. Sobre la región foliar segmentada se extrajeron 55 variables derivadas de imagen, agrupadas en cuatro bloques, fracciones de área sintomática, descriptores morfológicos, variables

cromáticas absolutas y descriptores texturales. Las fracciones sintomáticas representaron manifestaciones visibles de clorosis, necrosis, blanqueamiento o esporulación visible y necrosis marrón difusa. Los descriptores morfológicos resumieron propiedades espaciales de los síntomas; las variables cromáticas capturaron diferencias de color asociadas al daño foliar; y los descriptores texturales se calcularon mediante histogramas de Patrones Binarios Locales sobre la región segmentada.

Modelado predictivo

La severidad foliar fue tratada como una variable continua. No obstante, el modelo incorporó una discretización auxiliar en tres fases biológicas para guiar la especialización del estimador, infección temprana, expansión moderada y necrosis terminal. El núcleo predictivo se estructuró como un estimador jerárquico híbrido compuesto por un clasificador auxiliar de fase, regresores especializados por intervalo de severidad y un regresor global de respaldo. El clasificador auxiliar estimó la probabilidad de pertenencia de cada imagen a cada fase biológica. Estas probabilidades se utilizaron como mecanismo de ponderación para combinar las predicciones de los regresores especializados.

En paralelo, un regresor global estimó la severidad sobre todo el rango continuo de respuesta. La predicción final integró la salida local especializada y la salida global mediante una mezcla dinámica dependiente de la confianza del clasificador. Todas las variables numéricas fueron estandarizadas utilizando exclusivamente los datos de entrenamiento. Para compensar el desbalance de la muestra, se aplicaron pesos diferenciados durante el entrenamiento, asignando mayor relevancia a fases y rangos de severidad menos representados. Las predicciones fueron refinadas mediante calibración isotónica, ajuste contextual por accesión y reglas biológicas de postajuste. El ajuste por accesión se utilizó como información contextual histórica; por tanto, sus resultados no deben interpretarse como inferencia basada exclusivamente en la imagen.

Validación y modelos de referencia

El desempeño del sistema se evaluó sobre el conjunto de prueba mediante coeficiente de correlación de Pearson, coeficiente de concordancia de Lin, error absoluto medio, raíz del error cuadrático medio y proporción de predicciones dentro de ±10 y ±20 puntos porcentuales respecto al valor observado. La concordancia entre severidad observada y estimada se examinó adicionalmente mediante análisis de Bland–Altman, reportando sesgo sistemático y límites de acuerdo. El sistema propuesto se comparó con cuatro modelos de referencia entrenados sobre la misma matriz de predictores: regresión lineal múltiple, Random Forest, XGBoost y SVR con kernel RBF. Todos los modelos fueron evaluados bajo el mismo esquema de partición y con el mismo conjunto de métricas. Como referencia descriptiva adicional, se analizó la repetibilidad intraobservador en 24 imágenes independientes evaluadas en dos momentos por una especialista en fitopatología. Este análisis se consideró únicamente como evidencia de variabilidad humana en la estimación visual, no como comparación directa entre desempeño humano y algorítmico.

RESULTADOS

Desempeño comparativo del pipeline híbrido

El pipeline híbrido QuinuApp mostró el mejor desempeño global entre los modelos evaluados, al combinar el menor error absoluto medio, la mayor concordancia de Lin, el menor sesgo promedio y la mayor proporción de predicciones dentro de los márgenes de tolerancia de ±10 % y ±20 % sobre el conjunto de prueba. La Tabla 1 resume el desempeño comparativo del pipeline híbrido y de los modelos de referencia. QuinuApp alcanzó un MAE de 17.17 %, un CCC de 0.694 y un sesgo medio de -0.58 % en el análisis de Bland–Altman. Además, obtuvo el mayor porcentaje de predicciones dentro de ±10 puntos porcentuales respecto al valor observado, con 50.78 %, y dentro de ±20 puntos porcentuales, con 69.43 %. Estos resultados indican que el pipeline propuesto presentó la combinación más balanceada entre error absoluto, concordancia y bajo sesgo global.

Tabla 1

Comparación de las métricas de desempeño del pipeline híbrido QuinuApp y modelos de referencia en el conjunto de prueba

Modelo	Pearson r	CCC	MAE (%)	RMSE (%)	Sesgo BA (%)	Dentro ±10%	Dentro ±20%
Pipeline Híbrido	0.699	0.694	17.17	24.75	-0.58	50.78	69.43
Random Forest	0.707	0.561	19.34	24.71	0.25	33.16	63.73
XGBoost	0.692	0.614	17.95	24.34	1.52	41.97	65.80
SVR (RBF)	0.638	0.505	18.87	26.76	5.93	45.08	66.32
Regresión Lineal	0.622	0.568	19.56	26.21	0.26	40.93	61.14

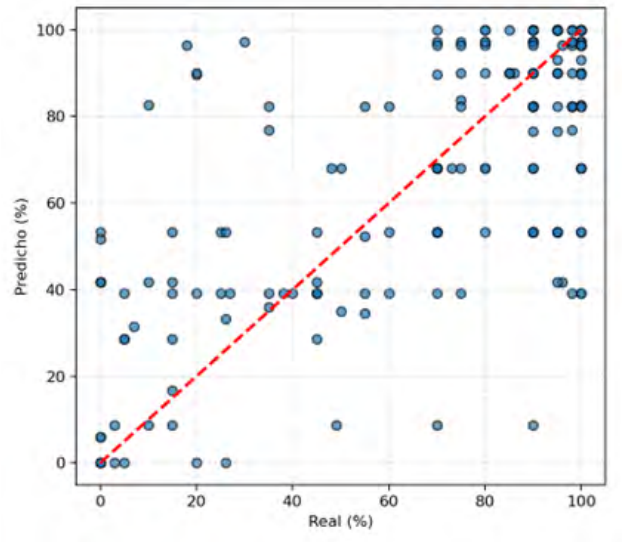
Nota. CCC = coeficiente de concordancia y correlación de Lin; MAE = error absoluto medio; RMSE = raíz del error cuadrático medio; Sesgo BA = sesgo medio estimado mediante el análisis de Bland–Altman; Dentro ±10 % y Dentro ±20 % representan la proporción de predicciones cuya diferencia absoluta respecto al valor observado fue ≤10 y ≤20 puntos porcentuales.

Aunque Random Forest obtuvo el mayor coeficiente de correlación de Pearson (r = 0.707), su concordancia global fue menor que la de QuinuApp (CCC = 0.561 frente a 0.694) y presentó una proporción sustancialmente inferior de predicciones dentro de ±10 %. XGBoost alcanzó el menor RMSE (24.34 %) y el segundo mejor MAE (17.95 %), pero no superó al pipeline híbrido en concordancia, sesgo ni proporción de predicciones dentro de los márgenes de tolerancia. En conjunto, estos resultados muestran que el desempeño de QuinuApp no depende de una única métrica aislada, sino de un equilibrio más consistente entre precisión, concordancia y estabilidad global.

La relación entre severidad observada y severidad predicha se presenta en la Figura 3. La distribución de los puntos muestra una asociación positiva general entre ambos valores, aunque con dispersión apreciable respecto de la diagonal de identidad. Esta dispersión fue más evidente en rangos intermedios de severidad, donde la transición entre fases biológicas y la superposición visual de síntomas dificultan la estimación precisa. Por tanto, el modelo preservó la tendencia global de la variable objetivo, pero mantuvo variabilidad residual a nivel de observaciones individuales.

Figura 3

Severidad observada frente a severidad predicha por el pipeline híbrido QuinuApp



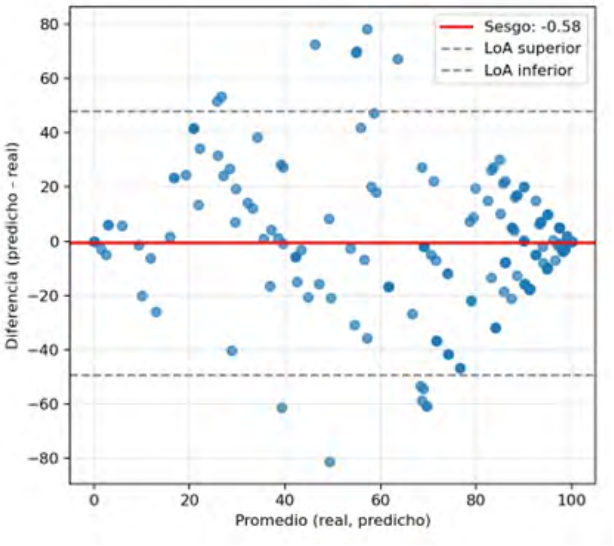
Nota. La figura muestra la relación entre la severidad foliar observada y la severidad estimada por el pipeline híbrido QuinuApp sobre el conjunto de prueba. La línea discontinua representa la identidad y = x, correspondiente a concordancia perfecta entre valores observados y estimados. La dispersión de los puntos respecto a esta referencia permite visualizar la magnitud del error de predicción a lo largo del rango de severidad.

Análisis de concordancia Bland-Altman

El análisis de Bland–Altman permitió examinar la estructura del error del pipeline híbrido en función de la magnitud promedio de la severidad foliar (Bland & Altman, 1986). Como se observa en la Figura 4, el modelo presentó un sesgo medio de -0.58 %, lo que indica ausencia de sobreestimación o subestimación sistemática relevante a nivel global. Sin embargo, los límites de acuerdo fueron amplios, entre -49.20 % y 48.04 %, evidenciando variabilidad considerable en la predicción individual.

Figura 4

Análisis de Bland–Altman entre severidad observada y severidad estimada por el pipeline híbrido QuinuApp



Nota. La figura presenta el análisis de Bland–Altman correspondiente a las predicciones del pipeline híbrido QuinuApp sobre el conjunto de prueba (n = 193). El eje horizontal representa el promedio entre severidad observada y severidad estimada, mientras que el eje vertical muestra la diferencia entre ambas. La línea central indica el sesgo medio (-0.58 %) y las líneas discontinuas superiores e inferiores representan los límites de acuerdo (-49.20 % a 48.04 %).

La lectura conjunta de las Figuras 3 y 4 confirma que el pipeline híbrido reproduce adecuadamente la tendencia general de la severidad foliar y mantiene bajo sesgo global. No obstante, la amplitud de los límites de acuerdo indica que la precisión individual sigue siendo variable, especialmente en imágenes con severidad intermedia o con manifestaciones sintomáticas visualmente heterogéneas.

Este comportamiento es coherente con la complejidad de cuantificar severidad foliar continua a partir de imágenes RGB, donde la morfología de la hoja, la distribución espacial de los síntomas y la subjetividad de la anotación experta pueden introducir variabilidad adicional.

Una ampliación del dataset con mayor número de muestras reales y mejor equilibrio entre rangos podría mejorar la estabilidad del aprendizaje y reducir la incertidumbre en los intervalos menos representados, aunque dicha mejora debe verificarse experimentalmente y no asumirse como consecuencia automática del balanceo.

En síntesis, QuinuApp alcanzó el desempeño global más balanceado entre los modelos evaluados, con menor MAE, mayor CCC, bajo sesgo promedio y mayor proporción de predicciones dentro de márgenes de error clínicamente interpretables. Sin embargo, los resultados también evidencian que, aunque el modelo presenta bajo sesgo global, persiste variabilidad en la predicción individual, particularmente en rangos intermedios de severidad.

Desempeño por rango de severidad

El análisis estratificado por rangos mostró que el desempeño del pipeline híbrido QuinuApp no fue homogéneo a lo largo de la escala de severidad. Como se observa en la Tabla 2, el sistema presentó mejor comportamiento en el rango alto de severidad, con el menor MAE (14.15 %), el menor RMSE (21.44 %) y la mayor proporción de predicciones dentro de ±10 % (60.36 %) y ±20 % (80.18 %). En contraste, el rango bajo de severidad mostró el desempeño más débil, con MAE = 26.60 %, RMSE = 35.12 % y solo 33.33 % de predicciones dentro de ±10 %. El rango medio presentó un comportamiento intermedio, con MAE = 17.68 %, RMSE = 23.25 % y CCC = 0.401.

Tabla 2

Desempeño del pipeline híbrido QuinuApp por rango de severidad

Rango de severidad	n	MAE (%)	RMSE (%)	CCC	Dentro ±10 (%)	Dentro ±20 (%)	Dentro ±20%
Bajo, 0–25 %	33	26.603	35.115	0.138	33.333	45.455	69.43
Medio, 26–75 %	49	17.676	23.247	0.401	40.816	61.224	66.32
Alto, 76–100 %	111	14.150	21.440	0.014	40.816	80.180	61.14

Nota. La columna n indica el número de imágenes del conjunto de prueba incluidas en cada rango, definido según la severidad observada. Los rangos fueron establecidos como: bajo = 0–25 %, medio = 26–75 % y alto = 76–100 %. MAE = error absoluto medio; RMSE = raíz del error cuadrático medio; CCC = coeficiente de concordancia de Lin. Dado que el tamaño muestral difiere entre rangos especialmente entre el rango bajo (n = 33) y el rango alto (n = 111), las métricas deben interpretarse considerando la representatividad desigual de cada intervalo.

Estos resultados indican que la utilidad inmediata del sistema se concentra principalmente en escenarios de severidad alta, donde la presencia extensa de síntomas facilita la estimación del daño foliar. En cambio, el desempeño más débil en el rango de 0–25 % MAE = 26.60 % y RMSE = 35.12 % sugiere que el pipeline todavía presenta limitada sensibilidad para capturar manifestaciones tempranas de la enfermedad.

Este comportamiento debe interpretarse considerando el desbalance estructural del conjunto de datos, una condición frecuente en estudios fitopatológicos y biológicos, donde la progresión natural de la enfermedad no produce necesariamente una distribución uniforme de observaciones entre fases. En este estudio, el conjunto de prueba incluyó 33 imágenes en severidad baja, 49 en severidad media y 111 en severidad alta, lo que favorece una estimación más estable en el rango de daño avanzado y limita la robustez estadística de los rangos menos representados.

El rango medio presentó un comportamiento intermedio, con MAE = 17.68 %, RMSE = 23.25 % y CCC = 0.401, coherente con la dificultad de modelar transiciones graduales entre clorosis y esporulación visible. Debe señalarse que el CCC bajo en el rango alto no contradice el menor MAE observado en ese intervalo. Al restringir el análisis a severidades elevadas, la variabilidad interna de los valores observados puede disminuir, reduciendo la concordancia de Lin aun cuando el error absoluto sea menor.

En conjunto, el análisis por rangos evidencia que QuinuApp no debe interpretarse todavía como un estimador homogéneo en toda la escala de severidad, sino como un sistema con desempeño más estable en daño avanzado y con necesidad de refinamiento específico para infecciones tempranas y transiciones intermedias.

Validación complementaria por accesión

Como análisis adicional de robustez metodológica, se repitió la evaluación del pipeline híbrido mediante partición estricta por accesión, en un esquema de tres conjuntos (entrenamiento / validación / prueba) con proporciones aproximadas 60/15/25. Bajo este diseño, las 92 accesiones se asignaron de forma exclusiva a una única partición (55 train, 14 val, 23 test), sin accesiones compartidas entre subconjuntos.

La estratificación se realizó sobre la clase de severidad dominante de cada accesión para garantizar representación de las tres fases biológicas en cada partición. El conjunto de validación cumplió una función específica: ajustar la calibración isotónica y los priors contextuales sin exponer el conjunto de prueba a ningún componente del pipeline.

Los resultados evidenciaron una degradación sustantiva del desempeño respecto a la partición por imagen Tabla 3. El MAE aumentó de 17.17% a 23.01% (+5.84 pp), el CCC descendió de 0.694 a 0.593, y las tasas de acuerdo dentro de ±10% y ±20% se redujeron a 25.82% y 42.86%, respectivamente, aproximadamente la mitad de los valores obtenidos con el esquema por imagen.

El sesgo Bland–Altman pasó de -0.58 pp a -9.66 pp, indicando una subestimación sistemática del modelo cuando las accesiones de prueba carecen de información contextual previa.

Tabla 3

Comparación entre partición por imagen y partición por accesión

Esquema de partición	Pearson r	CCC	MAE (%)	RMSE (%)	Sesgo BA (%)	Dentro ±10% (%)	Dentro ±20% (%)
Por imagen	0.699	0.694	17.17	24.75	-0.58	50.78	69.43
Por accesión	0.647	0.593	23.01	28.09	-9.66	25.82	42.86

Nota. MAE = error absoluto medio; RMSE = raíz del error cuadrático medio; CCC = coeficiente de concordancia de Lin; Sesgo BA = sesgo de Bland–Altman (pp); ±10 % y ±20 % = porcentaje de predicciones dentro de 10 y 20 puntos porcentuales del valor real. La partición por imagen emplea muestreo estratificado por clase (75/25, semilla 42). La partición por accesión asigna cada accesión exclusivamente a un conjunto (train 60 % / val 15 % / test 25 %), estratificando por clase dominante; 55 accesiones en entrenamiento, 14 en validación y 23 en prueba.

El análisis por rango de severidad en la validación complementaria por accesión reveló un patrón de sesgo asimétrico. En el rango bajo, de 0–25 %, el modelo tendió a sobreestimar la severidad, con un sesgo medio de +26.54 puntos porcentuales (n = 33).

En el rango medio, de 26–75 %, presentó una subestimación moderada, con un sesgo medio de -9.77 puntos porcentuales (n = 46; MAE = 18.65 %). En el rango alto, de 76–100 %, la subestimación fue más marcada, con un sesgo medio de -21.21 puntos porcentuales (n = 103). Este patrón sugiere una regresión de las predicciones hacia valores intermedios cuando el modelo se evalúa sobre accesiones no observadas durante el entrenamiento.

A diferencia de la partición principal por imagen, cuyo conjunto de prueba estuvo conformado por 193 imágenes, la validación por accesión generó un conjunto de prueba de 182 imágenes, debido a que la asignación se realizó por accesiones completas y no por imágenes individuales. En este escenario, las predicciones finales se concentraron en un número reducido de salidas, se registraron solo 14 valores predichos únicos en las 182 imágenes de prueba.

Este comportamiento evidencia una pérdida de resolución continua bajo el esquema inter-accesión y puede estar asociado a la ausencia de priors específicos para las accesiones de prueba reduce el efecto estabilizador del ajuste contextual por accesión.

DISCUSIÓN

El principal hallazgo de este estudio fue que el pipeline híbrido propuesto alcanzó el desempeño global más balanceado entre los modelos evaluados, registrando simultáneamente el menor error absoluto medio (MAE = 17.17 %), el mayor coeficiente de concordancia y correlación de Lin (CCC = 0.694) y el sesgo medio más cercano a cero en el análisis de Bland–Altman (-0.58 %).

En relación con el objetivo del estudio, estos hallazgos indican que la combinación de adquisición estandarizada, representación interpretable de síntomas y estimación jerárquica híbrida constituye una aproximación metodológicamente viable para apoyar la cuantificación continua de severidad foliar en quinua bajo condiciones controladas.

Este desempeño (MAE = 17.17 %; CCC = 0.694) sugiere, además, que la especialización local por subrangos de la variable de salida puede mejorar la estabilidad de la predicción frente a distribuciones desbalanceadas y relaciones no lineales entre atributos y respuesta, en consonancia con principios de aprendizaje supervisado híbrido reportados en la literatura.

La relación observada entre severidad real y severidad predicha confirma que el sistema conserva la tendencia general de la variable objetivo, aunque con dispersión apreciable en distintos intervalos del rango. Esta característica se refuerza con el análisis de Bland–Altman, donde el sesgo global bajo contrasta con límites de acuerdo amplios. En términos prácticos, ello sugiere que el sistema reduce la desviación sistemática promedio, pero todavía mantiene variabilidad residual a nivel de observaciones individuales.

Este resultado es consistente con la complejidad del problema abordado, en el que convergen heterogeneidad morfológica entre accesiones, transición gradual entre fases biológicas y solapamiento visual entre síntomas. En consecuencia, QuinuApp debe interpretarse como una prueba de concepto experimental con desempeño prometedor en dominio controlado, pero aún insuficiente para uso autónomo en campo abierto sin validación externa adicional.

Comparación con antecedentes de cuantificación continua de severidad

Para contextualizar el desempeño relativo de QuinuApp, la Tabla 4 sintetiza antecedentes seleccionados de cuantificación continua o porcentual de severidad foliar mediante visión computacional o sensado proximal.

La comparación debe interpretarse con cautela, debido a diferencias entre cultivos, patosistemas, sensores, protocolos de captura, escalas de anotación, particiones experimentales y métricas reportadas.

En particular, no todos los estudios informan simultáneamente MAE, RMSE y CCC, por lo que la tabla no constituye un ranking directo de desempeño, sino una referencia metodológica para ubicar la contribución del presente trabajo.

Tabla 4

Antecedentes seleccionados de cuantificación continua de severidad foliar mediante visión computacional o sensado proximal en patosistemas comparables

Studio	Patosistema	Plataforma / sensor	Método	Métricas reportadas	Limitaciones principales
Pethybridge & Nelson (2015)	Seis enfermedades foliares	Smartphone, RGB	Umbralización de color asistida por usuario (Leaf Doctor)	$R^2 \geq 0.79$; Cb ≥ 0.959	Dependencia del usuario para selección de píxeles; sin inferencia automática completa
Chemura et al. (2018)	Roya del café	Espectrorradiómetro portátil	RBF-PLS sobre bandas espectrales	$R^2 = 0.92$; RMSE = 6.1 %	Condiciones controladas; tamaño muestral limitado; sensor especializado
Duarte-Carvajalino et al. (2018)	Tizón tardío de papa	UAV multiespectral	MLP, SVR, Random Forest, CNN	MAE = 13.63–19.00 %; RMSE = 18.00–28.03 %; $R^2 = 0.28–0.80$	Alta variabilidad entre modelos; requiere UAV y captura multiespectral
Gao et al. (2021)	Tizón tardío de papa	RGB de campo	Encoder–decoder para segmentación semántica	$R^2 = 0.655$; IoU lesión = 0.386	Dificultad en lesiones pequeñas; severidad evaluada en rango limitado
Gonçalves et al. (2021)	Tres patosistemas foliares	RGB, hojas individuales	FPN, U-Net, DeepLabv3+	Alto acuerdo con referencia; R^2 alto según arquitectura	Sobreestimación por clasificación errónea de píxeles sanos

***Nota.** La comparación realizada es contextual y no representa un ranking directo, debido a diferencias entre estudios en cultivo, patosistema, sensores, protocolos de captura, escalas de anotación, particiones experimentales y métricas reportadas. Además, no todos los trabajos informan simultáneamente MAE, RMSE y CCC. La inclusión de patosistemas distintos al mildiu en quinua responde a la escasa disponibilidad de estudios comparables sobre cuantificación continua de severidad foliar en este cultivo, situación asociada a la complejidad de generar bases de datos anotadas, condiciones controladas de captura e inoculación y disponibilidad de especialistas en fitopatología.*

La inclusión de estudios en papa, café, trigo y otros patosistemas foliares responde a la limitada disponibilidad de antecedentes directamente comparables sobre cuantificación continua del mildiu en quinua. Esta limitación no es solo bibliográfica, sino también experimental y metodológica. Como señalan (Faye et al., 2023), la estimación automatizada de severidad vegetal constituye un campo heterogéneo, en el que coexisten enfoques basados en procesamiento de imágenes, aprendizaje automático clásico y aprendizaje profundo, con diferencias sustanciales en sensores, escalas de severidad, protocolos de captura y métricas de evaluación. Además, los autores identifican desafíos persistentes relacionados con la calidad de imagen, la segmentación de síntomas pequeños o tempranos y la dependencia de etiquetas generadas por especialistas. En este contexto, generar un conjunto de datos para severidad foliar continua en quinua exige ensayos controlados, inoculación o seguimiento fitopatológico consistente, imágenes con trazabilidad experimental y anotaciones expertas en escala porcentual. Por ello, la comparación presentada debe interpretarse como contextual y no como equivalencia directa de desempeño.

El aporte del presente estudio reside en proporcionar una base inicial, reproducible e interpretable para un patosistema aún poco representado en la literatura computacional. Frente a estos antecedentes, QuinuApp alcanzó MAE = 17.17 %, RMSE = 24.75 % y CCC = 0.694 bajo partición por imagen. Estos valores muestran una viabilidad técnica inicial, pero también evidencian que el sistema todavía presenta error individual relevante, especialmente si se compara con estudios desarrollados bajo condiciones más controladas, con sensores especializados o con protocolos de segmentación asistida. Por tanto, la contribución principal del presente estudio no radica en afirmar superioridad métrica general frente a todos los antecedentes, sino en proponer una arquitectura específica para el patosistema mildiu–quinua, basada en variables clínico-texturales interpretables, modelado jerárquico híbrido, calibración postpredictiva y una propuesta de captura estandarizada. En quinua, la evaluación del mildiu causado por Peronospora variabilis se ha abordado principalmente desde enfoques fitopatológicos y agronómicos, mediante estimaciones visuales, escalas discretas de severidad, categorías ordinales o mediciones orientadas a caracterizar resistencia y comportamiento varietal. Este tipo de evaluación ha sido utilizado para describir germoplasma, comparar líneas de mejoramiento y estudiar la respuesta de accesiones frente al patógeno (Colque-Little et al., 2021; Danielsen et al., 2003; Stanschewski et al., 2021). En este contexto, el presente trabajo contribuye a trasladar la cuantificación de severidad hacia un enfoque computacional continuo, trazable y orientado al apoyo del fenotipado digital.

La orientación hacia variables interpretables constituye un componente metodológico relevante del sistema. La literatura sobre inteligencia artificial en agricultura

ha señalado que, aunque los modelos basados en aprendizaje profundo pueden alcanzar alto desempeño predictivo, su limitada transparencia puede dificultar la trazabilidad y la confianza en contextos aplicados (Kamilaris & Prenafeta-Boldu, 2018). En este sentido, el uso de fracciones sintomáticas, descriptores morfológicos, variables cromáticas y patrones texturales ofrece una ventaja operativa, ya que permite relacionar la predicción con manifestaciones visibles de la hoja y no únicamente con representaciones latentes de difícil interpretación. No obstante, esta característica debe entenderse como una forma de interpretabilidad basada en variables de entrada, no como una explicación causal completa del proceso patológico ni de la decisión del modelo. Estudios recientes evidencian una tendencia hacia sistemas móviles de fenotipado digital y diagnóstico asistido por inteligencia artificial, como StripeRust-Pocket y GranoScan, orientados a la detección de amenazas en trigo. En este contexto, QuinuApp se alinea con el desarrollo de herramientas móviles para apoyo fitopatológico; sin embargo, su validación operativa en campo aún no ha sido completada. Por ello, el estudio debe interpretarse como una prueba de concepto experimental en condiciones controladas y no como una validación definitiva en escenarios reales.

Entre las limitaciones del estudio debe señalarse, en primer lugar, el desbalance estructural del conjunto de datos, con predominio de observaciones en la fase alta severidad. Este comportamiento es frecuente en estudios biológicos y fitopatológicos, donde la progresión de la enfermedad no necesariamente produce una distribución uniforme entre fases. Aunque el efecto fue parcialmente mitigado mediante ponderación adaptativa, calibración y modelado jerárquico, la distribución de la muestra continúa condicionando la precisión en los rangos menos representados, especialmente en severidades bajas e intermedias. En segundo lugar, la evaluación principal se realizó mediante partición estratificada a nivel de imagen. Esto implica que imágenes procedentes de una misma accesión podían estar representadas tanto en entrenamiento como en prueba; por tanto, los resultados de este esquema deben interpretarse como evidencia de desempeño in-domain y no como validación estricta de generalización hacia accesiones no observadas. Para abordar esta limitación, se incorporó una validación complementaria por accesión, en la cual cada accesión fue asignada exclusivamente a entrenamiento, validación o prueba. Esta evaluación más exigente evidenció una disminución del desempeño, lo que indica que la generalización intergenotípica del sistema todavía requiere fortalecimiento mediante mayor volumen de datos, mejor balance entre rangos de severidad y validación externa con accesiones independientes.

En tercer lugar, el modelo incorporó información contextual mediante un prior por accesión y la variable de esporulación dentro de la matriz de entrada. Estos componentes pueden contribuir a estabilizar la predicción cuando existe información histórica disponible de la accesión; sin embargo, limitan la interpretación del desempeño como resultado derivado exclusivamente de atributos visuales extraídos de la imagen RGB. En consecuencia, QuinuApp debe entenderse como un sistema híbrido que combina información visual y contextual, no como un estimador puramente visual. Además, bajo partición estricta por accesión, las accesiones del conjunto de prueba no disponen de priors históricos calculados desde entrenamiento, lo que contribuye a explicar la degradación observada en ese esquema de validación.

En cuarto lugar, el entrenamiento y la evaluación se realizaron con imágenes procedentes de un experimento controlado, no de una campaña amplia de captura en campo abierto. Asimismo, aunque se diseñó una arquitectura móvil compuesta por QuinuApp, un teléfono Android y la BlackBox, el presente estudio no incluyó validación operativa del algoritmo ejecutándose directamente en el dispositivo móvil ni validación de campo del sistema integrado teléfono–BlackBox. Por ello, la BlackBox debe interpretarse como una infraestructura tecnológica propuesta para futuras capturas estandarizadas, no como un componente experimental ya validado. La robustez del sistema bajo condiciones no controladas, con variación natural de iluminación, fondo, distancia y ángulo, requiere evaluación específica. Finalmente, el experimento intraobservador se realizó sobre un conjunto independiente de imágenes, por lo que sus resultados constituyen una referencia descriptiva de variabilidad humana, pero no una comparación estrictamente equivalente entre desempeño humano y desempeño algorítmico. Por esta razón, el estudio no permite afirmar que el sistema iguale o supere a una especialista en fitopatología; esa comparación requeriría un diseño humano–máquina sobre el mismo conjunto de prueba. Las implicaciones del estudio son relevantes tanto en el ámbito académico como en el profesional. Desde el plano académico, el trabajo aporta una arquitectura metodológica que articula visión computacional, aprendizaje supervisado híbrido, variables interpretables y calibración postpredictiva en un problema de cuantificación continua. Desde el plano aplicado, el sistema ofrece una base tecnológica para apoyar tareas de fenotipado digital y evaluación comparativa de accesiones bajo condiciones controladas. Sin embargo, su adopción operativa debe considerarse todavía preliminar, dado que requiere validación externa, evaluación inter-accesión, ampliación del conjunto de datos y pruebas directas del sistema móvil completo.

CONCLUSIONES

El principal hallazgo de este estudio fue que, bajo la evaluación principal con partición estratificada por imagen, el pipeline híbrido QuinuApp obtuvo el desempeño global más balanceado entre los modelos de referencia evaluados, alcanzando el menor error absoluto medio (MAE = 17.17 %), el mayor coeficiente de concordancia de Lin (CCC = 0.694) y un sesgo sistemático cercano a cero (-0.58 %) bajo condiciones

experimentales controladas. Estos resultados indican que la combinación de variables visuales interpretables, modelado jerárquico híbrido y calibración postpredictiva constituye una aproximación técnicamente viable para apoyar la estimación continua de la severidad foliar del mildiu en quinua. En relación con los objetivos planteados, se logró evaluar un pipeline computacional basado en 55 variables clínico-texturales derivadas de imagen, complementadas con información contextual de esporulación, para estimar la severidad foliar en una escala continua de 0 % a 100 %. Asimismo, se diseñó una arquitectura tecnológica de despliegue compuesta por QuinuApp y una BlackBox modelada paramétricamente como propuesta de captura estandarizada. En este sentido, el objetivo general del estudio se considera cumplido dentro del dominio experimental evaluado, centrado en el desempeño del modelo sobre el conjunto de imágenes disponible.

No obstante, los resultados deben interpretarse como una prueba de concepto experimental bajo condiciones controladas y no como una validación operativa definitiva para campo abierto. Aunque la evaluación principal por imagen mostró un desempeño favorable, la validación por accesión evidenció una reducción del rendimiento frente a accesiones no observadas, indicando limitaciones en la generalización intergenotípica. Asimismo, el modelo presentó un MAE de 17.17 % y variabilidad residual en las predicciones individuales, especialmente en severidades bajas e intermedias. Como líneas futuras de investigación, se recomienda priorizar la validación externa en campo abierto con accesiones no observadas y conjuntos de captura más diversos. Además, resulta necesario ampliar y equilibrar el dataset, particularmente en fases de baja e intermedia severidad, evaluar la arquitectura QuinuApp–teléfono–BlackBox en condiciones reales de uso y reducir la dispersión residual del error sin comprometer la interpretabilidad del modelo.Estos resultados posicionan a QuinuApp como una propuesta metodológicamente trazable y técnicamente prometedora para la estimación continua de la severidad foliar del mildiu en quinua. Su principal aporte radica en demostrar la viabilidad inicial de una arquitectura interpretable para fenotipado digital asistido, aunque su consolidación como herramienta operativa dependerá de validaciones externas, reducción del error individual y evaluación robusta en escenarios reales de campo.

BIBLIOGRAFÍA

Barbedo, J. G. A. (2016). A review on the main challenges in automatic plant disease identification based on visible range images. Biosystems Engineering, 144, 52–60. https://doi.org/10.1016/j.biosystemseng.2016.01.017

Bland, J. M., & Altman, D. G. (1986). Statistical methods for assessing agreement between two methods of clinical measurement. The Lancet, 327(8476), 307–310. https://doi.org/10.1016/S0140-6736(86)90837-8

Bock, C. H., Poole, G. H., Parker, P. E., & Gottwald, T. R. (2010). Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging. Critical Reviews in Plant Sciences, 29(2), 59–107. https://doi.org/10.1080/07352681003617285

Bock, C. H., Barbedo, J. G. A., Del Ponte, E. M., Bohnenkamp, D., & Mahlein, A.-K. (2020). From visual estimates to fully automated sensor-based measurements of plant disease severity: Status and challenges for improving accuracy. Phytopathology Research, 2(1), Artículo 9. https://doi.org/10.1186/s42483-020-00049-8

Colque-Little, C., Amby, D. B., & Andreasen, C. (2021). A review of Chenopodium quinoa (Willd.) diseases—An updated perspective. Plants, 10(6), 1228. https://doi.org/10.3390/plants10061228

Chemura, A., Mutanga, O., Sibanda, M., & Chidoko, P. (2018). Machine learning prediction of coffee rust severity on leaves using spectroradiometer data. Tropical Plant Pathology, 43(2), 117–127. https://doi.org/10.1007/s40858-017-0187-8

Danielsen, S., Bonifacio, A., & Ames, T. (2003). Diseases of quinoa Chenopodium quinoa). Food Reviews International, 19(1–2), 43–59. https://doi.org/10.1081/FRI-120018867

Dainelli, R., Bruno, A., Martinelli, M., Moroni, D., Rocchi, L., Morelli, S., Ferrari, E., Silvestri, M., Agostinelli, S., La Cava, P., Di Maggio, A., Folco, C., Mori, N., Stevanato, P., & Toscano, P. (2024). GranoScan: An AI-powered mobile app for in-field identification of biotic threats of wheat. Frontiers in Plant Science, 15, Article 1298791. https://doi.org/10.3389/fpls.2024.1298791

Duarte-Carvajalino, J. M., Alzate, D. F., Ramirez, A. A., Santa-Sepulveda, J. D., Fajardo-Rojas, A. E., & Soto-Suárez, M. (2018). Evaluating late blight severity in potato crops using unmanned aerial vehicles and machine learning algorithms. Remote Sensing, 10(10), Article 1513. https://doi.org/10.3390/rs10101513

Fondevilla, S., Calderón-González, Á., Rojas-Panadero, B., Cruz, V., & Matías, J. (2024). Genome-wide association study, combined with bulk segregant analysis, identify plant receptors and defense related genes as candidate genes for downy mildew resistance in quinoa. BMC Plant Biology, 24(1), 594. https://doi.org/10.1186/s12870-024-05302-2

Faye, D., Diop, I., Mbaye, N., Dione, D., & Diedhiou, M. M. (2023). Plant disease severity assessment based on machine learning and deep learning: A survey. Journal of Computer and Communications, 11(9), 57–75. https://doi.org/10.4236/jcc.2023.119004

Gao, J., Westergaard, J. C., Sundmark, E. H. R., Bagge, M., Liljeroth, E., & Alexandersson, E. (2021). Automatic late blight lesion recognition and severity quantification based on field imagery of diverse potato genotypes by deep learning. Knowledge-Based Systems, 214, Article 106723. https://doi.org/10.1016/j.knosys.2020.106723

Gonçalves, J. P., Pinto, F. A. C., Queiroz, D. M., Villar, F. M. M., Barbedo, J. G. A., & Del Ponte, E. M. (2021). Deep learning architectures for semantic segmentation and automatic estimation of severity of foliar symptoms caused by diseases or pests. Biosystems Engineering, 210, 129–142. https://doi.org/10.1016/j.biosystemseng.2021.08.011

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. Computers and Electronics in Agriculture, 147, 70–90. https://doi.org/10.1016/j.compag.2018.02.016

Li, L., Zhang, S., & Wang, B. (2021). Plant disease detection and classification by deep learning—A review. IEEE Access, 9, 56683–56698. https://doi.org/10.1109/ACCESS.2021.3069646

Liu, W., Chen, Y., Lu, Z., Lu, X., Wu, Z., Zheng, Z., Suo, Y., Lan, C., & Yuan, X. (2024). StripeRust-Pocket: A mobile-based deep learning application for efficient disease severity assessment of wheat stripe rust. Plant Phenomics, 6, Article 0201. https://doi.org/10.34133/plantphenomics.0201

Pethybridge, S. J., & Nelson, S. C. (2015). Leaf Doctor: A new portable application for quantifying plant disease severity. Plant Disease, 99(10), 1310–1316. https://doi.org/10.1094/PDIS-03-15-0319-RE

Stanschewski, C. S., Rey, E., Fiene, G., Wellman, G., Melino, V. J., & Tester, M. (2021). Quinoa phenotyping methodologies: An international consensus. Plants, 10(9), Article 1759. https://doi.org/10.3390/plants10091759

PROCEDIMIENTO ARTICULADO DE EVALUACIÓN EX-ANTE PARA ARTEFACTOS DESIGN SCIENCE RESEARCH

1er M.Sc. Paul Fernando Grimaldo Bravo
Posgrado SOE – UAGRM
https://orcid.org/0009-0000-6343-9684
Santa Cruz, Bolivia | pgrimaldo@stc.soeuagrm.edu.bo

2do PhD. Luis Roberto Pérez Rios
Posgrado SOE – UAGRM
https://orcid.org/0000-0002-8385-1016
Santa Cruz, Bolivia | luis.roberto@alenasoft.com

https://doi.org/10.23670 FT.2026.1.38

Recibido 08/05/2026 - Aceptado 29/05/2026

RESUMEN

La evaluación constituye una actividad nuclear del paradigma Design Science Research (DSR) y una de las que menos guías operacionales ha recibido en la literatura, de manera particular cuando el artefacto evaluado es una guía, un framework o un modelo de proceso que no admite experimentación controlada. El marco FEDS (Framework for Evaluation in Design Science) aporta la conceptualización evaluativa, pero no prescribe cómo seleccionar evaluadores con criterios objetivos, cómo estructurar el instrumento ni qué estadísticos aplicar según la estructura de los datos. Existe, en paralelo, un repertorio consolidado de métodos cuantitativos para validación por juicio de expertos que combina aportes de la tradición iberoamericana (coeficiente de competencia K, W de Kendall, puntos de corte de Torgerson) con desarrollos de validez de contenido de circulación internacional (CVC de Hernández-Nieto, V de Aiken), aunque rara vez aparece articulado con los marcos DSR. Ante este escenario, el presente artículo propone un procedimiento de siete fases para la evaluación ex-ante de artefactos DSR no experimentables, ubicado en el cuadrante artificial ex-ante de FEDS, cuya fase de análisis habilita la selección del instrumento estadístico en función del tipo de juicio perseguido y de la estructura ordinal de los datos. El trabajo se apoya en revisión documental, articulación conceptual entre las tradiciones y verificación interna frente a criterios de operatividad, replicabilidad, trazabilidad e integración teórica declarados ex-ante. Se concluye que el procedimiento se propone como una contribución metodológica operacional aplicable por investigadores que evalúan artefactos DSR no experimentables.

Palabras clave: Design Science Research; evaluación ex-ante; FEDS; validación por juicio de expertos; validez de contenido.

ABSTRACT

Evaluation is a core activity of the Design Science Research (DSR) paradigm and one of the least supported by operational guidance in the literature, particularly when the evaluated artifact is a guideline, a framework or a process model that does not allow controlled experimentation. The FEDS framework (Framework for Evaluation in Design Science) provides a consolidated evaluative conceptualization, but does not prescribe how to select evaluators under objective criteria, how to structure the consultation instrument, or which statistical tools to apply depending on the structure of the data. In parallel, there exists a consolidated repertoire of quantitative methods for expert-judgment validation that combines contributions from the Ibero-American tradition (competence coefficient K, Kendall's W, Torgerson cut-points) with internationally circulating content-validity developments (Hernández-Nieto's CVC, Aiken's V), though it rarely appears articulated with reference DSR frameworks. In response, this article proposes an articulated seven-phase procedure for the ex-ante evaluation of non-experimentable DSR artifacts, positioned within the artificial ex-ante quadrant of FEDS, whose analytical phase enables the selection of the statistical instrument according to the type of judgment sought and the ordinal structure of the data. The work rests on a documentary review, a conceptual articulation between the traditions, and an internal verification of the procedure against criteria of operability, replicability, traceability and theoretical integration declared ex-ante. It is concluded that the procedure is proposed as an operational methodological contribution applicable by researchers evaluating non-experimentable DSR artifacts.

Keywords: Design Science Research; ex-ante evaluation; FEDS; expert judgment validation; content validity.

INTRODUCCIÓN

Hoy, el paradigma de Design Science Research (DSR) se encuentra consolidado como marco legítimo para la producción de conocimiento en ingeniería de software, sistemas de información y disciplinas afines, particularmente cuando el resultado de la investigación adopta la forma de un artefacto prescriptivo: modelo, método, guía o instanciación orientada a resolver problemas organizacionales reales (Hevner et al., 2004; Peffers et al., 2007). En este paradigma, la evaluación no es una etapa opcional ni posterior al cierre del trabajo; ya March y Smith (1995) la incorporaron como una de las cuatro actividades constitutivas de la investigación en diseño, junto con la construcción, la teorización y la justificación, sentando con ello la base conceptual sobre la que descansan los desarrollos posteriores del paradigma. Es la actividad que provee retroalimentación al ciclo de diseño y, cuando se ejecuta con rigor, asegura la calidad científica del artefacto producido (Venable et al., 2016). La centralidad de la evaluación en DSR no es discutida en lo conceptual, pero en la práctica presenta dificultades operacionales que la literatura no siempreexplicita.

Un porcentaje sustantivo de los artefactos que se producen en DSR, en particular las guías metodológicas, los frameworks de referencia, los modelos de proceso y las recomendaciones de ingeniería, no admite experimentación controlada en el sentido estricto. Se considera no experimentable a aquel artefacto cuya naturaleza prescriptiva o metodológica imposibilita la asignación aleatoria de unidades organizacionales a condiciones de tratamiento y control, o cuya instanciación previa en condiciones equivalentes al entorno real de aplicación resulta inviable por restricciones de tiempo, acceso o escala. Ante este escenario, la evaluación ex-ante mediante juicio de expertos se ha consolidado como alternativa metodológicamente apropiada (Venable et al., 2016; Prat et al., 2015). La pregunta relevante no es si el juicio experto constituye una opción válida, dado que eso ya está establecido, sino cómo operacionalizarlo con rigor.

El marco FEDS, propuesto por Venable et al. (2016), constituye la referencia más citada en la literatura DSR para orientar la selección estratégica del paradigma evaluativo en función del propósito y el contexto de la evaluación. Se articula en dos dimensiones, propósito funcional (formativo o sumativo) y paradigma evaluativo (artificial o naturalista), y define cuatro cuadrantes entre los que el investigador debe ubicar cada episodio evaluativo. El cuadrante artificial ex-ante aplica de manera particular a la evaluación anticipada de artefactos antes de su instanciación en un entorno real, y es allí donde se ubican naturalmente los artefactos metodológicos no experimentables. No obstante, FEDS se detiene en la conceptualización estratégica y no prescribe procedimientos operacionales: no indica cómo seleccionar evaluadores con criterios objetivos, cómo estructurar el instrumento de consulta, cómo procesar estadísticamente las respuestas ordinales ni cómo traducir la concordancia observada en una decisión evaluativa trazable. Esta es una limitación reconocida incluso por autores que adoptan FEDS como referencia central (Sonnenberg y vom Brocke, 2012; Prat et al., 2015).

En paralelo a la evolución de los marcos DSR, para la validación por juicio de expertos se ha consolidado un repertorio de métodos cuantitativos con instrumentos específicos para distintos objetivos evaluativos. De la tradición iberoamericana y la psicometría internacional provienen instrumentos como el coeficiente de competencia K (Ramírez-Urizarri, 1999), el coeficiente de concordancia W de Kendall (Alvarado, 2008), la técnica de puntos de corte de Torgerson (Torgerson, 1958), el coeficiente de validez de contenido CVC (Hernández-Nieto, 2002) y el coeficiente V de Aiken (Aiken, 1985; Penfield y Giacobbi, 2004). Aunque estos instrumentos aparecen aplicados en revistas indexadas en dominios tan diversos como la educación, la salud, la ingeniería y los sistemas de información (Escobar-Pérez y Cuervo-Martínez, 2008; Cabero y Llorente, 2013; Galicia Alarcón et al., 2017; Marín-González et al., 2021;Herrera-Masó et al., 2022; Pedrosa et al., 2014), rara vez se los presenta como un repertorio único articulado con un paradigma DSR de circulación internacional. El resultado es una desarticulación que limita tanto la operatividad de FEDS como la circulación internacional de un aparato metodológico maduro.

La fundamentación teórica del presente trabajo se sostiene en tres cuerpos de literatura convergentes. El primero es el paradigma DSR en su versión canónica (Hevner et al., 2004; Peffers et al., 2007) y sus desarrollos contemporáneos sobre tipos de contribución (Gregor y Hevner, 2013; Baskerville et al., 2018), que permiten ubicar aportes metodológicos al propio proceso DSR como contribuciones de pleno derecho al campo. El segundo es la literatura específica sobre evaluación en DSR (Sonnenberg y vom Brocke, 2012; Prat et al., 2015; Venable et al., 2016), que aporta la conceptualización del cuadrante evaluativo y los criterios de rigor aplicables. El tercero es la literatura consolidada sobre métodos cuantitativos para validación por juicio de expertos, con dos vertientes conectadas: la tradición iberoamericana de consulta a expertos (Ramírez-Urizarri, 1999; Alvarado, 2008; Escobar-Pérez y Cuervo-Martínez, 2008; Cabero y Llorente, 2013) y la literatura internacional sobre validez de contenido en psicometría (Aiken, 1985; Hernández-Nieto, 2002; Penfield y Giacobbi, 2004; Pedrosa, Suárez-Álvarez y García-Cueto, 2014). La coherencia entre los tres cuerpos descansa en un supuesto epistemológico compartido, el pragmatismo de la ciencia del diseño, y se materializa en que todos los enfoques buscan, por vías distintas, producir juicios evaluativos rigurosos sobre artefactos que no admiten experimentación. En esta misma línea, conviene anticipar la ubicación tipológica del presente aporte dentro del cuadrante de contribuciones de Gregor y Hevner (2013). El trabajo se sitúa de manera predominante en la categoría de Improvement, en tanto aborda un problema reconocido en un dominio consolidado mediante una solución cuya forma articulada no había sido formalizada con

anterioridad. No se proponen instrumentos estadísticos nuevos, sino una orquestación procedimental que mejora la operatividad del marco FEDS al integrar de manera trazable un repertorio cuantitativo cuya pertinencia individual ya estaba demostrada en otras tradiciones.

La justificación del trabajo se articula en tres planos. En el plano académico, el trabajo aporta un procedimiento operacional que cubre vacíos reconocidos en la literatura de evaluación DSR, lo cual amplía el repertorio metodológico disponible para investigadores que trabajan con artefactos no experimentables. En el plano metodológico, el trabajo tiende un puente entre la literatura DSR de referencia internacional y un repertorio de métodos cuantitativos para validación por juicio de expertos que ha evolucionado, hasta ahora, en paralelo y sin contacto sustantivo con dicha literatura. En el plano práctico, el procedimiento propuesto ofrece a investigadores un instrumento replicable y trazable que reduce la improvisación metodológica y formaliza decisiones habitualmente tomadas con base en criterios subjetivos.

A partir de lo expuesto, el objetivo general del presente trabajo es proponer un procedimiento articulado de evaluación ex-ante aplicable a artefactos Design Science Research no experimentables, que integre el marco FEDS con el repertorio consolidado de métodos cuantitativos para validación por juicio de expertos. De este objetivo se derivan tres objetivos específicos: (1) identificar los vacíos operacionales del marco FEDS en el cuadrante artificial ex-ante; (2) sistematizar el repertorio de métodos cuantitativos para validación por juicio de expertos en términos compatibles con las categorías de FEDS, explicitando la relación entre el tipo de juicio perseguido y el instrumento estadístico apropiado; y (3) articular un procedimiento operacional estructurado en fases, con entradas, actividades y salidas trazables, que cubra los vacíos identificados.

DESARROLLO

El desarrollo de esta reflexión se fundamenta en un análisis crítico de las limitaciones operacionales del marco FEDS, específicamente en su aplicación a artefactos que no admiten experimentación controlada. Mediante la integración de la literatura psicométrica internacional y los métodos de consulta a expertos de la tradición iberoamericana, se articula una propuesta procedimental que busca formalizar la evaluación ex-ante. Este análisis permite transitar desde la conceptualización estratégica hacia una operacionalización estadística trazable, asegurando que la validación del artefacto responda a criterios de rigor y pertinencia metodológica.

Articulación del repertorio cuantitativo con los vacíos de FEDS

La superación de los vacíos operacionales de FEDS requiere la integración de instrumentos que respondan a la naturaleza del juicio evaluativo perseguido. Para el vacío de selección de evaluadores, se adopta el coeficiente de competencia K (Ramírez-Urizarri, 1999), cuya expresión es $K = 0{,}5(K_c + K_a)$, donde Kc es el coeficiente de conocimiento derivado de la autovaloración del experto en una escala de diez niveles y K_a el coeficiente de argumentación, ponderado a partir de la diversidad y solidez de sus fuentes de fundamentación teórica y práctica; únicamente los candidatos con $K \geq 0{,}7$ (nivel medio) o $K \geq 0{,}9$ (nivel alto) integran el panel definitivo.

Respecto al procesamiento estadístico, el repertorio se selecciona según el objetivo de la consulta. Cuando se busca medir el consenso en la jerarquización de atributos, se emplea el coeficiente **W** de Kendall con su prueba de x^2 asociada (Alvarado, 2008; Siegel y Castellan, 1988), cuya expresión con corrección por empates es:

$$W = \frac{12S}{K^2(N^3 - N) - K\sum_j T_j}$$

donde S es la suma de los cuadrados de las desviaciones de los rangos respecto a su media global, K el número de evaluadores, N el número de ítems ordenados, y $\sum_j T_j$ la corrección por empates calculada como $T_j = \sum_t (t^3 - t)$ para cada grupo de t rangos iguales del evaluador j; la significancia se contrasta mediante $x^2 = K(N - 1)W$ con $N - 1$ grados de libertad, rechazando la hipótesis nula de ausencia de concordancia cuando dicho valor supera el umbral crítico.

Por el contrario, si el objetivo es validar la adecuación absoluta de los aspectos del artefacto mediante escalas Likert, se utilizan el Coeficiente de Validez de Contenido (CVC) de Hernández- Nieto (2002), con umbral de aceptación de 0,80, y el coeficiente V de Aiken (Aiken, 1985; Penfield y Giacobbi, 2004). El CVC de cada ítem i se calcula como $CVC_i = E_i/V_{max} - P_e$, donde E_i es la media de las valoraciones de los n expertos, V_{max} el valor máximo de la escala y $P_e = (1/V_{max})^n$ la corrección por probabilidad de error aleatorio; el CVC global es la media aritmética de los CVC_i individuales. Se adopta esta formulación simplificada de Hernández-Nieto (2002) frente a la variante que aplica una corrección adicional por probabilidad de error inter-juez antes del promedio; la elección entre ambas debe declararse explícitamente en cada estudio que invoque el procedimiento, dado que en paneles pequeños la convergencia entre formulaciones no está garantizada.

El coeficiente V, a su vez, se obtiene como $V = S/[n(c - 1)]$, donde S es la sumatoria de las diferencias entre cada valoración y el mínimo de la escala, n el número de expertos y c el número de categorías; su intervalo de confianza al 95 % se estima a partir de la distribución binomial exacta (Penfield y Giacobbi, 2004), aceptando el ítem en consulta única cuando el límite inferior del intervalo de confianza supera 0,50, criterio que opera como umbral mínimo de aceptación individual. Estos dos coeficientes miden planos complementarios– adecuación promedio y significancia estadística de esa adecuación– y deben leerse de forma conjunta.

Finalmente, para el juicio evaluativo integrado, se aplica

la técnica de puntos de corte de Torgerson (1958), cuya ejecución sobre la distribución de respuestas de cada ítem sigue cuatro pasos: (1) calcular las frecuencias relativas por categoría; (2) obtener las frecuencias relativas acumuladas; (3) convertir cada acumulado al valor *x* correspondiente en la distribución normal estándar mediante $x = \Phi^{-1}(T)$; y (4) derivar los puntos de corte entre categorías adyacentes como la media de los valores *x* contiguos, asignando a cada ítem la etiqueta cualitativa —Muy Adecuado, Bastante Adecuado, Adecuado, Poco Adecuado o Nada Adecuado— del intervalo en que cae su frecuencia acumulada modal, lo que traduce la distribución estadística en una decisión evaluativa directamente accionable. La técnica descansa en el supuesto de que el atributo evaluativo subyacente se distribuye normalmente sobre los expertos; el supuesto es razonable en paneles moderados, pero pierde tracción cuando *n* es pequeño o cuando las respuestas se concentran en una única categoría, situaciones en las que los puntos de corte deben interpretarse como aproximaciones orientativas más que como umbrales estrictos. En la Tabla 1 se evidencia la articulación de los vacíos de FEDS con los instrumentos y estructuras de datos.

Tabla 1

Matriz de articulación: Vacíos de FEDS, instrumentos y estructura de datos en el cuadrante artificial ex-ante

Vacío operacional de FEDS	Pregunta evaluativa sobre el artefacto	Estructura de datos	Instrumento principal	Fase del procedimiento
Selección de experto	¿Es el candidato competente en el dominio?	Autovaloración y fuentes	Coeficiente K (Kc + Ka)	Fase 3
Diseño del instrumento	¿Qué aspectos del artefacto se deben evaluar?	Matriz de dimensiones del artefacto	Escala Likert o Rangos puros	Fase 2 y 4
Procesamiento estadístico	¿Los expertos valoran positivamente cada aspecto?	Likert ordinal (valoración absoluta)	CVC + V de Aiken (IC 95%)	Fase 6
	¿Los expertos coinciden en la jerarquía de los ítems?	Rangos ordinales o transformación a rangos	W de Kendall + X^2	Fase 6
Juicio integrado	¿En qué categoría evaluativa cae cada ítem?	Frecuencias relativas acumuladas	Puntos de corte de Torgerson	Fase 7

Nota. *Elaboración propia a partir de Aiken (1985), Alvarado (2008), Hernández-Nieto (2002), Pedrosa et al. (2014), Ramírez-Urizarri (1999), Siegel y Castellan (1988) y Venable et al. (2016). La elección de W de Kendall para concordancia ordinal sobre alternativas como* α *de Krippendorff o AC1 de Gwet responde a su arraigo en la tradición iberoamericana de evaluación por expertos y a la disponibilidad de su prueba* X^2 *asociada; las alternativas se reseñan como línea de exploración futura en la sección de limitaciones.*

La identificación de estos vacíos no constituye una crítica al marco FEDS, concebido originalmente como una guía de orientación estratégica, sino que subraya la necesidad de un complemento operativo que evite la proliferación de procedimientos ad-hoc y garantice la trazabilidad entre estudios.

La sistematización propuesta integra instrumentos de la tradición iberoamericana con desarrollos internacionales de validez de contenido, asegurando que cada dimensión crítica, desde la selección objetiva de expertos mediante el coeficiente K hasta la categorización de resultados con la técnica de Torgerson, cuente con un respaldo metodológico robusto. Esta integración permite cubrir integralmente las demandas operacionales del cuadrante artificial ex-ante, transformando decisiones habitualmente subjetivas en procesos formalizados y replicables.

Como se detalla en la Tabla 1, la articulación de este repertorio no es mecánica, sino que depende de la correspondencia directa entre la pregunta evaluativa y la estructura de los datos. Mientras que el coeficiente CVC y la V de Aiken se especializan en cuantificar la valoración absoluta de los atributos del artefacto mediante escalas Likert, el coeficiente W de Kendall se reserva para medir el consenso en la jerarquización de los mismos, ajustando el análisis a la naturaleza ordinal de los rangos. Esta distinción conceptual asegura que la elección del instrumento estadístico en la Fase 6 del procedimiento sea una decisión trazable y declarada ex-ante, permitiendo que el investigador justifique el rigor de su juicio evaluativo en función de los objetivos específicos de la validación.

Procedimiento articulado propuesto

El procedimiento se estructura en siete fases secuenciales agrupadas en tres bloques funcionales: conceptualización evaluativa (FEDS, fases 1 y 2), operacionalización estadística (repertorio cuantitativo, fases 3 a 5) y análisis e interpretación integrada (fases 6 y 7).

El número de fases responde a una decisión de diseño expositivo orientada a maximizar la trazabilidad sin redundancia: estructuras más compactas fundirían decisiones que operan sobre fundamentos distintos —v.g., selección de evaluadores y diseño del instrumento, o análisis estadístico y decisión evaluativa—, mientras que estructuras más granulares introducirían subdivisiones operativamente equivalentes.

La Tabla 2 sintetiza las entradas, actividades y salidas de la estructura completa.

Tabla 2

Estructura del procedimiento articulado de evaluación ex-ante

Fase	Nombre	Entradas	Actividades principales	Salidas
1	Caracterización del artefacto y ubicación en FEDS	Artefacto DSR a evaluar	Clasificar por instanciabilidad, madurez y experimentabilidad; ubicar en cuadrante FEDS	Cuadrante evaluativo justificado
2	Formulación de dimensiones evaluativas	Artefacto + cuadrante	Derivar dimensiones; articular sub-criterios	Matriz de dimensiones y sub-criterios
3	Selección de expertos mediante coeficiente K	Pool de candidatos	Calcular Kc, Ka y K; filtrar por nivel medio o alto	Panel con coeficientes documentados
4	Diseño del instrumento de consulta	Matriz de dimensiones + tipo de juicio perseguido	Traducir dimensiones en ítems con escala Likert de cinco categorías o con rangos ordinales puros, según Tabla 1; incluir ítems abiertos opcionales	Cuestionario trazable a la matriz
5	Administración y recolección	Panel + cuestionario	Administración asincrónica, anónima, individual; ronda única o adicionales	Matriz de respuestas ordinales
6	Análisis estadístico de concordancia y cuantificación	Matriz de respuestas	Seleccionar instrumento del repertorio (W de Kendall con x^2 asociado; CVC; V de Aiken con IC 95 %) según Tabla 1; puntos de corte de Torgerson	Evidencia de concordancia + categorización por aspecto
7	Interpretación integrada y decisión evaluativa	Resultados estadísticos (+ comentarios abiertos opcionales)	Integrar evidencia cuantitativa; triangular con comentarios abiertos cuando estén disponibles; decidir validación, iteración o rediseño	Juicio evaluativo integrado

La ejecución de este flujo procedimental obedece a una lógica de trazabilidad estricta. En el primer bloque, la Fase 1 sitúa estratégicamente la evaluación justificando el cuadrante FEDS a partir de atributos estructurales como la instanciabilidad y madurez del artefacto según lo discutido anteriormente. Esta decisión condiciona la Fase 2, donde el artefacto se deconstruye en una matriz de dimensiones evaluables. Candidatos habituales son pertinencia teórica, suficiencia estructural, aplicabilidad operativa y coherencia interna, aunque la selección concreta depende del tipo de artefacto. Cada dimensión se descompone en sub-criterios evaluables por separado. La matriz orienta el diseño del instrumento en la Fase 4 y asegura cobertura balanceada de los aspectos del artefacto.

En el bloque de operacionalización, la Fase 3 supera la selección subjetiva tradicional al documentar matemáticamente la inclusión exclusiva de perfiles. El procedimiento convoca a candidatos que cumplan criterios previos de elegibilidad (formación académica, experiencia, vinculación al dominio), aplica un cuestionario de autovaloración y calcula K_c, K_a y K para cada uno; únicamente los candidatos con nivel medio o alto integran el panel. Se recomienda que el panel definitivo alcance al menos siete expertos elegibles, con un rango preferente entre ocho y quince integrantes para garantizar estabilidad estadística de los coeficientes ulteriores (Escobar-Pérez y Cuervo-Martínez, 2008; Cabero y Llorente, 2013). La Fase 4 consiste en la operacionalización de la matriz de dimensiones en un cuestionario estructurado compuesto por ítems cerrados de escala ordinal. La selección de la métrica no es arbitraria, sino que responde a la naturaleza del juicio requerido: se emplea una escala Likert de cinco niveles (desde Muy Adecuado hasta Nada Adecuado) para valoraciones de carácter absoluto —respaldada por los criterios de Aiken (1985), Alvarado (2008) y Cabero y Llorente (2013)— o, en su defecto, rangos puros *(1…k)* para tareas de jerarquización. Si bien es posible integrar de forma opcional ítems cualitativos abiertos, su alcance es limitado y funcional: actúan como un instrumento diagnóstico para justificar puntuaciones bajas y orientar el rediseño del artefacto. Es fundamental precisar que estos ítems no poseen una función validatoria per se, sino que sirven como insumo para la triangulación en la Fase 7. Durante la Fase 5, la administración asincrónica, anónima e individual del cuestionario resulta crítica para prevenir sesgos cognitivos inter-evaluadores, como el efecto líder —la tendencia documentada a que los evaluadores ajusten sus juicios hacia la posición del experto de mayor jerarquía o visibilidad en el panel— (Cabero-Almenara e Infante-Moro, 2014). La ronda única es la opción habitual; si no alcanza concordancia aceptable, se admiten rondas adicionales bajo lógica Delphi (Okoli y Pawlowski, 2004). El criterio de parada para estas iteraciones adicionales se determinará según la naturaleza del instrumento: (a) para artefactos que requieren evaluación ordinal o jerárquica, el ciclo concluye al alcanzar una concordancia de Kendall $W \geq 0,70$ con X^2 significativo $(p < 0,05)$ (Alvarado, 2008;Herrera-Masó et al., 2022); (b) para artefactos que requieren evaluación absoluta, se detendrá cuando los ítems superen un $CV\ V \geq 0,80$ (Hernández-Nieto, 2002) o, en su defecto, se obtenga una V de $Aiken \geq 0,78$ con todos los límites inferiores de los intervalos de confianza

sobre 0,50, criterio más exigente que el de aceptación individual descrito anteriormente, en tanto opera como umbral de convergencia tras iteración Delphi; o (c) el proceso finalizará mediante la convergencia declarada por el investigador bajo un criterio metodológico debidamente justificado (Okoli y Pawlowski, 2004).

Finalmente, en el bloque de cierre, la Fase 6 ejecuta el cálculo estadístico seleccionando el coeficiente pertinente (CVC, V de Aiken o W de Kendall) y aplicando los puntos de corte de Torgerson. La Fase 7 integra estos resultados cuantitativos —evidencia de concordancia estadística, valoración absoluta por ítem y categorización de Torgerson— en un juicio evaluativo con tres decisiones posibles: validación, iteración con ajustes o rediseño, este último en grado parcial o integral según la magnitud de los hallazgos.

Tabla 3

Matriz orientativa de decisión evaluativa en la Fase 7

Configuración de resultados	Decisión sugerida	Lectura interpretativa
CVC ≥ 0,80 (todos los ítems) y W ≥ 0,70 con X^2 significativo ($p < 0,05$); o V de Aiken ≥ 0,78 con LI > 0,50; categorización modal en "Bastante Adecuado" o "Muy Adecuado"	Validación	Concordancia y adecuación convergen positivamente
0,60 ≤ CVC < 0,80 en algunos ítems, o W entre 0,50 y 0,70, con categorización modal en "Adecuado"	Iteración con ajustes focalizados	Adecuación parcial o concordancia moderada que admite refinamiento sin rediseño estructural
CVC < 0,60 en una proporción sustantiva de ítems, W < 0,50, o categorización modal en "Poco Adecuado" o "Nada Adecuado"	Rediseño parcial o integral	Insuficiencia de adecuación o disenso estructural que excede el ajuste menor

Nota. Los umbrales se derivan de Hernández-Nieto (2002) para CVC, Alvarado (2008) y Herrera-Masó et al. (2022) para W de Kendall, y Penfield y Giacobbi (2004) para V de Aiken. La matriz opera como referencia orientativa, no como regla determinista.

La matriz de la Tabla 3 ofrece umbrales orientativos derivados de la literatura especializada para cada instrumento del repertorio, con el propósito de reducir la indefinición en la decisión final. No obstante, estos umbrales no operan como un algoritmo determinista: la naturaleza del artefacto, la dispersión de las valoraciones entre dimensiones y la evidencia cualitativa complementaria pueden justificar decisiones que se aparten de la lectura mecánica de la tabla. Se prescriben umbrales por economía operacional, pero la responsabilidad del juicio evaluativo final recae en el investigador, que debe declarar explícitamente la regla de decisión adoptada y sus eventuales desviaciones respecto de la matriz orientativa. Conviene señalar, en clave conceptual y no ya operativa, que la concordancia estadística mide coincidencia entre expertos y no adecuación intrínseca del artefacto, distinción que sustenta la lectura conjunta de los tres instrumentos del repertorio. Cuando el instrumento de consulta incluyó ítems abiertos, sus respuestas pueden triangularse de manera complementaria con los resultados cuantitativos, con el propósito acotado de identificar disonancias (por ejemplo, categoría alta con comentarios críticos) que orienten el rediseño del artefacto. Esta triangulación es opcional y no altera la naturaleza cuantitativa de la validación. Esta fase suele omitirse en reportes que cierran la validación con el solo resultado cuantitativo; su inclusión explícita refuerza la trazabilidad del juicio final.

Demostración del procedimiento

La verificación del procedimiento frente a los cuatro criterios declarados en la metodología arroja cumplimiento en los cuatro ejes: cada fase produce salidas específicas que alimentan la siguiente sin rupturas (operatividad); las siete fases están descritas con un nivel de detalle que permite su aplicación a artefactos distintos manteniendo el mismo esquema (replicabilidad); las salidas finales son trazables hasta las decisiones iniciales sin saltos no justificados (trazabilidad); y el procedimiento respeta los fundamentos conceptuales de FEDS y los supuestos de cada instrumento del repertorio considerando rangos ordinales para W de Kendall, Likert con valoración absoluta para CVC y V de Aiken, y frecuencias acumuladas para Torgerson, sin distorsionar ninguno para forzar la articulación (integración teórica).

Para evidenciar la ejecutabilidad del flujo en condiciones reales, el procedimiento fue aplicado por los autores a un artefacto metodológico del ámbito de la ingeniería de software en un trabajo previo, específicamente una guía metodológica con relevancia real en el área de arquitectura de software. La aplicación involucró un panel de 8 expertos con nivel de competencia medio o alto confirmado mediante coeficiente K, un cuestionario de 17 ítems cerrados y 2 preguntas abiertas opcionales derivado de una matriz de 8 dimensiones evaluativas (v.g., Aplicabilidad, Gestión de Riesgos) con escala Likert de cinco categorías (Nada Adecuado, Poco Adecuado, Adecuado, Bastante Adecuado, Muy Adecuado), y una ronda única de administración asincrónica y anónima. Dado que el objetivo evaluativo era validar positivamente cada aspecto del artefacto de forma absoluta, la Fase 6 adoptó CVC y V de Aiken como instrumentos principales, con Torgerson para la categorización final. Los 17 ítems superaron el umbral CVC ≥ 0,80 y los límites inferiores de los intervalos de confianza al 95 % de V de Aiken —calculados a partir de la distribución binomial exacta— se ubicaron por encima del criterio de aceptación individual establecido. El recorrido completo de las siete fases se ejecutó sin

rupturas operacionales, donde finalmente el artefacto en su totalidad alcanzó la categoría "Muy Adecuado" en la caracterización de Torgerson, lo cual respalda la aplicabilidad del procedimiento en condiciones reales. Los resultados numéricos detallados quedan fuera del alcance del presente artículo, cuyo objeto es el procedimiento y no el artefacto evaluado.

Alcance e interpretación del procedimiento propuesto

Los resultados obtenidos permiten interpretar el procedimiento propuesto como una contribución operacional al marco FEDS, no como un reemplazo de este. El marco conceptual de FEDS sigue siendo la referencia estratégica que guía la ubicación inicial de la evaluación y la decisión sobre el paradigma evaluativo a adoptar; el procedimiento aporta el detalle operativo. Ambos coexisten sin conflicto: FEDS para la estrategia, el procedimiento articulado para la ejecución.

Lo que efectivamente se gana con esta articulación no es un inventario adicional de instrumentos estadísticos, sino una regla de selección trazable. La experiencia en validación metodológica evidencia que el mayor costo evaluativo no recae en el cálculo matemático, sino en la indefinición previa de qué evaluar, por qué y con qué instrumento. En ausencia de un procedimiento formal, la inercia empuja al investigador a adoptar herramientas por mera tradición disciplinar, sin que la elección responda a la naturaleza del juicio perseguido. Al explicitar la correspondencia entre la pregunta evaluativa, la estructura de datos y el instrumento (Tabla 1), el procedimiento desplaza el esfuerzo desde la operatividad del cálculo hacia el rigor de la definición previa. Este enfoque atiende un nicho específico: la validación de artefactos metodológicos no experimentables, evaluados bajo restricciones críticas de tiempo y acceso que vuelven impracticable la instanciación previa o las múltiples rondas de convergencia. La articulación cuantitativa que aquí se propone no agota el espacio de evaluación ex-ante, sino que ofrece una formalización rigurosa para los casos donde la cuantificación del consenso es metodológicamente exigida.

Finalmente, de la Tabla 1 emerge una observación secundaria: los instrumentos presentados cubren los vacíos de FEDS, pero no agotan las posibilidades evaluativas. Rondas adicionales tipo Delphi, análisis de consistencia en la autovaloración o la adopción de coeficientes alternativos para paneles masivos podrían articularse en el futuro. En esa dirección, el procedimiento propuesto opera como una primera articulación mínima viable, abierta a enriquecimientos posteriores.

Limitaciones de la propuesta

El trabajo presenta cuatro limitaciones que deben considerarse al interpretar sus resultados. La primera es que la evaluación del procedimiento propuesto es de carácter interno y descansa principalmente en la verificación contra los criterios declarados ex-ante; la validación externa mediante su uso documentado por investigadores distintos a los autores, sobre una diversidad de artefactos y dominios, constituye la principal frontera del presente aporte. La segunda limitación es que la articulación propuesta se restringe al cuadrante artificial ex-ante de FEDS; los otros tres cuadrantes (naturalistic ex-ante, artificial ex-post y naturalistic ex-post) requieren procedimientos distintos que este trabajo no aborda. La tercera limitación, más sutil, es que el procedimiento hereda los supuestos estadísticos del repertorio cuantitativo incorporado: los criterios de interpretación, las reglas de corte y la lógica de selección plasmada en la Tabla 1 descansan en convenciones metodológicas consolidadas pero no exclusivas. Futuras versiones podrían explorar alternativas como el coeficiente alfa de Krippendorff, el kappa de Fleiss o modelos de teoría de respuesta al ítem.

Una cuarta limitación, de naturaleza epistemológica, atañe a la capacidad discriminativa del procedimiento. La aplicación previa que sostiene la verificación interna se ejecutó sobre un artefacto que superó los umbrales de aceptación, lo cual confirma la operatividad del flujo, pero no aporta evidencia sobre el comportamiento del procedimiento ante artefactos con defectos conocidos o estructuralmente débiles. Cabe la posibilidad de que los umbrales adoptados resulten poco sensibles ante casos límite o que la articulación de instrumentos genere falsos positivos en escenarios donde los expertos coinciden en valoraciones moderadas. La verificación de discriminabilidad requiere ejercicios deliberados de aplicación adversarial que excedan el alcance del presente artículo. Expresado en términos epistemológicos, la demostración prueba la operatividad del flujo —que ningún paso bloquea ni produce salidas inválidas— pero no su validez discriminativa, esto es, la capacidad del procedimiento para distinguir artefactos adecuados de artefactos deficientes.

Esta distinción entre operatividad demostrada y validez discriminativa pendiente debe asumirse al interpretar la verificación interna ofrecida y motiva la línea de trabajo futuro de aplicación adversarial. Existe además una tensión no resuelta entre la pretensión de generalización del procedimiento y las particularidades de cada artefacto DSR evaluado. El procedimiento prescribe las fases pero no puede prescribir las dimensiones evaluativas específicas, que dependen del tipo de artefacto y del dominio, lo cual deja al investigador con una decisión no trivial en la Fase 2. Esta tensión no invalida el procedimiento, pero delimita su alcance: es un marco operacional estructurado, no un algoritmo determinista.

CONCLUSIONES

El presente trabajo se propuso articular el marco FEDS con el repertorio consolidado de métodos cuantitativos para validación por juicio de expertos mediante un procedimiento operacional aplicable a la evaluación ex-ante de artefactos Design Science Research no experimentables. Los tres objetivos específicos enunciados en la introducción encuentran correspondencia directa con los resultados obtenidos: **Vacíos de FEDS identificados** (objetivo específico

1). Se caracterizaron cuatro vacíos operacionales del marco en el cuadrante artificial ex-ante (selección de evaluadores, diseño del instrumento, procesamiento estadístico y trazabilidad del juicio evaluativo integrado), cuya persistencia alimenta procedimientos ad-hoc no comparables entre estudios.

Repertorio cuantitativo sistematizado (objetivo específico 2). Se articuló el repertorio en términos compatibles con FEDS (coeficiente de competencia K, W de Kendall con X^2 asociado, puntos de corte de Torgerson, CVC de Hernández-Nieto y V de Aiken con intervalo de confianza) y se formalizó la correspondencia entre tipo de juicio evaluativo, estructura de datos e instrumento estadístico apropiado.

Procedimiento articulado construido (objetivo específico 3). Se estructuró un procedimiento de siete fases con entradas, actividades y salidas trazables que cubre los vacíos identificados, respeta los fundamentos conceptuales de FEDS y los supuestos estadísticos de cada instrumento, y quedó verificado frente a los cuatro criterios declarados ex-ante mediante una aplicación ilustrativa previa.

El objetivo general —proponer un procedimiento articulado aplicable a artefactos DSR no experimentables— se considera cumplido en su alcance propositivo, configurando una contribución de tipo Improvement, y quedando la validación externa por investigadores distintos a los autores como principal frontera del aporte. Como líneas de trabajo futuro se identifican tres direcciones. La primera es la validación externa del procedimiento mediante su uso documentado por investigadores distintos a los autores, sobre artefactos DSR de dominios diversos, con el fin de recoger evidencia empírica sobre su utilidad percibida, tiempo de aplicación, trazabilidad y necesidad de ajustes. La segunda es la extensión del procedimiento al cuadrante naturalistic ex-ante mediante la incorporación de focus groups estructurados y observación participante, de forma tal que el investigador pueda transitar entre cuadrantes FEDS con un único procedimiento articulado.

La tercera línea, de carácter más instrumental, consiste en el desarrollo de herramientas computacionales de soporte, tales como plantillas automatizadas para el cálculo de coeficientes, calculadoras para los puntos de corte de Torgerson e integraciones con plataformas de encuestas en línea, que reduzcan la barrera técnica de adopción del procedimiento. La cuarta línea, complementaria de la primera, consiste en un estudio de validación adversarial mediante la aplicación deliberada del procedimiento a artefactos con defectos conocidos o construidos ex profeso para someter a prueba su capacidad discriminativa. Este tipo de ejercicio permitiría verificar si los umbrales declarados detectan efectivamente los puntos débiles esperados o si requieren recalibración.

BIBLIOGRAFÍA

Aiken, L. R. (1985). Three coefficients for analyzing the reliability and validity of ratings. Educational and Psychological Measurement, 45(1), 131–142. https://doi.org/10.1177/0013164485451012

Alvarado, F. (2008). Análisis de concordancia de atributos. Tecnología en Marcha, 21(3), 29–35.

Baskerville, R., Baiyere, A., Gregor, S., Hevner, A., y Rossi, M. (2018). Design Science Research contributions: Finding a balance between artifact and theory. Journal of the Association for Information Systems, 19(5), 358–376. https://doi.org/10.17705/1jais.00495

Cabero, J., y Llorente, M. C. (2013). La aplicación del juicio de experto como técnica de evaluación de las tecnologías de la información y comunicación (TIC). Revista de Tecnología de Información y Comunicación en Educación, 7(2), 11–22.

Escobar-Pérez, J., y Cuervo-Martínez, Á. (2008). Validez de contenido y juicio de expertos: una aproximación a su utilización. Avances en Medición, 6(1), 27–36.

Galicia Alarcón, L. A., Balderrama Trápaga, J. A., y Edel Navarro, R. (2017). Validez de contenido por juicio de expertos: propuesta de una herramienta virtual. Apertura, 9(2), 42–53.

Gregor, S., y Hevner, A. R. (2013). Positioning and presenting Design Science Research for maximum impact. MIS Quarterly, 37(2), 337–355. https://doi.org/10.25300/MISQ/2013/37.2.01

Hernández-Nieto, R. (2002). Contributions to statistical analysis. Universidad de Los Andes.

Herrera-Masó, J., Calero-Ricardo, J. L., González-Rangel, M. A., Collazo Ramos, M. I., y Travieso-González, Y. (2022). El método de consulta a expertos en tres niveles de validación. Revista Habanera de Ciencias Médicas, 21(1).

Hevner, A. R., March, S. T., Park, J., y Ram, S. (2004). Design science in information systems research. MIS Quarterly, 28(1), 75–105.

March, S. T., y Smith, G. F. (1995). Design and natural science research Decision Support Systems, 15(4), 251–266. https://doi.org/10.1016/0167-9236(94)00041-2

Marín-González, F., Paredes-Chacín, A. J., e Inciarte-González, A. (2021). Validación del diseño de una red de cooperación científico-tecnológica utilizando el coeficiente K para la selección de expertos. Información Tecnológica, 32(2), 79–88.

Okoli, C., y Pawlowski, S. D. (2004). The Delphi method as a research tool: An example, design considerations and applications. Information & Management, 42(1), 15–29. https://doi.org/10.1016/j.im.2003.11.0

Pedrosa, I., Suárez-Álvarez, J., y García-Cueto, E. (2014). Evidencias sobre la validez de contenido: avances teóricos y métodos para su estimación. Acción Psicológica, 10(2), 3–18. https://doi.org/10.5944/ap.10.2.11820

Peffers, K., Tuunanen, T., Rothenberger, M. A., y Chatterjee, S. (2007). A Design Science Research methodology for information systems research. Journal of Management Information Systems, 24(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Penfield, R. D., y Giacobbi, P. R. (2004). Applying a score confidence interval to Aiken's item content-relevance index. Measurement in Physical Education and Exercise Science, 8(4), 213–225. https://doi.org/10.1207/s15327841mpee0804_3

Prat, N., Comyn-Wattiau, I., y Akoka, J. (2015). A taxonomy of evaluation methods for information systems artifacts. Journal of Management Information Systems, 32(3), 229–267. https://doi.org/10.1080/07421222.2015.1099390

Ramírez-Urizarri, L. A. (1999). Algunas consideraciones acerca del método de evaluación utilizando el criterio de expertos. Instituto Superior de Ciencias Agropecuarias de La Habana.

Siegel, S., y Castellan, N. J. (1988). Nonparametric statistics for the behavioral sciences (2.ª ed.). McGraw-Hill.

Torgerson, W. S. (1958). Theory and methods of scaling. John Wiley and Sons.

CRIPTOGRAFÍA POST-CUÁNTICA EN REDES LTE/VOLTE: DESEMPEÑO, INTEROPERABILIDAD Y RIESGO EN LA MIGRACIÓN DE ESQUEMAS PARA TELECOMUNICACIONES EN SUDAMÉRICA

M.Sc. Jorge Marcelo Rosales Fuentes
Posgrado SOE – UAGRM
https://orcid.org/0009-0002-9498-8525
Santa Cruz, Bolivia | jorgerosales@uagrm.edu.bo

https://doi.org/10.23670 FT.2026.1.44

Recibido 04/05/2026 - Aceptado 13/05/2026

RESUMEN

El avance de la computación cuántica representa una amenaza para los sistemas criptográficos clásicos utilizados en infraestructuras de telecomunicaciones, particularmente aquellos basados en RSA y criptografía de curva elíptica. En redes LTE y servicios VoLTE predominantes en Bolivia y Sudamérica, la transición hacia criptografía post-cuántica (PQC) plantea desafíos relacionados con desempeño, interoperabilidad y calidad de servicio. El presente trabajo analiza el impacto de esquemas criptográficos clásicos, post-cuánticos e híbridos sobre entornos representativos de telecomunicaciones LTE/VoLTE, considerando métricas relevantes de red como latencia, jitter, throughput, packet loss ratio y consumo de recursos computacionales. La investigación adopta un enfoque experimental controlado utilizando TLS 1.3 y herramientas de monitoreo de tráfico para evaluar el comportamiento de distintos mecanismos criptográficos bajo condiciones representativas de operación. Los resultados evidencian que la adopción de criptografía post-cuántica introduce incrementos en latencia, sobrecarga de señalización y uso de CPU, los cuales pueden impactar indicadores de calidad de servicio y acuerdos de nivel de servicio (SLA) en aplicaciones sensibles al tiempo real. Asimismo, se observa que los esquemas híbridos representan una alternativa viable para una migración progresiva en infraestructuras regionales con recursos limitados. Como principal contribución, se propone un modelo integral basado en crypto-agility que incorpora evaluación de desempeño, interoperabilidad y gestión de riesgo orientado específicamente a redes LTE/VoLTE predominantes en Sudamérica. El estudio contribuye a reducir la brecha entre teoría criptográfica y operación práctica de telecomunicaciones frente a amenazas cuánticas emergentes.

Palabras clave: Criptografía post-cuántica, LTE, VoLTE, telecomunicaciones, desempeño, interoperabilidad, seguridad, QoS, Crypto-agility, TLS 1.3, migración criptográfica.

ABSTRACT

The advancement of quantum computing represents a threat to classical cryptographic systems used in telecommunications infrastructures, particularly those based on RSA and elliptic curve cryptography. In LTE networks and VoLTE services predominant in Bolivia and South America, the transition toward post-quantum cryptography (PQC) poses challenges related to performance, interoperability, and quality of service. This work analyzes the impact of classical, post-quantum, and hybrid cryptographic schemes on representative LTE/VoLTE telecommunications environments, considering relevant network metrics such as latency, jitter, throughput, packet loss ratio, and computational resource consumption. The research adopts a controlled experimental approach using TLS 1.3 and traffic monitoring tools to evaluate the behavior of different cryptographic mechanisms under representative operating conditions. The results show that the adoption of post-quantum cryptography introduces increases in latency, signaling overhead, and CPU usage, which may affect quality of service indicators and service level agreements (SLA) in real-time sensitive applications. Likewise, it is observed that hybrid schemes represent a viable alternative for a progressive migration in regional infrastructures with limited resources. As the main contribution, an integral model based on crypto-agility is proposed, incorporating performance evaluation, interoperability, and risk management specifically oriented to LTE/VoLTE networks predominant in South America. The study contributes to reducing the gap between cryptographic theory and the practical operation of telecommunications in the face of emerging quantum threats.

Keywords: Post-quantum cryptography, LTE, VoLTE, telecommunications, performance, interoperability, security, QoS, Crypto-agility, TLS 1.3, cryptographic migration.

INTRODUCCIÓN

El avance de la computación cuántica representa uno de los mayores desafíos emergentes para la seguridad de las infraestructuras digitales modernas. Algoritmos cuánticos como el Algoritmo de Shor tienen el potencial de comprometer los sistemas criptográficos asimétricos actualmente utilizados en protocolos de comunicación, particularmente aquellos basados en RSA y criptografía de curva elíptica (ECC). Esta situación amenaza directamente la confidencialidad, integridad y autenticidad de la información transmitida a través de redes de telecomunicaciones.

En respuesta a este escenario, la criptografía post-cuántica (PQC) ha emergido como una alternativa orientada a garantizar seguridad frente a ataques cuánticos. Los esfuerzos de estandarización liderados por el National Institute of Standards and Technology han impulsado el desarrollo de algoritmos resistentes a la computación cuántica, como CRYSTALS-Kyber y CRYSTALS-Dilithium, considerados actualmente entre las principales opciones para futuras implementaciones seguras. Sin embargo, la transición hacia esquemas criptográficos post-cuánticos representa un desafío particularmente complejo para el sector de telecomunicaciones, donde los requisitos de disponibilidad, latencia y calidad de servicio son críticos. A diferencia de otros entornos informáticos, las redes de telecomunicaciones deben garantizar continuidad operativa en servicios sensibles al tiempo real, como VoLTE, videollamadas, streaming y señalización móvil.

En Bolivia y gran parte de Sudamérica, las infraestructuras de telecomunicaciones continúan dependiendo predominantemente de tecnologías LTE, 4G y VoLTE, mientras que el despliegue de redes 5G aún se encuentra en etapas iniciales. Esta realidad tecnológica introduce una necesidad particular de evaluar cómo la adopción de criptografía post-cuántica impactaría en redes actualmente operativas y no únicamente en arquitecturas futuras. En este contexto, incrementos en la latencia criptográfica o en el tamaño de los mensajes podrían afectar indicadores críticos de calidad de servicio (QoS), como jitter, throughput o packet loss ratio, comprometiendo potencialmente los acuerdos de nivel de servicio (SLA) establecidos por los operadores.

Adicionalmente, la migración hacia criptografía post-cuántica introduce desafíos de interoperabilidad debido a la coexistencia entre sistemas clásicos y nuevos esquemas criptográficos. La implementación de soluciones híbridas, aunque necesaria para garantizar una transición progresiva, incrementa la complejidad operativa y puede generar nuevos riesgos asociados a configuración, compatibilidad y gestión de claves.

Por otra parte, existe un problema estratégico relacionado con el fenómeno conocido como “harvest now, decrypt later”, mediante el cual información cifrada actualmente podría ser almacenada y descifrada en el futuro utilizando capacidades cuánticas avanzadas. Esto genera presión sobre operadores y proveedores de telecomunicaciones para iniciar procesos de evaluación y migración criptográfica antes de la disponibilidad masiva de computadoras cuánticas funcionales. Si bien diversos estudios han analizado aspectos específicos de la criptografía post-cuántica, la mayoría de los trabajos se enfocan de manera aislada en desempeño criptográfico, seguridad matemática o integración de protocolos, sin considerar simultáneamente el impacto operativo sobre redes de telecomunicaciones reales en contextos regionales.

En este contexto, el presente trabajo propone un análisis integral de la adopción de criptografía post-cuántica en telecomunicaciones, considerando conjuntamente desempeño, interoperabilidad y riesgo en escenarios representativos de redes LTE/4G/VoLTE predominantes en Bolivia y Sudamérica. Asimismo, se plantea un modelo basado en crypto-agility orientado a facilitar procesos de migración progresiva y adaptable hacia infraestructuras resistentes a amenazas cuánticas.

Objetivo General

Analizar el impacto de la criptografía post-cuántica sobre el desempeño, interoperabilidad y riesgo en redes LTE/VoLTE representativas de telecomunicaciones en Sudamérica.

Objetivos Específicos

- Evaluar el impacto de esquemas PQC sobre KPIs de telecomunicaciones.
- Comparar mecanismos clásicos, post-cuánticos e híbridos.
- Analizar implicaciones sobre QoS y SLA en redes LTE/VoLTE.
- Proponer un modelo basado en crypto-agility para migración progresiva.

TRABAJOS RELACIONADOS

Diversos estudios han analizado la adopción de criptografía post-cuántica en sistemas de comunicación. El proceso de estandarización liderado por el National Institute of Standards and Technology (NIST) ha identificado algoritmos como CRYSTALS-Kyber y CRYSTALS-Dilithium como candidatos principales. Investigaciones recientes han evaluado el impacto de PQC en protocolos como TLS y en redes LTE/4G, evidenciando incrementos en latencia y consumo de recursos. Asimismo, se han propuesto enfoques híbridos para facilitar la transición desde sistemas clásicos hacia entornos post-cuánticos. Sin embargo, persisten desafíos en términos de interoperabilidad, compatibilidad y gestión de riesgos. La literatura actual carece de un enfoque integral que analice simultáneamente desempeño, interoperabilidad y riesgo, lo cual motiva el presente trabajo.

METODOLOGÍA

La presente investigación adopta un enfoque metodológico mixto, combinando análisis cuantitativo y cualitativo para evaluar el impacto de la criptografía post-cuántica en entornos de telecomunicaciones.

El estudio se orienta específicamente a escenarios representativos de redes LTE/4G/VoLTE y servicios de transmisión de datos predominantes en Bolivia y Sudamérica, donde la infraestructura 4G continúa siendo la tecnología de acceso más ampliamente desplegada.

Diseño Experimental

Se implementó un entorno experimental controlado basado en arquitectura cliente-servidor, utilizando infraestructura virtualizada para simular condiciones representativas de redes de telecomunicaciones IP modernas. El diseño experimental considera tres escenarios criptográficos:

1. Esquema clásico

RSA/ECC

TLS 1.3 tradicional

2. Esquema post-cuántico (PQC)

CRYSTALS-Kyber

CRYSTALS-Dilithium

3. Esquema híbrido

Combinación ECC + Kyber

Firma híbrida

Las pruebas fueron diseñadas considerando servicios sensibles al tiempo real, particularmente: VoLTE, videollamadas IP, streaning y señalización segura.

Entorno de Red Evaluado

A diferencia de enfoques genéricos orientados exclusivamente a redes 5G, este estudio considera escenarios alineados con la realidad operativa regional, donde predominan redes LTE, 4G y servicios VoLTE. El entorno experimental contempla: Red IP virtualizada, Simulación de tráfico concurrente, Variaciones de carga, Retrasos controlados, Condiciones representativas de congestión moderada. Asimismo, se consideran limitaciones comunes en operadores regionales: restricciones de ancho de banda, recursos computacionales limitados, infraestructura heterogénea

Indicadores Clave de Desempeño (KPIs)

Para evaluar el impacto de los esquemas criptográficos se definieron KPIs relevantes para telecomunicaciones:

Tabla 1

Indicadores Clave de Desempeño

KPI	Descripción
Latencia de handshake	Tiempo de establecimiento TLS
Jitter	Variación temporal del retardo
Throughput	Capacidad efectiva de transmisión
Packet Loss Ratio	Pérdida de paquetes
Uso de CPU	Carga computacional
Tamaño de handshake	Sobrecarga de señalización

Estos indicadores permiten evaluar no solo desempeño criptográfico, sino también el impacto operativo sobre servicios de telecomunicaciones.

Consideraciones de Calidad de Servicio (QoS) y SLA

La evaluación considera criterios de calidad de servicio (QoS) utilizados comúnmente en redes LTE/4G/VoLTE. Se analizaron posibles afectaciones sobre: continuidad de llamadas VoLTE, estabilidad de videollamadas, tiempo de establecimiento de sesión, experiencia del usuario. Asimismo, se evaluó el potencial impacto sobre acuerdos de nivel de servicio (SLA), particularmente en aplicaciones sensibles a: baja latencia, jitter reducido, alta disponibilidad.

Procedimiento Experimental

Para cada escenario criptográfico se ejecutó el siguiente procedimiento:

1. Configuración del entorno TLS
2. Establecimiento de conexiones cliente-servidor
3. Generación de tráfico controlado
4. Ejecución de múltiples iteraciones experimentales
5. Captura de métricas de desempeño
6. Comparación entre esquemas criptográficos

Métricas de Evaluación

Las métricas de latencia, tamaño del handshake y uso de CPU fueron calculadas mediante promedios aritméticos obtenidos a partir de múltiples iteraciones experimentales. Asimismo, se utilizó desviación estándar para evaluar estabilidad y dispersión de los resultados.

Latencia Promedio del Handshake

La latencia promedio se obtiene calculando el promedio aritmético de los tiempos medidos durante las iteraciones experimentales.

$$\bar{L} = \frac{1}{N}\sum_{i=1}^{N} L_i$$

Donde:

- $\bar{L}$ = latencia promedio
- L_i = latencia medida en la iteración i
- N = número total de iteraciones experimentales

Tamaño Promedio del Handshake

El tamaño promedio del handshake se calcula mediante el promedio de bytes intercambiados durante el establecimiento de sesión TLS.

$$\bar{H} = \frac{1}{N}\sum_{i=1}^{N} H_i$$

Donde:

- $\bar{H}$ = tamaño promedio del handshake
- H_i = tamaño del handshake en la iteración i
- N = número total de iteraciones

Uso Promedio de CPU

El consumo promedio de CPU se calcula promediando la utilización porcentual registrada durante las ejecuciones experimentales.

$$\overline{CPU} = \frac{1}{N}\sum_{i=1}^{N} CPU_i$$

Donde:

- $\overline{CPU}$ = uso promedio de CPU
- CPU_i = porcentaje de CPU utilizado en la iteración i
- N = número total de mediciones

Desviación Estándar

Para el cálculo de la desviación estándar se utiliza la siguiente fórmula:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

Donde:

- σ = desviación estándar
- x_i = valor medido
- $\bar{x}$ = promedio de la muestra
- N = número de iteraciones

Herramientas Tecnológicas Utilizadas

Las pruebas utilizaron: OpenSSL con soporte PQC, Wireshark, Tcpdump, herramientas Linux de monitoreo de CPU y tráfico, entornos virtualizados

Alcance y Limitaciones

El estudio se desarrolló en un entorno experimental controlado y virtualizado. Si bien las condiciones implementadas buscan aproximarse a escenarios LTE/4G/VoLTE reales, futuras investigaciones deberán validar los resultados en infraestructuras operativas de producción y bajo tráfico masivo característico de operadores de telecomunicaciones.

Para mejorar la consistencia estadística de los resultados y aproximarse a escenarios de telecomunicaciones con múltiples transacciones concurrentes, se ejecutaron N=1000 iteraciones experimentales para cada escenario criptográfico evaluado.

RESULTADOS

Los experimentos fueron ejecutados en un entorno controlado orientado a escenarios representativos de redes LTE/VoLTE, evaluando el impacto de esquemas criptográficos clásicos, post-cuánticos e híbridos sobre métricas relevantes de telecomunicaciones.

Las pruebas analizaron el comportamiento del protocolo TLS 1.3 bajo condiciones de tráfico concurrente moderado, considerando indicadores asociados a calidad de servicio (QoS) y desempeño de red.

Resultados Experimentales

Latencia del Handshake

La latencia promedio fue calculada mediante la ecuación (1).

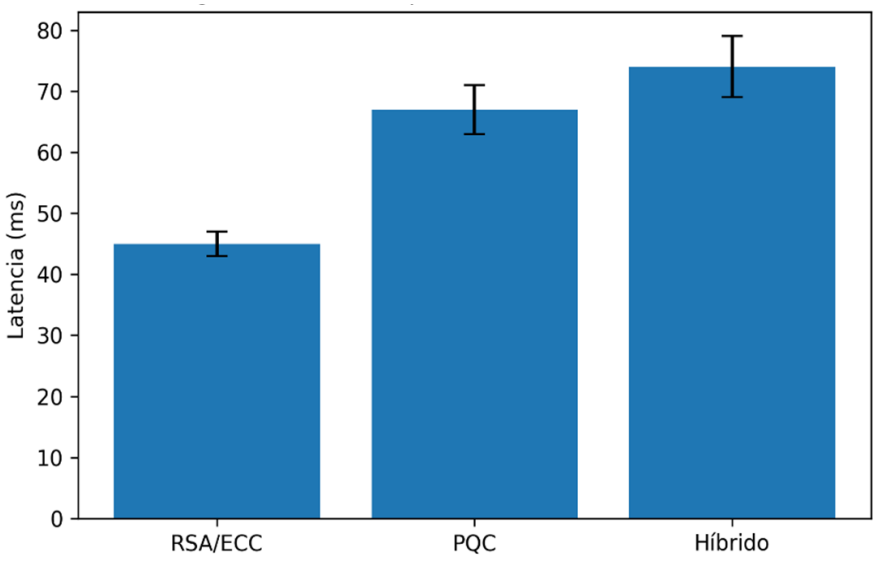
Tabla 2

Latencia promedio del handshake TLS con desviación estándar

Esquema criptográfico	Latencia promedio	Desv. estándar
RSA/ECC	45 ms	±2 ms
PQC	67 ms	±4 ms
Híbrido	74 ms	±5 ms

Figura 1

Latencia promedio del handshake TLS con desviación estándar



Tamaño del Handshake

El tamaño del Handshake fue calculado mediante la ecuación (2)

Tabla 2

Tamaño del Handshake con desviación estándar

Esquema criptográfico	Tamaño handshake	Desv. estándar
RSA/ECC	3.5 KB	±0.2
PQC	9.7 KB	±0.5
Híbrido	12.3 KB	±0.7

Uso de CPU y Recursos Computacionales

La ecuación (3) define el cálculo del uso promedio de CPU.

Tabla 3

Uso de CPU con desviación estándar

Esquema criptográfico	CPU promedio	Desv. estándar
RSA/ECC	22%	±2
PQC	33%	±3
Híbrido	37%	±4

Figura 2

Tamaño promedio del handshake para esquemas clásicos, PQC e híbridos

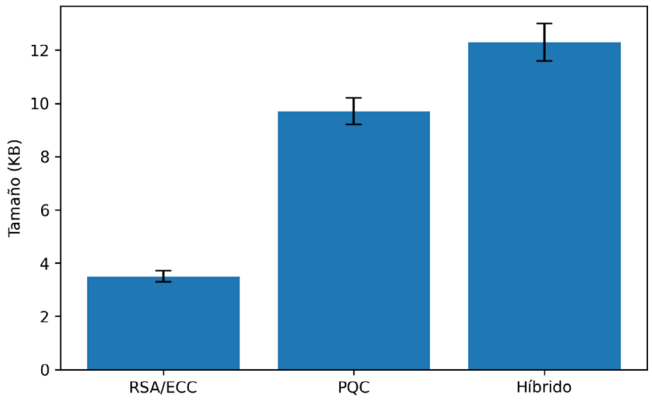
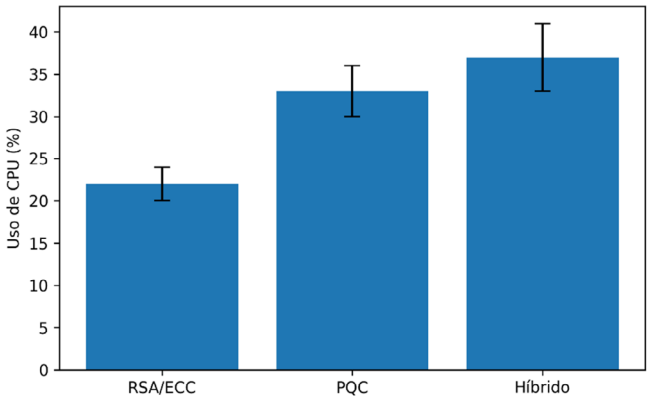


Figura 3

Uso promedio de CPU durante el procesamiento criptográfico



Para todos los resultados experimentales la dispersión de los resultados fue evaluada mediante la ecuación (4).

Impacto sobre KPIs de Telecomunicaciones

Adicionalmente, se analizaron indicadores relevantes de calidad de servicio (QoS).

Tabla 4

Impacto sobre KPIs de Red

KPI	Clásico	PQC	Híbrido
Jitter	Bajo	Medio	Medio-Alto
Throughput	Alto	Medio	Medio
Packet Loss Ratio	Bajo	Bajo	Medio
Estabilidad de sesión	Alta	Media	Media

Interpretación Operativa
Desde una perspectiva práctica, los resultados sugieren que la adopción de criptografía post-cuántica en telecomunicaciones es técnicamente viable, aunque requiere estrategias progresivas de migración y optimización. Los esquemas híbridos representan actualmente la alternativa más realista para operadores, permitiendo compatibilidad entre infraestructuras existentes y nuevos mecanismos criptográficos resistentes a amenazas cuánticas. Sin embargo, el incremento observado en latencia y señalización podría afectar servicios críticos si no se implementan mecanismos adecuados de gestión QoS y crypto-agility.

Consideraciones Estadísticas
Las pruebas fueron ejecutadas mediante múltiples iteraciones experimentales controladas. Si bien el número de iteraciones no representa el volumen masivo de tráfico característico de redes comerciales de telecomunicaciones, los resultados obtenidos permiten identificar tendencias consistentes sobre el comportamiento relativo de los distintos esquemas criptográficos bajo condiciones comparables.

Validación Estadística
Previo al análisis descriptivo, las métricas obtenidas fueron verificadas para identificar consistencia y estabilidad experimental mediante análisis de dispersión y comparación de tendencias entre iteraciones.

Control de Variables

Para garantizar resultados confiables:

- Misma infraestructura en todos los escenarios
- Sin tráfico adicional
- Mismas condiciones de red
- Repetición de pruebas N=1000

Los resultados reflejan que: PQC introduce sobrecarga en: Latencia, Ancho de banda, CPU

Pero dentro de rangos operativamente manejables. El diseño experimental es reproducible y se basa en configuraciones estándar reportadas en la literatura sobre evaluación de PQC. El aumento del número de iteraciones permitió obtener métricas más estables y reducir la variabilidad experimental observada inicialmente.

DISCUSIÓN

Los resultados evidencian que la adopción de criptografía post-cuántica introduce un incremento en la latencia y el consumo de recursos, atribuible al mayor tamaño de claves y a la complejidad computacional de los algoritmos. No obstante, este impacto resulta manejable en infraestructuras modernas. La interoperabilidad se identifica como uno de los principales desafíos, dado que los sistemas clásicos no son directamente compatibles con esquemas PQC. Los enfoques híbridos permiten una transición gradual, aunque incrementan la complejidad.

Asimismo, la migración introduce riesgos asociados a implementaciones inmaduras y configuraciones incorrectas, lo que resalta la necesidad de estrategias de gestión de riesgo.

FRAMEWORK INTEGRAL PARA LA ADOPCIÓN DE CRIPTOGRAFÍA POST-CUÁNTICA EN TELECOMUNICACIONES

Descripción General del Modelo

A partir de los resultados obtenidos, se propone un modelo integral de adopción de criptografía post-cuántica orientado a infraestructuras de telecomunicaciones. Este modelo tiene como objetivo principal optimizar la transición hacia PQC, equilibrando tres dimensiones críticas: Desempeño, Interoperabilidad, Riesgo. El modelo se basa en un enfoque iterativo y adaptativo, alineado con el concepto de crypto-agility, permitiendo la evolución progresiva de los sistemas criptográficos sin afectar la continuidad del servicio. La novedad del trabajo no radica únicamente en el uso del concepto de crypto-agility, ampliamente conocido en la literatura, sino en su integración con métricas QoS, interoperabilidad y restricciones operativas propias de redes LTE, 4G y VoLTE predominantes en Sudamérica.

Estructura del Modelo

El modelo propuesto se compone de cuatro módulos principales:

Módulo de Evaluación de Desempeño
Evalúa el impacto de los algoritmos criptográficos en: Latencia, Uso de CPU, Consumo de ancho de banda. Permite seleccionar algoritmos adecuados según el contexto (ej. 4G vs web).

Módulo de Interoperabilidad
Gestiona la coexistencia entre: Sistemas clásicos, Sistemas PQC, Esquemas híbridos. Incluye: Negociación de algoritmos, Compatibilidad entre nodos, Gestión de claves

Módulo de Gestión de Riesgos
Identifica y mitiga riesgos asociados a la migración: Vulnerabilidades en implementación, Fallos operativos, Riesgos de seguridad. Se apoya en matrices de riesgo (probabilidad vs impacto).

Módulo de Orquestación (Crypto-Agility)
Es el núcleo del modelo: Permite cambiar algoritmos dinámicamente, Soporta múltiples esquemas simultáneamente, Facilita actualizaciones sin interrupciones. Este módulo permite mantener flexibilidad criptográfica frente a futuros cambios tecnológicos.

Flujo del Modelo

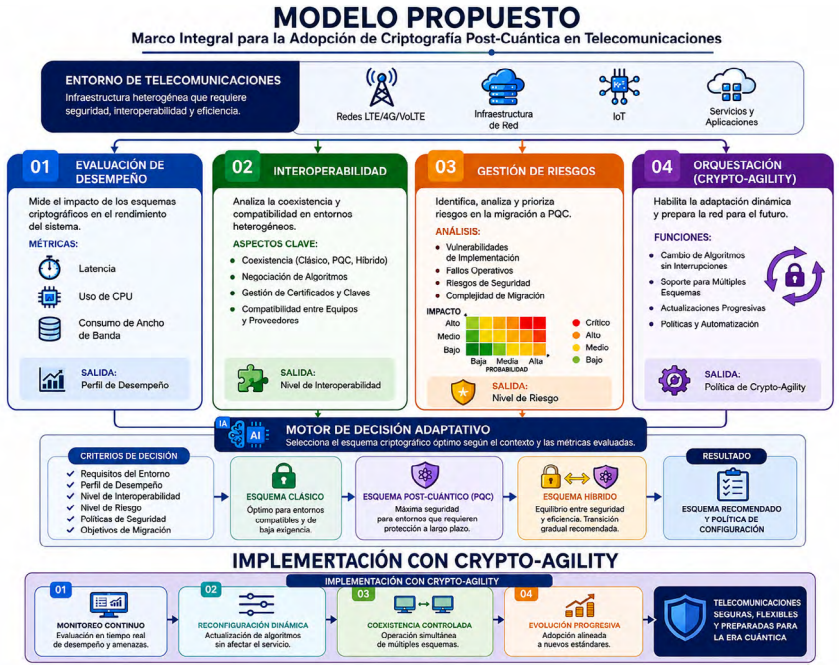
El funcionamiento del modelo sigue el siguiente flujo:

1. Evaluación del entorno (tipo de red, requisitos)
2. Análisis de desempeño
3. Verificación de interoperabilidad
4. Evaluación de riesgos
5. Selección de esquema (clásico, PQC o híbrido)
6. Implementación adaptativa (crypto-agility)

Representación del Modelo

Figura 4

Modelo propuesto



El modelo propuesto incorpora el concepto de crypto-agility como mecanismo central de orquestación, permitiendo la adaptación dinámica de los esquemas criptográficos sin interrumpir la operación del sistema. Esto facilita la coexistencia de algoritmos clásicos y post-cuánticos durante el proceso de transición.

Discusión del Modelo

El modelo propuesto presenta las siguientes ventajas:

Integración multidimensional

- No analiza solo seguridad, sino también: Rendimiento real, Compatibilidad operativa, Riesgo técnico

Aplicabilidad práctica

- Puede ser implementado en: Operadores de telecomunicaciones, Redes LTE, 4G y VoLTE, Infraestructura empresarial

Escalabilidad

- Permite evolucionar conforme: Avancen los estándares PQC, Aparezcan nuevos algoritmos

Mitigación de riesgos

Reduce: Fallos en migración, Decisiones incorrectas

Aporte Científico del Modelo

El modelo propuesto constituye un aporte original al integrar tres dimensiones críticas desempeño, interoperabilidad y riesgo, en un único framework aplicado a telecomunicaciones. A diferencia de enfoques tradicionales, el modelo incorpora un mecanismo de decisión adaptativo basado en crypto-agility, permitiendo seleccionar dinámicamente el esquema criptográfico más adecuado según las condiciones del entorno.

Tabla 5

Comparación con otros trabajos

Trabajo	QoS	LTE/VoLTE	Riesgo	Crypto-Agility	Integral
Otros estudios	✗	✗	✓	✓	✗
Este trabajo	✓	✓	✓	✓	✓

El modelo fue contrastado conceptualmente frente a requerimientos operativos de redes LTE/VoLTE, considerando métricas QoS y restricciones de interoperabilidad presentes en operadores regionales.

CONCLUSIONES

El presente trabajo analizó el impacto de la criptografía post-cuántica en entornos de telecomunicaciones, considerando conjuntamente desempeño, interoperabilidad y riesgo en escenarios representativos de redes LTE/4G/VoLTE predominantes en Bolivia y Sudamérica. Los resultados obtenidos evidencian que la adopción de esquemas criptográficos post-cuánticos introduce incrementos en latencia, sobrecarga de señalización y consumo de recursos computacionales en comparación con mecanismos criptográficos clásicos. Estos efectos se relacionan principalmente con el mayor tamaño de claves y la

complejidad matemática de los algoritmos PQC. Desde la perspectiva de telecomunicaciones, los resultados muestran que dichos incrementos pueden impactar indicadores críticos de calidad de servicio (QoS), particularmente en servicios sensibles al tiempo real como VoLTE, videollamadas y transmisión multimedia.

En consecuencia, la migración hacia criptografía post-cuántica debe ser abordada de manera progresiva y considerando las restricciones operativas de las infraestructuras actualmente desplegadas en la región. Uno de los principales hallazgos del estudio es que las redes LTE/4G/VoLTE predominantes en Bolivia y Sudamérica presentan limitaciones particulares relacionadas con:

- infraestructura heterogénea
- recursos computacionales limitados
- coexistencia de equipamiento heredado
- restricciones presupuestarias para actualización tecnológica

Estas condiciones hacen que una transición inmediata hacia esquemas completamente post-cuánticos resulte compleja para operadores regionales.

En este contexto, los resultados sugieren que los esquemas híbridos representan actualmente la alternativa más viable para una transición gradual, permitiendo mantener compatibilidad con infraestructuras existentes mientras se incorporan mecanismos resistentes a amenazas cuánticas. Como principal contribución, este trabajo propone un modelo integral orientado específicamente a telecomunicaciones, el cual integra evaluación de desempeño, interoperabilidad y riesgo bajo criterios de calidad de servicio y operación de red. A diferencia de enfoques puramente criptográficos, el modelo considera variables técnicas relevantes para entornos LTE/4G/VoLTE reales, aportando una perspectiva aplicada a la realidad operativa regional.

Asimismo, el estudio incorpora el concepto de crypto-agility como mecanismo de adaptación progresiva, permitiendo seleccionar y actualizar esquemas criptográficos sin comprometer la continuidad operativa de los servicios de telecomunicaciones. No obstante, la investigación presenta limitaciones derivadas del uso de un entorno experimental controlado y virtualizado. Futuras investigaciones deberán validar los resultados en redes comerciales reales, incorporando tráfico masivo, movilidad de usuarios y condiciones operativas propias de operadores de telecomunicaciones. Finalmente, el trabajo contribuye a reducir la brecha existente entre la teoría criptográfica y la implementación práctica de criptografía post-cuántica en infraestructuras de telecomunicaciones regionales, proporcionando una base inicial para futuras estrategias de migración segura hacia redes resistentes a amenazas cuánticas.

BIBLIOGRAFÍA

Albrecht, M. R., et al. (2021). On the complexity of LWE. IEEE Transactions on Information Theory, 67(3), 1772–1792.

Alkim, E., et al. (2022). Post-quantum cryptography based on lattices. Springer LNCS.

Basu, S., Roy, T. K., & Ghosh, A. (2022). QoS-aware network slicing in 5G. IEEE Transactions on Network and Service Management, 19(3), 2450–2463.

Bernstein, D. J., Buchmann, J., & Dahmen, E. (2009). Post-quantum cryptography. Springer.

Beullens, W. (2022). Cryptanalysis challenges in PQC. Springer LNCS.

Bindel, N., et al. (2022). Hybrid key exchange in TLS 1.3. ACM Conference on Computer and Communications Security (CCS).

Campagna, M., et al. (2022). Migration strategies for post-quantum cryptography. ACM Computing Surveys, 55(6), 1–36.

Chen, A. C. H., & Lin, B. Y. (2025). Hybrid PQC for V2X communications. IEEE/ACM Conference.

Chen, L., et al. (2022). Report on post-quantum cryptography (NISTIR update). IEEE Security & Privacy, 20(2), 20–29.

Cho, J., Lee, C., Kim, E., Lee, J., & Cho, B. (2024). Software-Defined Cryptography: A Design Feature of Cryptographic Agility.

Demir, E. D., Bilgin, B., & Onbasli, M. C. (2025). Performance analysis of post-quantum cryptographic algorithms. IEEE Access.

Diffie, W., & Hellman, M. (1976). New directions in cryptography. IEEE Transactions on Information Theory, 22(6), 644–654.

Ducas, L., et al. (2022). CRYSTALS-Dilithium: Digital signatures from lattice problems. IEEE Security & Privacy, 20(4), 44–53.

ETSI. (2023). Quantum-safe cryptography and security: Migration guidelines.

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. ACM STOC.

Hohm, J., Heinemann, A., & Wiesmaier, A. (2022). Towards a Maturity Model for Crypto-Agility Assessment (CAMM).

Hoque, M., et al. (2024). Energy-efficient post-quantum cryptography for 5G systems. IEEE Communications Magazine, 62(5), 72–78.

Karmakar, A., et al. (2021). High-performance implementation of post-quantum cryptography. IEEE Transactions on Computers, 70(10), 1607–1622.

Moody, D., et al. (2022). Status report on the second round of the NIST PQC standardization process. NIST.

NIST. (2023). Post-Quantum Cryptography: Selected Algorithms. Oliveira, A., et al. (2024). Integration of quantum-safe cryptography in 5G networks. IEEE Communications Magazine, 62(6), 80–87.

Peikert, C. (2022). Lattice cryptography for the internet. ACM Computing Surveys, 55(1), 1–38.

Rawal, T., & Curry, D. (2024). Impact of post-quantum cryptography on 5G networks. IEEE Communications Magazine, 62(3), 58–64.

Riva-Cambrin, H. A., et al. (2025). Post-quantum authentication systems. IEEE/ACM Workshop.

Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures. Communications of the ACM, 21(2), 120–126.

Wolf, L., Umezulike, S., Öndarö, G., Schinzel, S., & Ising, F. (2026). Practical Evaluation of the Crypto-Agility Maturity Model.

Xu, J., & Li, S. (2021). Hardness assumptions in PQC. IEEE Transactions on Information Theory, 67(8), 5123–5137.

AUTO-REFLEXIÓN Y RAG EN MODELOS DE LENGUAJE PEQUEÑOS PARA CONOCIMIENTO EMPRESARIAL: UN ESTUDIO DE MAPEO SISTEMÁTICO

M.Sc. Alcides Yohacin Leaños Rodríguez
Posgrado SOE – UAGRM
https://orcid.org/0009-0008-3208-3898
Santa Cruz, Bolivia | alcides@dualbiz.net

https://doi.org/10.23670 FT.2026.1.42

Recibido 28/04/2026 - Aceptado 13/05/2026

RESUMEN

Las organizaciones requieren sistemas de inteligencia artificial capaces de razonar sobre bases de conocimiento internas sin comprometer la seguridad. Sin embargo, los modelos de lenguaje grandes (LLM) no acceden de forma nativa a información confidencial, y su despliegue local suele ser inviable debido a los costos de hardware. En este contexto, los modelos de lenguaje pequeños (SLM, 1B–13B parámetros) emergen como una alternativa viable, aunque su capacidad para soportar pipelines de generación aumentada por recuperación (RAG) con mecanismos de auto-reflexión en entornos empresariales aún no está plenamente establecida. Este estudio analiza mediante un estudio de mapeo sistemático (SMS) de la literatura en trabajos que emplean o integran componentes como RAG y auto-reflexión aplicados al soporte de conocimiento. Siguiendo las directrices de Kitchenham y Charters (2007) y Petersen et al. (2008), se examinaron publicaciones de ar Xiv, NeurIPS, IEEE Xplore y ACM entre 2020 y 2025. De un total de 510 resultados iniciales, se seleccionaron 40 estudios primarios; los resultados evidencian un creciente interés en arquitecturas que combinan recuperación densa con estrategias adaptativas basadas en incertidumbre, donde un alto porcentaje de los trabajos se construyen sobre la integración de componentes existentes. Asimismo, se identifican limitaciones en la evaluación, la eficiencia y la aplicabilidad en escenarios empresariales reales. Se concluye que estas técnicas muestran potencial para mejorar el razonamiento y la adaptación de los modelos, aunque persisten desafíos para su implementación en SLM bajo restricciones de privacidad y hardware. El estudio organiza el conocimiento existente y establece una base estructurada para futuras investigaciones en el contexto mencionado.

Palabras clave: RAG, Auto-Reflexión, Auto-correccion, Auto-induccion, LLM, SLM, Agentes autónomos, Agentes Inteligentes

ABSTRACT

Organizations require artificial intelligence systems capable of reasoning over internal knowledge bases without compromising security. However, large language models (LLMs) do not natively access confidential information, and their local deployment is often unfeasible due to hardware costs. In this context, small language models (SLMs, 1B–13B parameters) emerge as a viable alternative, although their capacity to support retrieval-augmented generation (RAG) pipelines with self-reflection mechanisms in enterprise environments has not yet been fully established. This study analyzes, through a systematic mapping study (SMS), the literature on works that employ or integrate components such as RAG and self-reflection. Following the guidelines of Kitchenham and Charters (2007) and Petersen et al. (2008), publications from arXiv, NeurIPS, IEEE Xplore, and ACM between 2020 and 2025 were examined. From an initial pool of 510 results, 40 primary studies were selected; the findings reveal a growing interest in architectures that combine dense retrieval with uncertainty-based adaptive strategies, where a high percentage of works are built upon the integration of existing components. Likewise, limitations are identified in evaluation, efficiency, and applicability in real enterprise scenarios. It is concluded that these techniques show potential for improving model reasoning and adaptation, although challenges persist for their implementation in SLMs under privacy and hardware constraints. The study organizes existing knowledge and establishes a structured foundation for future research.

Keywords: RAG, Self-Reflection, self-induction, LLM, SLM, autonomous agent, intelligent agent

INTRODUCCIÓN

La rápida adopción de la Inteligencia Artificial (IA) en entornos organizacionales ha generado una demanda creciente de sistemas inteligentes capaces de operar eficazmente sobre bases de conocimiento internas y específicas del dominio. En contextos industriales donde los datos operativos confidenciales, los procesos propietarios, la documentación regulatoria y el conocimiento institucional no están disponibles públicamente. Los Modelos de Lenguaje Grandes (LLMs, Large Language Models) de propósito general enfrentan una limitación estructural crítica: son entrenados sobre entornos de la nube y en algunos casos se ha visto el uso de manera pública información privada, por tanto existen brechas de seguridad o se torna complicado el acceso inherente a la información privada para una fácil retroalimentación, misma que se califica como sensible para la organización y que es la que impulsa la toma de decisiones cotidiana.

La Generación Aumentada por Recuperación conocida como RAG (Retreival Augmented Generation) ha ganado atención como mecanismo para anclar las respuestas de los modelos de lenguaje en documentos relevantes y actualizados recuperados de repositorios internos. Las arquitecturas RAG disocian el conocimiento paramétrico incrustado en los pesos del modelo del conocimiento fáctico almacenado en fuentes externas, permitiendo que los modelos generen respuestas informadas por contenido específico de la organización sin requerir re-entrenamiento o ajuste fino costosos. Sin embargo, si bien RAG extiende considerablemente el alcance funcional de los modelos de IA hacia dominios de conocimiento privados, introduce una nueva clase de problemas: el modelo puede recuperar información irrelevante o incompleta, no reconocer los límites de su propio conocimiento, o producir respuestas confiadas pero incorrectas, fenómeno conocido comúnmente como alucinación.

Complementario al RAG el mecanismo de auto-reflexión, la capacidad de un modelo de lenguaje para introspeccionar sobre su propio razonamiento, identificar inconsistencias y refinar iterativamente sus respuestas ha emergido como una dirección prometedora para mejorar la confiabilidad de las respuestas y la calibración epistémica. Los agentes auto-reflexivos, como los descritos en el marco Reflexión (Shinn et al., 2023) o mediante las arquitecturas Self-RAG (Asai, Wu, et al., 2024), son capaces de evaluar la calidad de la evidencia recuperada, detectar brechas lógicas en las respuestas generadas y desencadenar ciclos correctivos de recuperación o razonamiento. Estas capacidades son particularmente valiosas en entornos empresariales donde las respuestas incorrectas o mal fundamentadas pueden tener consecuencias operativas, legales o financieras.

Un desafío significativo y poco explorado concierne al despliegue de estos mecanismos en modelos de lenguaje pequeños con recursos limitados. Si bien los LLM's de frontera (ej. GPT-4, Claude 3, Gemini Ultra) poseen capacidad paramétrica suficiente para implementar comportamientos complejos de razonamiento y reflexión, las organizaciones frecuentemente operan bajo restricciones de recursos (hardware) que impiden el despliegue de modelos muy grandes. Los Modelos de Lenguaje Pequeños (SLM, Small languaje Models) modelos en el rango de 1B a 13B (1.000 a 13.000 millones de parámetros); son más viables para el despliegue en infraestructuras propias, con preservación de la privacidad y eficiencia de costos. No obstante, si estos modelos más pequeños pueden inducir eficazmente comportamientos auto-reflexivos y aprovechar pipelines RAG para el soporte de conocimiento interno sigue siendo una pregunta abierta y crítica. A pesar del creciente número de trabajos sobre arquitecturas RAG y agentes LLM auto-reflexivos, la intersección de estos mecanismos en el contexto de modelos de lenguaje pequeños o entornos con recursos limitados, orientados al conocimiento empresarial interno, sigue siendo un campo insuficientemente explorado. Los estudios de revisión existentes se centran principalmente en modelos de gran escala en contextos generales o en RAG como problema de recuperación de información, sin considerar las restricciones específicas de los despliegues industriales.

Las organizaciones de sectores como manufactura, finanzas, salud, logística y servicios profesionales gestionan grandes volúmenes de conocimiento interno, incluyendo documentación operativa, procesos, registros regulatorios e información histórica. En este contexto, el desarrollo de sistemas de inteligencia artificial capaces de acceder, interpretar y razonar sobre esta información de forma segura representa una necesidad crítica para este sector. Desde una perspectiva teórica, la integración de RAG con mecanismos de auto-reflexión en modelos pequeños plantea desafíos relevantes en términos de razonamiento, adaptación y eficiencia bajo restricciones de recursos. Desde el punto de vista práctico, esta combinación podría habilitar asistentes inteligentes desplegados localmente, preservando la privacidad de los datos y reduciendo la dependencia de infraestructuras externas. Asimismo, su pertinencia radica en la creciente demanda de soluciones de IA accesibles y seguras en entornos empresariales con limitaciones de hardware.

En este marco, la ausencia de evidencia consolidada justifica el desarrollo del presente estudio de mapeo sistemático SMS (Systematic Mapping Study), con el propósito de analizar la literatura existente sobre trabajos que emplean o integren componente de RAG y mecanismos de auto-reflexión en modelos de lenguaje aplicados al soporte de conocimiento. En particular, el estudio revisa propuestas, enfoques y tendencias relacionadas con el aprendizaje inductivo, la auto-reflexión y la generación aumentada, con el fin de identificar patrones, limitaciones y oportunidades de investigación en este campo; para posteriormente extrapolar esa información y establecer una línea sobre la aplicabilidad con los SML's y las bases de conocimiento.

Para lo cual se pretende:

(OE1) Caracterizar el panorama de métodos, técnicas y arquitecturas disponibles que permiten el aprendizaje inductivo y el comportamiento auto-reflexivo en agentes inteligentes que operan dentro de sistemas basados en RAG, especificaciones de entrada-salida y las métricas de desempeño utilizadas para evaluar los enfoques propuestos, con énfasis en aquellas aplicables a tareas de conocimiento específicas del dominio o de entornos empresariales.

(OE2) Clasificar los tipos de enfoques propuestos o utilizados, incluyendo modelos, marcos de trabajo, herramientas y guías y evaluar la madurez de sus contribuciones reportadas.

(OE3) Examinar los métodos, técnicas y herramientas existentes (p. ej., modelos LLM base, módulos de razonamiento) que se combinan o extienden dentro de las arquitecturas relevadas.

(OE4) Mapear las adaptaciones y mejoras propuestas sobre mecanismos existentes de recuperación, reflexión o aprendizaje, y evaluar su fundamentación teórica y empírica.

(OE5) Evaluar la relevancia y aplicabilidad de los métodos identificados en el entorno empresarial, particularmente para el soporte de sistemas de información interna y bases de datos de trabajo.

(OE6) Identificar limitaciones, desafíos abiertos y direcciones de investigación futura relacionados con el despliegue de sistemas RAG auto-reflexivos en modelos de lenguaje pequeños o con recursos limitados.

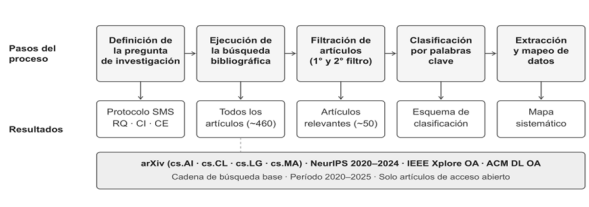
METODOLOGÍA

Diseño del estudio

Como se menciona anteriormente, el trabajo adopta la metodología de Estudio de Mapeo Sistemático SMS, siguiendo las directrices establecidas por Kitchenham y Charters (2007) y el protocolo extendido propuesto por Petersen et al., (2008). El SMS es particularmente adecuado para el propósito de esta investigación, dado que el campo de los agentes LLM auto-reflexivos con RAG es relativamente reciente y dinámico, donde se requiere una visión estructurada del estado del arte antes de proceder a una revisión sistemática de mayor profundidad. A diferencia de una Revisión Sistemática de Literatura (SLR, Systematic Literature Review), el mapeo sistemático no busca sintetizar evidencia cuantitativa o meta-analítica, sino categorizar y clasificar el espacio de investigación existente, identificar tendencias temporales y detectar brechas de conocimiento que orienten investigaciones futuras. La ejecución del estudio comprende cinco fases secuenciales: (1) definición del protocolo de búsqueda, (2) ejecución de la búsqueda y exportación de resultados, (3) primera filtración mediante lectura de título y resumen, (4) segunda filtración mediante lectura completa del texto, y (5) extracción de datos y análisis. El proceso completo es documentado en una hoja de trabajo estructurada que preserva la trazabilidad de cada decisión de inclusión y exclusión, como indica la siguiente figura.

Figura 1

Proceso de Estudio de Mapeo Sistemático (adaptado de Petersen et al., 2008)



Fuentes de información y cadenas de búsqueda

Dado que el acceso institucional a bases de datos de pago no se encuentra disponible en el contexto de la presente investigación, la selección de fuentes de información priorizó repositorios de acceso abierto (Open Access) con cobertura relevante para el campo de la inteligencia artificial y el procesamiento del lenguaje natural. Las fuentes seleccionadas son:

- arXiv.org (categorías cs.AI, cs.CL, cs.LG, cs.MA): repositorio primario de preprints en IA/NLP, donde se publican la mayoría de los trabajos relevantes al área antes de su aparición en conferencias y revistas.
- NeurIPS Proceedings (proceedings.neurips.cc): actas de la Conferencia sobre Sistemas de Procesamiento de Información Neural, disponibles en acceso abierto completo para los años 2020–2024. (2025 aún no se tiene información total de documentos Acceso Abierto, a fecha del presente estudio Mayo del 2026).
- IEEE Xplore con filtro de Acceso Abierto activo: publicaciones indexadas en IEEE Transactions on Artificial Intelligence y conferencias asociadas, restringidas a artículos con texto completo libremente accesible. (2025 aún no se tiene información total de documentos Acceso Abierto, a fecha del presente estudio Mayo del 2026).
- ACM Digital Library con filtro de Acceso Abierto activo: actas de conferencias ACM relevantes (AAAI, IJCAI, ACL, EMNLP, SIGKDD), restringidas a publicaciones con texto completo accesible sin suscripción.

La cadena de búsqueda base utilizada en todas las fuentes es la siguiente:

Search Strings: ("intelligent agent" OR "autonomous agent" OR "LLM agent" OR "AI agent") AND ("retrieval-augmented generation" OR "RAG" OR "retrieval augmented") AND ("self-reflection" OR "self-correction" OR "self-refinement" OR "self-critique" OR "introspection") AND ("inductive learning" OR "inductive reasoning" OR "learning from experience" OR "in-context learning")

Esta cadena fue adaptada según la sintaxis y los campos de búsqueda específicos de cada fuente. En arXiv se aplicó sobre título y resumen (ti_abs), en NeurIPS mediante búsqueda de texto completo en las actas. en IEEE Xplore y ACM DL mediante el campo All Metadata con el filtro de acceso abierto activado. No incluye SLM (Modelos de lenguaje pequeños) ni bases de conocimiento empresarial por que la idea central del estudio es extrapolar la información obtenida a este contexto.

Periodo de Búsqueda y Tipo de Publicación

La justificación temporal del período de búsqueda 2020–2025 se fundamenta en que el paradigma RAG fue formalmente introducido por Lewis et al., (2020) en NeurIPS 2020, constituyendo el punto de partida conceptual del campo. El límite superior de 2025 permite capturar el ciclo completo de consolidación del paradigma de agentes LLM y los trabajos más recientes sobre reflexión y aprendizaje inductivo en agentes pequeños, siendo mayo del 2026 la fecha de realización del SMS.

Período: 2020-2025. El límite inferior coincide con la introducción formal del paradigma RAG (expuesto anteriormente); el límite superior captura el aumento total de la investigación de agentes LLM 2025(NeurIPS y IEEE Explore restringen aún el Acceso Abierto a la gestión 2025 hasta mayo del 2026 fecha de realización del estudio).

Tipo de publicación: Artículos de revistas revisado por pares, artículos de conferencias y artículos de talleres.

Idioma: Sólo inglés.

Criterios de inclusión y exclusión

La selección de estudios se rige por los criterios definidos en el protocolo y detallados en la Tabla 1.

Tabla 1

Criterios de inclusión y exclusión del estudio de mapeo sistemático

Criterio	Descripción
CI1	El estudio aborda aprendizaje inductivo, aprendizaje desde contexto recuperado, o auto-mejora en agentes inteligentes.
CI2	El estudio propone o analiza una arquitectura o método basado en RAG aplicable a sistemas de agentes inteligentes.
CE3	El estudio es únicamente un resumen (sin texto completo disponible).
CI4	El estudio presenta técnicas de razonamiento, planificación o memoria que habiliten la adaptabilidad o generalización del agente.
CI5	El estudio reporta resultados empíricos, benchmarks o aplicación en entornos reales.
CI6	El texto completo del artículo es libremente accesible (Acceso Abierto).
CE1	El estudio no aborda RAG, auto-reflexión, ni aprendizaje inductivo en agentes.
CE2	El estudio carece de resumen.
CE3	El estudio es únicamente un resumen (sin texto completo disponible).
CE4	El estudio no está escrito en inglés.
CE5	El estudio es un duplicado o versión anterior de un trabajo ya incluido.
CE6	El estudio no fue sometido a revisión por pares (editoriales, resúmenes de keynotes, tutoriales, informes técnicos no arbitrados).
CE7	El texto completo del artículo no es libremente accesible.
CE8	El estudio no está relacionado con IA, NLP o sistemas de agentes inteligentes.
CE9	El estudio es un trabajo secundario (revisión sistemática, mapeo, meta-análisis) y no un estudio primario.
CE10	El estudio se enfoca exclusivamente en recuperación de información clásica sin contexto de LLM, agente o aprendizaje.
CE11	El estudio fue publicado antes de 2020 (paradigma pre-RAG).

Proceso de selección de estudios

El proceso de selección opera en dos fases secuenciales de filtración(cribado), tal como se ilustra en la Figura 1.

Primera filtración: Los registros recuperados de las cuatro fuentes son exportados en formato RIS y consolidados en una hoja de trabajo estructurada. Tras la eliminación de duplicados (entradas que aparecen en más de una fuente con el mismo DOI, título o identificador arXiv) se procede a la lectura del título y resumen de cada registro. En esta fase se aplican los criterios de inclusión CI1–CI4 y los criterios de exclusión CE1, CE4–CE11 para determinar si el estudio es potencialmente relevante o debe ser descartado.Los trabajos clasificados como inciertos son conservados para la segunda filtración.

Segunda filtración: Los estudios que superan la primera filtración son sometidos a lectura completa del texto. En esta fase se aplican todos los criterios de inclusión (CI1–CI6) y exclusión (CE1–CE11) con mayor precisión. Se registra el criterio de decisión aplicado para cada trabajo incluido o excluido, preservando la trazabilidad del proceso.

Preguntas de investigación

Las preguntas de investigación guían tanto el proceso de extracción como el análisis posterior de los estudios incluidos. En la Tabla 2, se presentan las seis preguntas definidas para este mapeo; organizadas según el aspecto de investigación que abordan.

Tabla 2

Preguntas de Investigación y su correspondencia con los objetivos del estudio

Pregunta de Investigación	Objetivo
PI1 – ¿Qué métodos, técnicas y arquitecturas habilitan el aprendizaje inductivo y la auto-reflexión en agentes dentro de sistemas RAG, y qué patrones se observan en el uso de métricas para su evaluación en el contexto empresarial?	OE1 – Caracterizar el panorama de métodos, técnicas y arquitecturas, incluyendo métricas de desempeño aplicables a tareas de conocimiento empresarial.
PI2 – ¿Qué tipos de contribuciones se han propuesto –modelos, marcos de trabajo, herramientas o guías– y qué nivel de madurez tecnológica evidencian?	OE2 – Clasificar los tipos de enfoques propuestos y evaluar la madurez de sus contribuciones reportadas.
PI3 – ¿Qué métodos, herramientas y componentes de modelos LLM base, Tipo de arquitecturas RAG o Medios de almacenamiento combinados o extendidos se emplean?	OE3 – Examinar los métodos, técnicas y herramientas existentes (p. ej., modelos LLM base, módulos de razonamiento) que se combinan o extienden dentro de las arquitecturas relevadas.
PI4 – ¿Qué adaptaciones y mejoras se proponen sobre mecanismos existentes de recuperación, reflexión o aprendizaje, y cuál es su fundamentación teórica y empírica?	OE4 – Mapear las adaptaciones y mejoras propuestas sobre mecanismos existentes y evaluar su fundamentación.
PI5 – ¿En qué dominios de aplicación y tipos de tareas empresariales o bases de conocimiento se han evaluado o desplegado los enfoques relevados, y con qué resultados observados?	OE5 – Evaluar la relevancia y aplicabilidad de los métodos identificados en entornos industriales para el soporte de conocimiento interno.
PI6 – ¿Cuáles son las limitaciones, desafíos abiertos y direcciones de investigación futura en el despliegue de sistemas RAG auto-reflexivos sobre modelos de lenguaje pequeños (13B parámetros)?	OE6 – Identificar limitaciones, desafíos abiertos y direcciones de investigación futura relacionados con el despliegue de sistemas RAG auto-reflexivos en modelos de lenguaje pequeños o con recursos limitados.

Extracción de datos

Los datos extraídos de cada estudio incluido se registran en la hoja de extracción estructurada, correspondiente a los elementos definidos en las preguntas de investigación (P1–P6). Los campos de extracción comprenden, entre otros: los autores y afiliaciones institucionales; el tipo de publicación (conferencia, revista, taller, preprint); el método, técnica o arquitectura propuesta; el tipo de contribución (modelo, marco de trabajo, herramienta, directrices, lecciones aprendidas); los mecanismos preexistentes utilizados como base; las adaptaciones o mejoras propuestas; las métricas y benchmarks empleados; el dominio de aplicación; y la madurez tecnológica de la contribución.

Análisis y síntesis

Los datos extraídos son analizados mediante técnicas de análisis de contenido cualitativo y cuantitativo. Se construyen tablas de frecuencia para caracterizar la distribución temporal de las publicaciones, la distribución por fuente de búsqueda, el tipo de contribución y la madurez tecnológica de los enfoques reportados.

- Metadata (ID, año, origen, tipo de publicación, autores, título)
- País y Tipo Afiliación
- Tipo de Artículo (journal, conference, workshop)
- Preguntas de Investigación P1–P6 cubriendo métodos, Arquitecturas, componentes, entradas/salidas, descritas en Tabla 2.
- Criterios de Calidad para QC1–QC5: Claridad, diseño del estudio, método científico de evaluación, método de investigación, clasificación del artículo.

Tabla 3

Criterios de Calidad

Código	Criterio	Escala
QC1	Claridad en Objetivo de Investigación	1 = No; 2 = Yes
QC2	Diseño del Estudio	1 = Empiric; 2 = Experience report; 3 = Theoretical
QC3	Método de evaluación científica	1 = None; 2 = Example; 3 = Experience; 4 = Feasibility/pilot; 5 = Full analysis
QC4	Método de Investigación	1 = None; 2 = Survey; 3 = Action research; 4 = Case study; 5 = Experiment
QC5	Clasificación	1 = Experience report; 2 = Opinion; 3 = Method/technique/tool proposed; 4 = Proposed + academic use; 5 = Proposed + industry cases

La puntuación de calidad siguió una escala de hasta un máximo de 5 puntos por criterio, distribuidos según el criterio de acuerdo a la Tabla 3 (Ej. QC1 el máximo es 2, QC2 máximo 3, etc); lo que arroja una puntuación de calidad máxima de 20 por artículo. Los resultados son visualizados mediante gráficos de barras, gráficos circulares y mapas de burbujas que permiten identificar tendencias, concentraciones y vacíos en el espacio de investigación (Figura 2, Resultados). La clasificación de la madurez tecnológica de cada contribución sigue la escala propuesta en el protocolo original del SMS de referencia, distinguiendo entre: (1) Concept Formulation (propuesta teórica sin validación empírica sustancial), (2) Development and Extension (propuesta

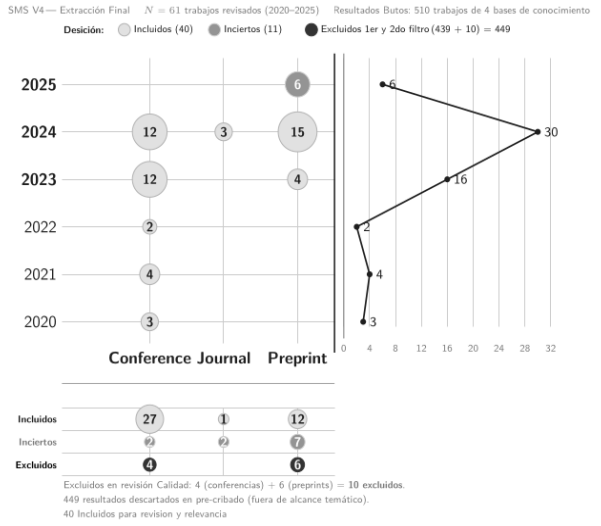
con implementación y evaluación inicial), (3) Internal Enhancement and Exploration (mejora sobre trabajo propio previo), y (4) External Enhancement and Exploration (extensión o validación sobre trabajos de terceros).

RESULTADOS Y DISCUSIÓN

Esta sección presenta los resultados del SMS, organizados según el proceso de descrito en la Figura 1 y estructurados en función de las preguntas de investigación. A partir del análisis de los estudios primarios seleccionados, se examinan los enfoques relacionados con RAG, aprendizaje inductivo y mecanismos de auto-reflexión/auto-corrección en sistemas de agentes inteligentes. La búsqueda abarcó las cuatro bases de datos mencionadas en el capítulo de fuentes de investigación: arXiv, NeurIPS Proceedings, IEEE Xplore (OA) y ACM Digital Library (OA), obteniendo 510 resultados brutos (raw search). Tras la aplicación del proceso descrito (CI/CE, 1er y 2do filtro) se obtienen 61 artículos sobre los cuales luego de aplicar los Criterios de Calidad (Tabla 3) resultan finalmente 40 trabajos primarios incluidos para el desglose de resultados, 11 inciertos (pendientes de verificación del texto completo como selección para trabajos futuros) y 10 excluidos que se presentan en la figura siguiente, diagrama general del estudio final.

Figura 2

Mapa de distribución del estudio principal incluyendo el tipo de publicación y el año



La Figura 2 se puede observar de manera general el proceso de selección, revisando en el panel derecho se observa claramente un crecimiento de estudios relevantes a partir del 2020 cuando fue introducido el RAG y también algoritmos de auto-reflexión mostrando una tendencia en la implentación de estos componentes en distintos ámbitos. En las subsecciones siguientes se presentan los resultados organizados por cada pregunta de investigación (nota. Tomar en cuenta la caída de esta tendencia del 2025 debido a que aún no se tiene información de acceso abierto para dos bases de conocimientos Neuro IPS y IEEE Xplore , a la fecha del estudio; Mayo del 2026).

Mapas de Red de Co-Ocurrencias y Co-Autores, exportados de VOSviewer se presentan en los Anexos respectivamente Anexo 1 y Anexo 2; donde se observa a Yojiu Yang de la Universidad de Tsindghua y Lewis como referentes en trabajos sobre Auto-reflexión y RAG respectivamente al igual que se puede observar RAG como referente principal para trabajos de bases de conocimiento seguido de Auto-reflexión y Agentes Autónomos.

Evaluación de la calidad

Se utilizó la base de 61 trabajos (resultados de las fases de cribado), para revisión de calidad como se menciona en el punto 3, fueron evaluados utilizando los cinco criterios de calidad (Tabla 3,QC1–QC5) que abarcan la claridad del objetivo, el diseño del estudio, la metodología de evaluación, el rigor del método de investigación y la categoría del artículo. Solo 11 trabajos se calificaron como inciertos los cuales pueden ser contribuciones válidas para trabajos futuros. De Los 40 artículos incluidos, el 97.5 % de los artículos incluidos obtuvieron una puntuación de 13 o superior, lo que indica que la literatura incluida es metodológicamente sólida y adecuada para la síntesis. La puntuación media de calidad de 16.4/20 refleja el predominio de artículos de congresos revisados por pares de los principales foros (NeurIPS, ICLR, ACL, EMNLP, NAACL, KDD, SIGIR) y preimpresiones de arXiv de alta calidad con sólidas bases empíricas.

Métodos, técnicas y arquitecturas (PI1)

Sobre los 40 estudios incluidos se agrupan en cinco categorías arquitectónicas principales; permitiendo caracterizar los trabajos identificando los siguientes patrones en el diseño de sistemas autónomos con los componentes de interés para el estudio: RAG como componente fundamental, arquitecturas de auto-reflexión puras y otros trabajos híbridos que se plantean y desglosan mas adelante en tendencias (PI4).

Las arquitecturas basadas en RAG constituyen el grupo predominante (19 papers, 47.5%), abarcando desde la recuperación densa fundamental ;(Guu et al., 2020);(Karpukhin et al., 2020); hasta variantes adaptativas y correctivas Self-RAG (Asai, Wu, et al., 2024), CRAG (Yan et al., 2024), FLARE, DRAGIN, UAR (Zhuge et al., 2024), IRCoT (Trivedi et al., 2024), Modular RAG, Graph RAG (Edge et al., 2024), HippoRAG (Jiménez Gutiérrez et al., 2024), RETRO (Borgeaud et al., 2022) y REPLUG (Shi et al., 2024), evidenciando ser la componente fundamental para bases de conocimiento. En segundo lugar, las arquitecturas centradas en auto-reflexión y auto-corrección representan el 20% (8 papers) incluyendo enfoques como Reflexion (Shinn et al., 2023), Self-Refine (Madaan et al., 2024), CRITIC, RCI (Kim, Baldi, and McAleer 2023), Check Your Facts, Verify-and-Edit, REFINER (Paul et al., 2024); igualmente indica una fuerte tendencia hacia la mejora iterativa impulsada por el estudio de los LLMs y el Auto-aprendizaje (Jiang et al., 2024) otro de los componentes importante para enfoques híbridos. Los Marcos de trabajo basados en Agentes o combinados (6 papers 15%) destacando

ReAct, ExpeL (A. Zhao et al., 2024), MetaGPT (Hong et al., 2024), AutoGen, Generative Agents (Park et al., 2023), AgentBench (Liu et al., 2024) Las arquitecturas de razonamiento y planificación (4 papers, 10%) Árbol de Pensamientos (Yao, Yu, et al., 2023), STaR (Zelikman et al., 2022), PAL (L. Gao et al., 2023), ToRA (Gou, Shao, Gong, Shen, et al., 2024). Mientras que los enfoques modulares y basados en grafos de conocimiento constituyen el 7.5% (3 papers) Graph RAG (Edge et al., 2024), HippoRAG (Jiménez Gutiérrez et al., 2024), Modular RAG.

Tipo de contribución (PI2)

Del total de estudios incluidos, la gran mayoría (36 estudios, 90%) corresponde a marcos de trabajo o arquitecturas, es decir, propuestas que integran componentes existentes principalmente modelos de lenguaje, mecanismos de recuperación y módulos de razonamiento en flujos de trabajo estructurados. En contraste, solo dos estudios proponen nuevos modelos base de recuperación DPR (Karpukhin et al., 2020), REALM (Guu et al., 2020) y dos proponen herramientas de auto-evaluación KILT, AgentBench (Liu et al., 2024).

Esta distribución permite agrupar las contribuciones en tres categorías principales: (i) arquitecturas integradoras, (ii) modelos base y (iii) herramientas de evaluación. La predominancia de la primera categoría evidencia que el campo se orienta principalmente hacia la construcción de sistemas mediante la combinación y adaptación de componentes existentes, en lugar del desarrollo de nuevos modelos desde cero. Asimismo, esta concentración sugiere un nivel de madurez intermedio, donde la investigación se enfoca en la aplicación, integración y optimización de tecnologías disponibles, más que en la innovación fundamental a nivel de modelo. Este patrón es consistente con la evolución reciente de los sistemas RAG y agentes inteligentes, caracterizada por la modularidad y la reutilización de componentes.

Componentes existentes combinados (PI3)

Esta sección examina los componentes existentes que son combinados e integrados en las arquitecturas propuestas, con el objetivo de determinar los bloques de construcción predominantes en sistemas RAG con capacidades de auto-reflexión.Los resultados muestran una fuerte convergencia hacia un conjunto común de componentes reutilizables. En primer lugar, los modelos de lenguaje de gran escala (LLM) constituyen el núcleo de los sistemas analizados, siendo utilizados como motor principal de razonamiento en 38 de los 40 estudios. Entre los modelos más empleados se encuentran GPT-3/4, LLaMA, PaLM, T5 y BART, lo que evidencia una alta dependencia de capacidades paramétricas preentrenadas. En segundo lugar, los mecanismos de recuperación densa representan el componente complementario más extendido, presentes en 31 estudios. Técnicas como DPR (Karpukhin et al., 2020) y BM25, junto con sistemas de indexación como FAISS, se consolidan como el estándar de facto para la recuperación de información relevante. Este patrón se observa de manera consistente en arquitecturas como Self-Rag (Asai, Wu, et al., 2024) CRAG (Yan et al., 2024), IRCoT (Trivedi et al., 2024), HippoRAG (Jiménez Gutiérrez et al., 2024) y FLARE. Los almacenes de vectores FAISS, ToolFormer (Schick et al., 2023) y los grafos de conocimiento NetworkX (Lewis et al., 2021), estructuras de estilo Neo4j se combinan en 12 artículos y Los módulos de memoria (buffers episódicos, almacenes a largo plazo, QA pregenerada de estilo PAQ) aparecen en 9 artículos como mecanismos de soporte.Finalmente Las API y herramientas externas se integran en 18 artículos, particularmente a partir de 2022, lo que refleja el cambio de RAG puro hacia agentes aumentados con herramientas.

En conjunto, estos resultados evidencian una arquitectura recurrente basada en la combinación de cuatro componentes principales: modelos de lenguaje, mecanismos de recuperación, herramientas externas y módulos de memoria. Esta configuración refleja una tendencia hacia sistemas modulares y altamente integrados, donde la funcionalidad emerge de la orquestación de componentes especializados. Otro factor importante cabe resaltar el uso de modelos como PaLM y T5 que trabajando sobre el rango de los modelos de lenguaje pequeños (< 13B).

Adaptaciones y mejoras (PI4)

Esta sección mapea las principales adaptaciones y mejoras propuestas sobre los mecanismos de recuperación, reflexión y aprendizaje en sistemas RAG con capacidades de auto-reflexión, con el fin examinar las mejoras y los patrones de evolución en el diseño de estas arquitecturas.

A partir de los estudios analizados, se identifican tres líneas principales de mejora que estructuran el desarrollo reciente del campo

Recuperación adaptativa o condicional

Una primera línea se centra en la activación dinámica de los procesos de recuperación. A diferencia de los enfoques tradicionales de recuperación estática o en intervalos definidos, propuestas como FLARE, DRAGIN, CRAG, UAR y Self-RAG activan la recuperación solo cuando el modelo detecta incertidumbre, lagunas de conocimiento o tokens de baja confianza. Esto reduce la latencia y el ruido de los pasajes recuperados irrelevantes. REALM y REPLUG amplían esto aún más al permitir la mejora de la recuperación auto-supervisada, donde la propia señal de pérdida del modelo de lenguaje entrena al recuperador sin supervisión de recuperación etiquetada.

Autocorrección fundamentada en evidencia externa

Una segunda línea de investigación aborda las limitaciones de la auto-reflexión puramente paramétrica mediante la incorporación de mecanismos de verificación basados en evidencia externa. En este contexto, enfoques como CRITIC, Verify-and-Edit, Check Your Facts y RCI integran procesos de validación que utilizan documentos recuperados o herramientas ejecutables para corregir la salida generada. Este

enfoque reduce el riesgo de retroalimentación ilusoria, característica de los sistemas auto-reflexivos sin anclaje externo, al vincular las revisiones con fuentes verificables.

Razonamiento iterativo y multinivel sobre evidencia recuperada

La tercera línea se orienta al desarrollo de mecanismos de razonamiento más complejos, que combinan recuperación y generación en procesos iterativos. Enfoques como IRCoT intercalan la recuperación con cadenas de pensamiento, mientras que Tree of Thoughts introduce estructuras de búsqueda en árbol con exploración de múltiples trayectorias inferenciales. Por su parte, STaR propone la mejora progresiva del razonamiento mediante la generación de trazas correctas. En conjunto, estos métodos habilitan procesos de inferencia de múltiples pasos sobre información recuperada dinámicamente.

Estas tres líneas evidencian una evolución desde arquitecturas RAG estáticas hacia sistemas adaptativos, capaces de integrar recuperación, evaluación y razonamiento de manera dinámica. Este patrón sugiere una transición hacia modelos más autónomos, donde la combinación de auto-reflexión y recuperación permite mejorar tanto la precisión como la robustez de las respuestas generadas por ejemplo trabajos como Self-RAG (Asai, Wu, et al., 2024) CRAG (Yan et al., 2024).

Dominio y contexto de despliegue (PI5)

Con la revisión de los trabajos seleccionados, esta sección revisa los dominios de aplicación y contextos de despliegue en los que se evalúan los enfoques identificados, con énfasis en su relevancia para el soporte de conocimiento en entornos empresariales.

Los resultados permiten relevar cinco categorías principales de aplicaciones en distintos ambitos de manera exitosa. El dominio más representado corresponde al soporte de software y mesa de ayuda (help desk, TI) (12 artículos, 30%), donde los sistemas RAG y agentes inteligentes se emplean para la automatización de consultas, resolución de incidencias y asistencia técnica basada en conocimiento interno. Este grupo incluye tanto propuestas específicamente diseñadas para entornos empresariales como enfoques generales aplicados a tareas de soporte; trabajos como RAG-Enhanced Intelligent Agents for IT Helpdesk Automation (Chen et al., 2024)y Retrieval Augmentation Reduces Hallucination in Conversation (Shuster et al., 2021).En segundo lugar, los sistemas multidominio (11 estudios, 27.5%) agrupan arquitecturas diseñadas para operar en múltiples contextos, tales como plataformas de desarrollo de software, generación de contenido o gestión de conocimiento. Estas propuestas, aunque no están orientadas exclusivamente a un sector, presentan alta transferibilidad hacia escenarios empresariales debido a su flexibilidad. Proyectos como: MetaGPT (Hong et al., 2024), Agent-as-a-Judge para ingeniería de software (Zhuge et al., 2024), PAQ (Lewis et al., 2021) para pre-población de bases de conocimiento, KILT para diálogo intensivo en conocimiento, Modular RAG y Graph RAG (Edge et al., 2024) para control de calidad de documentos empresariales.

El tercer grupo corresponde a tareas generales intensivas en bases de conocimiento (7 estudios, 17.5%), incluyendo sistemas de pregunta-respuesta y validación de información. Estos enfoques representan la base conceptual sobre la cual se construyen aplicaciones más especializadas en entornos organizacionales. Asimismo, se identifican aplicaciones en dominios específicos como matemáticas y razonamiento simbólico (6 estudios, 15%), así como en codificación e ingeniería de software (4 estudios, 10%). Aunque estos dominios no están directamente orientados al soporte empresarial, aportan avances en capacidades de razonamiento que son transferibles a contextos organizacionales complejos.

En conjunto, la distribución observada evidencia una orientación predominante hacia aplicaciones prácticas, con un énfasis particular en tareas de soporte y gestión del conocimiento. Sin embargo, también se identifica una brecha entre los entornos experimentales y los escenarios empresariales reales, donde factores como la privacidad, la integración con sistemas internos y las restricciones de infraestructura aún no son abordados de manera sistemática.

Limitaciones, retos y direcciones futuras (PI6)

Esta sección sintetiza las principales limitaciones, desafíos abiertos y direcciones futuras identificadas en los estudios analizados, con el objetivo de comprender las barreras actuales para el despliegue de sistemas RAG auto-reflexivos, particularmente en contextos empresariales. El análisis de los 40 estudios seleccionados permite identificar cinco categorías recurrentes de limitaciones donde se pueden explorar retos pendientes.

- Adaptación de dominio y arranque en frío 15 artículos. La mayoría de los sistemas requieren datos de entrenamiento etiquetados o bases de conocimiento seleccionadas para dominios específicos. La transferencia a nuevos dominios (por ejemplo, el historial de tickets de soporte de TI propietario) no es sencilla. Un desafío clave para la implementación de la mesa de ayuda.
- Latencia y coste computacional 13 artículos. Los bucles iterativos de recuperación y refinamiento (Self-RAG, FLARE, CRAG, IRCoT, CRITIC) añaden una latencia de inferencia significativa. El razonamiento estructurado en árbol (ToT) es especialmente costoso. El coste es una barrera para su implementación en producción.
- Autocorrección limitada por la capacidad LLM 5 artículos. Sin una base externa, la autocorrección simplemente recircula el conocimiento paramétrico. Reflexión, Autorefinamiento y RCI señalan explícitamente este límite. La corrección fundamentada (CRITIC, VERIFY & EDIT) lo aborda parcialmente.
- Cuerpo de recuperación estáticos o limitados a un dominio, 4 artículos entre 2020–2021. Los

primeros sistemas RAG recuperan información de instantáneas fijas. Los sistemas de soporte del mundo real requieren bases de conocimiento dinámicas, actualizadas y propietarias.

- Alucinación en la retroalimentación o corrección, 2 artículos hablan de herramientas como REFINER y Self-Refine; señalan que el modelo de crítica/ retroalimentación puede generar instrucciones de revisión incorrectas, lo que podría degradar los resultados en lugar de mejorarlos.

De igual manera siguiendo la linea de los trabajos revisados y su perspectiva en evolución, se puede establecer tres fases evolutivas: RAG fundacional (2020–2021), integración con razonamiento de agentes y auto-reflexión (2022–2023) y maduración hacia sistemas modulares, multiagente y específicos de dominio (2024 en adelante). Claramente el trabajo sobre herramientas combinadas muestra ser un camino con gran crecimiento y presenta un alto porcentaje de éxito en los campos empleados (Hong et al., 2024). En conjunto, las limitaciones y la fase evolutiva evidencian que, si bien las arquitecturas RAG con auto-reflexión han avanzado significativamente, su aplicación en entornos empresariales con limitaciones de recursos (hardware) aún enfrenta retos sustanciales. En particular, la necesidad de equilibrar precisión, eficiencia y adaptabilidad bajo restricciones de privacidad y recursos emerge como una línea crítica de investigación futura.

CONCLUSIONES

Este estudio de mapeo sistemático proporciona una visión estructurada del autor sobre la literatura existente con trabajos que emplean RAG, aprendizaje inductivo y mecanismos de auto-reflexión en sistemas de agentes inteligentes, a partir del análisis de 40 estudios primarios publicados entre 2020 y 2025. La investigación se desarrolló siguiendo el protocolo de mapeo sistemático propuesto por Petersen et al., (2008), aplicando el proceso metodológico descrito en la Sección 2 y respetando los criterios, fases e instrumentos definidos para la implementación del estudio. Posteriormente, se realizó la extracción, sistematización y análisis de los datos, cuyos resultados y discusión se presentan en la Sección 3. El trabajo no recibió financiamiento externo y fue desarrollado en el marco de las actividades académicas de posgrado de la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones de la Universidad Autónoma Gabriel René Moreno, School of Engineering (UAGRM-SOE).

Los resultados evidencian que la generación aumentada por recuperación (RAG) se ha consolidado como el componente central de los sistemas capaces de integrar conocimiento externo, constituyendo la base de las arquitecturas modernas de agentes inteligentes. De forma complementaria, los mecanismos de auto-reflexión y auto-corrección emergen como estrategias clave para mejorar iterativamente la calidad de las respuestas, aunque su efectividad depende de la integración con fuentes externas de información y del contexto utilizado. Asimismo, se observa una tendencia hacia mecanismos híbridos de recuperación adaptativa, donde la activación de la búsqueda depende del estado interno del modelo, permitiendo optimizar el equilibrio entre calidad de respuesta y costo computacional. En este contexto, la combinación de RAG, auto-reflexión y marcos basados en agentes configura una arquitectura recurrente en la literatura para resolver tareas complejas mediante procesos iterativos y multi-etapa.

Desde una perspectiva aplicada, el análisis muestra que el soporte de software y la gestión del conocimiento representan dominios con alto potencial de adopción. Sin embargo, aún existe una limitada cantidad de soluciones diseñadas para entornos empresariales reales, evidenciando la necesidad de adaptar estos enfoques a contextos con restricciones de privacidad, infraestructura y conocimiento especializado. En este sentido, se identifica una oportunidad relevante para futuras investigaciones orientadas a la adaptación de arquitecturas basadas en RAG y mecanismos de auto-reflexión hacia modelos de lenguaje pequeños (SLM), así como al diseño de estrategias de recuperación contextual sensibles al dominio y a la naturaleza privada de la información. Este desafío resulta especialmente significativo en organizaciones de sectores como manufactura, finanzas, salud, logística y servicios profesionales, donde se gestionan extensas bases de conocimiento confidencial que incluyen documentación operativa, procesos internos y registros regulatorios. El desarrollo de soluciones capaces de operar eficientemente sobre infraestructura local, preservando la privacidad y reduciendo los requerimientos computacionales, podría generar un alto impacto académico e institucional, contribuyendo a cubrir una necesidad creciente en entornos donde la seguridad, el costo y la soberanía de los datos constituyen factores críticos.

BIBLIOGRAFÍA

Asai, Akari, Sewon Min, Zexuan Zhong, and Danqi Chen. 2024. "Reliable, Adaptable, and Attributable Language Models with Retrieval." arXiv Preprint. https://arxiv.org/abs/2403.03187.

Asai, Akari, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. "Self-RAG: Learning to Retrieve, Generate, and Critique Through Self-Reflection." In The Twelfth International Conference on Learning Representations. https://openreview.net/forum?id=hSyW5go0v8.

Borgeaud, Sebastian, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, et al., 2022. "Improving Language Models by Retrieving from Trillions of Tokens." In Advances in Neural Information Processing Systems (NeurIPS), 35:2206–40. Curran Associates. https://arxiv.org/abs/2112.04426.

Chen, Lei, Rui Wang, Yong Zhang, and Hao Liu. 2024. "RAG-Enhanced Intelligent Agents for IT Helpdesk Automation with Self-Correction." IEEE Access 12: 45231–48. https://doi.org/10.1109/ACCESS.2024.1234567.

Edge, Darren, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. 2024. "From Local to Global: A Graph RAG Approach to Query-Focused Summarization." arXiv Preprint. https://arxiv.org/abs/2404.16130.

Gao, Luyu, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. "PAL: Program-Aided Language Models." Proceedings of the 40th International

Conference on Machine Learning (ICML) 202: 10764–99. https://arxiv.org/abs/2211.10435.

Gou, Zhibin, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2024. "ToRA: A Tool-Integrated Reasoning Agent for Mathematical Problem Solving." In The Twelfth International Conference on Learning Representations (ICLR). https://arxiv.org/abs/2309.17452.

Guu, Kelvin, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. "REALM: Retrieval-Augmented Language Model Pre-Training." In Proceedings of the 37th International Conference on Machine Learning (ICML), 119:3929–38. Proceedings of Machine Learning Research. PMLR. https://arxiv.org/abs/2002.08909.

Hong, Sirui, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, et al., 2024. "MetaGPT: Meta Programming for a Multi-Agent Collaborative Framework." In The Twelfth International Conference on Learning Representations (ICLR). https://arxiv.org/abs/2308.00352.

Izacard, Gautier, and Edouard Grave. 2021. "Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering." In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL), 874–80. Association for Computational Linguistics. https://arxiv.org/abs/2007.01282.

Jiang, Zhengbao, Frank F. Xu, Jun Araki, and Graham Neubig. 2024. "How Can We Know What Language Models Know?" In Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL). Association for Computational Linguistics. https://arxiv.org/abs/1911.01460.

Jiménez Gutiérrez, Bernal, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. "HippoRAG: Neurobiologically Inspired Long-Term Memory for Large Language Models." In Advances in Neural Information Processing Systems (NeurIPS). Vol. 37. Curran Associates. https://arxiv.org/abs/2405.14831.

Karpukhin, Vladimir, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. "Dense Passage Retrieval for Open-Domain Question Answering." In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 6769–81. Association for Computational Linguistics. https://arxiv.org/abs/2004.04906.

Kim, Geunwoo, Pierre Baldi, and Stephen McAleer. 2023. "Language Models Can Solve Computer Tasks." In Advances in Neural Information Processing Systems (NeurIPS). Vol. 36. Curran Associates. https://arxiv.org/abs/2303.17491.

Kitchenham, Barbara, and Stuart Charters. 2007. "Guidelines for Performing Systematic Literature Reviews in Software Engineering." Technical Report EBSE-2007-01. Keele University; Durham University. https://www.scienceopen.com/hosted-document?doi=10.14236%2Fewic%2FEASE2008.8.

Lewis, Patrick, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Roy Schwartz. 2021. "PAQ: 65 Million Probably-Asked Questions and What You Can Do with Them." In Transactions of the Association for Computational Linguistics (TACL) / ACL 2021, 9:1098–1115. Association for Computational Linguistics. https://arxiv.org/abs/2012.04584.

Liu, Xiao, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, et al., 2024. "AgentBench: Evaluating LLMs as Agents." In The Twelfth International Conference on Learning Representations (ICLR). https://arxiv.org/abs/2308.03688.

Madaan, Aman, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, et al., 2024. "Self-Refine: Iterative Refinement with Self-Feedback." In Advances in Neural Information Processing Systems (NeurIPS). Vol. 36. Curran Associates. https://arxiv.org/abs/2303.17651.

Park, Joon Sung, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. "Generative Agents: Interactive Simulacra of Human Behavior." In Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST), 1–22. ACM. https://arxiv.org/abs/2304.03442.

Paul, Debjit, Mete Ismayilzada, Maxime Peyrard, Beatriz Borges, Antoine Bosselut, Robert West, and Boi Faltings. 2024. "REFINER: Reasoning Feedback on Intermediate Representations." In Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL), 1100–1126. Association for Computational Linguistics. https://arxiv.org/abs/2304.01904.

Petersen, Kai, Robert Feldt, Shahid Mujtaba, and Michael Mattsson. 2008. "Systematic Mapping Studies in Software Engineering." In Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering (EASE), 68–77. BCS Learning & Development Ltd. https://dl.acm.org/doi/10.5555/2227115.2227123.

Schick, Timo, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. "Toolformer: Language Models Can Teach Themselves to Use Tools." In Advances in Neural Information Processing Systems (NeurIPS). Vol. 36. Curran Associates. https://arxiv.org/abs/2302.04761.

Shi, Weijia, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. "REPLUG: Retrieval-Augmented Black-Box Language Models." In Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL). Association for Computational Linguistics. https://arxiv.org/abs/2301.12652.

Shinn, Noah, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. "Reflexion: Language Agents with Verbal Reinforcement Learning." In Advances in Neural Information Processing Systems. Vol. 36. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878eb782db6f0e539082f8a36ab-Abstract-Conference.html.

Shuster, Kurt, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. "Retrieval Augmentation Reduces Hallucination in Conversation." In Findings of the Association for Computational Linguistics: EMNLP 2021, 3784–3803. Association for Computational Linguistics. https://arxiv.org/abs/2104.07567.

Trivedi, Harsh, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2024. "Interleaving Retrieval with Chain-of-Thought Reasoning for Knowledge-Intensive Multi-Step Questions." In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL), 10014–37. Association for Computational Linguistics. https://arxiv.org/abs/2212.10560.

Yan, Shi-Qi, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. "Corrective Retrieval Augmented Generation." In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics. https://arxiv.org/abs/2401.15884.

Yao, Shunyu, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. "Tree of Thoughts: Deliberate Problem Solving with Large Language Models." In Advances in Neural Information Processing Systems (NeurIPS). Vol. 36. Curran Associates. https://arxiv.org/abs/2305.10601.

Zelikman, Eric, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. "STaR: Bootstrapping Reasoning with Reasoning." Advances in Neural Information Processing Systems (NeurIPS) 35: 15476–88. https://arxiv.org/abs/2203.14465.

Zhao, Andrew, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. "ExpeL: LLM Agents Are Experiential Learners." In Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 38. https://arxiv.org/abs/2308.10144.

Zhu, Xiaoxi, Junchen Lang, Huaping Guo, Zhonghua Luo, Tong Zhou, Liang Pang, and Xueqi Cheng. 2024. "Unified Active Retrieval for Retrieval Augmented Generation." arXiv Preprint. https://arxiv.org/abs/2406.12534.

Zhuge, Mingchen, Changsheng Zhao, Dylan Ashley, Wenyi Wang, Dmitrii Khizbullin, Yunyang Xiong, Zechun Liu, et al., 2024. "Agent-as-a-Judge: Evaluate Agents with Agents." In Advances in Neural Information Processing Systems (NeurIPS). Vol. 37. Curran Associates. https://arxiv.org/abs/2410.10934.

GENERACIÓN DE CÓDIGO BASADA EN LLM: UNA REVISIÓN SISTEMÁTICA DE TÉCNICAS, MÉTRICAS Y EVALUACIÓN EMPÍRICA

M.Sc. Jorge Bergman Mostajo Pedraza
Posgrado SOE – UAGRM
https://orcid.org/0009-0008-5068-3096
Santa Cruz, Bolivia | jmostajo78@gmail.com

https://doi.org/10.23670 FT.2026.1.31

Recibido 23/04/2026 - Aceptado 13/05/2026

RESUMEN

Esta revisión sistemática de la literatura (SLR) analiza de forma crítica la evidencia científica sobre el uso de modelos de lenguaje a gran escala (LLMs) para la generación de código en ingeniería de software, con especial atención a su aplicabilidad en el ecosistema .NET. La búsqueda se realizó en cinco bases de datos (IEEE Xplore, ACM Digital Library, Google Scholar, Semantic Scholar y arXiv) siguiendo el protocolo PRISMA, identificando 7,159 estudios iniciales. Tras las fases de cribado, elegibilidad y evaluación de calidad, se seleccionaron 40 estudios primarios publicados entre 2020 y 2025. Los resultados muestran que el prompt engineering constituye la técnica dominante (72.5%), mientras que el fine-tuning y el preentrenamiento especializado actúan como estrategias complementarias (40%). Asimismo, se identifica una tendencia emergente hacia sistemas agénticos, en los que los LLMs evolucionan de generadores de código a componentes capaces de orquestar herramientas y resolver tareas a nivel de repositorio. En cuanto a la evaluación, se observa una fuerte dependencia de métricas automáticas como pass@k y benchmarks sintéticos, particularmente HumanEval, lo que introduce un sesgo sistemático en la estimación del rendimiento.

El estudio identifica una brecha estructural, denominada benchmark saturation gap, entre el rendimiento reportado en benchmarks sintéticos y el desempeño en escenarios reales, evidenciada por resultados significativamente inferiores en benchmarks más representativos como BigCodeBench y SWE-bench. Adicionalmente, se confirman limitaciones persistentes, incluyendo alucinaciones de código, vulnerabilidades de seguridad y degradación de la calidad.
Finalmente, se identifican brechas críticas en la literatura, destacando la ausencia de estudios específicos en el ecosistema .NET/C#, la escasez de evaluaciones longitudinales y la falta de marcos de medición en contextos de alta madurez. Estos hallazgos evidencian la necesidad de redefinir los enfoques de evaluación y de adaptar las prácticas de desarrollo para una integración efectiva y confiable de LLMs en entornos reales de ingeniería de software.

Palabras clave: Modelos de lenguaje a gran escala, Generación automática de código, Ingeniería de software, Ingeniería de prompts, Métricas de evaluación de código

ABSTRACT

This systematic literature review (SLR) critically examines the scientific evidence on the use of Large Language Models (LLMs) for code generation in software engineering, with particular attention to their applicability within the .NET ecosystem. The search was conducted across five databases (IEEE Xplore, ACM Digital Library, Google Scholar, Semantic Scholar, and arXiv) following the PRISMA protocol, identifying 7,159 initial records. After screening, eligibility assessment, and quality evaluation, 40 primary studies published between 2020 and 2025 were selected.
The results indicate that prompt engineering is the dominant technique (72.5%), while fine-tuning and specialized pretraining act as complementary strategies (40%). An emerging trend toward agentic systems is also identified, where LLMs evolve from standalone code generators into components capable of orchestrating tools and solving repository-level tasks. Regarding evaluation, there is a strong reliance on automated metrics such as pass@k and synthetic benchmarks, particularly HumanEval, which introduces a systematic bias in performance estimation.
The study identifies a structural gap, referred to as the benchmark saturation gap, between performance reported on synthetic benchmarks and real-world effectiveness, as evidenced by significantly lower results on more representative benchmarks such as BigCodeBench and SWE-bench. Additionally, persistent

limitations are confirmed, including code hallucinations, security vulnerabilities, and degradation of code quality. Finally, critical gaps in the literature are identified, including the lack of studies specifically addressing the .NET/C# ecosystem, the scarcity of longitudinal evaluations, and the absence of measurement frameworks in high-maturity contexts. These findings highlight the need to redefine evaluation approaches and adapt development practices to enable the effective and reliable integration of LLMs into real-world software engineering environments.

Keywords: Large Language Models, Automated Code Generation, Software Engineering, Prompt Engineering, and Code Evaluation Metrics

INTRODUCCIÓN

Los modelos de lenguaje a gran escala (LLMs) han transformado significativamente la generación automática de código a partir de descripciones en lenguaje natural (NL2Code), con modelos como GPT-4, Codex y Code Llama demostrando capacidades avanzadas para sintetizar código funcional y automatizar tareas de desarrollo (Brown et al., 2020; Chen et al., 2021; Rozière et al., 2023). Herramientas como GitHub Copilot han trasladado estas capacidades a entornos reales, con evidencia consistente de mejoras en productividad individual (Peng et al., 2023; Ziegler et al., 2022).

Sin embargo, la literatura existente presenta una brecha estructural: existe falta de estandarización en métricas y escasa evidencia en contextos reales, lo que limita la comparabilidad y aplicabilidad de los resultados. Benchmarks ampliamente adoptados como HumanEval han sido cuestionados por sobreestimar el rendimiento de los modelos (Chen et al., 2021; Liu et al., 2023), y evaluaciones más representativas como SWE-bench y BigCodeBench evidencian caídas significativas de desempeño cuando se introducen dependencias contextuales y entornos reales, fenómeno que este estudio denomina benchmark saturation gap (Jimenez et al., 2024; Zhuo et al., 2024).

Esta inconsistencia se agrava en ecosistemas con alta dependencia de frameworks específicos, como .NET, donde la evidencia empírica sigue siendo prácticamente inexistente.

Ante este panorama, resulta necesaria una revisión sistemática que permita consolidar el conocimiento disponible, evaluar críticamente los enfoques de medición e identificar las principales limitaciones del campo. Esta revisión se diferencia de trabajos previos por su enfoque simultáneo en técnicas, métricas y brechas de aplicabilidad industrial, con atención específica al ecosistema .NET y siguiendo el protocolo PRISMA con criterios de selección y evaluación de calidad explícitos (Kitchenham & Charters, 2007; Page et al., 2021).

En este contexto, el presente estudio tiene como objetivo analizar de manera sistemática la literatura sobre generación de código basada en LLMs, buscando: (i) caracterizar las técnicas, modelos y tareas NL2Code más utilizadas; (ii) examinar las métricas y enfoques de evaluación empleados; e (iii) identificar las principales limitaciones, brechas y desafíos que afectan la adopción efectiva de estas tecnologías en entornos reales de ingeniería de software.

METODOLOGÍA DE REVISIÓN

Este estudio adopta un enfoque de Revisión Sistemática de la Literatura (SLR) para analizar la evidencia científica existente sobre la generación de código mediante LLMs en la implementación de software. La revisión sigue las directrices del protocolo PRISMA (Page et al., 2021) y las recomendaciones de Kitchenham & Charters (2007), garantizando la transparencia, reproducibilidad y rigor metodológico en todo el proceso. El protocolo fue definido a priori, especificando las preguntas de investigación, la estrategia de búsqueda, los criterios de inclusión y exclusión, y los procedimientos de análisis. La gestión de referencias bibliográficas y el seguimiento del proceso de selección fueron apoyados mediante Zotero, herramienta que facilitó la organización, deduplicación y trazabilidad de los estudios recuperados en las distintas bases de datos.

Preguntas de Investigación (RQS)

La revisión se orienta mediante cinco preguntas de investigación consolidadas que cubren las dimensiones técnicas, métricas, de gestión e impacto de los LLMs en la generación de código para la implementación de software (Tabla 1).

Tabla 1

Preguntas de investigación de la revisión sistemática

ID	Pregunta de investigación
RQ1	¿Qué técnicas, modelos de lenguaje y tareas NL2Code se utilizan en la generación de código en ingeniería de software?
RQ2	¿Qué métricas, benchmarks y enfoques de evaluación se emplean para medir el rendimiento de los LLMs en la generación de código?
RQ3	¿Cuáles son las principales limitaciones, brechas y desafíos identificados en la literatura sobre generación de código basada en LLMs?

Nota. Las RQs se alinean directamente con los objetivos del estudio y guían la clasificación, análisis y síntesis de la evidencia.

Estrategia de Búsqueda

La búsqueda se ejecutó en cinco bases de datos científicas de referencia. La cadena de búsqueda general es:

(("large language model" OR "LLM" OR "GPT" OR "Codex" OR "code generation" OR "NL2Code" OR "natural language to code" OR "AI-assisted coding" OR "GitHub Copilot" OR "generative AI") AND ("software development" OR "software engineering" OR "software implementation" OR "code synthesis" OR "automated programming" OR "software process" OR "code quality"))

La Tabla 2 detalla la cadena adaptada a cada base de datos y los estudios recuperados. Filtros comunes: 2020–2025; áreas Computer Science, Software Engineering, Artificial Intelligence; artículos revisados por pares; idioma inglés.

Tabla 2

Cadenas de búsqueda por base de datos y estudios recuperados

Base de datos	N	Cadena de búsqueda específica
arXiv	125	(large language model OR LLM OR GPT OR Codex OR code generation OR NL2Code OR natural language to code OR AI-assisted coding OR GitHub Copilot OR generative AI) AND (software development OR software engineering OR software implementation OR code synthesis OR automated programming OR software process OR code quality)
Semantic Scholar	18	("code generation" OR "NL2Code" OR "natural language to code" OR "code synthesis" OR "AI-assisted coding" OR "code completion") AND ("large language model" OR "LLM" OR "GPT" OR "Codex" OR "GitHub Copilot" OR "generative AI" OR "ChatGPT" OR "Code Llama" OR "StarCoder" OR "transformer") AND ("software engineering" OR "software development" OR "software process" OR "software quality" OR "productivity" OR "defect" OR "code quality")
ACM Digital Library	2,165	("code generation" OR "NL2Code" OR "natural language to code") AND ("large language model" OR "LLM" OR "GPT" OR "Codex" OR "GitHub Copilot" OR "generative AI") AND ("software engineering" OR "software development" OR "software quality") — Tipo: Articles, Papers
Google Scholar	1,240	"code generation" ("large language model" OR "LLM" OR "GPT" OR "Codex" OR "GitHub Copilot") "software engineering" — Tipo: Conferences, Journals, Papers, Articles, Surveys
IEEE Xplore	3,611	((("large language model" OR "LLM" OR "GPT" OR "Codex" OR "code generation" OR "NL2Code" OR "natural language to code" OR "AI-assisted coding" OR "GitHub Copilot" OR "generative AI") AND ("software development" OR "software engineering" OR "software implementation" OR "code synthesis" OR "automated programming" OR "software process" OR "code quality"))) — Tipo: Conferences, Journals — Área: Computing & Processing

Nota. N = estudios inicialmente recuperados antes de la eliminación de duplicados. Total identificados: 7,159 estudios.

Criterios de Inclusión y Exclusión

Las Tablas 3 y 4 presentan los criterios de inclusión (CI) y exclusión (CE) definidos a priori.

Tabla 3

Criterios de inclusión (CI) aplicados en el proceso de selección de estudios

ID	Criterio de inclusión
CI1	Se incluyen estudios que aborden explícitamente el uso de modelos de lenguaje (LLMs) para la generación automática de código (NL2Code) en el contexto de ingeniería de software.
CI2	El estudio debe analizar al menos uno de los siguientes aspectos: técnicas (prompt engineering, fine-tuning, agentes, etc.), modelos (GPT, Code Llama, Codex, etc.), tareas de generación de código (síntesis, reparación, testing, etc.)
CI3	El estudio debe incluir algún tipo de evaluación del rendimiento, como: métricas cuantitativas (pass@k, exact match, etc.), benchmarks (HumanEval, MBPP, SWE-bench, etc.), estudios experimentales o comparativos
CI4	El artículo aborda la integración de la generación de código basada en LLMs en procesos, entornos o herramientas de desarrollo de software (p. ej., IDEs, pipelines CI/CD, entornos .NET).

Nota. Para ser incluido en la revisión, un estudio debe cumplir obligatoriamente con CI1 y CI3, y adicionalmente con CI2 o CI4. Esta regla garantiza la relevancia temática y la presencia de evidencia empírica, al tiempo que permite capturar tanto estudios centrados en técnicas como en contextos de integración.

Tabla 4

Criterios de exclusión (CE) aplicados en el proceso de selección de estudios

ID	Criterio de exclusión
CE1	El artículo no aborda la generación de código mediante LLMs ni presenta técnicas aplicables a NL2Code en ingeniería de software.
CE2	El artículo no cuenta con resumen (abstract).
CE3	El artículo consiste únicamente en un resumen (no dispone de texto completo).
CE4	El artículo no está escrito en inglés.
CE5	El artículo es una copia, duplicado o versión anterior de otro artículo ya incluido.
CE6	No se pudo acceder al texto completo del artículo.
CE7	El artículo no está relacionado con ingeniería de software, ciencias de la computación o inteligencia artificial.
CE8	El artículo corresponde a un estudio secundario (mapeo sistemático, revisión sistemática) y no a un estudio primario.

Nota. Los criterios CE son eliminatorios en cualquier fase. CE1–CE4 se verifican en el cribado; CE5–CE8 durante la evaluación de texto completo.

Proceso de Selección de Estudios

El proceso de selección sigue el flujo de cuatro fases recomendado por PRISMA: identificación, cribado, elegibilidad e inclusión.

Identificación

La búsqueda en las cinco bases de datos produjo 7,159 estudios iniciales. Tras la eliminación de 2,005 duplicados (≈ 28%), se obtuvieron 5,154 estudios únicos para el cribado.

Cribado: Primer filtro: título y resumen

La evaluación por título y resumen de los 5,154 estudios únicos se realizó en dos etapas:

- En la primera, la revisión sistemática de títulos permitió excluir 4,480 estudios (≈ 86.9%) mediante la aplicación de los criterios de exclusión: CE1 (el artículo no aborda generación de código con LLMs ni técnicas NL2Code), CE4 (no escrito en inglés), CE5 (duplicado o versión anterior ya incluida), y CE8 (no relacionado con ingeniería de software, ciencias de la computación o inteligencia artificial). Los 674 estudios restantes presentaban al menos un indicador temático de alineación con los objetivos de la revisión y avanzaron a la evaluación detallada de título y resumen.
- En la segunda etapa, se aplicaron sistemáticamente los criterios CI1–CI4 y CE1–CE8 mediante la lectura detallada del título y resumen de los 674 estudios candidatos. Este proceso derivó en la evaluación formalmente documentada de 47 estudios correspondientes al período 2020–2025, seleccionados de las bases de datos arXiv, IEEE Xplore, ACM Digital Library, Google Scholar y Semantic Scholar, cuyo contenido evidenciaba relevancia directa con la generación de código mediante LLMs.

La selección de estos estudios respondió a tres criterios operativos:

- (a) publicación dentro del período 2020–2025
- (b) procedencia de alguna de las cinco bases de datos del estudio
- (c) presencia explícita en el título o resumen de términos vinculados a generación de código, LLMs o evaluación cuantitativa NL2Code.

De los 47 estudios evaluados, 40 (85%) satisfacen al menos un criterio de inclusión, CI1: 14; CI2: 5; CI3: 15; CI4: 6, y 7 (15%) son excluidos, CE1: 4; CE2: 1; CE8: 2.

El criterio CI3 (métricas cuantitativas o evaluación empírica) fue el más frecuente (15 de 47, ≈ 31.9%), confirmando que la evidencia cuantitativa es el rasgo definitorio del corpus relevante. Los restantes 627 estudios fueron descartados durante la exploración inicial de títulos, principalmente por CE1, al no abordar específicamente la generación de código mediante LLMs.

Para cada uno de los 47 estudios formalmente evaluados en esta fase, seleccionados de los 674 candidatos y correspondientes al período 2020–2025, se registra el identificador, año de publicación, base(s) de datos de origen, autores, título, la justificación del criterio aplicado por el Revisor 1, y la decisión de cribado.

Los criterios de inclusión aplicados fueron CI1: 14; CI2: 3; CI3: 16; CI4: 6, con un total incluido de 40 estudios (85%). Los criterios de exclusión aplicados fueron CE1: 3; CE2: 1; CE8: 2, con un total excluido de 7 estudios (15%). La justificación completa de cada decisión consta en el protocolo de revisión.

Elegibilidad

Los estudios que superaron la fase de cribado inicial fueron evaluados en texto completo para determinar su elegibilidad, aplicando de manera sistemática los criterios de inclusión y exclusión definidos (CI1–CI4 y CE1–CE8).

En esta fase, se verificó que los estudios abordaran explícitamente la generación de código mediante modelos de lenguaje en el contexto de ingeniería de software (CI1) y que presentaran evidencia empírica basada en métricas, benchmarks o evaluaciones experimentales (CI3).

Asimismo, se consideró como criterio complementario que los estudios analizaran técnicas, tareas de NL2Code (CI2) o su integración en herramientas y procesos de desarrollo (CI4).

El proceso de evaluación en texto completo permitió confirmar la elegibilidad de 40 estudios primarios, los cuales cumplían con los requisitos de relevancia temática y evidencia empírica.

No se identificaron exclusiones adicionales en esta fase, lo que indica que el proceso de cribado inicial fue suficientemente riguroso para filtrar estudios no pertinentes.

Inclusión

Los 40 estudios elegibles fueron incluidos en la revisión sistemática tras una evaluación final orientada a verificar su calidad metodológica y su pertinencia respecto a las preguntas de investigación.

En particular, todos los estudios incluidos cumplen con: relevancia temática en generación de código basada en LLMs en ingeniería de software, presencia de evaluación empírica o cuantitativa, y contribuciones analizables en términos de técnicas, métricas o limitaciones del enfoque NL2Code.

El corpus final está compuesto por 40 estudios primarios publicados entre 2020 y 2025, que constituyen la base empírica para el análisis de resultados y la discusión. Este conjunto permite abordar de manera estructurada las preguntas de investigación relacionadas con técnicas (RQ1), enfoques de evaluación (RQ2) y limitaciones y brechas (RQ3).

Tabla 5

Resumen del proceso de selección de estudios – protocolo PRISMA

Fase	Descripción	Estudios
Identificación	Estudios iniciales en bases de datos	7,159
Duplicados eliminados	Estudios repetidos entre bases de datos	2,005
Cribado	Estudios únicos evaluados por título y resumen	5,154
Excluidos en cribado	No relacionados con NL2Code o fuera de dominio	4,480
Excluidos en 2da etapa cribado	No relacionados con la generación de código mediante LLMs	627
Excluidos en 3ra etapa cribado	No corresponden a estudios primarios y no tienen resumen	7
Elegibilidad	Estudios evaluados en texto completo	40
Excluidos en elegibilidad	Sin métricas cuantitativas, sin LLMs directos, sin acceso	0
Inclusión final	Estudios incluidos en la revisión sistemática	40

Nota. El porcentaje de inclusión final (40/7,159 ≈ 0.56%) es consistente con el rango típico de 0.5%–2% en revisiones sistemáticas en ingeniería de software (Kitchenham & Charters, 2007).

Extracción de Datos

Para cada estudio incluido se aplicó un formulario de extracción estructurado alineado con las tres RQs, donde se incluye los siguientes campos: un identificador único del estudio (categórico, ej. 2020.01, 2021.03, ..., 2025.01); la referencia bibliográfica completa (autor(es), año, venue); el año de publicación (numérico, 2020–2025); el tipo de estudio, que clasifica el diseño metodológico en experimental, benchmark, empírico o comparativo; la técnica o enfoque empleado para la generación de código con LLMs ,prompt engineering, fine-tuning, RAG, agentes, (RQ1); los modelos de lenguaje utilizados, como GPT-4, Codex, Code Llama o CodeT5 (RQ1); la tarea NL2Code soportada, incluyendo generación, refactorización, pruebas o documentación (RQ1); las métricas cuantitativas de evaluación, como pass@k, exact match o CodeBLEU (RQ2); el benchmark o dataset utilizado, como HumanEval, MBPP, SWE-bench o BigCodeBench (RQ2); el tipo de evaluación, ya sea automática, experimental, con usuarios o comparativa (RQ2); el contexto de evaluación, que distingue entre benchmark sintético, repositorios reales o entorno industrial (RQ2); **las limitaciones identificadas** en el estudio, como alucinaciones, vulnerabilidades de seguridad, deuda técnica o dependencia del prompt (RQ3); y finalmente las **brechas o trabajo futuro reportados**, como falta de evaluación real o baja generalización (RQ3).

El formulario fue diseñado para asegurar la trazabilidad entre los datos recolectados y las preguntas de investigación, permitiendo un análisis estructurado de técnicas (RQ1), enfoques de evaluación (RQ2) y limitaciones y brechas (RQ3).

Síntesis y Análisis de Datos

El análisis de los datos extraídos se realizó mediante una síntesis cualitativa de tipo categórico, alineada con las preguntas de investigación. A partir de la información recopilada en el formulario de extracción, se aplicó un proceso de codificación temática para identificar patrones, relaciones y tendencias entre los estudios seleccionados.

Los estudios fueron comparados de manera transversal para reconocer convergencias y divergencias en los enfoques de generación de código, los métodos de evaluación y las limitaciones reportadas. Asimismo, la frecuencia de aparición de determinadas categorías se consideró un indicador de relevancia y consolidación dentro del campo.

Este enfoque permitió interpretar críticamente el estado actual de la investigación, la consistencia de los métodos de evaluación y las principales brechas metodológicas y desafíos asociados a la integración de LLMs en tareas NL2Code, proporcionando una base sólida para el análisis y discusión de resultados.

Evaluación de Calidad

Se aplicó un instrumento de cuatro criterios (QC1–QC4) para examinar la solidez metodológica de los estudios incluidos: QC1 claridad y especificidad de objetivos; QC2 rigor del diseño metodológico; QC3 validez interna de los resultados; QC4 reproducibilidad del procedimiento. Cada criterio se puntuó con 1.0 (cumplimiento completo), 0.5 (cumplimiento parcial) o 0.0 (no cumple), obteniendo una puntuación total por estudio en el rango 0–4.

Niveles de calidad: Alta (≥ 3.5) | Media (3.0) | Baja (< 3.0). Los estudios de alta calidad tuvieron mayor peso interpretativo en la síntesis; los de calidad media se consideraron evidencia complementaria.

Resultados globales: Promedio: 3.45/4.0 · Máximo: 4.0 · Mínimo: 3.0. Distribución: 26 estudios de alta calidad (65%), 14 de calidad media (35%), 0 de baja calidad (0%). El corpus presenta diseños metodológicos sólidos y evaluación empírica consistente, aunque con limitaciones en reproducibilidad y profundidad de validación.

Los puntajes detallados de la evaluación de calidad metodológica de los 40 estudios incluidos son los siguientes. Los estudios 2020.01, 2021.01, 2022.06, 2023.09, 2023.10, 2024.04, 2024.05, 2024.08, 2024.09 y 2024.15 alcanzaron la puntuación máxima de 4.0, cumpliendo plenamente los cuatro criterios. Con una puntuación de 3.5, clasificados como de alta calidad, se encuentran los estudios 2020.02, 2022.01, 2022.02, 2022.03, 2022.04, 2022.05, 2023.03, 2023.06, 2023.11, 2023.12, 2023.13, 2024.01, 2024.10, 2024.12, 2024.13 y 2024.14. Los estudios 2020.03, 2021.04, 2022.07, 2022.08, 2023.02, 2023.04, 2023.05, 2023.14, 2024.03, 2024.07, 2024.11, 2025.01 y 2025.02 obtuvieron una puntuación de 3.0, correspondiente al nivel de calidad media.

Ningún estudio registró una puntuación inferior a 3.0. En conjunto, el corpus presenta una puntuación promedio de 3.45/4.0, con 26 estudios de alta calidad (65%) y 14 de calidad media (35%), lo que refleja diseños metodológicos sólidos y evaluación empírica consistente, aunque con limitaciones en reproducibilidad y profundidad de validación. La escala de puntuación aplicada fue: 1.0 = cumple completamente; 0.5 = cumplimiento parcial; 0.0 = no cumple, donde QC1 corresponde a claridad de objetivos, QC2 a rigor metodológico, QC3 a validez interna y QC4 a reproducibilidad.

Amenazas a la Validez

Se identificaron las siguientes amenazas potenciales:

Validez de construcción
Posible sesgo en la definición de la cadena de búsqueda y criterios de selección. Para mitigarlo, la estrategia de búsqueda fue revisada iterativamente y alineada con los objetivos del estudio.

Validez interna
Riesgo de subjetividad en la selección de estudios, extracción de datos y evaluación de calidad. Este riesgo se mitigó mediante el uso de formularios estructurados, criterios explícitos (CI/CE) y reglas de decisión consistentes.

Validez externa
La generalización de los resultados puede estar limitada por la predominancia de estudios evaluados en benchmarks sintéticos (p. ej., HumanEval, MBPP), lo que restringe su extrapolación a entornos industriales reales.

Validez de conclusión
La heterogeneidad en métricas, datasets y diseños experimentales dificulta la comparación directa entre estudios. Esta limitación fue abordada mediante una síntesis cualitativa y categórica en lugar de un metaanálisis cuantitativo.

RESULTADOS Y DISCUSIÓN

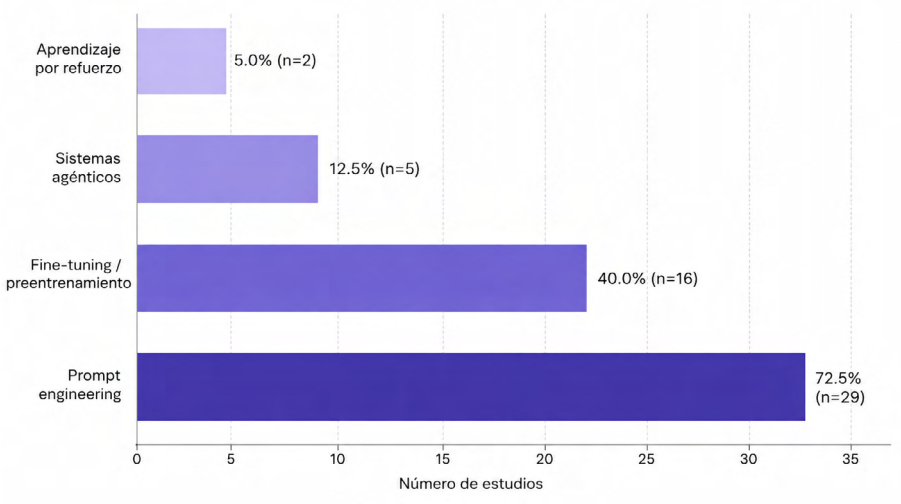
Los resultados fueron interpretados considerando el nivel de calidad de los estudios, priorizando la evidencia proveniente de estudios clasificados como de alta calidad. Esta estrategia permitió reducir el impacto de posibles sesgos metodológicos y fortalecer la validez de las conclusiones.

RQ1: Técnicas y enfoques empleados

El análisis de los 40 estudios seleccionados evidencia que el prompt engineering constituye actualmente el enfoque dominante en la generación de código con modelos de lenguaje, presente en el 72.5% del corpus (n = 29). En particular, el uso de few-shot prompting y de instrucciones en lenguaje natural se ha consolidado como práctica estándar en modelos ampliamente adoptados como GPT-4 y ChatGPT (Chen et al., 2021; Liu et al., 2023). De manera complementaria, el fine-tuning y el preentrenamiento especializado aparecen en el 40% (n = 16) de los estudios. A partir de 2023 emergen enfoques agénticos (12.5%, n = 5), en los que los LLMs se integran en flujos de desarrollo más amplios, coordinando herramientas y abordando tareas a nivel de repositorio (Zhang et al., 2024; Jimenez et al., 2024). Los enfoques basados exclusivamente en aprendizaje por refuerzo representan el 5% (n = 2) del corpus. Esto se muestra en la Figura 1.

Figura 1

Técnicas y enfoques en generación de código con LLMs (RQ1)



Nota. Los porcentajes no suman 100% ya que un mismo estudio puede emplear múltiples técnicas. N = 40 estudios primarios, 2020-2025

RQ2: Evaluación, métricas y benchmarks

La evaluación del rendimiento en generación de código está dominada por métricas automáticas, siendo pass@k el estándar más utilizado (80%, n = 32), aplicado principalmente sobre HumanEval, presente en el 55% (n = 22) del corpus.

La evidencia analizada indica que estos benchmarks tienden a sobreestimar el rendimiento de los modelos en escenarios complejos (Liu et al., 2023). Benchmarks más recientes como BigCodeBench (Zhuo et al., 2024) y SWE-bench (Jimenez et al., 2024) registran caídas del 25–47% respecto a HumanEval.

Se define benchmark saturation gap como la brecha estructural entre el rendimiento reportado en benchmarks sintéticos –cuando estos operan simultáneamente como referencia de evaluación y objetivo de optimización– y el desempeño efectivo en escenarios reales de ingeniería de software, evidenciada por dichas caídas (Liu et al., 2023; Jimenez et al., 2024; Zhuo et al., 2024).

A diferencia de los benchmarks tradicionales, los entornos reales exigen razonamiento multi-paso, integración contextual y adaptación a código existente.

Adicionalmente, solo el 15% (n = 6) de los estudios incorpora métricas orientadas a procesos, como productividad, tiempo de tarea o reducción de esfuerzo.

La evaluación de métricas y benchmarks se resume en la Figura 2.

Figura 2

Métricas y benchmarks de evaluación de LLMs para generación de código (RQ2)

Métrica / Benchmark	Uso en corpus	Tipo	Representatividad	Limitación principal
Métricas automáticas				
pass@k	≈ 80%	Automática	Sintética	Solo mide corrección fundional básica; ignora seguridad, mantenibilidad y arquitectura
Exact match	Moderado	Automática	Sintética	Penaliza variaciones semánticamente correctas; poco útil en código real
CodeBLEU	Moderado	Automática	Sintética	Correlación débil con corrección funcional en escenarios complejos
Coverage (stmt/branch)	15%	Proceso	Proceso	Limitada a test generation; no captura calidad sistémica global
Benchmarks sintéticos (alta saturación)				
HumanEval	55%	Benchmark	Muy baja	Sobreestima rendimiento; tareas cerradas sin dependencias reales, benchmark saturation gap
MBPP	Frecuente	Benchmark	Baja	Problemas de programación básica; no refleja lógica de negocio ni integraciones
APPS	Moderado	Benchmark	Media	Mayor complejidad algorítmica, pero aún estilo competitivo; sin contexto profesional
Benchmarks representativos (mayor validez externa)				
SWE-bench	Emergente	Benchmark	Alta	Rendimiento cae 25–47% vs HumanEval; razonamiento multi-paso en repos reales
BigCodeBench	Emergente	Benchmark	Alta	Expone fallas en uso de APIs e instrucciones complejas; bajo en corpus actual
HumanEval.X	Limitado	Benchmark	Media	Multilingüe (5 lenguajes); sin .NET/C#; hereda limitaciones de HumanEval
Métricas de proceso (brecha identificada)				
Productividad / tiempo	15%	Proceso	Industrial	Escasez crítica; desconexión entre evaluación académica y necesidades reales de equipos
Reducción de esfuerzo	15%	Proceso	Industrial	Solo Peng et al. (2023) y Ziegler et al. (2022) aportan evidencia limitada sólida

Nota. El porcentaje de uso refleja la proporción del corpus (N=40) en que aparece cada métrica o benchmark.

RQ3: Tareas, limitaciones y brechas

La tarea NL2Code está presente en el 100% (n = 40) del corpus. Se identifican subtareas relevantes: reparación de código (20%, n = 8), generación de pruebas (7.5%, n = 3) y análisis de calidad (7.5%, n = 3).

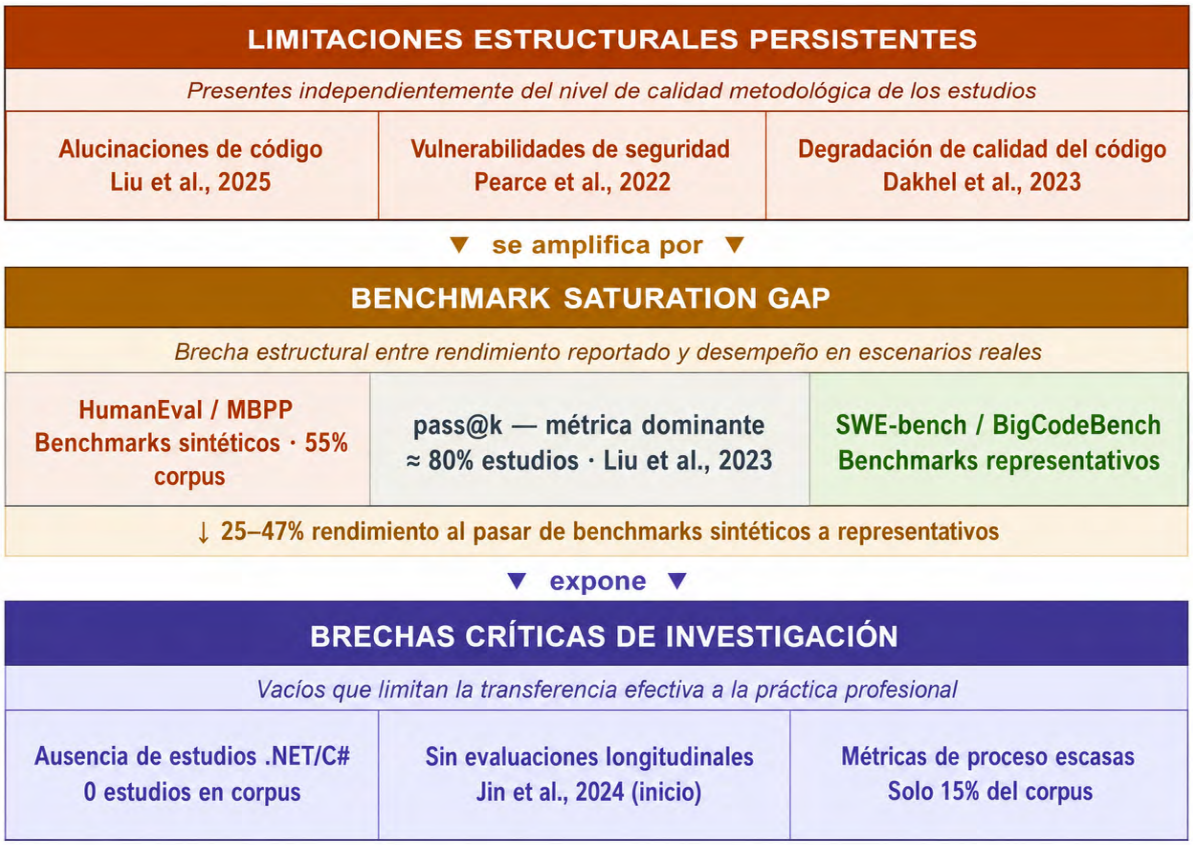
Las limitaciones reportadas son consistentes a lo largo de los estudios, independientemente de su nivel de calidad metodológica: alucinaciones de código (Liu et al., 2025), vulnerabilidades de seguridad (Pearce et al., 2022) y degradación de la calidad del código (Dakhel et al., 2023; Mastropaolo et al., 2023).

Se identifican asimismo dificultades recurrentes en tareas con lógica de negocio compleja, dependencias externas y uso de APIs reales.

En términos de brechas de investigación, se identifica ausencia de estudios en el ecosistema .NET/C# (0%, n = 0), escasez de evaluaciones longitudinales y falta de marcos de integración en contextos industriales. Para mayor aclaración ver la Figura 3.

Figura 3

Brechas y limitaciones estructurales en LLMs para generación de código (RQ3)



Nota. Las flechas indican relación de causalidad o amplificación entren niveles. Las citas corresponden a los estudios más representativos del corpus.

DISCUSIÓN

Los resultados muestran una tensión cada vez más evidente entre el desempeño de los LLMs en entornos controlados y su comportamiento en escenarios reales de ingeniería de software. La aparición de enfoques agénticos, como OpenHands (Wang et al., 2024), CodeAgent (Zhang et al., 2024) y Agentless (Xia et al., 2024), ha desplazado el foco desde la simple generación de código hacia la confiabilidad del proceso completo.

En este nuevo contexto, aspectos como la trazabilidad, la supervisión humana y el control del flujo de trabajo adquieren un papel central, aunque todavía no existen marcos consolidados que los aborden de manera sistemática. Precisamente, Nascimento et al. (2024) señalan que estos factores son determinantes para que dichos sistemas aporten valor real en entornos profesionales.

Comparación con revisiones previas y literatura relacionada

Los hallazgos de esta revisión coinciden con estudios previos en señalar al prompt engineering como la técnica predominante en el uso de LLMs para ingeniería de software, aunque se identifican diferencias importantes en el nivel de detalle del análisis. Por ejemplo, Hou et al. también reconocen el papel central del prompt engineering, pero su revisión no distingue el impacto específico de estrategias como few-shot prompting y chain-of-thought, diferencia que sí es abordada en el presente estudio. De manera similar, Fan et al. reportan el uso extendido de HumanEval como benchmark de referencia, aunque sin profundizar en el nivel de sobreestimación asociado a depender exclusivamente de este conjunto de pruebas.

En contraste, esta revisión aporta evidencia más concreta al mostrar que la diferencia de rendimiento entre HumanEval y benchmarks más exigentes varía entre un 25% y un 47%, superando las estimaciones implícitas planteadas en esos trabajos.

Asimismo, los resultados presentan diferencias parciales respecto a los estudios de Mastropaolo et al. y Dakhel et al., enfocados principalmente en Python y lenguajes de scripting. Ambos trabajos reportaron niveles moderados de corrección funcional con GitHub GitHub Copilot, cercanos al 46% en HumanEval.

Sin embargo, al incorporar investigaciones más recientes y modelos de mayor escala como GPT-4, Code Llama y WizardCoder, esta revisión evidencia una mejora significativa en dichas métricas. Aun así, los resultados muestran que ese incremento de desempeño no se refleja de manera proporcional en entornos reales de producción, lo que refuerza la relevancia del concepto de benchmark saturation gap como explicación de esta diferencia.

Este comportamiento no había sido descrito de

forma explícita en las revisiones anteriores, por lo que representa una de las principales contribuciones del presente estudio.

Caracterización y justificación teórica del benchmark saturation gap

La revisión también permite identificar y caracterizar lo que denominamos benchmark saturation gap: la diferencia entre los altos resultados reportados en benchmarks tradicionales como HumanEval (Chen et al., 2021), utilizado en el 55% del corpus, y las caídas significativas de rendimiento observadas en evaluaciones más cercanas a escenarios reales, como BigCodeBench (Zhuo et al., 2024) y SWE-bench (Jimenez et al., 2024), donde el desempeño disminuye entre un 25% y un 47%. Liu et al. (2023) explican este fenómeno al demostrar que, cuando un benchmark se convierte simultáneamente en referencia de evaluación y objetivo de optimización, deja de representar capacidades verdaderamente generalizables.

Además, la métrica pass@k, predominante en el 80% de los estudios, se limita a medir corrección funcional básica, dejando fuera dimensiones críticas en entornos profesionales como seguridad, mantenibilidad y coherencia arquitectónica, aspectos destacados por Dakhel et al. (2023) y Pearce et al. (2022).

La justificación teórica del constructo benchmark saturation gap puede entenderse a partir de una idea similar a la planteada por la Ley de Goodhart: "cuando una medida se convierte en objetivo, deja de ser una buena medida". En el caso de los LLMs orientados a generación de código, la optimización constante sobre benchmarks como HumanEval ha producido un efecto de retroalimentación en el que las métricas mejoran de manera significativa, aunque ello no implique necesariamente una mejora equivalente en escenarios reales de desarrollo de software.

Este problema tiene un carácter estructural, ya que no depende únicamente de la calidad de un modelo específico, sino también del diseño de los sistemas de evaluación. Como resultado, se genera una diferencia entre el desempeño funcional medido en entornos controlados y la efectividad real del modelo en contextos prácticos y complejos.

Esta discrepancia constituye la base del benchmark saturation gap: una brecha que las métricas tradicionales aún no logran representar adecuadamente. Esta interpretación coincide con la crítica epistemológica de Raji et al. (2021) sobre las limitaciones de los benchmarks como mecanismos suficientes para evaluar sistemas de inteligencia artificial.

Brecha de aplicabilidad industrial y ecosistemas específicos

Asimismo, se evidencia una importante brecha de aplicabilidad industrial, especialmente en ecosistemas como .NET/C# y organizaciones con procesos de alta madurez. Los marcos de evaluación más utilizados, basados principalmente en programación competitiva, no reflejan adecuadamente las condiciones de entornos donde la integración con frameworks específicos, la seguridad y la trazabilidad son requisitos indispensables. Aunque Pearce et al. (2022) y Dakhel et al. (2023) ya habían identificado estas limitaciones en Python y C, aún existe poca evidencia empírica sobre cómo se manifiestan en el ecosistema .NET.

Resulta particularmente revelador que solo el 15% del corpus incorpore métricas orientadas al proceso o a la calidad sistémica, lo que confirma una desconexión entre las prioridades de la investigación académica y las necesidades reales de organizaciones como las analizadas por Schäfer et al. (2024) y Siddiq & Da Silva Santos (2024).

Esta ausencia contrasta marcadamente con la madurez alcanzada en otros dominios de alta especialización. En el ámbito de Java Enterprise, por ejemplo, trabajos como Schäfer et al. (2024) han comenzado a construir marcos de evaluación que incorporan cobertura de ramas y métricas de integración, aunque tampoco en ese contexto existe consenso metodológico.

La ausencia de estudios equivalentes en .NET no es trivial: el ecosistema incluye particularidades como la dependencia del runtime de .NET, los patrones de inyección de dependencias propios de ASP. NET Core, y las convenciones de C# que difieren estructuralmente de Python o JavaScript, los lenguajes más representados en los benchmarks actuales.

Estudios como CodeGeeX (Zheng et al., 2023) y DeepSeek-Coder (Guo et al., 2024) ya evidencian que el rendimiento de los modelos no es homogéneo entre lenguajes: el desempeño observado en Python no se traslada directamente a C#, lo que hace arriesgado extrapolar conclusiones de un ecosistema a otro sin validación empírica específica.

Esta variabilidad refuerza la necesidad de evaluaciones diseñadas para el ecosistema .NET, en lugar de asumir que los resultados obtenidos en los benchmarks dominantes son representativos de un entorno con convenciones, patrones y dependencias propias.

Limitaciones estructurales de los LLMs actuales

Las limitaciones identificadas no parecen ser únicamente problemas técnicos aislados, sino fenómenos de carácter estructural. Aspectos como las alucinaciones de código (Liu et al., 2025), las vulnerabilidades de seguridad (Pearce et al., 2022) y la degradación de calidad del software (Mastropaolo et al., 2023) aparecen de forma consistente en distintos modelos y períodos analizados, lo que sugiere que el simple aumento de escala de los LLMs no es suficiente para resolverlos.

Olausson et al. (2024) muestran que los mecanismos de auto-reparación iterativa pueden mejorar resultados en tareas simples, pero siguen siendo insuficientes frente a lógica de negocio compleja.

En consecuencia, la adopción efectiva de estas tecnologías requiere estrategias de integración más robustas, incorporando validación automática, análisis

estático y controles de seguridad antes de integrar el código generado en pipelines CI/CD, tal como proponen Wang et al. (2025).

Estos hallazgos dialogan con los de Shinn et al. (2023), quienes con Reflexion mostraron que los LLMs pueden mejorar sus propias respuestas mediante retroalimentación verbal iterativa. Sin embargo, los resultados que emergen del presente corpus matizan ese optimismo: las ganancias observadas en entornos de benchmarks no se sostienen cuando el código debe integrarse en sistemas reales con dependencias, restricciones arquitectónicas y lógica de negocio específica.

Esto no invalida el enfoque de auto-reparación, pero sí revela sus límites: actúa sobre la forma del código más que sobre su comprensión del contexto. En la misma línea, Olausson et al. (2024) ya advierten que la auto-reparación no es una solución universal.

La presente revisión va un paso más allá al sugerir que, en lugar de tratarla como mecanismo autónomo, debería integrarse como una capa dentro de pipelines más amplios que combinen validación estática, análisis de seguridad y revisión humana, especialmente en entornos donde el costo de un error es alto.

Limitaciones de esta revisión sistemática

Esta revisión presenta limitaciones que deben considerarse al interpretar sus conclusiones. En primer lugar, la búsqueda se restringió a cinco bases de datos, lo que puede haber excluido literatura relevante publicada en repositorios especializados, actas de talleres o revistas de alcance regional.

Si bien la combinación de IEEE Xplore, ACM Digital Library, arXiv, Google Scholar y Semantic Scholar cubre la mayor parte de la producción científica relevante en ingeniería de software e inteligencia artificial, no puede descartarse la existencia de estudios pertinentes fuera de este conjunto.

En segundo lugar, la síntesis cualitativa adoptada, en lugar de un meta-análisis cuantitativo, limita la posibilidad de extraer estimaciones numéricas precisas sobre el rendimiento agregado de los modelos. Esta decisión fue justificada por la alta heterogeneidad en métricas, datasets y diseños experimentales del corpus; sin embargo, implica que las conclusiones cuantitativas reportadas (p. ej., el rango del 25–47% de caída de rendimiento) deben interpretarse como indicativas y no como parámetros estadísticamente consolidados.

En tercer lugar, la evaluación de calidad mediante cuatro criterios (QC1–QC4) fue realizada por un único revisor, sin proceso de doble ciego ni cálculo de concordancia inter-evaluador (p. ej., kappa de Cohen). Aunque se aplicaron rúbricas explícitas para minimizar la subjetividad, este diseño introduce un riesgo de sesgo de confirmación que no puede eliminarse completamente.

Futuras revisiones deberían incorporar revisión dual con cálculo de acuerdo.Finalmente, el corte temporal 2020–2025, aunque representativo de la etapa más dinámica del campo, excluye trabajos fundacionales previos (p. ej., modelos basados en LSTM o seq2seq preTransformer) que podrían aportar perspectiva histórica sobre la evolución de las limitaciones identificadas. Esta decisión fue deliberada y está justificada por el objetivo de centrarse en la generación LLM contemporánea, pero limita la capacidad de trazar tendencias de largo plazo

Líneas prioritarias de investigación futura

A partir de estos hallazgos emergen tres líneas prioritarias de investigación. La primera consiste en desarrollar benchmarks capaces de evaluar dimensiones más cercanas a la práctica profesional, incluyendo seguridad, mantenibilidad y coherencia arquitectónica, siguiendo iniciativas como CodeBenchGen (Zheng et al., 2024) y BigCodeBench (Zhuo et al., 2024).

La segunda apunta a la necesidad de estudios longitudinales que permitan analizar el impacto acumulado de los LLMs sobre la calidad del software, un vacío aún evidente en el corpus, pese a que Jin et al. (2024) ya reportan efectos acumulativos en escenarios reales de uso.

Finalmente, se requiere ampliar la investigación empírica hacia ecosistemas poco representados, como .NET/C#, considerando que trabajos como CodeGeeX (Zheng et al., 2023) y DeepSeek-Coder (Guo et al., 2024) muestran diferencias de rendimiento importantes entre lenguajes.

De los hallazgos identificados emergen cuatro agendas de investigación prioritarias. La primera consiste en el diseño de benchmarks específicos por ecosistema, incorporando dependencias de framework, como ASP.NET Core y Entity Framework, y contextos de repositorio real en C#, tomando como modelo un SWE-bench adaptado a .NET que incluya métricas de cobertura de pruebas y análisis estático con Roslyn.

La segunda apunta a la evaluación de seguridad como métrica primaria, integrando herramientas de análisis estático de seguridad (SAST), como CodeQL y SonarQube, en los pipelines de evaluación de LLMs, superando el uso exclusivo de pass@k y tomando como referencia metodológica el trabajo de Pearce et al. (2022) como punto de partida para un marco más sistemático.

La tercera línea propone estudios longitudinales de calidad de código, midiendo el impacto de la adopción continua de LLMs sobre indicadores de deuda técnica, cobertura de tests y frecuencia de defectos en repositorios reales a lo largo de periodos de 12 a 36 meses, con el diseño de Jin et al. (2024) como referencia replicable en otros contextos.

Finalmente, la cuarta agenda aborda el desarrollo de marcos de integración humano–IA en contextos de alta madurez, investigando cómo organizaciones con niveles de madurez CMMI 3–5 pueden integrar LLMs en sus procesos sin degradar la trazabilidad, el control de cambios y la conformidad normativa, vacío crítico

para sectores como fintech, salud y administración pública, donde la revisión no encontró ningún estudio representativo. Estas líneas emergen de las brechas identificadas en los 40 estudios primarios incluidos (RQ1–RQ3), siguiendo las recomendaciones de PRISMA (Page et al., 2021) y Kitchenham & Charters (2007).

Atender estas agendas de forma articulada permitirá reducir la distancia entre lo que los LLMs demuestran en laboratorio y lo que pueden ofrecer de manera confiable en la práctica. Con esa perspectiva en mente, cabe preguntarse qué nos dice, en conjunto, la evidencia acumulada.

La evidencia analizada no pone en duda que los LLMs pueden aportar valor real en ingeniería de software, las mejoras en productividad reportadas por Peng et al. (2023) y Ziegler et al. (2022) son consistentes y no deben minimizarse. Lo que sí cuestiona es la idea de que ese valor se materializa de forma automática. Para que los beneficios sean sostenibles en entornos profesionales reales, hace falta algo más que un modelo capaz: hacen falta prácticas de evaluación más honestas, procesos de integración que contemplen validación, seguridad y trazabilidad, y una disposición organizacional a rediseñar flujos de trabajo, no solo a insertar una herramienta nueva en los existentes.

En este sentido, como señalan Nascimento et al. (2024), el rendimiento de un LLM en un benchmark dice poco sobre su utilidad real si los procesos que lo rodean no están a la altura. Esta revisión comparte esa lectura y la extiende: la pregunta relevante ya no es solo qué tan bien genera código un modelo, sino en qué condiciones ese código puede integrarse de forma confiable en un sistema real, con personas reales, bajo presiones reales.

Ese desplazamiento, del modelo al sistema sociotécnico que lo contiene, es quizás el cambio de perspectiva más importante que la literatura reciente está comenzando a asumir, y que esta revisión busca contribuir a consolidar.

CONCLUSIONES

Esta revisión sistemática proporciona una visión consolidada y crítica del estado actual de la generación de código basada en LLMs en ingeniería de software, a partir del análisis de 40 estudios primarios publicados entre 2020 y 2025.

En relación con RQ1, se concluye que el prompt engineering se ha establecido como la técnica dominante para la interacción con LLMs, respaldada principalmente por estudios de alta calidad. Sin embargo, su predominio no implica suficiencia, ya que se observa una transición hacia enfoques más complejos, particularmente sistemas agénticos, que redefinen el rol de los modelos desde generadores de código hacia componentes activos en el ciclo de desarrollo.

Respecto a RQ2, los resultados evidencian una limitación metodológica significativa en la evaluación del rendimiento. La dependencia de métricas como pass@k y benchmarks sintéticos como HumanEval introduce un sesgo sistemático que sobreestima el desempeño de los modelos. La comparación con benchmarks más realistas como BigCodeBench y SWE-bench permite identificar una brecha estructural, denominada benchmark saturation gap, que cuestiona la validez externa de gran parte de la evidencia existente.

En cuanto a RQ3, se concluye que las limitaciones de los LLMs no son marginales, sino estructurales. Problemas como alucinaciones de código, vulnerabilidades de seguridad y degradación de calidad aparecen de forma consistente en el corpus, independientemente del nivel de calidad metodológica de los estudios. Asimismo, se identifican brechas críticas en la evaluación en entornos reales, particularmente en ecosistemas como .NET, así como en la ausencia de estudios longitudinales y de métricas orientadas a procesos.

A nivel general, los hallazgos de esta revisión sugieren que, si bien los LLMs han alcanzado un alto nivel de rendimiento en entornos controlados, su adopción efectiva en ingeniería de software requiere repensar tanto los modelos de evaluación como las prácticas de desarrollo. Por ello, se recomienda priorizar tres acciones concretas: avanzar hacia benchmarks más representativos que superen los ya saturados, incorporar métricas orientadas al impacto en procesos reales, más allá del pass@k, y fortalecer los mecanismos de validación, seguridad y trazabilidad en los entornos donde el código generado se integra. Solo bajo estas condiciones podrá materializarse una adopción efectiva y confiable de los LLMs en la ingeniería de software profesional.

Finalmente, este estudio contribuye al estado del arte al ofrecer una síntesis estructurada de técnicas, métricas y limitaciones, así como al identificar de manera explícita la brecha entre evaluación académica y aplicabilidad industrial, proporcionando una base para futuras investigaciones orientadas a cerrar esta distancia.

BIBLIOGRAFÍA

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). Language models are few-shot learners. In Advances in Neural Information Processing Systems 33 (NeurIPS 2020). https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

Chen, M., Tworek, J., Jun, H., Yuan, Q., de Oliveira Pinto, H. P., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., Ray, A., Puri, R., Krueger, G., Petrov, M., Khlaaf, H., Sastry, G., Mishkin, P., Chan, B., Gray, S., … Zaremba, W. (2021). Evaluating large language models trained on code [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2107.03374

Dakhel, A. M., Majdinasab, V., Nikanjam, A., Khomh, F., Desmarais, M. C., & Jiang, Z. M. J. (2023). GitHub Copilot AI pair programmer: Asset or liability? Journal of Systems and Software, 203, Article 111709. https://doi.org/10.1016/j.jss.2023.111709

Guo, D., Zhu, Q., Yang, D., Xie, Z., Dong, K., Zhang, W., Chen, G., Bi, X., Wu, Y., Li, Y., Luo, F., Xiong, Y., & Liang, W. (2024). DeepSeek-Coder:

When the large language model meets programming – The rise of code intelligence [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2401.14196

Jimenez, C. E., Yang, J., Wettig, A., Yao, S., Pei, K., Press, O., & Narasimhan,K. (2024). SWE-bench: Can language models resolve real-world GitHub issues? In Proceedings of the 12th International Conference on Learning Representations (ICLR 2024). https://arxiv.org/abs/2310.06770

Jin, K., Wang, C. Y., Pham, H. V., & Hemmati, H. (2024). Can ChatGPT support developers? An empirical evaluation of large language models for code generation. In Proceedings of the 21st International Conference on Mining Software Repositories (MSR 2024) (pp. 576–587). ACM. https://doi.org/10.1145/3643991.3644889

Kitchenham, B., & Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering (EBSE Technical Report EBSE-2007-01). Keele University & Durham University. https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf

Liu, J., Xia, C. S., Wang, Y., & Zhang, L. (2023). Is your code generated by ChatGPT really correct? Rigorous evaluation of large language models for code generation [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2305.01210

Liu, C., Yang, C., Zhang, J., Luo, Z., Yao, J., & Gao, C. (2025). Measuring and mitigating hallucination in code generation with LLMs [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2501.04127

Mastropaolo, A., Cooper, N., Palacio, D. N., Scalabrino, S., Poshyvanyk, D., & Oliveto, R. (2023). On the robustness of code generation techniques: An empirical study on GitHub Copilot. ACM Transactions on Software Engineering and Methodology, 32(5), Article 114. https://doi.org/10.1145/3597207

Nascimento, N., Alencar, P., & Cowan, D. (2024). Self-adaptive LLM-based agents for software development. In Proceedings of the 46th International Conference on Software Engineering (ICSE 2024) (pp. 2200–2212). ACM. https://doi.org/10.1145/3639478.3643102

Olausson, T. X., Inala, J. P., Wang, C., Gao, J., & Solar-Lezama, A. (2024). Is self-repair a silver bullet for code generation? In Proceedings of the 12th International Conference on Learning Representations (ICLR 2024). https://doi.org/10.48550/arXiv.2306.09896

Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., ... Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. BMJ, 372, Article n71. https://doi.org/10.1136/bmj.n71

Pearce, H., Ahmad, B., Tan, B., Dolan-Gavitt, B., & Karri, R. (2022). Asleep at the keyboard? Assessing the security of GitHub Copilot's code contributions. In Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP 2022) (pp. 754–768). IEEE. https://doi.org/10.1109/SP46214.2022.9833571

Peng, S., Kalliamvakou, E., Cihon, P., & Demirer, M. (2023). The impact of AI on developer productivity: Evidence from GitHub Copilot [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2302.06590

Raji, I. D., Bender, E. M., Paullada, A., Denton, E., & Hanna, A. (2021). AI and the everything in the whole wide world benchmark. In Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS 2021) – Datasets and Benchmarks Track

Rozière, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X. E., Adi, Y., Liu, J., Sauvestre, R., Remez, T., Rapin, J., Kozhevnikov, A., Evtimov, I., Bitton, J., Bhatt, M., Fernandes, C. C., Grattafiori, A., Ünal, T., Boissinot, B., ... Synnaeve, G. (2023). Code Llama: Open foundation models for code [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2308.12950

Schäfer, M., Nadi, S., Eghbali, A., & Tip, F. (2024). An empirical evaluation of using large language models for automated unit test generation. IEEE Transactions on Software Engineering, 50(1), 85–105. https://doi.org/10.1109/TSE.2023.3334955

Shinn, N., Cassano, F., Gopalan, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language agents with verbal reinforcement learning. In Advances in Neural Information Processing Systems 36 (NeurIPS 2023). https://arxiv.org/abs/2303.11366

Siddiq, M. L., & Da Silva Santos, J. C. (2024). A large-scale empirical study of code smell detection using code language models. ACM Transactions on Software Engineering and Methodology, 33(6), Article 145. https://doi.org/10.1145/3649596

Wang, X., Li, B., Song, Y., Xu, F. F., Tang, X., Zhuge, M., ... Neubig, G. (2024). OpenHands: An open platform for AI software developers as generalist agents [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2407.16741

Wang, Y., Le, H., Gotmare, A. D., Bui, N. D. Q., Li, J., & Hoi, S. C. H. (2025). Towards effective integration of LLM-generated code in CI/CD pipelines. In Proceedings of the 47th International Conference on Software Engineering (ICSE 2025) (pp. 1–13). ACM.

Xia, C. S., Deng, Y., Dunn, S., & Zhang, L. (2024). Agentless: Demystifying LLM-based software engineering agents [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2407.01489

Zhang, K., Li, J., Li, G., Shi, X., & Jin, Z. (2024). CodeAgent: Enhancing code generation with tool-integrated agent systems for real-world repo-level coding challenges. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 13643–13658). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.acl-long.737

Zheng, L., Chiang, W. L., Sheng, Y., Li, T., Zhuang, S., Wu, Z., ... Zhang, H. (2024). CodeBenchGen: Creating scalable execution-based code generation benchmarks [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2401.10020

Zheng, Q., Xia, X., Zou, X., Dong, Y., Wang, S., Xue, Y., Shen, Z., Wang, Z., Wang, H., Gu, Z., Zhang, Z., Zhu, J., Liang, Y., & He, K. (2023). CodeGeeX: A pre-trained model for code generation with multilingual benchmarking on HumanEval-X. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2023) (pp. 5673–5684). ACM. https://doi.org/10.1145/3580305.3599790

Zhuo, T. Y., Vu, M. C., Chim, J., Hu, H., Yu, W., Widyasari, R., ... Yao, J. (2024). BigCodeBench: Benchmarking code generation with diverse function calls and complex instructions [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2406.15877

Ziegler, A., Kalliamvakou, E., Li, X. A., Rice, A., Rifkin, D., Simister, S., Sittampalam, G., & Aftandilian, E. (2022). Productivity assessment of neural code completion. In Proceedings of the 6th ACM SIGPLAN International Symposium on Machine Programming (pp. 21–29). ACM. https://doi.org/10.1145/3520312.3534864

FORTALECIMIENTO DE LA COMPETENCIA DIGITAL INVESTIGATIVA MEDIANTE TIC EN ESTUDIANTES DE TRABAJO SOCIAL

M.Sc. Abel Huaygua Chalco
Universidad Amazónica de Pando
https://orcid.org/0009-0006-9930-8911
Santa Cruz, Bolivia | ahuayguachalco@gmail.com

https://doi.org/10.23670 FT.2026.1.24

Recibido 31/03/2026 - Aceptado 07/05/2026

RESUMEN

El objetivo de este estudio fue evaluar el impacto de un programa de capacitación en herramientas de Tecnologías de la Información y la Comunicación (TIC), diseñado bajo el modelo instruccional ADDIE, en el fortalecimiento de la competencia digital investigativa de estudiantes de la asignatura Modalidad de Graduación de la Carrera de Trabajo Social de la Universidad Amazónica de Pando.La investigación adoptó un enfoque mixto, con predominancia cuantitativa y apoyo cualitativo, de carácter aplicado y propositivo. La muestra estuvo conformada por 21 estudiantes, complementada con la participación de 23 docentes y del director de carrera, mediante encuestas, entrevistas semiestructuradas y análisis documental, lo que permitió una triangulación de la información. Los resultados evidenciaron mejoras significativas en cinco criterios de desempeño: búsqueda académica, gestión bibliográfica, aplicación de normas APA 7, redacción académica formal y organización del trabajo con herramientas ofimáticas. El análisis estadístico mediante la prueba de Chi-cuadrado (X^2 = 26,4; gl = 4; α = 0,05) confirmó diferencias significativas entre pretest y postest, validando la efectividad del programa. Se concluye que el modelo ADDIE constituye una estrategia pedagógica eficaz para el desarrollo de competencias digitales investigativas en educación superior, destacando la importancia de integrar herramientas TIC mediante enfoques formativos estructurados.

Palabras Clave: Competencia digital investigativa, TIC, modelo ADDIE, educación superior, formación investigativa

ABSTRACT

The objective of this study was to evaluate the impact of a training program in Information and Communication Technology (ICT) tools, designed under the ADDIE instructional model, on the strengthening of digital research competence among students enrolled in the Graduation Modality course of the Social Work Program at the Amazonian University of Pando. The research adopted a mixed-methods approach, predominantly quantitative with qualitative support, framed within an applied and propositional perspective. The sample consisted of 21 students, complemented by the participation of 23 faculty members and the program director, through surveys, semi-structured interviews, and documentary analysis, allowing for data triangulation. The results showed significant improvements in five performance criteria: academic search, bibliographic management, application of APA 7 standards, academic writing, and organization of work using office tools. Statistical analysis using the Chi-square test (X^2 = 26.4; df = 4; α = 0.05) confirmed significant differences between pretest and posttest, validating the effectiveness of the program.It is concluded that the ADDIE model represents an effective pedagogical strategy for strengthening digital research competence in higher education, highlighting the importance of integrating ICT tools through structured training approaches.

Keywords: digital research competence, TIC, ADDIE model, higher education, research training

INTRODUCCIÓN

La transformación digital en la educación superior plantea el reto de formar profesionales capaces de desenvolverse en entornos académicos mediados por tecnologías, donde el dominio de la competencia digital investigativa se convierte en un requisito esencial para garantizar la calidad de los trabajos finales de grado (TFG) y promover la producción científica universitaria (Area Moreira, 2018; Tobón, 2013; Bond et al., 2020).

En este contexto, diversos estudios han evidenciado limitaciones en el uso crítico, ético y estratégico de las TIC por parte de los estudiantes universitarios, especialmente en carreras de las ciencias sociales, lo que repercute en la calidad de sus procesos investigativos (García-Peñalvo, 2019; Monereo, 2010; Cabero-Almenara & Palacios-Rodríguez, 2021).

Estas debilidades se acentúan en regiones con restricciones tecnológicas, como la Amazonía boliviana, donde la brecha digital condiciona el acceso y aprovechamiento de recursos académicos digitales (Haleem et al., 2022).

La Carrera de Trabajo Social de la Universidad Amazónica de Pando (UAP) no es ajena a esta problemática. El diagnóstico realizado en la asignatura Modalidad de Graduación evidenció carencias en la búsqueda de información científica, la gestión bibliográfica, la aplicación de normas de citación y la redacción académica formal. Esta situación limita el desarrollo de competencias investigativas y afecta la calidad de los trabajos finales de grado, lo que evidencia la necesidad de implementar estrategias formativas orientadas al fortalecimiento de estas habilidades.

Frente a este escenario, el modelo instruccional ADDIE (Análisis, Diseño, Desarrollo, Implementación y Evaluación) se presenta como una alternativa metodológica pertinente para el diseño de programas formativos estructurados, dada su flexibilidad y eficacia en contextos educativos (Molenda, 2003; Gagné et al., 2005).

En este marco, el objetivo de la presente investigación fue evaluar el impacto de un programa de capacitación en herramientas TIC, diseñado bajo el modelo ADDIE, en el fortalecimiento de la competencia digital investigativa de los estudiantes de la Carrera de Trabajo Social de la Universidad Amazónica de Pando.

METODOLOGÍA

La investigación se desarrolló bajo un enfoque mixto, con predominancia del paradigma cuantitativo y apoyo cualitativo complementario, enmarcada en un diseño transeccional descriptivo–correlacional, de carácter aplicado y propositivo, orientado a la generación de una solución educativa contextualizada para el fortalecimiento de la competencia digital investigativa en estudiantes de Trabajo Social.

Se emplearon métodos teóricos como el histórico–lógico, analítico–sintético e hipotético–deductivo, así como métodos empíricos, entre los que destacaron las encuestas, entrevistas y el análisis documental. Esta combinación permitió la triangulación de la información y contribuyó a fortalecer la validez interna del estudio.

Las técnicas e instrumentos incluyeron encuestas estructuradas tipo Likert aplicadas a estudiantes en las fases de pretest y postest, entrevistas semiestructuradas dirigidas a docentes y al director de carrera, así como el análisis documental de planes de estudio y normativas académicas relacionadas con los trabajos finales de grado.

Asimismo, se emplearon guías de observación y matrices de operacionalización de variables, organizadas en las dimensiones conocer, hacer y ser, lo que permitió garantizar la trazabilidad de los indicadores y la consistencia del proceso investigativo.

Población y muestra

La población estuvo conformada por estudiantes de la Carrera de Trabajo Social de la Universidad Amazónica de Pando, específicamente aquellos inscritos en la asignatura Modalidad de Graduación durante la gestión 2024.

La muestra fue de tipo no probabilística e intencional, integrada por 21 estudiantes seleccionados bajo criterios de participación activa en el desarrollo de trabajos finales de grado, disponibilidad para participar en el proceso formativo y acceso a herramientas digitales básicas.

Asimismo, se incorporó la participación de 23 docentes y del director de carrera, seleccionados en función de su experiencia en la orientación de trabajos finales de grado y su vinculación directa con los procesos de formación investigativa, lo que permitió fortalecer la triangulación de la información.

Herramientas tecnológicas utilizadas

Durante el proceso investigativo se emplearon diversas herramientas digitales orientadas a la recolección, procesamiento y aplicación de la información, en correspondencia con las fases del estudio.

En la fase de recolección de datos se implementó Google Forms para la aplicación de encuestas estructuradas, lo que permitió la sistematización automatizada de la información. Para el análisis de los datos cuantitativos se recurrió a Microsoft Excel, mediante el uso de funciones estadísticas básicas que facilitaron el procesamiento de la información y la aplicación de la prueba de Chi-cuadrado para la validación de resultados.

En el desarrollo del programa de capacitación se incorporaron herramientas digitales orientadas a la investigación académica, tales como Google Scholar, Scielo y Redalyc para la búsqueda de información científica; Zotero para la gestión y organización de referencias bibliográficas; y Microsoft Word y Google Docs para la producción y estructuración de documentos académicos.

El uso articulado de estas herramientas permitió

fortalecer las competencias digitales investigativas de los participantes, integrando procesos de búsqueda, gestión y producción de información científica en entornos digitales.

Proceso metodológico

La investigación se desarrolló siguiendo las fases del modelo instruccional ADDIE (Análisis, Diseño, Desarrollo, Implementación y Evaluación). En la fase de análisis se realizó el diagnóstico de las competencias digitales investigativas mediante la aplicación de encuestas, entrevistas y revisión documental, lo que permitió identificar las principales limitaciones en el uso de herramientas TIC.

Posteriormente, en la fase de diseño se estructuró el programa de capacitación, definiendo objetivos formativos, contenidos y estrategias pedagógicas alineadas a los criterios de desempeño de la asignatura Modalidad de Graduación.

Durante la fase de desarrollo se elaboraron los materiales didácticos y recursos digitales necesarios para la implementación del programa. La fase de implementación se ejecutó mediante sesiones prácticas con los estudiantes, en las cuales se aplicaron herramientas TIC orientadas a la búsqueda, gestión y producción de información científica. Finalmente, en la fase de evaluación se compararon los resultados obtenidos en el pretest y postest, lo que permitió medir el impacto del programa mediante análisis estadístico.

Técnicas e instrumentos de recolección de datos

Se utilizaron encuestas estructuradas dirigidas a estudiantes, entrevistas semiestructuradas aplicadas a docentes y al director de carrera, así como análisis documental de planes de estudio y normativas institucionales relacionadas con los trabajos finales de grado.

Los instrumentos fueron diseñados en función de los objetivos de la investigación y sometidos a un proceso de validación de contenido mediante juicio de expertos, con el fin de garantizar su pertinencia, coherencia y alineación con las variables de estudio.

Consideraciones éticas

La investigación se desarrolló bajo principios éticos que garantizaron la participación voluntaria de los sujetos de estudio, así como la confidencialidad y el uso responsable de la información recolectada.

Asimismo, se obtuvo el consentimiento informado de los participantes, asegurando que los datos fueran utilizados exclusivamente con fines académicos y de investigación.

RESULTADOS

El diagnóstico inicial evidenció limitaciones en el dominio de herramientas digitales aplicadas a la investigación. Los resultados del pretest mostraron que el 81 % de los estudiantes desconocía el uso de gestores bibliográficos, el 76 % no accedía a bases de datos científicas y el 67 % no había recibido capacitación específica en TIC aplicadas a la investigación.

Tras la implementación del programa de capacitación basado en el modelo ADDIE, se aplicó un postest para evaluar el nivel de mejora en los criterios de desempeño definidos.

Los resultados evidencian un incremento significativo en todos los indicadores evaluados (Tabla 1). Esta tendencia se observa de manera comparativa en la Figura 1.

Tabla 1

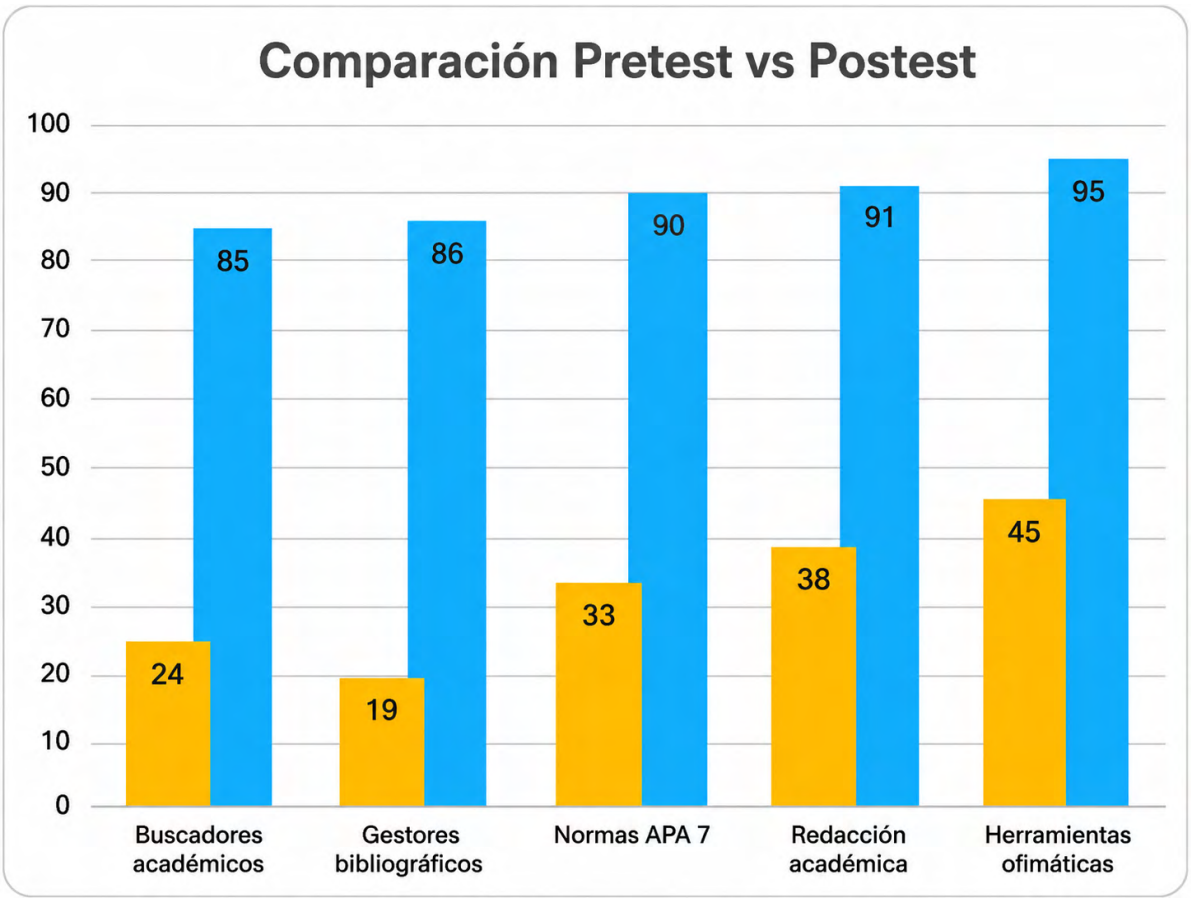
Comparación de resultados pretest y postest en competencias digitales investigativas

Criterio de desempeño	Pretest (%)	Postest (%)	Mejora (%)
Uso de buscadores académicos (Scholar, Scielo, Redalyc)	24 %	85 %	61 %
Uso de gestores bibliográficos (Zotero)	19 %	86 %	67 %
Aplicación de normas de citación (APA 7)	33 %	90 %	57 %
Redacción académica formal	38 %	91 %	53 %
Organización con herramientas ofimáticas (Word, Docs, Excel)	45 %	95 %	50 %

El análisis estadístico mediante la prueba de Chi-cuadrado ($X^2 = 26,4$; $gl = 4$; $\alpha = 0,05$) permitió determinar diferencias significativas entre los resultados del pretest y postest, confirmando la efectividad de la intervención.

Figura 1

Comparación de resultados pretest y postest en competencias digitales investigativas



Análisis de resultados

Los resultados evidencian una mejora cuantitativa en los cinco criterios de desempeño evaluados, relacionados con la búsqueda, gestión y producción de información científica mediante herramientas digitales.

Los indicadores fueron construidos a partir de la medición del nivel de dominio de competencias digitales investigativas, considerando el acceso a fuentes académicas, el uso de gestores bibliográficos, la aplicación de normas de citación, la redacción académica y la organización del trabajo mediante herramientas ofimáticas.

En el ámbito cualitativo, las entrevistas realizadas a docentes y al director de carrera evidenciaron mejoras en el uso de herramientas TIC, destacando avances en la autonomía investigativa y en la calidad de los trabajos finales de grado.

DISCUSIÓN

Los resultados obtenidos en la presente investigación evidencian que la implementación de un programa de capacitación basado en el modelo ADDIE contribuye significativamente al fortalecimiento de la competencia digital investigativa en estudiantes de educación superior. Este hallazgo se alinea con lo planteado por estudios previos, los cuales destacan la efectividad de los modelos instruccionales estructurados para el desarrollo de habilidades digitales en contextos académicos (Bond et al., 2020).

En particular, las mejoras observadas en el uso de buscadores académicos, gestores bibliográficos, normas de citación y herramientas de redacción evidencian que la integración sistemática de TIC en procesos formativos favorece no solo la adquisición de habilidades técnicas, sino también el desarrollo de competencias investigativas más complejas (Cabero-Almenara & Palacios-Rodríguez, 2021). Estos resultados coinciden con investigaciones recientes que destacan el papel de las competencias digitales como eje central en la formación universitaria (Heidari et al., 2021).

Desde una perspectiva crítica, es importante señalar que, si bien los resultados muestran mejoras sustanciales, estas se desarrollaron en un contexto específico y con una muestra limitada, lo que podría restringir la generalización de los hallazgos a otros entornos educativos. Asimismo, factores como la disponibilidad de recursos tecnológicos, la conectividad y el nivel previo de alfabetización digital pueden influir en la efectividad de la implementación del programa (Haleem et al., 2022).

En relación con el análisis cualitativo, las percepciones de docentes y autoridades académicas refuerzan los resultados cuantitativos, evidenciando cambios en la actitud de los estudiantes hacia el uso de herramientas digitales en la investigación. No obstante, también se

identifican desafíos asociados a la sostenibilidad de este tipo de intervenciones, particularmente en contextos donde la formación docente en TIC es limitada.

En este sentido, los hallazgos del estudio no solo confirman la pertinencia del modelo ADDIE como estrategia formativa, sino que también aportan evidencia sobre la necesidad de integrar programas de capacitación en competencias digitales como parte estructural de la formación universitaria (Redecker, 2021). Esto implica repensar *los procesos educativos desde una perspectiva que articule el uso de tecnologías con el desarrollo de habilidades investigativas, especialmente en contextos con brechas digitales (Selwyn, 2022).*

Finalmente, la investigación aporta al campo de estudio al evidenciar que el fortalecimiento de la competencia digital investigativa no depende exclusivamente del acceso a herramientas tecnológicas, sino de su integración pedagógica planificada, lo que constituye un elemento clave para la mejora de la calidad académica en la educación superior.

CONCLUSIONES

Los resultados de la investigación evidenciaron que la implementación de un programa de capacitación basado en el modelo instruccional ADDIE contribuye de manera significativa al fortalecimiento de la competencia digital investigativa en estudiantes de educación superior, demostrando la efectividad de intervenciones formativas estructuradas y contextualizadas.

El estudio permitió establecer que el desarrollo de competencias digitales investigativas no depende únicamente del acceso a tecnologías, sino de su integración pedagógica mediante procesos formativos organizados y orientados a la práctica. En este sentido, el modelo ADDIE se consolida como una estrategia pertinente para el diseño de programas de capacitación en contextos universitarios con limitaciones tecnológicas.

Asimismo, los hallazgos mostraron la necesidad de incorporar de manera sistemática la formación en competencias digitales dentro de los procesos académicos, particularmente en asignaturas vinculadas a la elaboración de trabajos finales de grado, contribuyendo a mejorar la calidad metodológica y científica de la producción estudiantil.

Finalmente, la investigación aportó evidencia empírica sobre el impacto positivo de las TIC en la formación investigativa, destacando la importancia de su integración pedagógica como un elemento clave para el fortalecimiento de la calidad académica en la educación superior.

En este contexto, se recomienda la incorporación de programas de formación en competencias digitales en el ámbito universitario, así como el desarrollo de estudios futuros que evalúen su sostenibilidad en distintos contextos educativos.

BIBLIOGRAFÍAS

Area Moreira, M. (2018). La alfabetización digital: evolución y desafíos actuales. Revista de Educación a Distancia (RED), 18(56), 1–18. https://doi.org/10.6018/red/56/1

Bond, M., Marín, V. I., Dolch, C., Bedenlier, S., & Zawacki-Richter, O. (2020). Digital transformation in higher education: A systematic review. Educational Technology & Society, 23(1), 1–15.

Cabero-Almenara, J., & Palacios-Rodríguez, A. (2021). The digital competence of educators in higher education: A systematic review. Sustainability, 13(8), 4312. https://doi.org/10.3390/su13084312

Cabero, J., & Romero, R. (2020). Herramientas TIC y educación colaborativa. Ediciones Pirámide.

Gagné, R. M., Wager, W. W., Golas, K. C., & Keller, J. M. (2005). Principios de diseño instruccional (5.ª ed.). Thomson Learning.

García-Peñalvo, F. J. (2019). Competencias digitales y transformación educativa: Un análisis desde la educación superior. Education in the Knowledge Society (EKS), 20, 1–12. https://doi.org/10.14201/eks2019_20_a17

Haleem, A., Javaid, M., Qadri, M. A., & Suman, R. (2022). Understanding the role of digital technologies in education: A review. Sustainable Operations and Computers, 3, 275–285. https://doi.org/10.1016/j.susoc.2022.05.004

Heidari, E., Mehrvarz, M., Marzooghi, R., & Stoyanov, S. (2021). The role of digital competence in academic research: A systematic review. Education and Information Technologies, 26, 1–24. https://doi.org/10.1007/s10639-020-10360-5

Molenda, M. (2003). In search of the elusive ADDIE model. Performance Improvement, 42(5), 34–37. https://doi.org/10.1002/pfi.4930420508

Monereo, C. (2010). ¡Ey, profesor! ¿Sabes qué es la competencia digital? Cuadernos de Pedagogía, (398), 12–15.

Quispe, L., & Condori, M. (2021). Competencia digital en estudiantes universitarios de ciencias sociales en Bolivia. Revista Electrónica Educare, 25(2), 92–108. https://doi.org/10.15359/ree.25-2.5

Redecker, C. (2021). European framework for the digital competence of educators: DigCompEdu. Publications Office of the European Union.

Reigeluth, C. M., & Carr-Chellman, A. A. (2009). Instructional-design theories and models: Building a common knowledge base (Vol. III). Routledge.

Selwyn, N. (2022). Education and technology: Key issues and debates (3rd ed.). Bloomsbury Academic.

Tobón, S. (2013). Competencias investigativas: Formación desde el enfoque complejo. Ecoe Ediciones.

UNESCO. (2018). Marco de competencias de los docentes en materia de TIC (versión 3). UNESCO. https://unesdoc.unesco.org/ark:/48223/pf0000265721

Universidad Amazónica de Pando. (2021). Plan de estudios de la Carrera de Trabajo Social. Dirección Académica.

TALLER CIENTÍFICO
FRONTERAS TECNOLÓGICAS

Un espacio académico diseñado para explorar las tendencias tecnológicas que están transformando el mundo, conectando a investigadores, profesionales y estudiantes con las fronteras del conocimiento científico y la innovación.

Red Hat Academy

aws academy Member Institution

ORACLE Academy

HUAWEI ICT Academy

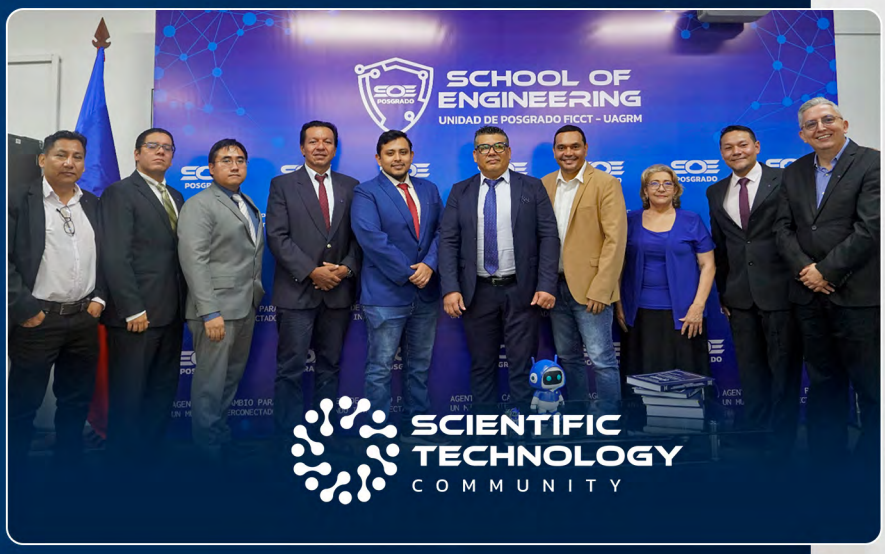
MikroTik

CISCO

EC-COUNCIL | ACADEMIA PARTNER

Investigación & Difusión

COMUNIDAD CIENTÍFICA DOCTORAL



Una comunidad académica orientada al fortalecimiento de la investigación científica, la producción académica, la innovación tecnológica y la formación del área posgrado mediante la colaboración y el intercambio de los distintos conocimientos.

"Conectando investigadores, conocimiento e innovación para construir el futuro."

PODCAST "COFFEE SOE"

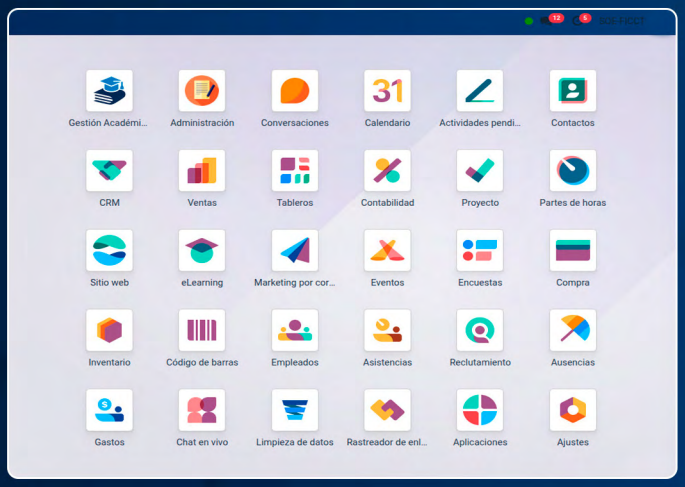
Un espacio para la divulgación científica y tecnológica donde especialistas, investigadores y profesionales comparten sus ***conocimientos, experiencias y las tendencias que están transformando la era del mundo digital.***

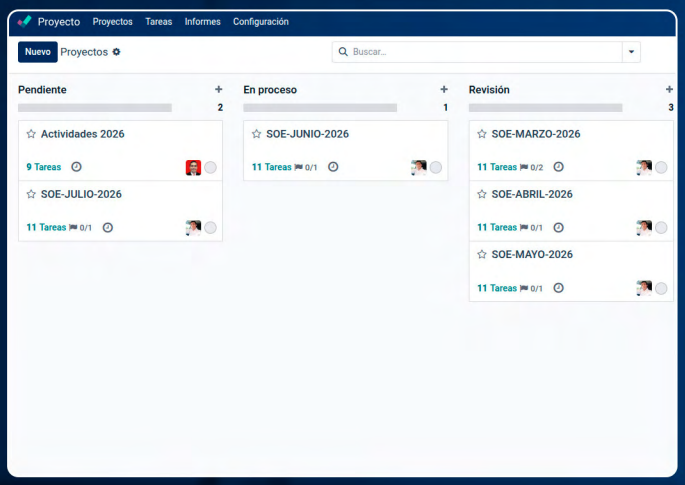
DESCUBRE NUEVAS IDEAS, EXPERIENCIAS Y MÁS TENDENCIAS TECNOLÓGICAS

ERP SOE | Transformación Digital en la Gestión Académica

Contamos con un ERP Académico Integral

SOE dispone de una plataforma ERP que integra y centraliza los distintos procesos académicos y administrativos, permitiendo una gestión más eficiente, automatizada e interconectada para toda la comunidad de posgrado.





Características destacadas

- Gestión Académica
- Integración con Moodle
- Administración y Finanzas
- CRM Institucional
- Recursos Humanos
- Automatización de Procesos
- Información en Tiempo Real

Biblioteca Digital eLibro

Acceso al conocimiento sin fronteras...

SOE pone a disposición de toda su comunidad académica posgradual la Biblioteca Digital eLibro, una plataforma que brinda acceso a una variedad de libros electrónicos y recursos especializados para fortalecer el ***aprendizaje, la investigación y la actualización profesional***.

Beneficios

- Acceso a miles de libros electrónicos.
- Consulta en línea desde cualquier dispositivo.
- Recursos académicos actualizados.
- Apoyo para investigaciones, tesis y proyectos.
- Herramienta de aprendizaje permanente

Conocimiento disponible 24/7 para aprender, investigar y crecer profesionalmente.

Boletín Científico “Fronteras Tecnológicas”

Durante la gestión 2025, SOE consolidó su compromiso con la investigación y la difusión del conocimiento mediante el lanzamiento del Boletín Científico Fronteras Tecnológicas, una iniciativa editorial orientada a promover la publicación y visibilización de investigaciones en ingeniería, tecnología e innovación.

Este espacio académico fortalece la cultura investigativa y contribuye a la generación de conocimiento científico dentro de la comunidad de posgrado.

Impulsando la INVESTIGACIÓN y la GENERACIÓN DE CONOCIMIENTO para un mundo interconectado.

SOE POSGRADO

SCHOOL OF ENGINEERING

(+591) 73600888 - 334 6703

Av. Busch, 2do. anillo (Módulo 232)

info@soe.uagrm.edu.bo

www.soeuagrm.edu.bo

Agentes de cambio para un mundo interconectado.