REVISTA CIENTÍFICA
Ciencia y Método
ISSN 3103-1072

EDITORIAL GRUPO AEA

Artículo Científico

Modelos clásicos vs. Transformers en la detección de texto generado

Classic models vs. Transformers in detecting generated text

Espin-Riofrio, César [1]
https://orcid.org/0000-0001-8864-756X
cesar.espinr@ug.edu.ec
Universidad de Guayaquil, Ecuador, Guayaquil.

Mendoza-Morán, Verónica [2]
https://orcid.org/0000-0001-7520-3505
veronica.mendozam@ug.edu.ec
Universidad de Guayaquil, Ecuador, Guayaquil.

Vergara-Bello, Oswaldo [3]
https://orcid.org/0009-0008-7679-4894
oswaldo.vergarab@ug.edu.ec
Universidad de Guayaquil, Ecuador, Guayaquil.

Bazurto-Velasco, Leonardo [4]
https://orcid.org/0009-0009-4842-1545
leonardo.bazurtovel@ug.edu.ec
Universidad de Guayaquil, Ecuador, Guayaquil.

Guim-Echeverria, Yuan [5]
https://orcid.org/0009-0008-4380-5737
yuan.guimech@ug.edu.ec
Universidad de Guayaquil, Ecuador, Guayaquil.

Autor de correspondencia [1]

DOI / URL: https://doi.org/10.55813/gaea/rcym/v4/n1/163

Resumen: La sofisticación de los modelos de lenguaje generativo plantea desafíos significativos para la detección automática de contenido generado artificialmente en contextos académicos y educativos. Esta investigación compara sistemáticamente modelos tradicionales (SVM, Random Forest, MLP, XGBoost, Voting Classifier) y arquitecturas Transformer (BERT, DeBERTa, RoBERTa) empleando características fraseológicas, sintácticas, semánticas, vectores de estilo y representaciones TF-IDF sobre el dataset PAN 2025 en inglés. Se evaluó el impacto del conjunto completo de características (186) versus selección óptima mediante feature selection (33 atributos). El Voting Classifier con feature selection alcanzó el mejor rendimiento (F1-score: 0.992101, accuracy: 0.991930), superando en 3.2 puntos porcentuales a DeBERTa (F1-score: 0.959864). Los resultados demuestran que la ingeniería de características combinada con ensambles tradicionales puede superar arquitecturas profundas manteniendo interpretabilidad y eficiencia, contribuyendo al desarrollo de herramientas robustas para la integridad académica y verificación textual automatizada.

Palabras clave: texto generado, características lingüísticas, TF-IDF, transformers, procesamiento de lenguaje natural.

Check for updates

Received: 15/Ene/2026
Accepted: 04/Feb/2026
Published: 28/Feb/2026

Cita: Espin-Riofrio, C., Mendoza-Morán, V., Vergara-Bello, O., Bazurto-Velasco, L., & Guim-Echeverria, Y. (2026). Modelos clásicos vs. Transformers en la detección de texto generado. *Revista Científica Ciencia Y Método*, *4*(1), 460-476. https://doi.org/10.55813/gaea/rcym/v4/n1/163

Revista Científica Ciencia y Método (RCyM)
https://revistacym.com
revistacym@editorialgrupo-aea.com
info@editoriagrupo-aea.com

Abstract:

The sophistication of generative language models poses significant challenges for automatic detection of artificially generated content in academic and educational contexts. This research systematically compares traditional models (SVM, Random Forest, MLP, XGBoost, Voting Classifier) and Transformer architectures (BERT, DeBERTa, RoBERTa) employing phraseological, syntactic, semantic features, style vectors, and TF-IDF representations on the PAN 2025 English dataset. The impact of the complete feature set (186) versus optimal selection through feature selection (33 attributes) was evaluated. The Voting Classifier with feature selection achieved the best performance (F1-score: 0.992101, accuracy: 0.991930), surpassing DeBERTa by 3.2 percentage points (F1-score: 0.959864). Results demonstrate that feature engineering combined with traditional ensembles can outperform deep architectures while maintaining interpretability and efficiency, contributing to the development of robust tools for academic integrity and automated text verification.

Keywords: generated text, linguistic features, TF-IDF, transformers, natural language processing.

1. Introducción

En los últimos años, el uso de modelos de lenguaje generativo como GPT ha crecido exponencialmente en ámbitos como la educación, la investigación, el periodismo y la producción de contenido digital. Si bien estas tecnologías aportan beneficios significativos, también plantean retos en cuanto a la autenticidad e integridad del contenido, especialmente cuando los textos generados artificialmente se presentan como escritos por humanos. En este contexto, la detección de texto generado por inteligencia artificial se ha consolidado como una línea de investigación clave dentro del procesamiento de lenguaje natural (PLN).

Las investigaciones previas han abordado este desafío desde múltiples perspectivas. Los enfoques basados en arquitecturas Transformer han demostrado alta efectividad: Prova (2024) alcanzó 93% de precisión con BERT, superando a XGB (84%) y SVM (81%); Wang et al. (2023) lograron 98% con RoBERTa en detección de noticias generadas por ChatGPT; y Preda et al. (2023) obtuvieron resultados competitivos en contextos multilingües mediante ensembles de Transformers con aprendizaje multitarea. Los trabajos de Bafna et al. (2024) y Espin-Riofrio, Charco, et al. (2024) exploraron arquitecturas híbridas combinando embeddings de BERT/RoBERTa con capas BiLSTM y optimización de hiperparámetros, aunque evidenciaron limitaciones en escenarios prácticos.

Paralelamente, diversos estudios han demostrado la efectividad de características lingüísticas explícitas. Espin-Riofrio, Ortiz-Zambrano, et al. (2024) integraron métricas

de perplejidad con rasgos estilométricos y clasificadores tradicionales, logrando resultados competitivos frente a modelos de deep learning. Shah et al. (2023) y Najjar et al. (2025) implementaron técnicas de IA explicable (XAI) mediante LIME y SHAP, alcanzando ~93% de precisión con transparencia interpretativa. Espin-Riofrio et al. (2023) demostraron la relevancia del filtrado léxico y la normalización gramatical en tareas de caracterización estilométrica.

Otros enfoques innovadores incluyen: DetectGPT (Mitchell et al., 2023), que identifica textos sintéticos mediante curvatura de probabilidad sin datos etiquetados; GLTR (Yan Wu & Segura-Bedmar, 2025), con bajo costo computacional y rendimiento competitivo; y métodos basados en coherencia factual mediante grafos de entidades (Zhong et al., 2020), superando a RoBERTa. Estudios en idiomas de bajos recursos (Sani et al., n.d.) y datasets especializados como CHEAT (Yu et al., 2023) y M4 (Y. Wang et al., 2023) han ampliado el alcance de la detección a contextos multilingües y heterogéneos.

La literatura también señala factores críticos: la longitud de secuencias influye en la precisión (Gaggar et al., 2023), los sistemas automáticos superan a evaluadores humanos especialmente con técnicas de muestreo engañosas (Ippolito et al., 2019), y la colaboración humana puede incrementar exactitud, aunque con desafíos operativos (Uchendu et al., 2023). Ardeshirifar (2025) concluye que los enfoques híbridos son óptimos, combinando la capacidad predictiva de Transformers con la interpretabilidad de características manuales.

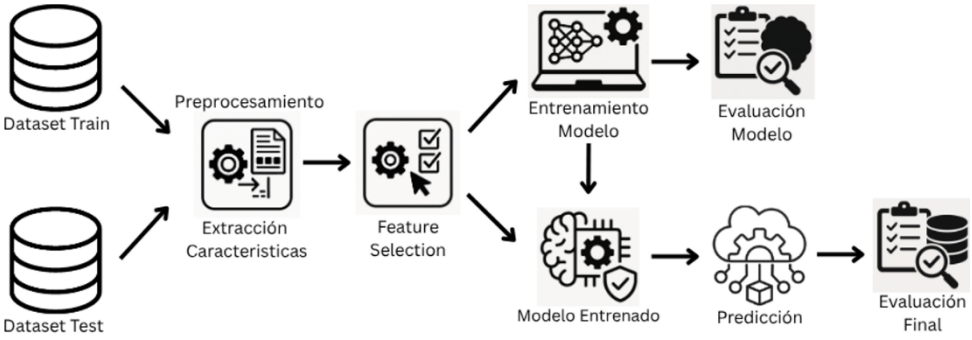
Pese a estos avances, persiste la necesidad de estrategias que integren sistemáticamente la potencia contextual de los modelos Transformer con la transparencia de las características lingüísticas. Este estudio plantea como hipótesis que dicha integración ofrece mejores resultados que el uso exclusivo de cualquier enfoque. El objetivo principal es comparar modelos tradicionales y arquitecturas Transformer, evaluando el impacto de características fraseológicas, sintácticas, semánticas, vectores de estilo y representaciones TF-IDF en la detección eficiente de textos generados automáticamente.

2. Materiales y métodos

Esta investigación evaluó comparativamente modelos tradicionales y arquitecturas Transformers en la detección de textos generados automáticamente mediante diferentes combinaciones de características lingüísticas. La Figura 1 presenta el esquema metodológico propuesto.

Figura 1

Esquema propuesto para esta investigación



Nota: Se muestra el proceso esquematizado de investigación (Autores, 2026).

Pipeline experimental: El enfoque de aprendizaje supervisado comprende: (1) preprocesamiento mediante limpieza léxica y normalización textual; (2) extracción de características fraseológicas, sintácticas y semánticas; (3) selección de atributos más relevantes mediante técnicas de reducción dimensional; y (4) división en conjuntos de entrenamiento, validación y prueba para optimización de hiperparámetros.

Enfoques experimentales: Se implementaron dos estrategias: (1) clasificadores tradicionales con el conjunto completo de características (186), y (2) clasificadores tradicionales y Transformers con características seleccionadas (33 atributos).

Evaluación: Ambos enfoques fueron evaluados mediante métricas estándar (accuracy, precision, recall, F1-score) y matrices de confusión para análisis exhaustivo del rendimiento predictivo**.**

Dataset

En el desarrollo de esta investigación se utilizó el dataset desarrollado por (Bevendorff et al., 2025) para la competición PAN 2025. El mismo fue elaborado con el objetivo de evaluar la capacidad de sistemas de detección de texto generado por humanos o por modelos de lenguaje.

El dataset, correspondiente a la Tarea 1 de la competencia, está completamente en idioma inglés y se organiza en dos subconjuntos: entrenamiento y prueba. Cada entrada en el dataset contiene un identificador único, el contenido textual, el modelo generador, la etiqueta binaria (0 para texto humano y 1 para texto generado) y el dominio de los textos.

En la Tabla 1 se muestra el número de instancias del dataset.

Tabla 1

Información del dataset.

Tipo	Entrenamiento	Validación
Generado	14606	2312
Humano	9101	1277
Total	23707	3589

Nota: Se muestra el proceso esquematizado de investigación (Autores, 2026).

Preprocesamiento

El preprocesamiento aplicó técnicas de limpieza y normalización textual para optimizar los datos previo al análisis y modelado. El proceso incluyó las siguientes etapas:

Normalización básica: Conversión a minúsculas para unificar el formato y evitar duplicidades por variaciones de capitalización.

Eliminación de ruido: Se removieron emojis, menciones de usuarios, hashtags, direcciones web, enlaces, números y caracteres no alfabéticos. Los signos de puntuación con valor sintáctico relevante fueron preservados. Se depuraron espacios en blanco redundantes, tabulaciones y saltos de línea excesivos.

Normalización lingüística: Se aplicó lematización mediante un modelo especializado para inglés, reduciendo cada palabra a su forma base mientras se preservaba su categoría gramatical. Finalmente, se eliminaron las palabras funcionales (stopwords) del idioma, ya que estas aportan valor informativo limitado para la tarea de clasificación.

Extracción de características

1) Fraseológicas:

Las características fraseológicas capturan hábitos estilísticos y patrones recurrentes de redacción mediante el análisis de aspectos superficiales del texto: frecuencia léxica, diversidad vocabular, estructuras repetitivas y secuencias de n-gramas (bigramas y trigramas).

Estas métricas resultan especialmente discriminativas en la detección de textos generados automáticamente, ya que revelan regularidades características de patrones algorítmicos frente a la variabilidad estilística humana. Los rasgos fraseológicos se centran en la forma estructural del lenguaje independientemente del contenido, proporcionando información complementaria crucial para la clasificación.

La Tabla 2 detalla las características fraseológicas extraídas durante el análisis.

Tabla 2
Características fraseológicas

Riqueza léxica	Proporción de palabras únicas
Longitud media de palabra	Promedio de caracteres por palabra
Proporción de stopwords	Porcentaje de palabras funcionales
Diversidad de n-gramas	Relación entre bigramas/trigramas únicos y total de palabras
Frecuencia de n-gramas	Promedio de aparición de combinaciones de bigramas y trigramas en el texto
Total de conectores	Cantidad absoluta de conectores discursivos presentes en el texto
Densidad de conectores	Proporción de conectores en relación con el número total de palabras
Conectores únicos	Número de conectores distintos que aparecen en el texto
Frecuencia de conectores	Frecuencia de repetición de los conectores

Nota: Lista de características fraseológicas implementadas (Autores, 2026).

2) Sintácticas:

Las características sintácticas evalúan la complejidad y variabilidad gramatical del texto, capturando diferencias estructurales entre la escritura humana y los textos generados automáticamente. Mientras la redacción humana exhibe mayor diversidad sintáctica y estructuras complejas, los sistemas de generación automática tienden a producir patrones repetitivos y construcciones simplificadas. La Tabla 3 detalla las características sintácticas extraídas.

Tabla 3
Características sintácticas

Longitud media de oración	Promedio de palabras por oración, refleja complejidad estructural
Complejidad sintáctica	Proporción de cláusulas subordinadas respecto al total de oraciones
Proporciones gramaticales	Distribución de categorías gramaticales como sustantivos, verbos, adjetivos, etc.
Conteo de puntuaciones	Cantidad total de signos de puntuación en el texto
Longitud promedio de oración	Longitud media de las oraciones en número de palabras
Número de tokens	Cantidad total de tokens en el texto.
Número de oraciones	Número total de oraciones presentes en el texto
Longitud promedio de palabras	Promedio de longitud de palabra en caracteres
Profundidad de dependencias	Profundidad máxima en los árboles de dependencias sintácticas
Longitud de dependencias	Distancia promedio entre palabras y sus dependencias en la estructura sintáctica
Longitud de frases	Promedio de palabras por frase en el texto

Nota: Lista de características sintácticas implementadas (Autores, 2026).

3) Semánticas:

Las características semánticas capturan el significado, coherencia y naturalidad del texto, permitiendo identificar incoherencias o desconexiones temáticas típicas de textos generados automáticamente. Se incluyen métricas de perplejidad (predictibilidad de secuencias mediante modelos de lenguaje), polaridad y subjetividad del sentimiento, y diversidad léxica. La Tabla 4 detalla las características semánticas extraídas.

Tabla 4
Características semánticas

Polaridad	Valor de polaridad del sentimiento general
Subjetividad	Nivel de subjetividad del texto
Polaridad VADER	Puntaje de sentimiento calculado mediante el analizador VADER
Conteo de vocales	Número total de vocales presentes en el texto

Nota: Lista de características semánticas implementadas (Autores, 2026).

4) Selección de Características:

El proceso de extracción generó un total de 186 características fraseológicas, sintácticas y semánticas. Sin embargo, la inclusión de atributos redundantes o poco informativos puede degradar el rendimiento y eficiencia del modelo, contrariamente a la intuición de que más características conducen a mejor desempeño.

Para mitigar este problema, se aplicó feature selection combinando Random Forest (estimación de importancia) y RidgeCV (refinamiento de la selección). Esta técnica redujo el conjunto inicial a 33 características significativas, optimizando el rendimiento

computacional y la capacidad de generalización. La Tabla 5 presenta las características seleccionadas.

Tabla 5

Características seleccionadas

1. Riqueza léxica	11. Longitud de frases
2. Longitud media de palabra	12. Longitud promedio de oración
3.Proporción de stopwords	13. Número de tokens
4. Longitud media de oración	14. Número de oraciones
5. Complejidad sintáctica	15. Total de conectores
6. Subjetividad	16. Densidad de conectores
7. Proporciones gramaticales	17. Profundidad de dependencias
8. Conectores	18. Longitud de dependencias
9. Puntuaciones	19. Frecuencia promedio de bigramas
10. Conteo de vocales	20. Frecuencia promedio de trigramas

Nota: Características de mayor peso luego de aplicar feature selection (Autores, 2026).

5) Embeddings de estilo:

Como parte del análisis, se incorporó el modelo preentrenado (StyleDistance, s. f.), el cual convierte los textos en representaciones vectoriales que capturan aspectos estilísticos y estructurales del lenguaje, obteniéndose un vector de embeddings de longitud 768, ejemplo [-0.63746583, 0.526328, 0.46823588, 0.21428992…]. Estas representaciones permiten comparar textos en términos de estilo, facilitando la detección de diferencias entre escritura humana y generada automáticamente.

6) TF-IDF de n-gramas:

Se aplicó TF-IDF sobre los textos preprocesados para generar representaciones vectoriales que capturan la importancia relativa de cada término, ponderando su frecuencia local en el documento contra su frecuencia global en el corpus. Se analizaron unigramas y bigramas para capturar información léxica y colocacional.

Para optimizar la relación información-eficiencia, se definió un rango incremental de características entre 500 y 3000, conservando únicamente los n-gramas más discriminativos y evitando la inclusión de términos poco informativos.

Entrenamiento

Una vez obtenidas todas las características del conjunto de datos, fue necesario definir los modelos con los que se realizaría la comparativa. A continuación, se hace un desglose de los enfoques seleccionados.

1) Modelos Tradicionales:

En base a experiencia en investigaciones previas, se seleccionaron algoritmos que han demostrado un alto rendimiento en tareas de detección de texto generado automáticamente. Además, se empleó la técnica de Voting Classifier, que permite combinar las predicciones de los mejores modelos base, con el objetivo de mejorar la precisión global del sistema.

En el contexto de este estudio, se emplearon los modelos de clasificación detallados en la Tabla 6.

Tabla 6

Modelos clasificadores utilizados y sus parámetros

Modelos de clasificación	Parámetros por Defecto
Support Vector Classifier (SVC)	dual='auto'
Support Vector Machine (SVM)	kernel='linear', probability=True
Random Forest (RF)	n_estimators=100, random_state=42
Multilayer Perceptron (MLP)	hidden_layer_sizes=(100), max_iter=500
XGBoost (XGB)	n_estimators=100

Nota: Parámetros implementados en los modelos clasificadores (Autores, 2026).

Posteriormente para el entrenamiento, se procedió con la concatenación de las características lingüísticas extraídas, las matrices generadas mediante la representación TF-IDF de n-gramas (unigramas y bigramas), y los vectores de estilo. Esta integración permitió construir una representación enriquecida y multidimensional de los textos, capturando tanto aspectos superficiales como estructurales y estilísticos.

Una vez construido el conjunto de características, se dividió en conjuntos de entrenamiento y prueba con una proporción 80/20. Antes del entrenamiento, se imputaron los valores faltantes mediante la media y se aplicó la normalización con MinMaxScaler para asegurar una escala uniforme entre todas las variables. Finalmente, se utilizó la técnica *smote* para balancear el conjunto de entrenamiento, generando instancias sintéticas de la clase minoritaria. Esto permitió reducir el sesgo del modelo hacia la clase mayoritaria y mejorar su capacidad de generalización.

2) Transformers

Estos modelos son ampliamente utilizados en tareas de procesamiento de lenguaje natural por su capacidad para capturar contextos complejos y relaciones semánticas profundas en los textos. Para esta investigación se seleccionaron variantes modernas y robustas de BERT y otros modelos multilingües, los cuales se listan en la Tabla 7.

Tabla 7

Modelos Transformer utilizados

Deberta
Bert Base Uncased
Roberta
Modern Bert

Nota: Se mencionan los modelos Transformes a entrenar (Autores, 2026).

Debido a que se realizó un proceso de fine-tuning y estos modelos requieren configuración específica, se estableció una versión base predeterminada para cada uno. Las configuraciones utilizadas se detallan en la Tabla 8.

Tabla 8

Hiperparámetros de fine-tuning

Parámetros	Valor
Batch size	16
Dropout	0.3
Función de activación interna	ReLU (Rectified Linear Unit)
Función de activación final	LogSoftmax
Optimizador	AdamW
Learning rate	2e-5

Nota: Valores de hiperparámetros implementados (Autores, 2026).

Los hiperparámetros utilizados en el proceso de entrenamiento fueron definidos manualmente como configuración por defecto para esta investigación. Para enriquecer la representación de los textos, se diseñó un enfoque que combina la salida contextual del token [CLS] del modelo transformador con un vector de características lingüísticas adicionales. Este vector incluye métricas textuales y una representación TF-IDF de 3000 dimensiones generada a partir de n-gramas (1,2).

La arquitectura del clasificador está compuesta por una capa densa que recibe como entrada la concatenación del vector [CLS] con las características lingüísticas. Esta capa es seguida por una activación ReLU, una operación de dropout para mitigar el sobreajuste y una segunda capa densa con activación LogSoftmax para producir las probabilidades de clase. Tanto los pesos del modelo base (transformador) como los del clasificador fueron ajustados durante el entrenamiento.

El modelo fue entrenado utilizando el optimizador AdamW, junto con un planificador de tasa de aprendizaje lineal. Como función de pérdida se empleó NLLLoss con ponderación de clases, para abordar el desbalance de clases en los datos. Además, se implementó una estrategia de early stopping basada en la pérdida de validación, con una paciencia de 3 épocas y un máximo de 100 ciclos de entrenamiento.

Predicción y evaluación

La evaluación del desempeño de modelos tradicionales y Transformers se realizó sobre el conjunto de prueba utilizando métricas estándar para clasificación binaria: accuracy, precision, recall y F1-score, que cuantifican desde la proporción global de aciertos hasta el equilibrio entre falsos positivos y negativos.

Complementariamente, se generaron reportes de clasificación detallados por clase y matrices de confusión para visualizar gráficamente los patrones de aciertos y errores, facilitando la interpretación de fortalezas y limitaciones de cada modelo.

3. Resultados

Es Una vez completado el proceso de entrenamiento y de evaluación, se procede a profundizar en el rendimiento de ambos enfoques.

3.1. Modelos tradicionales

A continuación, se presentan la Tabla 9 y Tabla 10 que muestran los resultados obtenidos de los modelos clasificadores con la experimentación previa con todas las características y la siguiente utilizando las de mayor peso obtenidas con feature selection.

Tabla 9
Evaluación con todas las características

Modelo	Accuracy	Precision	Recall	F1-Score	Max Features
VC	0.992756	0.992099	0.992099	0.992099	1000
VC	0.988298	0.985058	0.989690	0.987298	500
SVC	0.988019	0.988763	0.985092	0.986883	3000
SVM	0.984675	0.984208	0.982321	0.983253	1000
SVM	0.983840	0.978769	0.986756	0.982523	500

Nota: Resultados de predicción de modelos clasificadores (Autores, 2026).

En el entrenamiento con todas las características, el modelo con mejor rendimiento fue el Voting Classifier al utilizar el TFIDF de 1000 n-gramas alcanzando un F1-Score de 0.992099.

Tabla 10
Evaluación con feature selection

Modelo	Accuracy	Precision	Recall	F1-Sore	Max Features
VC	0.992756	0.991930	0.992274	0.992101	500
VC	0.992198	0.991491	0.991491	0.991491	1000
VC	0.991920	0.991102	0.991274	0.991188	3000
SVM	0.991362	0.989515	0.991718	0.990600	500
SVC	0.990527	0.989667	0.989667	0.989667	3000

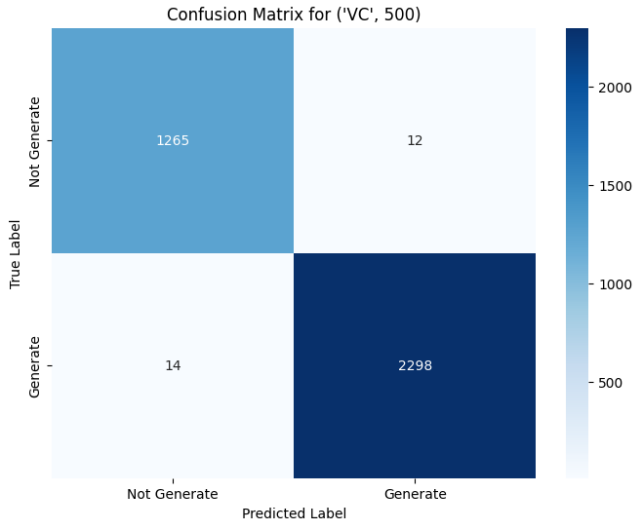
Nota: Resultados de predicción aplicando feature selection (Autores, 2026).

Como se puede observar, el modelo con mejor desempeño fue el Voting Classifier utilizando TFIDF de 500 n-gramas. La aplicación de feature selection permitió una leve mejora en el rendimiento, alcanzando un F1-score de 0.992101, ligeramente superior al obtenido sin selección de características (F1-score de 0.992099).

En la Figura 2 se muestra la matriz de confusión correspondiente al mejor modelo Voting Classifier, donde se observa la distribución de positivos y falsos en las predicciones realizadas.

Figura 2

Matriz de confusión de la predicción



Nota: Resultados de predicción representados en matriz de confusión (Autores, 2026).

El modelo presenta resultados muy satisfactorios, los valores erróneos son mínimos (26 errores de 3589 ejemplos), lo que demuestra una alta capacidad de detección tanto para textos generados como para no generados.

Para ampliar a mayor detalle se presenta la Tabla 11, con el reporte de clasificación.

Tabla 11

Reporte de clasificación del modelo VC con max features de 500

	Precision	Recall	F1-score	Support
0	0,989054	0,990603	0,989828	1277
1	0,994805	0,993945	0,994375	2312
accuracy			0,992756	3589
macro avg	0,991930	0,992274	0,992101	3589
weighted avg	0,992759	0,992756	0,992757	3589

Nota: Resultados de predicción conjunto de prueba (Autores, 2026).

El modelo alcanzó un F1-Score de 0.989828 para la clase 0 y 0.994375 para la clase 1. El promedio macro del F1-Score fue 0.9921 y el promedio ponderado, 0.992757, sobre un total de 3589 ejemplos.

3.2. Modelos Transformers

A continuación, se presenta la Tabla 12 con los resultados obtenidos tras el proceso de evaluación.

Tabla 12

Métricas de evaluación de modelos Transformers

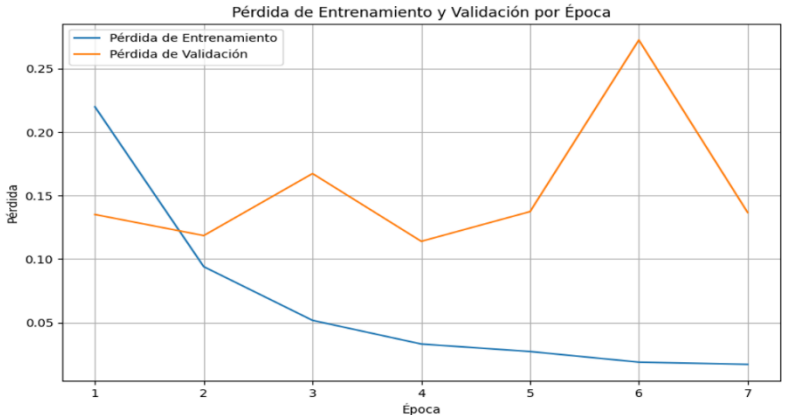
Modelo	Epoch	Accuracy	Precision	Recall	F1-Macro
Deberta	4	0.962942	0.956868	0.963174	0.959864
Bert Base Uncased	1	0.958763	0.953025	0.957652	0.955254
Roberta	5	0.954305	0.945639	0.957171	0.950823
Modern Bert	2	0.944274	0.936731	0.942900	0.939655
Bert Multi	2	0.939537	0.931005	0.938873	0.934666

Nota: Mejores resultados de predicción con los modelos Transformers (Autores, 2026).

El modelo DeBERTa obtuvo los mejores resultados entre los modelos evaluados, logrando un F1-Score de 0.959864. Por otro lado, el BERT MULTI presentó el rendimiento más bajo con un F1-score de 0.934666.

Para profundizar en los resultados del modelo DeBERTa, se presenta la Figura 3 de la curva de pérdida del modelo durante el entrenamiento y evaluación a lo largo de las épocas.

Figura 3
Curva de pérdida de entrenamiento y validación

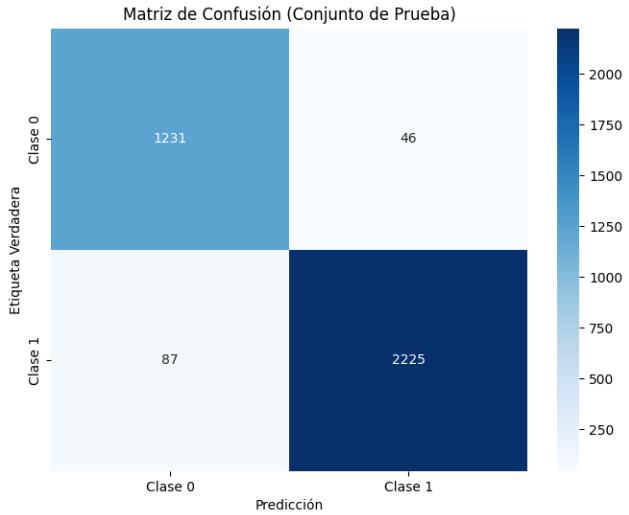


Nota: Curvas de pérdidas de entrenamiento y evaluación (Autores, 2026).

Se observa que la pérdida de entrenamiento disminuye de manera constante a lo largo de las épocas. Sin embargo, a partir de la tercera época, la pérdida de validación comienza a mostrar fluctuaciones hacia arriba.

En la Figura 4 se muestra la matriz de confusión, lo que permite visualizar el desempeño del modelo en términos de verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos.

Figura 4
Matriz de confusión de DeBERTa



Nota: Representación de predicción mediante matriz de confusión (Autores, 2026).

La matriz de confusión confirma los excelentes resultados obtenidos, donde comete muy pocos errores, tan solo 133 de 3589 ejemplos. Pero si presenta un leve sesgo a producir falsos negativos.

Para complementar estos resultados, a continuación, se presenta la Tabla 13 de reporte de clasificación.

Tabla 13
Reporte de clasificación DeBERTa

	Precision	Recall	F1-score	Support
0	0,933991	0,963978	0,948748	1277
1	0,979745	0,962370	0,970980	2312
accuracy			0,962942	3589
macro avg	0,956868	0,963174	0,959864	3589
weighted avg	0,963465	0,962942	0,963069	3589

Nota: Predicción sobre el conjunto de prueba utilizando DeBERTa (Autores, 2026).

El modelo obtuvo un F1-Score del 0.959864, donde la clase 1 presento mejores métricas, probablemente debido a su mayor representación en el dataset.

3.3. Comparativa entre modelos tradicionales y Transformers

Por último, con el objetivo de comparar el desempeño de los dos mejores modelos obtenidos, se presenta la Tabla 14, la cual resume los resultados alcanzados por ambos enfoques. Esta comparación permite identificar cuál ofrece un rendimiento superior en la tarea de detección de texto generado.

Tabla 14
Comparativas de resultados entre métodos

Modelo	Accuracy	Precision	Recall	F1-Macro
DeBERTa	0.962942	0.956878	0.963174	0.959864
Voting Classifier	0.992756	0.991930	0.992274	0.992101

Nota: Métricas comparativas de métodos tradicionales y Transformers (Autores, 2026).

Como se puede observar, ambos enfoques presentan resultados sobresalientes a pesar de sus diferencias arquitectónicas. No obstante, los modelos tradicionales, en particular el Voting Classifier utilizando 500 n-gramas, demostraron una ventaja consistente en términos de rendimiento.

4. Discusión

El análisis comparativo revela hallazgos significativos sobre la efectividad de diferentes enfoques metodológicos en la detección de textos generados automáticamente.

Impacto de feature selection: La reducción dimensional mediante selección de características no solo mantiene el rendimiento, sino que lo optimiza. El Voting Classifier mejoró marginalmente su F1-score de 0.992099 (1000 n-gramas, sin selección) a 0.992101 (500 n-gramas, con selección), evidenciando mayor eficiencia computacional sin sacrificar precisión. La matriz de confusión confirma un rendimiento

sobresaliente con errores mínimos y predicciones balanceadas entre ambas clases, demostrando alta capacidad discriminativa sin sesgos significativos.

Rendimiento de Transformers: Entre los modelos basados en arquitecturas profundas, DeBERTa alcanzó el mejor desempeño con F1-score macro de 0.959864. La curva de pérdida muestra convergencia óptima en la época 4, aunque la matriz de confusión revela un leve sesgo hacia falsos negativos (textos generados clasificados como humanos).

Comparación general: Ambos enfoques exhiben alta efectividad en la tarea, sin embargo, el Voting Classifier con feature selection supera a los Transformers en 3.2 puntos porcentuales de F1-score (0.992101 vs 0.959864), ofreciendo además ventajas sustanciales en eficiencia computacional y requerimientos de recursos. Estos resultados demuestran que la ingeniería de características combinada con ensambles de clasificadores tradicionales puede superar a arquitecturas profundas en escenarios donde la interpretabilidad y eficiencia son prioritarias.

5. Conclusiones

Este estudio demuestra que la integración multidimensional de características fraseológicas, sintácticas, semánticas, TF-IDF y embeddings estilísticos resulta esencial para la detección efectiva de textos generados automáticamente.

El Voting Classifier alcanzó el mejor rendimiento (F1-score: 0.992101), superando en 3.2 puntos porcentuales a DeBERTa (0.959864), demostrando que la ingeniería de características combinada con ensambles tradicionales puede superar arquitecturas profundas manteniendo interpretabilidad y eficiencia computacional. La reducción de 186 características iniciales a 33 atributos clave mediante feature selection optimizó simultáneamente rendimiento y eficiencia, validando que la calidad supera la cantidad en la representación de datos.

Los Transformers evidencian capacidad para capturar relaciones contextuales profundas, pero implican costos computacionales sustancialmente mayores sin ventajas significativas en precisión para esta tarea. Como trabajo futuro, se propone explorar arquitecturas híbridas que integren la potencia contextual de Transformers con la eficiencia de modelos clásicos, así como extender el sistema a contextos multilingües. En síntesis, la selección inteligente de características y arquitecturas apropiadas optimiza la detección de textos generados, evitando complejidad innecesaria y asegurando viabilidad en aplicaciones reales.

CONFLICTO DE INTERESES

“Los autores declaran no tener ningún conflicto de intereses”.

Referencias Bibliográficas

Ardeshirifar, R. (2025). Comparing hand-crafted and deep learning approaches for detecting AI-generated text: Performance, generalization, and linguistic insights. *AI and Ethics, 5*, 4197–4209. https://doi.org/10.1007/s43681-025-00699-4

Bafna, J., Mittal, H., Sethia, S., Shrivastava, M., & Mamidi, R. (2024). Mast Kalandar at SemEval-2024 Task 8: On the trail of textual origins: RoBERTa-BiLSTM approach to detect AI-generated text. In A. Kr. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, & A. Rosá (Eds.), *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)* (pp. 1627–1633). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.semeval-1.231

Bevendorff, J., Wiegmann, M., Potthast, M., & Stein, B. (2025). *PAN'25/26 generative AI detection: Voight-Kampff AI detection sensitivity* (Version v1) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.14962653

Espin-Riofrio, C., Charco, J. L., Preciado-Maila, D. K., Ramos-Ramírez, L., Camacho-Villalva, H., & Montejo-Ráez, A. (2024). Embeddings of initial tokens from BERT-based models to identify human-written or automatically generated text. In M. M. Larrondo Petrie, J. Texier, & R. A. Rivas Matta (Eds.), *Sustainable engineering for a diverse, equitable, and inclusive future at the service of education, research, and industry for a society 5.0.: Proceedings of the 22nd LACCEI International Multi-Conference for Engineering, Education and Technology (LACCEI 2024)*. Fundacion LACCEI. https://doi.org/10.18687/LACCEI2024.1.1.108

Espin-Riofrio, C., Ortiz-Zambrano, J., & Montejo-Ráez, A. (2023). An approach to lexicon filtering for author profiling. *Procesamiento del Lenguaje Natural, 71*, 75–86. https://doi.org/10.26342/2023-71-6

Espin-Riofrio, C., Ortiz-Zambrano, J., & Montejo-Ráez, A. (2024). SINAI at IberAuTexTification in IberLEF 2024: Perplexity metrics and text features for classifying automatically generated text. In S. M. Jiménez-Zafra, L. Chiruzzo, F. Rangel, F. Balouchzahi, U. B. Corrêa, A. Bonet Jover, H. Gómez-Adorno, J. Á. González Barba, D. I. Hernández Farías, A. Montejo Ráez, P. Moral, C. Rodríguez Abellán, M. E. Vallecillo Rodríguez, M. Taulé, & R. Valencia-García (Eds.), *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2024) co-located with the Conference of the Spanish Society for Natural Language Processing (SEPLN 2024)* (*CEUR Workshop Proceedings*, Vol. 3756). CEUR-WS.org. https://ceur-ws.org/Vol-3756/IberAuTexTification2024_paper1.pdf

Gaggar, R., Bhagchandani, A., & Oza, H. (2023). Machine-generated text detection using deep learning [Preprint]. *arXiv*. https://doi.org/10.48550/arXiv.2311.15425

Guggilla, C., Roy, B., Chavan, T. R., Rahman, A., & Bowen, E. (2025). AI generated text detection using instruction fine-tuned large language and transformer-based models [Preprint]. *arXiv*. https://doi.org/10.48550/arXiv.2507.05157

Ippolito, D., Duckworth, D., Callison-Burch, C., & Eck, D. (2020). Automatic detection of generated text is easiest when humans are fooled. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 1808–1822). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.164

Mitchell, E., Lee, Y., Khazatsky, A., Manning, C. D., & Finn, C. (2023). DetectGPT: Zero-shot machine-generated text detection using probability curvature. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *Proceedings of the 40th International Conference on Machine Learning* (*Proceedings of Machine Learning Research*, Vol. 202, pp. 24950–24962). PMLR. https://proceedings.mlr.press/v202/mitchell23a.html

Najjar, A. A., Ashqar, H. I., Darwish, O. A., & Hammad, E. (2025). Detecting AI-generated text in educational content: Leveraging machine learning and explainable AI for academic integrity [Preprint]. *arXiv*. https://doi.org/10.48550/arXiv.2501.03203

Preda, A.-A., Cercel, D.-C., Rebedea, T., & Chiru, C.-G. (2023). UPB at IberLEF-2023 AuTexTification: Detection of machine-generated text using transformer ensembles. In M. Montes-y-Gómez, F. Rangel, S. M. Jiménez-Zafra, M. Casavantes, B. Altuna, M. Á. Álvarez-Carmona, G. Bel-Enguix, L. Chiruzzo, I. de la Iglesia, H. J. Escalante, M. Á. García-Cumbreras, J. A. García-Díaz, J. Á. González Barba, R. Labadie Tamayo, S. Lima, P. Moral, & F. M. Plaza del Arco (Eds.), *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023) co-located with the Conference of the Spanish Society for Natural Language Processing (SEPLN 2023)* (CEUR Workshop Proceedings, Vol. 3496). CEUR-WS.org. https://ceur-ws.org/Vol-3496/autextification-paper19.pdf

Prova, N. (2024). Detecting AI generated text based on NLP and machine learning approaches [Preprint]. *arXiv*. https://doi.org/10.48550/arXiv.2404.10032

Sani, B., Soy, A., Hafiz Imam, S., Mustapha, A., Aliyu, L. J., Abdulmumin, I., Ahmad, I. S., & Muhammad, S. H. (n.d.). Who Wrote This? Identifying Machine vs Human-Generated Text in Hausa. Retrieved July 8, 2025, from https://github.com/TheBangis/hausa_corpus

Shah, A., Ranka, P., Dedhia, U., Prasad, S., Muni, S., & Bhowmick, K. (2023). Detecting and unmasking AI-generated texts through explainable artificial intelligence using stylistic features. *International Journal of Advanced Computer Science and Applications, 14*(10). https://doi.org/10.14569/IJACSA.2023.01410110

StyleDistance. (s. f.). *styledistance* [Model]. Hugging Face. Recuperado el 7 de julio de 2025, de https://huggingface.co/StyleDistance/styledistance

Uchendu, A., Lee, J., Shen, H., Le, T., Huang, T.-H. K., & Lee, D. (2023). Does human collaboration enhance the accuracy of identifying LLM-generated deepfake texts? *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing, 11*(1), 163–174. https://doi.org/10.1609/hcomp.v11i1.27557

Wang, Y., Mansurov, J., Ivanov, P., Su, J., Shelmanov, A., Tsvigun, A., Whitehouse, C., Mohammed Afzal, O., Mahmoud, T., Sasaki, T., Arnold, T., Aji, A. F., Habash, N., Gurevych, I., & Nakov, P. (2024). M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection. In Y. Graham & M. Purver (Eds.), *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1369–1407). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.eacl-long.83

Wang, Z., Cheng, J., Cui, C., & Yu, C. (2023). Implementing BERT and fine-tuned RobertA to detect AI generated news by ChatGPT [Preprint]. *arXiv*. https://doi.org/10.48550/arXiv.2306.07401

Yan Wu, L., & Segura-Bedmar, I. (2025). AI-generated text detection with a GLTR-based approach [Preprint]. *arXiv*. https://doi.org/10.48550/arXiv.2502.12064

Yu, P., Chen, J., Feng, X., & Xia, Z. (2025). CHEAT: A large-scale dataset for detecting ChatGPT-written abstracts. *IEEE Transactions on Big Data, 11*(3), 898–906. https://doi.org/10.1109/TBDATA.2025.3536929

Zhong, W., Tang, D., Xu, Z., Wang, R., Duan, N., Zhou, M., Wang, J., & Yin, J. (2020). Neural deepfake detection with factual structure of text. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 2461–2470). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.193