Artículo Científico / Scientific Paper

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR
INGENIUS
Revista de Ciencia y Tecnología

pISSN: 1390-650X / eISSN: 1390-860X

Mejora de la precisión en la detección de noticias falsas de política en español: un estudio sobre algoritmos de optimización aplicados a la regresión logística

Enhancing Precision in the Detection of Spanish Political Fake News: A Study on Optimization Algorithms Applied to Logistic Regression

Santiago Tene-Castillo [1], Luis Chamba-Eras [2,*]

Resumen

En la era digital, la propagación de noticias falsas en el ámbito político ha representado un desafío para la opinión pública y los procesos democráticos. La facilidad con que circula información no verificada ha impulsado la necesidad de desarrollar herramientas eficientes que permitan identificar y mitigar la desinformación. Este trabajo se centró en mejorar la detección de noticias falsas de política en español utilizando un modelo de regresión logística. El objetivo fue optimizar dicho modelo mediante algoritmos basados en gradiente (GD, SGD, MBGD, AdaGrad, Adam y RMSProp) para elevar la precisión en la clasificación de noticias verdaderas y falsas. Para abordar el problema, se creó un nuevo conjunto de datos que integró registros de noticias de política en español de Kaggle y noticias de periódicos ecuatorianos, alcanzando un total de 82,355 registros. Se aplicaron técnicas de limpieza, normalización y balanceo de clases para obtener un dataset de calidad. Posteriormente, se entrenó un modelo de regresión logística con diferentes optimizadores. Los experimentos revelaron que el gradiente descendente estocástico (SGD) alcanzó la mayor precisión con un 80%, superando a la regresión logística sin optimización, que registró un 73.7%. Estos resultados sugieren que la optimización de la regresión logística contribuye significativamente a la detección de noticias falsas, proporcionando un modelo optimizado para ayudar a mitigar la desinformación en el ámbito político. Además, este enfoque ha abierto la puerta a futuras investigaciones que incluyan un mayor volumen de datos y metodologías más avanzadas para mejorar la precisión y robustez de los modelos.

Palabras clave: Aprendizaje automático, optimización de hiperparámetros, algoritmos de clasificación, detección de texto, noticias falsas.

Abstract

In the digital era, the spread of fake news in the political sphere has posed a challenge to public opinion and democratic processes. The ease with which unverified information circulates has driven the need to develop efficient tools to identify and mitigate misinformation. This work focused on improving the detection of political fake news in Spanish using logistic regression model. The aim was to optimize this model with gradient-based algorithms (GD, SGD, MBGD, AdaGrad, Adam, and RMSProp) to increase precision in classifying true and false news. To address this problem, a new dataset was created by integrating Spanish-language political news records from Kaggle and news from Ecuadorian newspapers, resulting in a total of 82,355 records. Data cleaning, normalization, and class balancing techniques were applied to obtain a high-quality dataset. Subsequently, a logistic regression model was trained using different optimizers. The experiments revealed that stochastic gradient descent (SGD) achieved the highest precision at 80%, outperforming logistic regression without optimization, which recorded 73.7%. These results suggest that optimizing logistic regression significantly contributes to fake news detection by providing an enhanced model that helps mitigate misinformation in the political domain. Moreover, this approach has opened the door to future research involving larger datasets and more advanced methodologies to further improve the precision and robustness of the models.

Keywords: Machine learning, Hyperparameter optimization, Classification algorithms, Text detection, Fake news, Machine learning algorithms.

[1] Carrera de Computación, Universidad Nacional de Loja, Loja, Ecuador, email: santiago.tene@unl.edu.ec
[2,*] Carrera de Computación, GITIC, Universidad Nacional de Loja, Loja, Ecuador. Autor para correspondencia: lachamba@unl.edu.ec

1. Introducción

En la era digital, la proliferación de noticias falsas ha generado un reto significativo en el ámbito político, donde su rápida difusión puede alterar la percepción pública y afectar la estabilidad de los procesos democráticos. Este fenómeno se ha intensificado con el auge de plataformas digitales, las cuales facilitan la propagación de información sin verificación. Ante este desafío, se requiere el desarrollo de modelos de detección más precisos para mitigar la desinformación [1] [2].

Estudios recientes han explorado diversos enfoques basados en machine learning (ML) para detectar noticias falsas. Por ejemplo, en [3] emplearon modelos preentrenados como BERT y ELMo para clasificar noticias falsas en español, logrando una precisión máxima del 80%. Sin embargo, se identificaron limitaciones relacionadas con la incapacidad de estos modelos para abordar las especificidades lingüísticas del español en contextos locales, como el ecuatoriano. Por otro lado, en [4] demostraron que algoritmos de optimización como Gradiente Descendente Estocástico (SGD), combinados con representaciones basadas en TF-IDF y unigramas, alcanzaron una precisión del 94.2%, destacando la relevancia de las técnicas de optimización para mejorar los modelos de clasificación.

En [5] se han implementado variantes del Gradiente Descendente (GD), SGD y Adam, para optimizar modelos de Regresión Logística (RL), logrando incrementos significativos en precisión, desde un 87% hasta más del 93%, dependiendo del conjunto de datos. Además, en investigaciones aplicadas a idiomas como el árabe han demostrado la adaptabilidad de estos métodos en contextos multilingües, logrando precisiones superiores al 87%.

De manera similar, en [6] aplicaron SGD en redes neuronales recurrentes (RNN), mejorando la precisión de clasificación del 97% al 98% mediante procesos de optimización. Estos resultados destacan la eficacia de las técnicas de optimización incluso en escenarios complejos. Por su parte, en [7] integraron Gradiente Descendente Gaussiano con RL alcanzaron precisiones superiores al 94%, subrayando la importancia de una configuración precisa de hiperparámetros y la optimización continua para maximizar el rendimiento de los modelos.

El estudio [8] aborda la detección de spam en redes sociales y evalúa la precisión de modelos de RL con y sin GD. Los resultados muestran que, mientras el modelo sin optimización alcanzó una precisión del 80%, la inclusión de GD incrementó este valor al 95% en la detección de perfiles falsos. Este hallazgo refuerza la idea de que, aunque la RL puede ser limitada por sí sola, la incorporación de algoritmos de optimización mejora considerablemente su rendimiento, especialmente al reducir las funciones de costo.

En [9] se abordó la detección de noticias falsas mediante algoritmos de optimización como SGD, mejorando la eficacia de los modelos en tareas de clasificación y detección. Para ello, se utilizó el conjunto de datos AFND (Arabic Fake News Detection). Luego de la optimización, los resultados registraron una precisión del 87% en la detección. Dichos modelos pudieron integrarse en plataformas de redes sociales para disminuir la propagación de noticias falsas, contribuyendo a una mejora en la calidad de la información que circula en línea.

En [10] implementaron un modelo de RL para clasificar artículos o declaraciones de noticias como genuinas o falsas, utilizando la técnica de optimización GD para minimizar la función de pérdida. Sus resultados resaltaron la importancia de optimizar la RL, ya que esto contribuyó de forma notable a reducir el error en las predicciones .

Por lo tanto, en la detección de noticias falsas en español, se emplean técnicas de procesamiento de texto y ML. Ajustar los modelos con algoritmos de optimización Adam demuestra un aumento de la precisión, lo que resalta la importancia de la optimización en los modelos de clasificación de ML [11].

La regresión logística es un algoritmo de machine learning con un alto potencial para la detección de noticias falsas en el ámbito político en español. Sin embargo, su aplicación en este campo aún no ha sido suficientemente optimizada. La limitada experimentación con algoritmos de optimización como GD, SGD, Gradiente Descendente por Mini-Lote (MBGD), Algoritmo de Gradiente Adaptativo (AdaGrad), Estimación de Momento Adaptativo (Adam) y Propagación de la Raíz del Promedio Cuadrático (RMSProp) dificulta evaluar con precisión su impacto en el rendimiento del clasificador. Esta falta de optimización, debido a la escasez de pruebas sistemáticas, ha resultado en una precisión limitada del modelo.

El principal objetivo de esta investigación es entrenar y evaluar un modelo de RL optimizado mediante los algoritmos GD, SGD, MBGD, AdaGrad, Adam y RMSProp, con el fin de clasificar noticias falsas de contenido político en español. El estudio se enfoca en noticias publicadas entre 2018 y 2024, con el propósito de desarrollar un modelo optimizado que sea aplicable tanto en el ámbito académico como en entornos prácticos para mitigar la desinformación. El dataset empleado en esta investigación consta de 74,276 registros equilibrados entre noticias verdaderas y falsas, obtenidos tras un proceso de integración de fuentes, limpieza, estandarización y preprocesamiento

de datos.

Este trabajo ofrece importantes contribuciones en el ámbito de la optimización en ML:

1. Desarrollo de un dataset especializado: La integración de dos conjuntos de datos en español, complementados con información recopilada de medios ecuatorianos, proporciona una base sólida para el análisis y entrenamiento del modelo [12].

2. Optimización de modelos: La implementación de algoritmos como GD, SGD, y Adam ha permitido mejorar significativamente la precisión del modelo de RL, alcanzando valores del 82.74%, 92.79%, 79.24% respectivamente.

3. Reducción de la propagación de noticias falsas: El modelo optimizado desarrollado en este trabajo contribuye a mitigar la difusión de desinformación, sirviendo como base para el desarrollo de herramientas destinadas a combatir la propagación de noticias falsas en medios digitales.

La estructura del trabajo es la siguiente: la sección 2 aborda los desafíos que representa la desinformación en el ámbito político, su impacto en la opinión pública y la necesidad de aplicar herramientas de detección de noticias falsas basadas en modelos optimizados de ML. La sección 3 describe el procedimiento seguido, incluyendo la personalización del dataset utilizado para el entrenamiento del modelo, la ingeniería de datos y modelos, así como el ajuste de hiperparámetros en los métodos de optimización. En la sección 4, se comparan los resultados obtenidos con el estado actual de la clasificación de noticias falsas a partir de trabajos relacionados. Además, se analizan las limitaciones del estudio y se proponen líneas de investigación futuras. Finalmente, la sección de 5 resume los hallazgos más relevantes de este trabajo.

2. Antecedentes

La propagación de noticias falsas en el ámbito político ha sido ampliamente estudiada debido a su impacto en la opinión pública y los procesos electorales. Investigaciones previas han demostrado que la desinformación se propaga más rápido que las noticias verificadas, lo que ha motivado el desarrollo de modelos de ML optimizados, capaces de mejorar y automatizar su detección. En esta sección, se presenta una revisión del impacto de la desinformación, los métodos de clasificación utilizados previamente y la relevancia de la optimización en la detección automatizada de noticias falsas [13] [1] [14] [2].

2.1. Propagación de Noticias Falsas y su Impacto

La rápida difusión de noticias falsas en plataformas digitales y redes sociales ha sido ampliamente documentada en estudios recientes. Factores como la facilidad de acceso a la información, la velocidad de propagación y la dificultad de verificación han incrementado el alcance de la desinformación. Investigaciones previas han demostrado que la propagación de noticias falsas en redes sociales ocurre más rápido que la de noticias verificadas, lo que resalta la urgencia de desarrollar modelos efectivos para su detección [8]. Además, en el contexto político, la desinformación puede influir en la percepción de los votantes y en la estabilidad de los procesos electorales [3].

2.2. Algoritmos de Machine Learning en la Detección de Noticias Falsas

Los enfoques tradicionales para la detección de noticias falsas han utilizado modelos basados en reglas y análisis manual, pero estos métodos han resultado ineficaces debido a la gran cantidad de datos generados diariamente. En consecuencia, se han implementado técnicas de ML para abordar este problema. Modelos como la RL, Máquinas de Soporte Vectorial (SVM) y Redes Neuronales han demostrado su utilidad en la clasificación de noticias falsas [4]. Investigaciones recientes han explorado la aplicación de modelos preentrenados como BERT y ELMo, logrando mejoras en la detección de noticias falsas en distintos idiomas, incluidos el inglés y el español [5].

Sin embargo, estos enfoques presentan limitaciones cuando se aplican a contextos específicos, como el ecuatoriano, donde la estructura lingüística y los sesgos informativos pueden afectar la precisión de los modelos. Además, la escasez de datasets balanceados en español ha dificultado la generalización de estos modelos a escenarios locales [11].

2.3. Optimización en Modelos de Clasificación de Texto

La optimización de modelos de ML ha demostrado ser un factor determinante en la mejora de la precisión y eficiencia de los clasificadores. Técnicas de optimización como GD y sus variantes, incluyendo SGD, MBGD, AdaGrad, Adam y RMSProp, han sido ampliamente utilizadas para ajustar los parámetros de los modelos y reducir el error de clasificación [15].

Estudios previos han demostrado que el uso de optimizadores adecuados puede incrementar significativamente la precisión de los modelos de detección de

noticias falsas. Por ejemplo, investigaciones han reportado mejoras del 5% al 10% en precisión al comparar modelos de RL optimizados con Adam y SGD frente a aquellos sin optimización [9]. Estos hallazgos subrayan la importancia de la elección del optimizador en tareas de clasificación de texto y detección de desinformación.

Uno de los principales desafíos en la detección de noticias falsas en español ha sido la escasez de conjuntos de datos especializados en desinformación política. Recientemente, se ha desarrollado el Spanish Political Fake News Dataset [16], un corpus de 57,231 noticias recopiladas a través de web scraping y generación sintética, con el objetivo de mejorar la precisión de modelos de detección automatizada. Este dataset ha sido utilizado para entrenar modelos basados en Transformers, como BERT y RoBERTa, obteniendo tasas de precisión superiores al 98% en datos sintéticos, aunque con una disminución en desempeño al aplicarse a noticias reales no curadas. Estos hallazgos subrayan la importancia de contar con datos específicos y representativos para el entrenamiento de modelos de ML en la detección de desinformación política. Este dataset se toma como base para la construcción de un corpus personalizado, ajustado al contexto de esta investigación.

3. Materiales y métodos

Se usó la metodología CRISP-ML, la cual propone un proceso iterativo para guiar proyectos de ML, abarcando desde la definición del problema y la exploración de los datos hasta la selección, optimización y despliegue de modelos. Cada etapa establece objetivos claros, lo que permite abordar la complejidad y variedad de la información y asegurar resultados confiables. Se destacan tres fases esenciales: la ingeniería de datos, que reúne y transforma la información; la ingeniería de modelos, enfocada en el diseño, entrenamiento y ajuste de algoritmos; y la evaluación, que utiliza métricas e indicadores para medir la eficacia y solidez de los enfoques implementados [17]. Esta metodología garantiza la trazabilidad de los resultados y proporciona un marco replicable para futuras investigaciones en la detección de desinformación.

3.1. Fase 1: Ingeniería de datos

Se llevaron a cabo cinco actividades para la generación del dataset como para su procesamiento.

(a) Selección de datos (personalización del dataset): Se creó un dataset al integrar información de tres fuentes distintas. Dos de esos conjuntos, Spanish Political Fake News [16] y Spanish Political News Dataset, se obtuvieron de Kaggle con acceso libre, mientras que el tercero se generó mediante técnicas de web scraping. Como resultado, se obtuvo un total de 63,602 registros que incluyen noticias tanto verdaderas como falsas.

(b) Limpieza de datos: Consistió en la eliminación de valores nulos, registros vacíos y valores duplicados.

(c) Normalización de datos: Se aplicaron diversas técnicas para garantizar la estandarización de las noticias destinadas al procesamiento por el modelo. Estas técnicas incluyeron la conversión de texto a minúsculas, la eliminación de caracteres especiales y la tokenización.

(d) Equilibrio de clases: Se realizó un balanceo en la clase Label, que es binaria y está compuesta por los valores 0 (para noticias falsas) y 1 (para noticias verdaderas). Este procedimiento se realizó con el objetivo de evitar sesgos en los datos, ajustando el número de registros a 23,847 para cada clase.

(e) División de datos: Se realizó la partición del conjunto en un 70% para entrenamiento, 15% para validación y 15% para evaluación. Este procedimiento se realizó utilizando la biblioteca scikit-learn.

3.2. Fase 2: Ingeniería de modelos

Durante esta fase, se entrenó el modelo en condiciones similares para todos los métodos de optimización. Se ajustaron los hiperparámetros con el objetivo de exprimir al máximo el potencial de cada optimizador.

(a) Ajuste de hiperparámetros: Se llevó a cabo mediante un enfoque de prueba y error, con el objetivo de identificar la configuración óptima que permitiera al modelo alcanzar el mayor nivel de precisión posible, sin descuidar métricas clave como el recall.

(b) Entrenamiento de modelo: Se entrenaron diversos algoritmos de optimización, incluyendo Gradiente Descendente (GD), Gradiente Descendente Estocástico (SGD), Gradiente Descendente por Mini Lotes (MBGD), Algoritmo de Gradiente Adaptativo (AdaGrad), Estimación de Momento Adaptativo (Adam) y Propagación de la Raíz del Promedio Cuadrático (RMSProp) sobre el modelo base de Regresión Logística.

3.3. Fase 3: Evaluación del modelo

Los modelos se evaluaron y compararon para determinar qué método de optimización brindaba mejores resultados al aplicarse al modelo de regresión logística.

(a) Evaluación del modelo: Se evaluó el rendimiento del modelo utilizando métricas como sensibilidad, especificidad, precisión, recall y F1 Score, las cuales se calcularon a partir de la matriz de confusión. El análisis permitió medir la capacidad del modelo

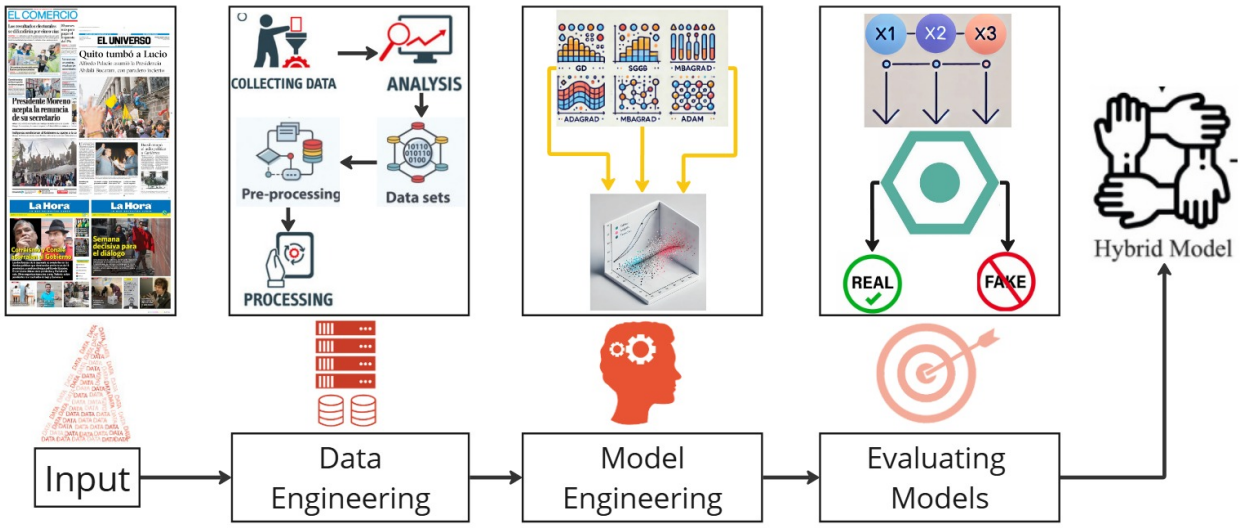

Figura 1. Arquitectura para la clasificación de noticias. Comienza con la ingesta de datos (noticias), seguida de ingeniería de datos para su preprocesamiento. Luego, en la ingeniería de modelos, se entrenan algoritmos de optimización (GD, SGD, MBGD, AdaGrad, Adam) y se evalúan en la etapa de modelos híbridos, clasificando las noticias como verdadera o falsa.

para clasificar noticias en español. Los resultados obtenidos durante las fases de entrenamiento y validación fueron documentados con el objetivo de seleccionar el modelo que presentara el mejor desempeño en función de dichas métricas.

4. Resultados y discusión

En esta sección se presentan y analizan los resultados obtenidos al aplicar diferentes métodos de optimización para la regresión logística. El objetivo principal fue evaluar la precisión de cada método sobre el modelo.

Para ello, se realizaron múltiples experimentos sobre un conjunto de datos preprocesado y balanceado, asegurando condiciones uniformes para cada método de optimización. Se monitorearon métricas como precisión (precision), pérdida (loss function), además de la estabilidad de la convergencia durante el entrenamiento del modelo.

La comparación de los resultados permitirá identificar el mejor método de optimización para la identificación de noticias de política en español en contexto ecuatoriano. Asimismo, se discutirá el impacto del tamaño del lote, la tasa de aprendizaje (learning rate) y otros hiperparámetros relevantes en el desempeño del modelo.

4.1. Resultados

4.1.1. Fase 1: Ingeniería de datos

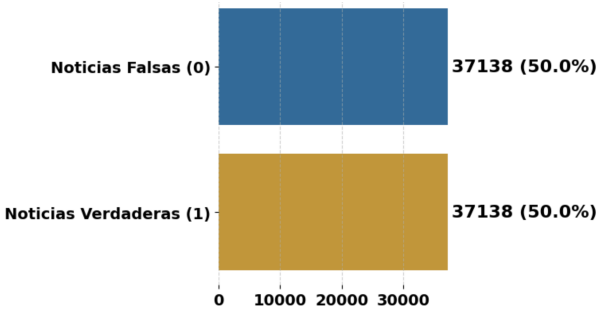

Figura 2. Conjunto de datos perfectamente equilibrado entre dos categorías. Esto sugiere que el modelo de clasificación tendrá las mismas oportunidades de aprender de ambas clases, evitando sesgos hacia una de ellas.

Se consolidaron datos provenientes de tres fuentes: Spanish Political Fake News [18], Spanish Political News Dataset [19] y registros obtenidos por web scraping, alcanzando un total inicial de 82.355 registros (45.045 noticias verdaderas y 37.310 falsas). Tras un proceso de limpieza que eliminó valores nulos, duplicados y registros vacíos, se obtuvieron 44.903 noticias verdaderas y 37.138 falsas. Para evitar sesgos, se equilibraron las clases, ajustando el número de registros a 37.138 por categoría (Figura 2).

Tabla 1. Parámetros utilizados por los algoritmos de optimización GD, SGD, MBGD, AdaGrad, Adam, RMSProp.

Hiperparámetro	**GD**	**SGD**	**MBGD**	**AdaGrad**	**Adam**	**RMSProp**
Learning Rate	10^{-2}	10^{-4}	5×10^{-3}	10^{-4}	10^{-4}	5×10^{-5}
Batch Size	1	1	64	128	256	256
Momentum	0.9	0.9	0.9	-	-	-
Beta 1	-	-	-	-	0.9	-
Beta 2	-	-	-	-	0.999	-
Epsilon	-	-	-	10^{-8}	10^{-8}	10^{-8}
Decay Rate	10^{-12}	10^{-12}	-	-	-	0.99
Regularization	-	-	10^{-4}	10^{-2}	-	-
Epochs	5,000	100,000	3,000	1,210	602	1,165
Stopping Tolerance	10^{-5}	-	-	10^{-4}	10^{-4}	10^{-4}

Número de épocas (**Epochs**); Tasa de aprendizaje (**Learning Rate**); Tamaño del lote (**Batch Size**); Inercia del gradiente (**Momentum)**; Promedio primer momento (**Beta 1**); Promedio segundo momento (**Beta 2**); Evitar división cero (**Epsilon**); Disminución tasa aprendizaje (**Decay Rate**); Evitar sobreajuste (**Regularization**); Parada temprana (**Stopping Tolerance**)

Además, se normalizaron los datos mediante técnicas como conversión a minúsculas, eliminación de caracteres especiales y tokenización, dejándolos listos para la tarea de división de datos. Finalmente, los datos se dividieron en tres conjuntos, con el objetivo de evaluar el modelo tanto con optimización como sin ella. Las noticias fueron seleccionadas de manera aleatoria para asegurar la representatividad de cada conjunto. La división se realizó mediante un código implementado en Python, lo que permitió generar tres subconjuntos específicos: el conjunto de entrenamiento (train_data), destinado al ajuste del modelo; el conjunto de validación (val_data), utilizado para ajustar hiperparámetros y evaluar el rendimiento durante el desarrollo; y el conjunto de pruebas (test_data), empleado para medir el desempeño final del modelo (Figura 3).

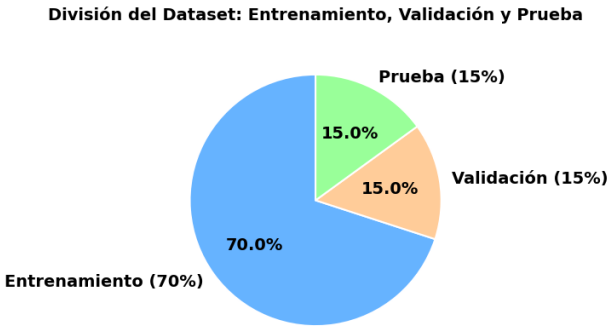

Figura 3. Distribución del conjunto de datos en las fases de entrenamiento, validación y prueba. El 70% de los datos se destinó al entrenamiento del modelo, mientras que el 15% se asignó a la validación y el 15% restante a la prueba. Esta división garantiza un adecuado balance entre el ajuste del modelo y su capacidad de generalización.

4.1.2. Fase 2: Ingeniería de modelos

Los experimentos realizados aplicaron diversos algoritmos de optimización para entrenar un modelo de regresión logística con el propósito de clasificar noticias falsas en español dentro del ámbito político ecuatoriano. Se presentan configuraciones de hiperparámetros, donde se combinan tasas de aprendizaje, tamaños de lote, optimizadores con momentum y epsilon. Algunas configuraciones incluyen regularización, tasas de decaimiento y un criterio de parada. Con estas configuraciones se evaluó su convergencia y rendimiento (Tabla 1).

Tabla 2. Parámetros y precisión de los algoritmos de optimización. Los métodos de optimización son GD, SGD, MBGD, AdaGrad, Adam y RMSProp. Cada uno de estos métodos fue entrenado con distintas épocas, tasas de aprendizaje y tamaños de batch, y el rendimiento de cada algoritmo se representa con la métrica de precisión.

Algoritmo	Epochs	Learning Rate	Batch Size	Momentum	Partition	Precision (%)
GD	5,000	10^{-2}	1	0.9	70:15:15	80.0%
SGD	100,000	10^{-4}	1	0.9	70:15:15	81.3%
MBGD	3,000	5×10^{-3}	64	0.9	70:15:15	76.5%
AdaGrad	1,210	10^{-4}	128	-	70:15:15	71.5%
Adam	602	10^{-4}	256	-	70:15:15	79.0%
RMSProp	1,165	5×10^{-5}	256	-	70:15:15	79.2%

Número de épocas (**Epochs**); Tasa de aprendizaje (**Learning rate**); Tamaño del lote (**Batch size**); Inercia del gradiente (**Momentum)**; Precisión (**Precision**); Gradiente Descendente (**GD**); Gradiente Descendente Estocástico (**SGD**); Gradiente descendente por Mini-Lote (**MBGD**); Algoritmo de Gradiente Adaptativo (**AdaGrad**); Estimación de Momento Adaptativo (**Adam**); Propagación de la Raíz del Promedio Cuadrática (**RMSProp**)

Con la configuración de hiperparámetros se realizó la validación de los optimizadores en donde el Gradiente Descendente (GD) y el Gradiente Descendente Estocástico (SGD) mostraron un desempeño destacado, alcanzando precisiones del 81.3% y 80.0%, respectivamente. El Mini-Batch Gradient Descent (MBGD), alcanzó una precisión menor, del 76.55%.

Por su parte, los algoritmos adaptativos, como AdaGrad, Adam y RMSProp, obtuvieron precisiones inferiores, entre 71.51% y 79.24%, a pesar de un riguroso ajuste mediante prueba y error de sus hiperparámetros, especialmente en lo referente a la tasa de aprendizaje y el tamaño del batch. Estos resultados subrayan la relevancia de ajustar adecuadamente los algoritmos de optimización para mejorar la precisión en tareas de clasificación de noticias falsas (Tabla 2).

De los algoritmos de optimización evaluados, el Gradiente Descendente Estocástico (SGD) obtuvo la mayor precisión en la fase de validación con un 81.3%, superando ligeramente al Gradiente Descendente (GD), que alcanzó un 80.0%. Por lo tanto, el algoritmo SGD destacó como la opción más eficiente en términos de precisión (Tabla 2).

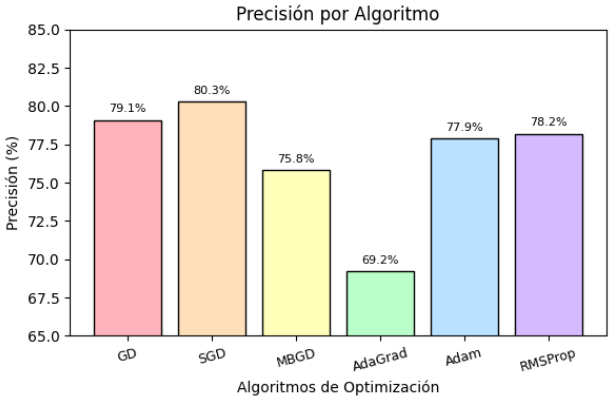

Figura 4. La visualización permite identificar de manera clara la efectividad de cada algoritmo de optimización en el conjunto de datos de evaluación (test), destacando la superioridad del método SGD para su selección.

4.1.3. Fase3: Evaluación del modelo

En la evaluación del modelo, se utilizaron los mismos hiperparámetros (épocas, tasa de aprendizaje, tamaño de lote y momento) y la misma partición del conjunto de datos, de acuerdo con lo descrito en la Tabla (1). En esta fase, el conjunto de prueba test_data fue empleado para medir el rendimiento de los distintos métodos a través de la matriz de confusión. Los resultados confirmaron lo observado en la fase de validación: el modelo que aplicó gradiente descendente estocástico (SGD) obtuvo la mayor precisión (Figura 4).

Por lo tanto, se determinó el modelo más adecuado tanto para validación como para pruebas, siendo el gra-

diente descendente estocástico (SGD) la mejor opción, al superar las métricas del modelo base. Asimismo, se generó un archivo .pt [20] que incluyó el modelo resultante empaquetado, junto con el proyecto completo [21], con el fin de que la comunidad interesada en la optimización de modelos de machine learning pueda reproducir y verificar los resultados obtenidos.

Con el objetivo de optimizar el proceso de evaluación, se implementó un prototipo de interfaz web desarrollado en Flask, diseñado para que los usuarios puedan verificar la autenticidad de noticias políticas en español mediante la introducción de texto. La herramienta ofrece la posibilidad de generar informes personalizables, permitiendo entre 5 y 20 predicciones, los cuales pueden exportarse en PDF para su revisión posterior.

El prototipo desarrollado busca maximizar la facilidad de uso, mostrando solo la información fundamental para el análisis. Si bien aún no se ha aplicado una evaluación formal de la experiencia de usuario, se planea que en futuras iteraciones la plataforma sea intuitiva y ágil. Por el momento, la aplicación solo está disponible para ejecución en entornos locales, lo que restringe su acceso desde otras ubicaciones. Sin embargo, se contempla a futuro su despliegue en la nube utilizando servicios como AWS (Amazon Web Services), GCP (Google Cloud Platform) o Heroku, especializados en el hospedaje de aplicaciones web. Otra alternativa explorada es el uso de Docker para empaquetar la solución, agilizando su implementación en servidores externos. Dado que el modelo en formato .pt tiene un peso reducido, la interfaz no demandaría una infraestructura compleja para su funcionamiento (Figura 5).

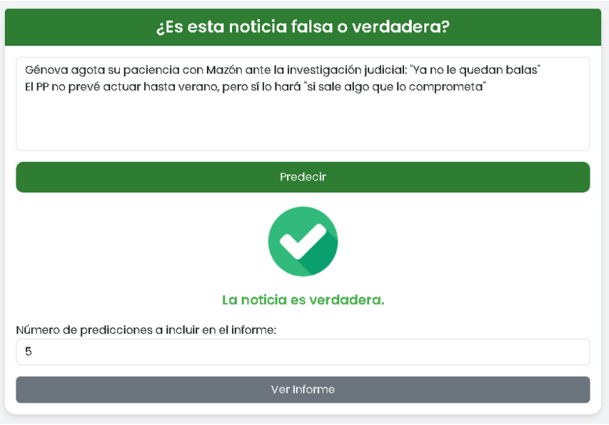

Figura 5. Prototipo de la interfaz web para verificación de noticias políticas. Permite al usuario ingresar texto, obtener una predicción sobre su veracidad y generar un informe personalizable descargable en PDF.

4.2. Discusión

A diferencia de la mayoría de los estudios [15] [5] [4] [9] que se basan en conjuntos de datos en inglés, en este trabajo se adaptaron las fases de CRISP-ML para, a través de la ingeniería de datos, generar un nuevo dataset de noticias en español enriquecido con información de noticias de periódicos ecuatorianos. Este recurso se empleó posteriormente en tareas de optimización, ofreciendo una alternativa diferenciada frente a otros trabajos relacionados.

Al evaluar los modelos mediante la matriz de confusión, se observó que el modelo sin optimización (regresión logística) quedó 6.6% por debajo de aquel entrenado con gradiente descendente estocástico (SGD), el cual alcanzó la mayor precisión. En concreto, la regresión logística obtuvo una precisión de 75.1% en validación y 73.7% en evaluación, mientras que el modelo con SGD logró 81.3% en validación y 80.3% en evaluación. Este hallazgo coincide con la mayoría de estudios similares [4, 5, 8], lo que confirma que la aplicación de métodos de optimización (Gradiente Descendente, Gradiente Descendente Estocástico, Gradiente Descendente por Mini Lotes, Algoritmo de Gradiente Adaptativo, Estimación de Momento Adaptativo y Propagación de la Raíz del Promedio Cuadrático) incrementa de manera significativa el desempeño de las métricas, especialmente en términos de precisión.

De manera similar, el método Adam-LR alcanzó una precisión de 77.95%, superando el 74.8% obtenido por Adam en la detección de noticias falsas en árabe [9]. Asimismo, el trabajo de Rajalaxmi et al. reporta que Adam incrementó la precisión en un 9.54% para el conjunto de datos DS1 y que RMSProp lo hizo en un 5.87% para DS4 [15]. En contraste, en nuestro estudio Adam mostró un aumento de 4.25% y RMSProp de 4.5%. Estos resultados difieren de los hallazgos de Sai Raja et al., donde la combinación de regresión logística con gradiente descendente (LR-GD) logró un incremento del 15% en sus métricas, especialmente en la precisión [8]. Por su parte, en este trabajo, la aplicación de gradiente descendente (GD) mejoró el desempeño en un 5.4%.

No obstante, la capacidad de procesamiento disponible constituyó una limitación durante el entrenamiento, impidiendo aprovechar por completo el potencial de los métodos de gradiente descendente y sus variantes, especialmente los adaptativos. Las limitaciones de hardware (memoria RAM, GPU y CPU) dificultaron evaluar con mayor amplitud el desempeño de estos algoritmos. Dichas dificultades podrían mitigarse mediante la paralelización de hardware con múltiples núcleos o múltiples GPU, y optimizadores como SGD pueden beneficiarse de esta estrategia para

manejar grandes volúmenes de datos, tal como se lo discute en [22].

Además, se identificaron limitaciones referentes a la calidad y la disponibilidad de datos libres sobre noticias políticas en español. Para atenuar este inconveniente, se combinaron diversos conjuntos de datos con información obtenida mediante web scraping, en contraste con otros estudios, por ejemplo, [5] (44,898 registros), [3] (51,233 registros) y [15] (12,800 registros). Esta situación pudo afectar la generalización de los resultados. Se planteó, asimismo, la posibilidad de alcanzar una detección más robusta de noticias falsas si se incorporaran métodos de clasificación más avanzados, como SVM, Random Forest o redes neuronales con diferentes configuraciones, tal como sugiere en [11].

De cara a trabajos futuros, se debe investigar la aplicación de estos modelos optimizados en entornos reales donde la mitigación de desinformación sea prioritaria, además de enriquecer el dataset con más registros y características. Asimismo, podría considerarse la incorporación de optimizadores más avanzados o el uso combinado de métodos de gradiente de primer y segundo orden para mejorar la convergencia y reducir costos computacionales, incluso la sustitución de la regresión logística por enfoques más potentes, como redes neuronales.

5. Conclusiones

El modelo de regresión logística basado en gradiente descendente estocástico (SGD) alcanzó una precisión del 80,3%, superando al modelo sin optimización, cuya precisión fue de 73,7%. Este resultado evidenció la mejora significativa obtenida mediante la optimización y confirmó que la regresión logística requiere un proceso de ajuste para lograr una clasificación más fiable en el ámbito de las noticias falsas de política en español.

Para el entrenamiento, se creó un nuevo conjunto de datos con 82,355 registros resultado de fusionar dos bases anteriores y añadir noticias de política ecuatoriana, asegurando diversidad de fuentes, volumen de datos y mayor riqueza informativa. Dicho proceso, en conjunto con el ajuste de los hiperparámetros, contribuyó a optimizar la precisión del modelo tras la fase de optimización.

Finalmente, la evaluación del mejor modelo obtenido en la fase de validación, mediante matrices de confusión, ofreció un análisis detallado de su rendimiento. Métricas como sensibilidad, especificidad, precisión, exactitud y F1 confirmaron que el Gradiente Descendente Estocástico (SGD) presentó el desempeño más alto frente a Gradiente Descendente (GD), Gradiente Descendente por Mini Lotes (MBGD), Algoritmo de Gradiente Adaptativo (AdaGrad), Estimación de Momento Adaptativo (Adam) y Propagación de la Raíz del Promedio Cuadrático (RMSProp).

Agradecimientos

Extendemos nuestro más sincero reconocimiento a la Universidad Nacional de Loja (UNL) https://unl.edu.ec y, en particular, a la Carrera de Computación por su invaluable apoyo y los recursos proporcionados para la realización de esta investigación. Este artículo es el resultado de un esfuerzo académico desarrollado en el marco de la asignatura "Composición de Textos Científicos en Ingeniería", impartida durante el período académico comprendido entre septiembre 2024 - febrero 2025. Agradecemos la guía y el estímulo intelectual recibidos, que han sido fundamentales para la consecución de este trabajo y para nuestro crecimiento como investigadores en el campo de la ingeniería computacional.

Declaración de IA generativa y tecnologías asistidas por IA en el proceso de redacción

Durante la elaboración de este artículo, se emplearon herramientas como ChatGPT y Bard, orientadas a la generación de ideas innovadoras, la búsqueda de información relevante en los documentos consultados, la traducción del resumen y la redacción de versiones preliminares. Tras la utilización de dichos recursos, se revisó, modificó y amplió la información redactada. En consecuencia, los autores asumieron la plena responsabilidad del contenido final presentado en la investigación.

Referencias

[1] D. Y. Quintero Perozo and J. A. Ortega Riveros, “Detección automática de noticias falsas en español con técnicas de machine learning,” B.S. thesis, Universidad de los Andes, Bogotá, Colombia, 2020.

[2] G. A. Chavarría Muñoz, “Prototipo de aplicación web para la detección de noticias falsas utilizando machine learning y técnicas de procesamiento de lenguaje natural,” B.S. thesis, Universidad del Istmo, Ciudad de Guatemala, Guatemala, 2022.

[3] K. Martínez-Gallego, A. M. Álvarez Ortiz, and J. D. Arias-Londoño, “Fake news

detection in spanish using deep learning techniques," pp. 1–10, 2021. [Online]. Available: https://arxiv.org/abs/2110.06461

[4] J. Asha and A. Meenakowshalya, "Fake news detection using n-gram analysis and machine learning algorithms," *Journal of Mobile Computing, Communications & Mobile Networks*, vol. 8, no. 1, pp. 33–43, 2021.

[5] I. Ahmad, M. Yousaf, S. Yousaf, and M. O. Ahmad, "Fake news detection using machine learning ensemble methods," *Complexity*, vol. 2020, no. 1, p. 8885861, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2020/8885861

[6] S. Basharat, S. Afzal, A. M. Bamhdi, S. Khurshid, and M. Chachoo, "Predicting and mitigating the effect of skewness on credibility assessment of social media content using machine learning: A twitter case study," *International Journal of Computer Theory and Engineering*, vol. 15, no. 3, pp. 101–110, 2023. [Online]. Available: https://www.ijcte.org/show-132-1644-1.html

[7] K. Passi and A. Shah, "Distinguishing fake and real news of twitter data with the help of machine learning techniques," in *Proceedings of the 26th International Database Engineered Applications Symposium*, ser. IDEAS '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 18. [Online]. Available: https://doi.org/10.1145/3548785.3548811

[8] E. V. Sai Raja, B. L V S Aditya, and S. N. Mohanty, "Fake profile detection using logistic regression and gradient descent algorithm on online social networks," *EAI Endorsed Transactions on Scalable Information Systems*, vol. 11, no. 1, pp. 1–10, Nov. 2023. [Online]. Available: https://publications.eai.eu/index.php/sis/article/view/4342

[9] M. Alsafad, "Stance classification for fake news detection with machine learning," *The Eurasia Proceedings of Science Technology Engineering and Mathematics*, vol. 22, p. 191198, 2023. [Online]. Available: https://doi.org/10.55549/epstem.1344457

[10] A. Sultana, M. Islam, M. Hasan, and F. Ahmed, "Fake news detection using machine learning techniques," in *2023 IEEE/ACIS 21st International Conference on Software Engineering Research, Management and Applications (SERA)*, 2023, pp. 98–103. [Online]. Available: https://ieeexplore.ieee.org/document/10197712

[11] N. J. Mafla Checa, "Identificación automática de noticias falsas en español utilizando técnicas de minería de datos y procesamiento del lenguaje natural." B.S. thesis, Escuela Politécnica Nacional, Quito, Ecuador, 2021.

[12] S. Tene, "Dataset (noticias política española & ecuatoriana)," 2025, repositorio GitHub. [Online]. Available: https://bit.ly/3WLBTZ2

[13] A. E. V. Lozano and M. F. C. Muñoz, "Noticias falsas la otra cara de la pandemia. caso: Ecuador." *Espíritu Emprendedor TES*, vol. 7, no. 1, pp. 52–74, 2023. [Online]. Available: https://doi.org/10.33970/eetes.v7.n1.2023.325

[14] M. A. Espejel-Rivera, R. Calderón-Suárez, R. M. Ortega-Mendoza, C. J. Camacho-Bello, and M. A. Máquez-Vera, "Detección automática de noticias falsas usando representaciones textuales tradicionales y soluciones basadas en aprendizaje profundo," *Pädi Boletín Científico de Ciencias Básicas e Ingenierías del ICBI*, vol. 10, no. Especial3, pp. 120–127, ago. 2022. [Online]. Available: https://doi.org/10.29057/icbi.v10iEspecial3.9008

[15] R. Rajalaxmi, L. Narasimha Prasad, B. Janakiramaiah, C. Pavankumar, N. Neelima, and V. Sathishkumar, "Optimizing hyperparameters and performance analysis of lstm model in detecting fake news on social media," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, Mar. 2022, just Accepted. [Online]. Available: https://doi.org/10.1145/3511897

[16] Y. Blanco-Fernández, J. Otero-Vizoso, A. Gil-Solla, and J. García-Duque, "Enhancing misinformation detection in spanish language with deep learning: Bert and roberta transformer models," *Applied Sciences*, vol. 14, no. 21, 2024. [Online]. Available: https://doi.org/10.3390/app14219729

[17] S. Studer, T. B. Bui, C. Drescher, A. Hanuschkin, L. Winkler, S. Peters, and K.-R. Müller, "Towards crisp-ml(q): A machine learning process model with quality assurance methodology," *Machine Learning and Knowledge Extraction*, vol. 3, no. 2, pp. 392–413, 2021. [Online]. Available: https://doi.org/10.3390/make3020020

[18] Kaggle, "Spanish political fake news dataset," 2023. [Online]. Available: http://bit.ly/40GdwgJ

[19] ——, "Spanish political news dataset," 2023. [Online]. Available: https://bit.ly/4jMUMoj

[20] S. Tene, "Modelo optimizado empaquetado (sgd-lr)," 2025, repositorio GitHub. [Online]. Available: https://bit.ly/40Nz7nt

[21] ——, "Proyecto optimización completo," 2025, repositorio GitHub. [Online]. Available: https://bit.ly/3EhYTsf

[22] Y. Tian, Y. Zhang, and H. Zhang, "Recent advances in stochastic gradient descent in deep learning," *Mathematics*, vol. 11, no. 3, 2023. [Online]. Available: https://doi.org/10.3390/math11030682