Modelo hibrido en cascada para la asignación de revisores en artículos científicos

A cascading hybrid model for assigning reviewers to scientific articles

Mg(c). Liseth Paola Claro Ascanio[1], **Mg. Cristian Camilo Tirado Cifuentes**[1]

[1] **Universidad de La Salle,** Facultad de Ingenierías, Maestría en Inteligencia Artificial en Profundización, Bogotá, Distrito Capital, Colombia.

Correspondencia: lisethclaro@gmail.co

Recibido: 11 mayo 2026. **Revisado:** 27 junio 2026. **Aceptado:** 09 julio 2026. **Publicado:** 16 julio 2026.

Cómo citar: L. P. Claro Ascanio and C. C. Tirado Cifuentes, "Modelo hibrido en cascada para la asignación de revisores en artículos científicos", *Revista Colombiana de Tecnologías de Avanzada (RCTA)*, vol. 2, no. 48, pp. 140–154, jul. 2026, doi: 10.24054/rcta.v2i48.4492

Resumen: Las revistas científicas enfrentan demoras en las publicaciones de nuevos números, debido a la escasez de revisores expertos y el aumento de sometimiento de artículos. Este estudio propone una combinación de modelos con diseño en cascada que utilizó como técnicas de procesamiento del lenguaje natural (PLN) y aprendizaje automático (AA) para automatizar la asignación de revisores. Se adoptó un enfoque cuantitativo con diseño experimental sobre registros históricos extraídos de la plataforma OJS de una revista científica real del área de Ciencias Sociales centrado en las disciplinas Administración de Empresas, contabilidad, económicas, pedagogía aplicada al fomento empresarial, emprendimiento, desarrollo local, innovación, tecnología (2021-2025). El modelo se estructura en tres etapas secuenciales: clasificación de aprobación del resumen, clasificación del campo disciplinar y la clasificación de la rapidez del revisor. Los resultados fueron: se evaluaron 100 combinaciones de representaciones textuales, clasificadores y técnicas de balanceo, con validación cruzada estratificada de 5 folds; Etapa 1, F1-macro =0.663 (SBERT+SMOTE+Random Forest); Etapa 2, F1-macro =0.966(TF-IDF + SMOTE+XGBoost); y Etapa 3, F1= 0.9578 (TF-IDF + NumCat+ RUS+ Random Forest). En conclusión, este diseño en cascada demostró ser viable, modular e interpretable para automatizar la asignación de revisores con datos reales (213 artículos, 117 evaluadores). La arquitectura secuencial permitió que cada etapa opere con la representación óptima para sus datos, contener errores y validar el modelo con datos reales.

Palabras clave: aprendizaje automático (AA); revisión por pares; inteligencia artificial (IA); asignación de revisores; modelos híbridos en cascada; procesamiento del lenguaje natural (PLN).

Abstract: Scientific journals face delays in publishing new issues due to a shortage of expert reviewers and an increase in article submissions. This study proposes a combination of models with a cascading desing that utilizes natural language processing (NLP) and machine learning (ML) tecniques to automate the assignment of reviewers. A quantitative approach with an experimental design was adopted, using historical records extracted from the OJS platform of a real scientific journal in the field of Social Sciences, focusing on the disciplines of Business administration, accounting, economics, pedagogy applied to

business development, entrepreneurship, local development, innovation and technology (2021-2025). The model is structured in three sequential stages: abstract acceptance classification, disciplinary field classification, and reviewer speed classification. The results were as follows: 100 combinations of textual representations, classifiers, and data balancing techniques were evaluated using 5-fold stratified cross-validation; stage 1, F1-macro=0.663 (SBERT+SMOTE+Random Forest); Stage 2, F1-Macro=0.966(TF-IDF+SMOTE+XGBoost); and stage 3, F1=0.9578 (TF-IDF+NumCat+RUS+Random Forest). In conclusión, this cascade design proved to be viable, modular, and interpretable for automating reviewer assignment using real-world data (213 articles, 117 evaluators). The sequential architecture allowed each stage to operate with the optimal representation for its data, contain errors, and validate the model with real-world data.

Keywords: machine learning (ML); peer review; artificial intelligence (AI); reviewer assignment; hybrid cascade models; natural language processing (NLP).

1. INTRODUCCIÓN

Las revistas científicas son las responsables de publicar artículos de alto impacto para la comunidad académica y científica. Para cumplir con este objetivo, los equipos editoriales desarrollan procesos, que no solo involucran al comité editorial, sino que también requieren la participación de revisores externos considerados expertos en el área temática del manuscrito, lo cual indica la importancia de su papel en la validación y aseguramiento de la calidad de los artículos publicados [1].

El sistema tradicional de revisión por pares o semiautomatizados, enfrentan desafíos a nivel global; por ejemplo, el aumento de envíos de artículos crea una sobrecarga en los equipos editoriales, dificultando la identificación de revisores adecuados y comprometiendo la calidad de las evaluaciones [2].

Además, estas dificultades se incrementan en América Latina, particularmente en Colombia, donde la escasez de revisores especializados, la falta de infraestructura tecnológica y la baja visibilidad internacional de las revistas empeoran la problemática [3], [4]; incluso factores como barreras idiomáticas, la concentración de redes académicas locales, la dificultad para identificar afinidades temáticas y retrasos en los tiempos de evaluación, han generado un aumento en el riesgo de sesgos y conflictos de intereses, disminuyendo la transparencia de los procesos editoriales [5], [6], [7].

En la Fig. 1, se muestra el tiempo que toma el proceso de revisión por pares.

Fig. 1. *Línea de tiempo: proceso de revisión por pares.*
Fuente: *elaboración propia.*
Nota: *estudios para realizar la línea de tiempo [8], [9], [10], [11], [12].*

Respecto, a la automatización de la asignación de revisores, se han planteado algoritmos que combinan: referencias a fines y peso de publicaciones para identificar evaluadores con afinidad temática [13]; enfoques basados en profundidad y amplitud del conocimiento disciplinar, reduciendo barreras para localizar expertos [14]; sistemas como PEERRec y PEERAssist que han explorado la predicción de decisiones editoriales a partir de interacciones entre manuscritos y revisores [2], [15]; evaluaciones automatizadas de la calidad de las revisiones [16]; el análisis de características lingüísticas y semánticas de artículos [17]. Todas estas soluciones apuntan a generalizar y fortalecer la integridad del arbitraje a través del uso de técnicas de Inteligencia artificial (IA) y procesamiento del lenguaje natural (PLN), permitiendo transformar los artículos en estructuras válidas para que los algoritmos ayuden a la toma de decisiones.

Si bien existen avances en ese campo, aún persisten limitaciones en la integración de múltiples fuentes de información, arquitecturas que integren de forma secuencial, y en la precisión de los modelos, como

metadatos bibliométricos, perfiles académicos y contenido textual, lo que reduce su capacidad de capturar la complejidad multidimensional de la experticia de un revisor [18], [19]. En segundo lugar, la precisión de los modelos actuales se ve limitada por la variabilidad en los puntajes de similitud, así mismo la escasez de datos etiquetados de calidad que permitan una evaluación de los algoritmos propuestos [20], [21].

Por consiguiente, la presente investigación propone un diseño híbrido en cascada que combina técnicas de PLN y aprendizaje automático (AA) para la asignación de revisores en revistas científicas, donde el aporte diferencial con respecto a los estudios analizados en esta investigación no radica en las técnicas empleadas (SBERT, TF-IDF, SMOTE, Random Forest, XGBoost o Regresión Logística), sino en encontrar la combinación más óptima según la naturaleza de los datos, ya que los estudios se orientaron principalmente a bases de datos de acceso abierto derivado de conferencias como ICLR y NeurIPS, y no sobre revistas científicas que mantiene esta información de manera confidencial. Este diseño se realizó empleando una secuencia de tres etapas y en su validación sobre datos operativos reales dentro de un contexto editorial latinoamericano con recursos limitados e información escasa, con el fin de mejorar la eficiencia y calidad del proceso de revisión por pares.

2. MARCO TEÓRICO

2.1. Los desafíos en la revisión por pares y las limitaciones en el proceso manual de la gestión editorial

El proceso de revisión por pares es el componente principal para la validación del conocimiento científico [22]. Pero ha enfrentado desafíos que comprometen su eficiencia como lo es el incremento de manuscritos sometidos, la escasez de revisores y las inconsistencias en las evaluaciones [2], [23], generando sobrecargas: aproximadamente el 20% de los revisores ejecutan más del 70% de las revisiones a nivel global [11]. Por ejemplo, en revistas como las de enfermería, requieren entre 25 y 30 invitaciones para lograr una única revisión efectiva [24].El escaso reconocimiento a revisores por ser una actividad voluntaria, el uso inapropiado de herramientas de IA y los sesgos por afiliación institucional, género o nacionalidad deterioran aún más la confianza del sistema [23], [25]. El estudio realizado por NeurIPS y citado en [2], documentó que el 57% de los artículos aceptados por un comité fueron rechazados por otros, y solo el 23% de los revisores afirman estar completamente seguros de sus recomendaciones [26].

Estos procesos se realizan de manera manual empeorando esta situación [27], al limitar la capacidad de procesar información de manera eficiente y generando asignaciones menos precisas [13], [28], [29]. Aunque existen sistemas de gestión editorial, estos no capturan la complejidad semántica del conocimiento científico [30]. En America Latina y Colombia, donde hay una menor densidad de investigadores en comparación con regiones como Europa o Norteamérica, el cual restringe la capacidad de respuesta del sistema [3]. La concentración de la evaluación científica en ciertos contextos geográficos y lingüísticos dificultan el acceso a revisores internacionales, aumentando la carga sobre los pocos especialistas bilingües disponibles [6], quienes son convocados de manera recurrente con la consecuente disminución en la calidad de las evaluaciones [4]. Estas prácticas tienden a privilegiar revisores de países con mayor producción científica, reduciendo la diversidad de perspectivas [31], y generando una baja visibilidad internacional de las revistas, lo que restringe su posicionamiento en índices globales [5], [32].

2.2. Inteligencia artificial, modelos de representación semántica y métricas de rendimiento en sistemas de asignación de revisores

La Inteligencia artificial (IA), apoyada en técnicas de procesamiento del lenguaje natural (PLN) y aprendizaje automático (AA), emergen como una alternativa para optimizar múltiples etapas del proceso editorial, permitiendo extraer representaciones semánticas que facilitan la comparación entre manuscritos y perfiles de revisores [22], [33], [34]. A partir de la revisión sistemática de la literatura y aplicación de la metodología prisma en bases de datos de Scopus, IEEE y Springer con las palabras claves “Automated Peer Review Recommendation, Reviewer Assignment using NLP, AI for Academic Peer Review”, se identificaron 137 artículos relacionados. Después de aplicar criterios de inclusión (IA / ML / PLN aplicado, últimos 5 años, escasez de revisores, sobrecarga de evaluadores) y exclusión (sin evidencia empírica, fuera del tema central, Sin propuesta aplicable, debates sin alternativas), se obtuvieron 22 artículos que fueron analizados, donde el 90% emplearon PLN como componente central, destacando modelos pre-

entrenados como BERT, SBERT, GPT y BART, así como técnicas de similitud semántica como TF-IDF, Word2Vec y similitud coseno [2], [14], [23].

En cuanto a las representaciones vectoriales de texto y modelos de palabras incrustadas (Word embeddings) como Word2Vec, han sido ampliamente utilizados para medir la afinidad temática [14], [35]. Sin embargo, presentan limitaciones al capturar la complejidad contextual de los textos. Los más recientes como BERT y SBERT generan embeddings contextualizados que mejoran sustancialmente la interpretación del contenido [2], [36]. Asimismo, el uso de umbrales dinámicos de similitud coseno y la combinación de métricas cuantitativas, han demostrado mejorar la precisión en la asignación de revisores [14], [35].

De acuerdo, con la literatura las técnicas de balanceo de clases, deben estar acorde con el nivel de los datos como lo es el sobremuestreo aleatorio (ROS) y SMOTE, que buscan compensar la subrepresentación de la clase minoritaria mediante duplicación o generación sintética de instancias [37], [38]; el submuestreo aleatorio (RUS), que reduce la clase mayoritaria [39]; y enfoque generativos más recientes basados en redes adversariales condicionales (CTGAN), orientados a capturar las distribuciones complejas en datos tabulares mixtos [40], [41]. Donde la selección entre estas técnicas suele depender del tamaño muestral, la severidad del desbalance y su interacción con el clasificador utilizado [42], [43].

En cuanto a las métricas de rendimiento más utilizadas se encuentra: F1 con un 45%, en tareas con clases desbalanceadas como la distinción entre revisiones de alta y baja calidad, donde la exactitud global resultaría engañosa al favorecer a la clase mayoritaria [16], [28], [44]; ROUGE el 27%, en sistemas de generación de texto, permitiendo comparaciones entre distintos enfoques [45], [46], [47]; la precisión y recall con el 23%, en contextos donde la interpretabilidad de cada componente es prioritaria [13], [35]. Por otro lado, el área bajo la curva (AUC-ROC), la cual válida la capacidad discriminativa del modelo a lo largo de distintos umbrales de clasificación, y la Exactitud (accuracy) que hace referencia al desempeño global en conjuntos de datos más equilibrados con un 14% [23], [44], [48].

2.3. Diseño de modelos híbridos en cascada para la asignación de evaluadores y el estado del conocimiento frente al estudio

Dada la diversidad de tipos de datos que contiene la metadata capturada por las revistas al momento de postular un artículo; el rendimiento de un único modelo que integre diferentes tareas se vuelve complejo y con bajo rendimiento [47]. Por esta razón, destacan los modelos híbridos en cascada como una estrategia para dividir la complejidad de un modelo en varios que funcionen de forma secuencial. Estos modelos pueden abordar temas frente a la asignación de evaluadores, estructurándose en varias fases: (i) preprocesamiento y representación semántica del manuscrito, utilizando técnicas como SBERT o BERT; (ii) extracción de características relevantes, incluyendo tópicos, palabras claves o aspectos específicos de contenidos; (iii) cálculo de similitud o afinidad entre el articulo y los perfiles de revisores y (iv) clasificación o ranking de evaluadores [49]. Esta arquitectura, la salida de cada etapa constituye en la entrada de la siguiente, lo cual se distingue de arquitecturas de extremo a extremo, donde todas las etapas se optimizan de forma conjunta [21], [50].

Los estudios revisados, han utilizado como base de datos las conferencias de ICLR o NeurIPS, que consta de revisión abierta y abundante información [2], [23], lo que limita su aplicabilidad en revistas pequeñas o medianas con recursos restringidos. Por consiguiente, el presente trabajo propone un sistema de asignación automática de revisores mediante un modelo con diseño híbrido en cascada que incorpora técnicas de PLN y AA, adaptado a un proceso editorial real.

3. METODOLOGÍA

3.1. Enfoque y diseño de la investigación

La investigación adoptó un enfoque cuantitativo con un diseño experimental orientado a la construcción y validación de un modelo híbrido en cascada, basado en el PLN y AA para la automatización de la asignación de revisores. Este enfoque es coherente con investigaciones propuestas sobre la automatización del proceso de revisión por pares donde se basa en modelos predictivos y evaluaciones cuantitativas [2], [13], [23].

En ese sentido, el modelo desarrollado fue híbrido en cascada, siguiendo principios de diseño modular y patrones de arquitectura para sistemas de aprendizaje automático que favorecen la reproducibilidad, interpretabilidad y escalabilidad [51]. Para la evaluación del desempeño del sistema se realizó mediante métricas que aseguran la validez y confiabilidad de los resultados de la investigación.

3.2. Recolección y preprocesamiento de los datos

Los datos fueron extraídos de la plataforma Open Journal Systems (OJS) en la versión 3.3.0.6, correspondientes al periodo 2021 al 2025, obteniendo 221 registros iniciales. Las variables recolectadas se agruparon en dos datasets principales: uno orientado a las características de los artículos y otro en la de los evaluadores, con el fin de poder facilitar el análisis diferenciado de cada dimensión de las tareas.

Dado que el alcance de la revista y las características de los artículos que son enviados para su evaluación y posible publicación, las áreas del conocimiento fueron reclasificadas conforme al Manual de Frascati de la OCDE[1] y al modelo de clasificación y reconocimiento de revistas científicas nacionales Publindex 2026[2], consolidándose en 6 categorías: Ciencias Sociales, Educación, Ciencias Económicas. Administrativas y Contables, Ciencias Agrícolas y Veterinarias y Ciencias Exacta. Dicha reclasificación se realizó debido a que las áreas de interés originalmente registradas en el sistema editorial no seguían una taxonomía estandarizada, lo cual dificultaba su uso directo como variable de clasificación.

No obstante, el uso de esta taxonomía podría generar un posible sesgo de granularidad frente a áreas de conocimiento con fronteras disciplinares difusas: la revista analizada se encuentra orientada a las Ciencias Sociales, con una concentración marcada en el área de Ciencias Económicas y Contables (56%) y una representación considerablemente menor de las demás áreas (44%), lo cual pudo favorecer artificialmente la discriminabilidad léxica observada en la etapa 2, al beneficiar la separación de la clase mayoritaria frente a la minoritaria. La variable objetivo de la etapa 1, presentó un desbalance: 90% artículos aprobados y 10% no aprobados. Este comportamiento ha sido también reportado en estudios similares sobre la predicción editorial en contextos académicos [2], [44].

El preprocesamiento incluyó: (1) la depuración y eliminación de los registros incompletos, duplicados y las recomendaciones canceladas por los revisores; (2) la normalización de variables categóricas mediante estandarización de cadenas de texto; (3) transformación de variables temporales para el cálculo de tiempo de respuesta y cumplimiento; y (4) tratamiento de los datos textuales (resúmenes, áreas de interés y recomendaciones), mediante técnicas de PLN, como la limpieza de texto, eliminación de caracteres especiales, *tokenizacion*, lematización y vectorización con múltiples técnicas de representación, con el propósito de convertir la información no estructurada en representaciones numéricas aptas para el análisis estadístico y el entrenamiento del modelo. Tras este proceso se obtuvieron 213 registros, divididos en dos datasets: (1) artículos (213 registros, 6 áreas de interés por Modelo de Clasificación y Reconocimiento de Revistas Científicas Nacionales Publindex 2026) y (2) evaluadores (117 registros únicos), cuya distribución se presenta en la Tabla 1.

Tabla 1: *Distribución de evaluadores por áreas de interés MinCiencias Publindex 2026.*

Areas de interés (Minciencias Publindex 2026)	**N° Evaluadores**	**% del total**
Ciencias Económicas, Administrativas y Contables	72	61.5%
Educación.	28	23.9%
Ciencias Agrícolas y Veterinarias	3	2.6%
Ciencias Sociales	12	10.3%
Ciencias exactas	2	1.7%
Total	**117**	**100%**

Fuentes: *autores del proyecto*

Con el fin de garantizar la reproducibilidad del modelo, la Tabla 2 resume cada etapa, las variables de entrada empleadas y la de objetivo correspondiente. Todas las variables se obtuvieron directamente de los registros históricos del sistema OJS y se normalizaron para la etapa 3 mediante StandardScaler (variables numéricas) y OneHotEncoder (variables categóricas), ajustados sobre el subconjunto de entrenamiento de cada partición experimental.

Tabla 2: *variables de entrada y objetivo de cada etapa.*

Etapa	Variable entrada	Variable objetivo
E1- clasificacion de aprobación del resumen	Resumen textual del artículo, vectorizado mediante representación textual más óptima	Recomendación: Aprobado (1) / No aprobado (0)
E2- clasificación del campo disciplinar	Resumen textual del artículo, vectorizado mediante representación textual más óptima	Campo disciplinar: Ciencias Económicas, Administrativas y Contables (1) / Otras áreas (0)
E3 – clasificación	(i) Área disciplinar del evaluador vectorizado	Rapidez del evaluador:

[1] [52]

[2] [53]

del evaluador	mediante representación textual; (ii) 8 variables numéricas escaladas con StandardScaler, índice H de Google Scholar, promedio de días de asignación, N° de evaluaciones históricas, % de evaluaciones entregadas antes del límite y 4 estadísticas de dispersión temporal (máximo, mínimo, desviación estándar y mediana de días respecto al límite); (iii) 2 variables categóricas nominales (escolaridad, nacionalidades codificadas con OneHotEncoder)	Rápido (1) /No rápido (0) construido a partir del percentil 33 del promedio de días respecto al límite de entrega.

3.3. Tratamiento del desbalance de clases, división de los datos y estrategia de validación

Se identificó en los datasets utilizados en este estudio un desbalance severo en la etapa 1 (aprobación: razón 9.1:1) y leve en la etapa 2 (campo disciplinar: 1.3:1); en evaluadores la razón fue aproximadamente 2:1. Según [54], un dataset desbalanceado ocurre cuando una o varias clases que están representadas con menor frecuencia que otras, generando predicciones sesgadas; para mitigarlo existen técnicas como al nivel de datos, algoritmos y evaluación.

Se implementaron cuatro técnicas de balanceo: ROS (*Random Oversampling*) que duplica ejemplos existentes de la clase minoritaria sin generar nueva información [37]; SMOTE genera ejemplos sintéticos interpolando muestras cercanas de la clase minoritaria [38], con k_neighbors=2 para la clasificación de artículos y k_neighbors=3 para evaluadores; RUS (*Random Undersampling*), reduce la clase mayoritaria eliminando muestras de forma aleatoria [39]; y CTGAN (*Conditional Tabular Generative Adversarial Network*), una red generativa adversarial condicional que produce datos sintéticos tabulares capturando distribuciones complejas en variables numéricas y categóricas [40], [41]. La selección de la técnica más óptima se determinó empíricamente evaluando cada técnica en combinación con las distintas representaciones textuales y clasificadores del modelo [42], [43], utilizando métricas de evaluación como accuracy, precisión, recall y F1-macro [44], [48].

En la etapa de clasificación de artículos se empleó la validación cruzada estratificada de 5 folds sobre los 213 registros, sin división previa train/test fijo, por lo que todas las métricas reportadas corresponden al promedio y desviación estándar de los cinco folds, garantizando que la proporción de clases se mantuvieran representativa en cada iteración [55]. Las representaciones TF-IDF y Word2Vec fueron ajustadas únicamente sobre los datos de entrenamiento de cada fold; los embeddings pre-entrenados (BERT, SBERT, Gemini) se generaron con caché antes de la partición, sin introducir leakage [56]. La métrica utilizada fue F1-macro, que penalizó el colapso sobre la clase minoritaria frente al F1-binary [57]. Y para el modelo de clasificación de evaluadores, se asumió una partición estratificada 80/20 (train/test) con validación cruzada de 5 folds sobre el subconjunto de entrenamiento. Se aplicó ajuste de hiperparámetros, y se realizó la búsqueda mediante gridSearchCV con StratifiedKfold interno de 3 folds como criterio. Las métricas manejadas fueron: F1-weighted sobre el test externo, precisión, recall y AUC-ROC. La variable objetivo (Rápido=1 / No rápido = 0), se construyó a partir del percentil 33 del promedio de días respecto al límite de entrega.

Con el fin de garantizar la reproducibilidad de los experimentos se fijó una semilla de aleatoriedad común (*random_state* = 42) en todas las divisiones del conjunto de datos, así como en los algoritmos de clasificación empleados. El balanceo de clases se aplicó únicamente en el subconjunto de entrenamiento de cada fold, preservando el subconjunto de prueba libre de contaminación por datos sintéticos [58].

3.4. Técnicas de representación textual y clasificadores

Los textos fueron convertidos a variables numéricas mediante técnicas de vectorización descritas en la Tabla 3.

Tabla 3: *Técnicas de representación textual empleadas.*

Técnica	Descripción	Aplicación
TF-IDF	Term Frequency-Inverse Document Frequency; vectorización (n-gramas 1-2, 5000 características)	Clasificación de artículos y evaluadores
Word2Vec	Embedding distribuido entrenado sobre el corpus de entrenamiento de cada fold (size = 300, Windows=5, epochs =30)	
BERT	Bert-base-multilingual-cased;representacion contextual [CLS] de 768	

	dimensiones, con cache en disco
SBERT	Paraphrase-multilingual-MiniLM-Ll2-V2; embeddings de oraciones para similitud semántica.
Gemini Emb. 2	Models/Gemini-embedding-2-preview; salida de 768 dimensiones, tipo de tarea CLASSIFICATION

Para el modelo de clasificación de evaluadores se integraron ocho variables numéricas (índice h Google. Días de asignacion, evaluaciones históricas y días respecto al límite), estandarizadas con StandardScaler y dos variables categóricas nominales (Escolaridad y Nacionalidad) codificadas con OneHotEncoder para evitar relaciones numéricas artificiales [59]. Las features de texto, numéricas y categóricas se concatenaron horizontalmente antes del entrenamiento. Se evaluaron cuatro clasificadores con ajuste de hiperparámetros mediante GridSearchCV, resumida en la Tabla 4.

Tabla 4: *Clasificadores y configuración de hiperparámetros.*

Clasificador	Configuración principal	Ajuste
Regresion Logistica	solver= 'saga', class_weight= 'balanced', max_iter=2000	GridSearchCV(C ∈ {0.1,1,5,10})
SVM	Kernel= ' rbf', probability =true, class_weight= 'balanced'	GridSearchCV (C, gamma)
Random Forest	N_estimators = 200-300, class_weight= 'balanced'	GridSearchCV(n_estimator, max_depth, learning_rate)
XGBoost	Scale_pos_weight; calculado sobre y_train original del fold; eval_metric='logloss '	GridSearchCV(n_estimators, max_depth, learning_rate)

3.5. Arquitectura del diseño híbrido en cascada

La arquitectura del modelo se encuentra enmarcada por tres etapas de clasificación binaria secuencialmente encadenadas donde la salida de cada etapa condiciona la activación de la siguiente (Fig. 2).

En la etapa 1 (clasificación de aprobación del resumen): el resumen textual del artículo se vectorizó y se clasificó en Aprobado (1) y No aprobado (0), determinando si el articulo continua en el sistema o recibe respuesta inmediata (ver tabla 2).

En la etapa 2 (Clasificación del campo disciplinar), el resumen de articulo aprobado en la etapa anterior se vectorizó y se clasificó en el campo disciplinar Ciencias Económicas, Administrativas y Contables (1) y otras áreas (0), funcionando como filtro temático que delimita el subconjunto de evaluadores candidatos en la etapa siguiente (ver tabla 2).

En la etapa 3 (clasificación de evaluadores), se construyó un vector de características por evaluador a partir de tres fuentes de información: representaciones textuales de palabra claves temáticas, variables numéricas categóricas nominales (ver tabla 2), con las cuales se clasificó la rapidez de respuesta (Rápido (1) / No rápido (0)), construida a partir del percentil 33 del promedio de días respecto al límite de entrega. La probabilidad posterior p(Rápido) del modelo genero un ranking de los cinco evaluadores con mayor probabilidad de respuesta oportuna. Para cada una de las tres etapas se evalúo de manera independiente el conjunto de representaciones textuales, técnicas de balaceo y clasificadores descritos en la sección 3.4, seleccionando en cada caso la combinación con el desempeño más optimo según la naturaleza de los datos de cada etapa; los resultados de esta comparación y la configuración final seleccionada se presenta en la sección 4.4.

En cuanto a las variables predictoras seleccionadas para la Etapa 3 constituyen aproximaciones (proxies) imperfectas de la experticia y disponibilidad real del evaluador, y deben ser interpretadas con cautela. El índice h de Google Scholar, aunque ampliamente disponible, no refleja de forma directa la especialidad temática del evaluador: un índice h reducido no implica que necesariamente menor experticia, particularmente en áreas de conocimiento altamente especializadas o donde la producción científica es menor pero la pertinencia temática puede llegar a ser alta. De manera similar, los tiempos históricos de respuesta empleados como variable predictiva presentan una variabilidad considerable y con sensibles factores externos no controlados por el modelo, tales como la carga laboral del evaluador, su disponibilidad temporal, compromisos institucionales y la estacionalidad académica (por ejemplo, periodos de vacaciones o cierre de semestre). Estas limitaciones no invalidan el uso de chas variables, pero condicionan el alcance interpretativo de las predicciones en la Etapa 3 y deben tenerse en cuenta al desplegar el modelo en un entorno de producción.

La Fig. 2, resume las diferentes combinaciones realizadas en cada una de las etapas descritas.

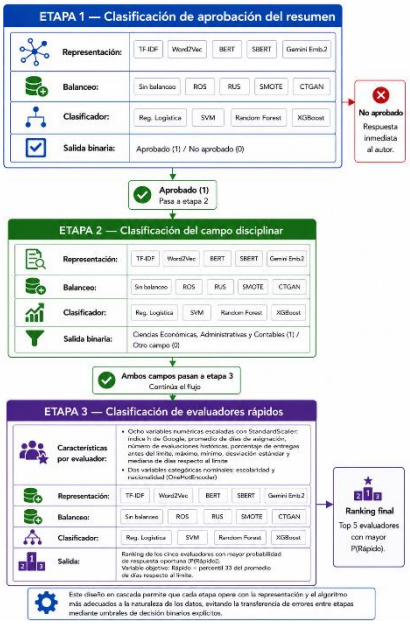

Fig. 2. *Arquitectura híbrida en cascada.*
Fuente: *autores del proyecto.*

4. RESULTADOS

4.1. Características de los conjuntos de datos

El Dataset 1 de artículos, es destinado para las etapas 1 (E1) y 2 (E2), con 213 registros únicos, con cuatro variables: identificador, resumen, área de interés y recomendación editorial. El dataset 2 de evaluadores, para la etapa 3 (E3) con 213 registros de revisores y consolidadas en 117 evaluadores únicos con 11 variables originales ampliadas a 16 tras el preprocesamiento. La variable objetivo binaria de la E3 se construyó con el percentil 33(umbral =-1.0 días): un evaluador es clasificado como Rápido (1) si su promedió de días respecto al límite es ≤ -1.0, indicando entrega anticipada.

4.2. Experimentación de las Etapas 1 y 2

Para la construcción del modelo implicó explorar un espacio de búsqueda de 100 configuraciones únicas por etapas (5 representaciones x 4 clasificadores x 5 estrategias de balanceo) sobre el dataset de artículos, usando F1-macro como métrica principal con un detector automático de colapso: F1-min < 0.10 en la clase minoritaria en todos los folds, la combinación fue excluida del ranking.

En la Etapa 1 (Clasificación aprobación del resumen) (con desbalance severo 9.1:1) (E1): Se generó un total de 5 combinaciones con colapso total en los 5 folds, todas involucrando TGAN + Regresión Logística o SVM. La combinación más óptima fue SBERT+SMOTE+Random Forest con F1-macro = 0.663±0.122, seguida de TF-IDF+SMOTE+Random Forest (0.659) y TF-IDF+SMOTE+Regresion logística (0.656). El valor de la desviación estándar demostró la dificultad del problema: 21 instancias negativas, presentando entre 3 a 4 ejemplos de la clase minoritaria en prueba. El desempeño del clasificador global acumulado fue precisión macro =0.63, recall macro = 0.63, exactitud = 87%. Esta diferencia entre la exactitud alta y los valores de precisión y recall considerablemente menores demuestra que el efecto del desbalance de clases (90% aprobados frente al 10% no aprobados): la exactitud se vea favorecida por la clase mayoritaria y resulta poco informativa sobre la capacidad real del modelo para identificar artículos no aprobados, mientras que las métricas macro, al pondera ambas clases por igual, reflejan con mayor fidelidad la dificultad del problema. Por esta razón se escogió F1-macro como métrica principal de selección de modelos en esta etapa. Un patrón similar, aunque menos pronunciado se observa en la distribución de áreas de interés (56% Ciencias Económicas, Administrativas y Contables, 44% otras áreas), la cual pudo favorecer ligeramente el desempeño de las combinaciones evaluadas en la etapa 2. En la Tabla 5, muestra el ranking de las mejores combinaciones.

Tabla 5: *Top 6 combinaciones Etapa 1 (F1-macro, sin colapso total)*

Embedding	Balanceo	Clasificador	F1-macro	F1 macro std
SBERT	SMOTE	Random Forest	0.6633	0.1220
TF-IDF	SMOTE	Random Forest	0.6586	0.1248
TF-IDF	SMOTE	Regresión logística	0.6560	0.1295
BERT	Sin balanceo	Random Forest	0.6452	0.1170
Gemini Emb. 2	Sin balanceo	Random Forest	0.6452	0.1170
Word2Vec	Sin balanceo	Random Forest	0.6452	0.1170

Nota. *La fila en color verde: combinación seleccionada para el modelo en cascada.*
Fuente*: Elaboración propia.*

La Tabla 6, compara el F1 por técnica de balanceo en ambas etapas. ROS obtuvo el mayor F1 en E1 (0.6011), seguido de SMOTE (0.5948), ambos mayores que sin balanceo (0.5848). TGAN presentó mayor frecuencia de colapso (2.65 folds en promedio) y RUS redujo el recall de la clase

mayoritaria. La diferencia entre ROS (0.6011) y SMOTE (0.5948) equivalen a 0.0063 de F1-macro, una dimensión inferior a la desviación estándar entre folds observada en ambas técnicas, por lo que no resulta estadísticamente concluyente por si sola. También, se debe mencionar que, en la columna de folds colapsadas de la tabla 5, ROS presento un promedio ligeramente menor que SMOTE en la Etapa 1; en ese sentido, la selección de SMOTE no se apoya en una superioridad empírica frente a ROS en estos dos indicadores puntuales, sino en un criterio teórico de generalización: mientras que ROS duplica ejemplos existentes de la clase minoritaria sin aportar variabilidad adicional, lo que lleva incrementar el riesgo de sobreajuste a instancias repetidas, SMOTE genera ejemplos sintéticos mediante interpolación entre vecinos cercanos, favoreciendo una frontera de decisión más suave y potencialmente más generalizable fuera de la muestra de validación cruzada utilizada. Dado que la diferencia numérica entre ambas técnicas es marginal, reconociendo que ROS podría llegar a constituir una alternativa igualmente válida frente a validaciones con conjuntos de datos de mayor tamaño.

***Tabla 6**: F1-macro promedio por técnicas de balanceo (E1 y E2, 100 combinaciones).*

Balanceo	F1-macro E1	F1-macro E2	Folds col.
ROS	0.6011	0.8718	0.85
SMOTE	0.5948	0.8717	0.90
Sin balanceo	0.5848	0.8752	1.00
TGAN	0.5632	0.8748	2.65
RUS	0.3156	0.8528	0.55

***Fuente.** Elaboración propia*

La Etapa 2 (Clasificación campo disciplinar) (con desbalance leve 1.3:1) (E2): Obtuvo el conjunto más equilibrado (120 vrs. 93 registros) permitiendo que todas las combinaciones de F1-macro fueran >0.90 sin colapso. La combinación **TF-IDF + SMOTE+XGBoost** alcanzó F1-macro = 0.9662±0.0363, con un AUC-ROC =0.9915 y F1-min_clase =0.9615. Los folds 2 y 3 lograron F1=1.0000, indicando la alta discriminabilidad léxica del vocabulario económico-administrativo en la lematización con spaCy. La Tabla 7 muestra las mejores combinaciones en esta etapa.

***Tabla 7:** Top5 combinaciones Etapa 2(F1-macro, 5 folds CV)*

Emb	Bal	Clasif	F1-macro	F1 std
TF-IDF	SMOTE	XGBoost	0.9662	0.0363
Gemini Emb2	Sin balanceo	XGBoost	0.9621	0.0322
SBERT	Sin Balanceo	Random Forest	0.9613	0.0247
Word2Vec	ROS	XGBoost	0.9516	0.0264
BERT	SMOTE	XGBoost	0.9466	0.0446

***Fuente**. Elaboración propia.*

***Nota**. Emb=Embeddings, Bal=balanceo, Clasif=clasificador, el sombreado verde es la combinación seleccionada para el diseño en cascada.*

4.3. Experimentación de la Etapa 3 Clasificación evaluadores (E3)

Se evaluaron 5 técnicas de representación combinadas con 4 clasificadores sobre 117 evaluadores únicos, con partición estratificada 80/20 (train =93, test=24). El tamaño reducido del conjunto es consistente en la limitación general del dataset (213 registros en Etapa 1 y 2, 117 evaluadores y un conjunto de prueba externo de solo n=24 en la Etapa 3), que condiciona la robustez estadística y la generalización de métricas reportadas en la etapa, cuya confiabilidad resulta sensible al tamaño del conjunto de prueba; los resultados obtenidos en esta etapa deben ser interpretados, como muestras de una prueba de concepto y no como una estimación de desempeño estable en producción. Igualmente, se evaluaron 4 técnicas de balanceo por remuestreo (ROS; SMOTE, RUS y CTGAN), para cada una de las representaciones textuales, en lugar de aplicar a priori una estrategia única fija. Esta comparación mostró que la técnica óptima varía según la representación: RUS resultó preferible para TF-IDF+ NumCat, SBERT+ NumCat, BERT+NumCat y Gemini+NumCat, mientras que CTGAN más adecuado para Word2Vec+NumCat, lo que demuestra que el remuestreo por submuestreo (RUS) tiende a favorecer a la mayoría de las representaciones evaluadas en este conjunto de datos, aunque esta relación debe confirmarse en estudios futuros con un conjunto de datos de mayor tamaño [63].

Es importante indicar; Primero, el umbral del percentil 33 (p33) utilizado para crear la variable objetivo (Rápido/no rápido), se calculó sobre el conjunto de datos completo, cuando se debió limitarse al subconjunto de entrenamiento en cada partición experimental; esta decisión introduce una fuga de información (leakage) parcial en la decisión de la variable objetivo, ya que el umbral incorpora información del conjunto de prueba, lo que puede llegar a comprometer en parte la validez de la evaluación reportada. La corrección completa de este procedimiento implicaría repetir la experimentación de la Etapa 3 con el umbral recalculado por partición, las conclusiones derivadas de esta etapa deben interpretarse con un

alcance inferencial limitado, como resultado preliminar de factibilidad técnica y no como una estimación insesgada del desempeño del modelo en producción. Segundo, para Gemini Embedding 2 se emplearon representaciones de 256 dimensiones en lugar de 768 que son las habituales para el modelo, debido a restricciones de la API utilizada; esta reducción dimensional estableció una limitación experimental que afectó la comparabilidad de Gemini frente a las demás representaciones evaluadas y pudo estar sesgado, desfavoreciendo su posicionamiento relativo dentro del ranking de desempeño de la Tabla 8.

La combinación TF-IDF + Numcat +RUS+ Random Forest fue la más óptima con F1 =0.957 (precisión=0.9609, recall=0.9583), obteniendo una buena estabilidad en validación cruzada (CV F1=0.9475±0.0471), con respecto al resto de combinaciones evaluadas. Le siguieron BERT+ NumCat CTGAN con regresión Logística (F1=0.9179) y Word2Vec+NumCat con ROS y Random Forest (F1=0.9141). SBERT+Numcat y Gemini+NumCat, ambas con RUS y Regresión Logística, alcanzaron un desempeño intermedio y exactamente idéntico entre si (F1=0.8761), lo cual es consistente con la limitación ya señalada sobre el tamaño reducido del conjunto de prueba (n=24). El Ensemble Voting Soft (RF+XGB+SVM+LR) obtuvo F1 =0.873, sin aportar mejora sobre el modelo seleccionado. La Tabla 8 se compara las principales combinaciones evaluadas.

Tabla 8: *Comparación de las técnicas en la etapa 3 (conjunto de prueba externo, n=24)*

Técnica	Bal.	Clasif	F1-w	AUC	CV F1
TF-IDF+NumCat	RUS	RF	0.9578	0.9926	0.9475±0.0471
BERT+NumCat	CTGAN	RL	0.9179	0.8741	0.8507±0.068
Word2Vec+NumCat	RUS	RF	0.9141	0.9704	0.9570±0.039
SBERT+Numcat	RUS	RL	0.8761	0.9185	0.8301±0.076
Gemini+NumCat	RUS	RL	0.8761	0.8963	0.8518±0.084
Ensemble(soft)	*SMOTE*	*RF+XGB+SVM+RL*	*0.8733*	*0.9556*	*0.9047±0.0513*

Nota. *RL=Regresión Logística, RF=Random Forest, Bal. = balanceo, CV F1=validación cruzada en el subconjunto de entrenamiento (5 folds). La fila de color azul claro es la combinación seleccionada del modelo más óptimo para esa etapa (TF-IDF+NumCat con RUS y Random Forest, F1=0.9578). Las combinaciones SBERT+NumCat y gemini+NumCat, ambas con RUS y Regresión Logística, obtuvieron exactamente el mismo desempeño en el conjunto de prueba; dado que el tamaño reducido del conjunto de prueba (n=24), esta coincidencia no corresponde a un error de transcripción, sino a que ambos modelos clasificaron correctamente el mismo subconjunto de evaluadores.*

4.4. Desempeño del sistema en cascada

En la Tabla 9, se resumen las combinaciones escogidas para cada etapa, predominando el mayor F1-macro (E1 y E2) y F1-weighted (E3) sin colapso sobre la clase minoritaria, garantizando que el sistema no derive en un único resultado para artículos rechazados o evaluadores lentos.

Tabla 9: *Combinaciones seleccionadas para el sistema hibrido en cascada.*

Etapa	Combinación optima	Métrica	Validación
1	SBERT+SMOTE+ Random Forest	F1-macro: 0.663±0.122	CV-5 estratificada
2	TF-IDF+SMOTE+XG Boost	F1-macro:0.966± 0.036	CV-5 estratificada
3	TF-IDF+NumCat+RUS+RandomForest	F1: 0.9578	80/20 externo

Fuente. *Elaboración propia.*
Nota. *CV-5= validación cruzada estratificada de 5 folds, 80/20= partición externa.*

Este diseño en cascada permitió que cada etapa operará con la representación y el algoritmo más adecuados a la naturaleza de los datos, evitando la transferencia de errores entre etapas mediante umbrales de decisión binarios explícitos [62]. La Fig. 3, resumen el proceso de tratamiento de los conjuntos de datos para cada modelo generado en cada etapa.

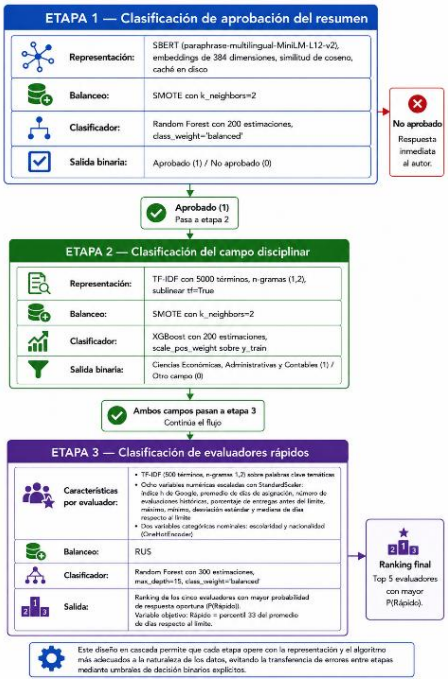

Fig 3. *Arquitectura del diseño hibrido en cascada con los resultados en cada una de las etapas.*
Fuente. *autores del proyecto.*

La prueba se realizó con artículos reales postulados a la revista científico objetivo, seleccionando aquellos que contenían decisiones contradictorias de los evaluadores. En esta prueba, la probabilidad posterior estimada por el clasificador de la E1 para la clase de "Aprobado" fue de 75.8%; esta cifra corresponde a la salida probabilística del modelo (no a la medida de exactitud ni a un intervalo de confianza estadístico) y debe ser interpretado como el grado de certeza del clasificador sobre ese caso puntual, en el marco del desempeño global reportado para la Etapa 1 (F1_macro=0.663). De manera análoga, la E2 asignó al campo disciplinar una probabilidad posterior de 54.8% hacia el campo de Ingeniería y la E3 generó un ranking de 5 posibles evaluadores según su probabilidad estimada de respuesta oportuna, como se observa en la Fig 4, y frente a la aplicabilidad operativa del sistema, es necesario precisar sus condiciones de funcionalidad. El modelo depende de variables históricas (número de evaluaciones previas, tiempo de respuestas pasado) que no están disponibles para evaluadores nuevos sin historial en el sistema; en estos casos, el modelo de la Etapa 3 no podrá generar una predicción confiable y necesitará una regla de respaldo (por ejemplo, asignación aleatoria ponderada por afinidad temática) hasta acumular historial suficiente. Asimismo, en escenarios de sobrecarga editorial, picos de sometimiento de artículos y ranking de evaluadores producido por el modelo no incorporará de forma dinámica la disponibilidad actual del evaluador, por lo que su uso debe complementarse con una verificación manual de disponibilidad antes de la invitación formal. Por consiguiente, el sistema puede considerarse funcional como herramienta de apoyo a la decisión editorial en el contexto experimental evaluado (213 artículos, 117 evaluadores de una revista real), pero su despliegue en producción a mayor escala requerirá una validación adicional con evaluadores nuevos y bajo condiciones de carga variable.

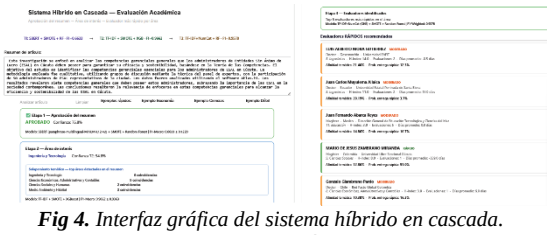

Fig 4. *Interfaz gráfica del sistema híbrido en cascada.*
Fuente. *Elaboración propia.*

5. CONCLUSIONES

El diseño híbrido en cascada construido en este estudio demostró tener un desempeño funcional para la asignación automática de revisores bajo las condiciones evaluadas en un conjunto de datos reducidos (213 artículos, 117 evaluadores), con validación cruzada estratificada de 5 folds en las etapas 1 y 2, y la partición externa de solo 24 casos en la Etapa 3. En las limitaciones y justificaciones señaladas sobre el cálculo del umbral p 33, el tamaño muestral y la naturaleza de proxy de variables como el índice h, donde la arquitectura permitió asignar representaciones óptimas por subproblema, contener errores entre etapas y validar el modelo con datos operativos reales, sin que implique una garantía de desempeño equivalente en escenarios de producción a mayor escala o con evaluadores nuevos. La selección de las representaciones textuales, técnicas de balanceo y clasificadores por etapa fue determinante: SBERT resultó óptimo en E1 por su capacidad de capturar similitud semántica de oraciones; TF-IDF con lematización dominó en E2 por la discriminabilidad léxica del vocabulario; y la Random Forest en E3 en prueba, demostrando que no existe una representación universalmente superior [64], sino que todo depende de la naturaleza de los datos de cada tarea. Adicionalmente, con el desbalance severo en el dataset de artículos de E1 (9.1:1), el balanceo de clases de SMOTE(K=2) fue más estable que CTGAN (2.65 folds colapsadas) y RUS, mientras que para embeddings de alta dimensión resultó más viable en la búsqueda de hiperparámetros. La lematización con spaCy y el ajuste del vectorizador sobre los datos de entrenamiento, fueron condiciones necesarias para estimaciones de generalización confiables. El perfil textual enriquecido temáticamente de E3 mostró que el diseño semántico del conjunto de datos puede compensar limitaciones de tamaño muestral en representaciones dispersas [58].

5.1. Limitaciones

El tamaño del conjunto de los datos utilizados en este estudio es reducido, el cual generó una alta varianza inter-fold en E1 (21 instancias negativas); el umbral p33 de E3, se calculó sobre el dataset completo y en producción debe restringirse al subconjunto de entrenamiento; y Gemini Embedding 2 se generó con 256-d (vrs. 768-d habituales) por restricciones de la API, afectando potencialmente su posición en el ranking.

5.2. Trabajos futuros

Implementar Fine-tuning de BERT multibiligue sobre el corpus de resúmenes en español para mejorar el F1-macro de la E1; ampliación del dataset con otras revistas para extender la cobertura

disciplinar; implementar la actualización incremental del modelo E3 para mitigar concept drift en el historial de evaluadores; y aplicar evaluación de LLMs como clasificadores zero-shot en E1 para contextos de datos muy escasos. Por otro lado, se deberán implementar acciones como: recalcular el umbral p 33 únicamente sobre el subconjunto de entrenamiento en cada partición para eliminar el sesgo actualmente reconocido en la Etapa 3; profundizar en la selección y validación de variables predictiva alternativas a los proxies utilizados (índice h, tiempos históricos de respuesta); evaluar la generalización temporal del modelo mediante validación con datos periodos posteriores al del entrenamiento (2021-2025); y explorar mecanismos que permitan asignar evaluadores nuevos sin historial previo para adaptar el ranking a la disponibilidad real en escenarios de sobrecarga editorial.

Agradecimientos: Los autores agradecen al editor de la revista que participó en este proyecto, cuya información constituyó al insumo central. Este trabajo se realizó en la modalidad de trabajo de grado para obtener el título de Maestría en Inteligencia Artificial de la Universidad de la Salle. Se utilizó la herramienta de IA (CHATGPT) solamente para orientar el diseño de material gráfico (Fig.1 y Fig 2); este instrumento no participó en la redacción del texto, en el análisis de los datos, la programación de los modelos ni en la interpretación de los resultados presentados en este manuscrito, los cuales fueron elaborados completamente por los autores.

REFERENCIAS

[1] J. P. Gisbert y M. Chaparro, «Reglas y consejos para ser un buen revisor por pares de manuscritos científicos», *Gastroenterol. Hepatol.*, vol. 46, n.º 3, pp. 215-235, mar. 2023, doi: 10.1016/j.gastrohep.2022.03.005.

[2] P. K. Bharti, T. Ghosal, M. Agarwal, y A. Ekbal, «PEERRec: An AI-based approach to automatically generate recommendations and predict decisions in peer review», *Int. J. Digit. Libr.*, vol. 25, n.º 1, pp. 55-72, mar. 2024, doi: 10.1007/s00799-023-00375-0.

[3] J. A. Huete-Perez y N. Salvatierra, «Assessing biomedical research capacities in selected countries of Latin America: challenges, opportunities, and recommendations», *Front. Res. Metr. Anal.*, vol. 10, jul. 2025, doi: 10.3389/frma.2025.1594303.

[4] A. Severin y J. Chataway, «Overburdening of peer reviewers: A multi-stakeholder perspective on causes and effects», *Learn. Publ.*, vol. 34, n.º 4, pp. 537-546, 2021, doi: 10.1002/leap.1392.

[5] A. Becerril García y S. Córdoba González, Eds., *Conocimiento abierto en América Latina: trayectoria y desafíos*. en Colección Grupos de trabajo. Erscheinungsort nicht ermittelbar: CLACSO, 2021.

[6] J. A. C. Osorio, «Problemáticas de las revistas científicas colombianas»:, *Sci. Tech.*, vol. 29, n.º 4, pp. 145-147, dic. 2024, doi: 10.22517/23447214.25746.

[7] J. C. Calderón, «Dear Editor», *Colomb. Medica*, vol. 42, n.º 3, pp. 406-407, 2011, doi: 10.25100/cm.v42i3.889.

[8] N. Felisi, R. Vecchio, y A. Odone, «When peer review drags on: the harm to early career researchers», *Front. Res. Metr. Anal.*, vol. 11, feb. 2026, doi: 10.3389/frma.2026.1740381.

[9] B.-C. Björk y D. Solomon, «The publishing delay in scholarly peer-reviewed journals», *J. Informetr.*, vol. 7, n.º 4, pp. 914-923, oct. 2013, doi: 10.1016/j.joi.2013.09.001.

[10] J. Huisman y J. Smits, «Duration and quality of the peer review process: the author's perspective», *Scientometrics*, vol. 113, n.º 1, pp. 633-650, oct. 2017, doi: 10.1007/s11192-017-2310-5.

[11] Publons, «Publons' Global State Of Peer Review 2018», Publons, London, UK, sep. 2018. doi: 10.14322/publons.GSPR2018.

[12] B. Aczel, B. Szaszi, y A. O. Holcombe, «A billion-dollar donation: estimating the cost of researchers' time spent on peer review», *Res. Integr. Peer Rev.*, vol. 6, n.º 1, p. 14, nov. 2021, doi: 10.1186/s41073-021-00118-2.

[13] T. A. Mahmud, B. M. M. Hossain, y D. Ara, «Automatic Reviewers Assignment to a Research Paper Based on Allied References and Publications Weight», en *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, Greater Noida, India: IEEE, dic. 2018, pp. 1-5. doi: 10.1109/CCAA.2018.8777730.

[14] X. Zheng, P. Li, y X. Wu, «An Automatic Paper-Reviewer Recommendation Algorithm Based on Depth and Breadth», *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 9, n.º 1, pp. 607-616, feb. 2025, doi: 10.1109/TETCI.2024.3402694.

[15] P. K. Bharti, S. Ranjan, T. Ghosal, M. Agrawal, y A. Ekbal, «PEERAssist: Leveraging on Paper-Review Interactions to Predict Peer Review Decisions», en *Towards Open and Trustworthy Digital Societies*,

Springer, Cham, 2021, pp. 421-435. doi: 10.1007/978-3-030-91669-5_33.

[16] L. Ramachandran, E. F. Gehringer, y R. K. Yadav, «Automated Assessment of the Quality of Peer Reviews using Natural Language Processing Techniques», *Int. J. Artif. Intell. Educ.*, vol. 27, n.º 3, pp. 534-581, sep. 2017, doi: 10.1007/s40593-016-0132-x.

[17] A. Perković Paloš *et al.*, «Linguistic and semantic characteristics of articles and peer review reports in Social Sciences and Medical and Health Sciences: analysis of articles published in Open Research Central», *Scientometrics*, vol. 128, n.º 8, pp. 4707-4729, ago. 2023, doi: 10.1007/s11192-023-04771-w.

[18] J. Jovanovic y E. Bagheri, «Reviewer assignment problem: A scoping review», 13 de mayo de 2023, *arXiv*: arXiv:2305.07887. doi: 10.48550/arXiv.2305.07887.

[19] X. Zhao y Y. Zhang, «Reviewer assignment algorithms for peer review automation: A survey», *Inf. Process. Manag.*, vol. 59, n.º 5, p. 103028, sep. 2022, doi: 10.1016/j.ipm.2022.103028.

[20] C. Cousins, J. Payan, y Y. Zick, «Into the Unknown: Assigning Reviewers to Papers with Uncertain Affinities», arXiv.org. Accedido: 4 de mayo de 2026. [En línea]. Disponible en: https://arxiv.org/abs/2301.10816v2

[21] D. Mimno y A. McCallum, «Expertise modeling for matching papers with reviewers», en *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, en KDD '07. New York, NY, USA: Association for Computing Machinery, ago. 2007, pp. 500-509. doi: 10.1145/1281192.1281247.

[22] H. Bauchner y F. P. Rivara, «Use of artificial intelligence and the future of peer review», *Health Aff. Sch.*, vol. 2, n.º 5, p. qxae058, may 2024, doi: 10.1093/haschl/qxae058.

[23] I. Boukhris y C. Zaâbi, «A GAN-BERT based decision making approach in peer review», *Soc. Netw. Anal. Min.*, vol. 14, n.º 1, p. 107, may 2024, doi: 10.1007/s13278-024-01269-y.

[24] S. M. Kadri, N. Dorri, M. Osaiweran, P. Garyali, y M. Petkovic, «Scientific Peer Review in an Era of Artificial Intelligence», en *Scientific Publishing Ecosystem*, Springer, Singapore, 2024, pp. 397-413. doi: 10.1007/978-981-97-4060-4_23.

[25] T. Ghosal, S. Kumar, P. K. Bharti, y A. Ekbal, «Peer review analyze: A novel benchmark resource for computational analysis of peer reviews», *PLOS ONE*, vol. 17, n.º 1, p. e0259238, ene. 2022, doi: 10.1371/journal.pone.0259238.

[26] A. Barnett, L. Allen, A. Aldcroft, T. L. Lash, y V. McCreanor, «Examining uncertainty in journal peer reviewers' recommendations: a cross-sectional study», *R. Soc. Open Sci.*, vol. 11, n.º 9, p. 240612, doi: 10.1098/rsos.240612.

[27] W. Yuan, P. Liu, y G. Neubig, «Can We Automate Scientific Reviewing?», 30 de enero de 2021, *arXiv*: arXiv:2102.00176. doi: 10.48550/arXiv.2102.00176.

[28] R. Verma, K. Shinde, H. Arora, y T. Ghosal, «Attend to Your Review: A Deep Neural Network to Extract Aspects from Peer Reviews», en *Neural Information Processing*, Springer, Cham, 2021, pp. 761-768. doi: 10.1007/978-3-030-92310-5_88.

[29] C. Candal-Pedreira, J. Rey-Brandariz, L. Varela-Lema, M. Pérez-Ríos, y A. Ruano-Ravina, «Challenges in peer review: how to guarantee the quality and transparency of the editorial process in scientific journals», *An. Pediatría Engl. Ed.*, vol. 99, n.º 1, pp. 54-59, jul. 2023, doi: 10.1016/j.anpede.2023.05.006.

[30] S. P. J. M. (. S. Horbach y W. (. W. Halffman, «The changing forms and expectations of peer review», *Res. Integr. Peer Rev.*, vol. 3, n.º 1, p. 8, sep. 2018, doi: 10.1186/s41073-018-0051-5.

[31] J. P. Tennant *et al.*, «A multi-disciplinary perspective on emergent and future innovations in peer review», 29 de noviembre de 2017, *F1000Research*. doi: 10.12688/f1000research.12037.3.

[32] M. Salatino y G. Banzato, «Confines históricos del Acceso Abierto latinoamericano», *Conoc. Abierto En América Lat. Trayectoria Desafíos Ciudad Autónoma B. Aires CLACSO Univ. Autónoma Estado México 2021 P 79-115*, 2021, Accedido: 19 de septiembre de 2025. [En línea]. Disponible en: https://www.memoria.fahce.unlp.edu.ar/library?a=d&c=libros&d=Jpm4982

[33] M. Majadly y M. Last, «Leveraging peer-review aspects for extractive and abstractive summarization of scientific articles», *Int. J. Data Sci. Anal.*, vol. 19, n.º 3, pp. 537-550, abr. 2025, doi: 10.1007/s41060-024-00665-z.

[34] A. Checco, L. Bracciale, P. Loreti, S. Pinfield, y G. Bianchi, «AI-assisted peer review», *Humanit. Soc. Sci. Commun.*, vol. 8, n.º 1, p. 25, ene. 2021, doi: 10.1057/s41599-020-00703-8.

[35] S. Azad *et al.*, «A Reviewer Recommender System for Scientific Articles Using a New Similarity Threshold Discovery Technique», en *Proceedings of the Fourth International*

Conference on Trends in Computational and Cognitive Engineering, Springer, Singapore, 2023, pp. 503-518. doi: 10.1007/978-981-19-9483-8_42.

[36] J. Devlin, M.-W. Chang, K. Lee, y K. Toutanova, «BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding», presentado en Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), jun. 2019, pp. 4171-4186. doi: 10.18653/v1/N19-1423.

[37] M. Kubát y S. Matwin, «Addressing the Curse of Imbalanced Training Sets: One-Sided Selection», presentado en International Conference on Machine Learning, 1997. Accedido: 21 de noviembre de 2025. [En línea]. Disponible en: https://www.semanticscholar.org/paper/Addressing-the-Curse-of-Imbalanced-Training-Sets%3A-Kub%C3%A1t-Matwin/ebc3914181d76c817f0e35f788b7c4c0f80abb07

[38] N. V. Chawla, K. W. Bowyer, L. O. Hall, y W. P. Kegelmeyer, «SMOTE: Synthetic Minority Over-sampling Technique», *J. Artif. Intell. Res.*, vol. 16, pp. 321-357, jun. 2002, doi: 10.1613/jair.953.

[39] T. Wongvorachan, S. He, y O. Bulut, «A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining», *Information*, vol. 14, n.º 1, p. 54, ene. 2023, doi: 10.3390/info14010054.

[40] L. Xu, M. Skoularidou, A. Cuesta-Infante, y K. Veeramachaneni, «Modeling Tabular data using Conditional GAN», 28 de octubre de 2019, *arXiv*: arXiv:1907.00503. doi: 10.48550/arXiv.1907.00503.

[41] M. E. S.-G. González-Pérez Pedro P., «Addressing the class imbalance in tabular datasets from a generative adversarial network approach in supervised machine learning - Máximo E Sánchez-Gutiérrez, Pedro P González-Pérez, 2023», *J. Algorithms Comput. Technol.*, nov. 2023, Accedido: 5 de mayo de 2026. [En línea]. Disponible en: https://journals.sagepub.com/doi/10.1177/17483026231215186

[42] J. M. Johnson y T. M. Khoshgoftaar, «Survey on deep learning with class imbalance», *J. Big Data*, vol. 6, n.º 1, p. 27, mar. 2019, doi: 10.1186/s40537-019-0192-5.

[43] G. Kovács, «An empirical comparison and evaluation of minority oversampling techniques on a large number of imbalanced datasets», *Appl. Soft Comput.*, vol. 83, p. 105662, oct. 2019, doi: 10.1016/j.asoc.2019.105662.

[44] A. C. Ribeiro, A. Sizo, H. Lopes Cardoso, y L. P. Reis, «Acceptance Decision Prediction in Peer-Review Through Sentiment Analysis», en *Progress in Artificial Intelligence*, Springer, Cham, 2021, pp. 766-777. doi: 10.1007/978-3-030-86230-5_60.

[45] P. K. Bharti, T. Ghosal, M. Agrawal, y A. Ekbal, «How Confident Was Your Reviewer? Estimating Reviewer Confidence from Peer Review Texts», en *Document Analysis Systems*, Springer, Cham, 2022, pp. 126-139. doi: 10.1007/978-3-031-06555-2_9.

[46] A. Kumar, T. Ghosal, y A. Ekbal, «A Deep Neural Architecture for Decision-Aware Meta-Review Generation», en *2021 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, sep. 2021, pp. 222-225. doi: 10.1109/JCDL52503.2021.00064.

[47] A. Kumar, T. Ghosal, S. Bhattacharjee, y A. Ekbal, «Towards automated meta-review generation via an NLP/ML pipeline in different stages of the scholarly peer review process», *Int. J. Digit. Libr.*, vol. 25, n.º 3, pp. 493-504, sep. 2024, doi: 10.1007/s00799-023-00359-0.

[48] P. K. Bharti, M. Navlakha, M. Agarwal, y A. Ekbal, «PolitePEER: does peer review hurt? A dataset to gauge politeness intensity in the peer reviews», *Lang. Resour. Eval.*, vol. 58, n.º 4, pp. 1291-1313, dic. 2024, doi: 10.1007/s10579-023-09662-3.

[49] A. Kumar, T. Ghosal, S. Bhattacharjee, y A. Ekbal, «Investigations on Meta Review Generation from Peer Review Texts Leveraging Relevant Sub-tasks in the Peer Review Pipeline», *Lect. Notes Comput. Sci. Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinforma.*, vol. 13541 LNCS, pp. 216-229, 2022, doi: 10.1007/978-3-031-16802-4_17.

[50] C.-H. Lai y P.-Y. Peng, «A Hybrid Deep Learning Method to Extract Multi-features from Reviews and User–Item Relations for Rating Prediction», *Int. J. Comput. Intell. Syst.*, vol. 16, n.º 1, p. 109, jun. 2023, doi: 10.1007/s44196-023-00288-5.

[51] V. Lakshmanan, S. Robinson, y M. Munn, «Machine Learning Design Patterns».

[52] «Microsoft Word - ANEXO 7. AREAS OCDE». Accedido: 21 de noviembre de 2025. [En línea]. Disponible en:

https://minciencias.gov.co/sites/default/files/upload/convocatoria/anexo_7._areas_ocde_0.pdf

[53] «Convocatoria de Clasificación y Reconocimiento de Revistas Científicas Nacionales - Publindex 2026», Minciencias. Accedido: 8 de julio de 2026. [En línea]. Disponible en: https://minciencias.gov.co/convocatorias/convocatoria-clasificacion-y-reconocimiento-revistas-cientificas-nacionales-publindex

[54] H. He y E. A. Garcia, «Learning from Imbalanced Data», *IEEE Trans. Knowl. Data Eng.*, vol. 21, n.º 9, pp. 1263-1284, sep. 2009, doi: 10.1109/TKDE.2008.239.

[55] R. Kohavi, «A study of cross-validation and bootstrap for accuracy estimation and model selection», en *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, en IJCAI'95. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., ago. 1995, pp. 1137-1143.

[56] J. Dodge, G. Ilharco, R. Schwartz, A. Farhadi, H. Hajishirzi, y N. Smith, «Fine-Tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping», arXiv.org. Accedido: 7 de mayo de 2026. [En línea]. Disponible en: https://arxiv.org/abs/2002.06305v1

[57] M. Sokolova y G. Lapalme, «A systematic analysis of performance measures for classification tasks», *Inf. Process. Manag.*, vol. 45, n.º 4, pp. 427-437, jul. 2009, doi: 10.1016/j.ipm.2009.03.002.

[58] G. C. Cawley y N. L. C. Talbot, «On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation», *J Mach Learn Res*, vol. 11, pp. 2079-2107, ago. 2010.

[59] J. T. Hancock y T. M. Khoshgoftaar, «CatBoost for big data: an interdisciplinary review», *J. Big Data*, vol. 7, n.º 1, p. 94, nov. 2020, doi: 10.1186/s40537-020-00369-8.

[60] L. Yang y A. Shami, «On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice», arXiv.org. Accedido: 7 de mayo de 2026. [En línea]. Disponible en: https://arxiv.org/abs/2007.15745v3

[61] L. Breiman, «Random Forests», *Mach. Learn.*, vol. 45, n.º 1, pp. 5-32, oct. 2001, doi: 10.1023/A:1010933404324.

[62] R. Gopalan, D. Onniyil, G. Viswanathan, y G. Samdani, «Hybrid models combining explainable AI and traditional machine learning: A review of methods and applications», *World J. Adv. Eng. Technol. Sci.*, vol. 15, pp. 1388-1402, may 2025, doi: 10.30574/wjaets.2025.15.2.0635.

[63] R. Blagus y L. Lusa, «SMOTE for high-dimensional class-imbalanced data», *BMC Bioinformatics*, vol. 14, n.º 1, p. 106, mar. 2013, doi: 10.1186/1471-2105-14-106.

[64] D. H. Wolpert, «La falta de distinciones a priori entre algoritmos de aprendizaje | Computación neuronal | MIT Press», vol. 8, n.º 7, Neural Computation, 1 de octubre de 1996. Accedido: 8 de mayo de 2026. [En línea]. Disponible en: https://direct.mit.edu/neco/article-abstract/8/7/1341/6016/The-Lack-of-A-Priori-Distinctions-Between-Learning?redirectedFrom=fulltext