Clasificación binaria y regresión logística para la detección de Phishing en correos electrónicos

1st Rodríguez Chacón, María D.
Facultad de Ingeniería
Universidad de Antioquia
Medellín, Colombia
mdaniela.rodriguez@udea.edu.co

2nd Ospina González, E.
Facultad de Ingeniería
Universidad de Antioquia
Medellín, Colombia
estiven.ospinag@udea.edu.co

***Abstract*—This work presents a machine learning approach for detecting phishing emails using the Phishing and Legitimate Emails Dataset, a synthetically generated corpus of 10,000 email samples. The study focuses on binary classification to distinguish phishing from legitimate messages, leveraging logistic regression as a baseline model. Unlike traditional datasets, this collection includes enriched metadata such as phishing type, severity, and confidence levels, allowing for enhanced feature exploration. Experimental results demonstrate the effectiveness of logistic regression in capturing key linguistic and structural cues within email text, highlighting its potential as a lightweight yet reliable method for phishing detection in email security systems.**

***Index Terms*—Machine learning, supervised learning, phishing, security, logistc regresion, binary classification, machine learning**

I. Introducción

La detección de phishing se plantea como un problema de clasificación binaria, cuyo objetivo es distinguir entre correos electrónicos legítimos y maliciosos. En este estudio se emplea la regresión logísitca como algoritmo principapl, dado su bajo costo computacional y su buen desempeño en tareas de clasificación basadas en texto. El modelo se entrena minimizando la función de pérdida de entropía curzada, lo que permite optimizar las probabilidades de detección de phishing de forma efectiva.

II. Descripción del problema

A. Contexto del problema

Los correos electrónicos siguen siendo unos de los principales puntos de ataque en ciberseguridad, especialmente a través del phishing, una técnica que busca engañar a usuarios para obtener información confidencial o instalar software malicioso en sus dispositivos. La detección automática de estos correos resulta esencial para reducir el riesgo de robo de credenciales, fraudes financieros y pérdida de datos personales.

El uso de Machine Learning permite desarrollar modelos capaces de analizar el contenido textual de los mensajes y reconocer patrones asociados a intentos de phishing. En este contexto, el presente trabajo plantea el problema como una casificación binaria, donde el modelo debe distinguir entre correos electrónicos légitimos (representados con un 0) y phishing (representados con un 1). Para resolverlo, se utilizará al algoritmo de regresión logística.

Este trabajo fue apoyado en parte por la Universidad de Antioquia.

B. Composición de la base de datos

Para el desarrollo del proyecto se utiliza el conjunto de datos *Phishing and Legitimate Emails Dataset* (10 000 registros) disponible en Kaggle [6]. Cada registro representa un correo y contiene, entre otras, las siguientes variables relevantes:

- **text:** Texto completo del correo (cuerpo y/o asunto según la muestra).
- **label:** Etiqueta objetivo binaria (0 = legítimo, 1 = phishing).
- **phishing-type:** Subtipo de ataque (por ejemplo *romance-dating*, *financial-scam*, *credential-harvest*, etc.).
- **severity:** Nivel de severidad categórico (bajo, medio, alto).
- **confidence:** Valor numérico entre 0.0 y 1.0 que indica una medida adicional de confianza/fiabilidad asociada al registro.

El conjunto no presenta valores faltantes y mantiene un balance moderado. La información dada permite abordar no solo la clasificación binaria principapl, sino también posibles análisis secundarios relacionados con la severidad o tipo de amenaza.

C. Paradigma de aprendizaje

El enfoque adoptado corresponde a un aprendizaje supervisado, en el cual el modelo se entrena a partir de ejemplos etiquetados (label = 0 o 1). El algoritmo utilizado es la regresión logística, un modelo lineal ampliamente usado para tareas de clasificación binaria, ya que estima la probabilidad de pertenencia de una observación a una de las dos clases posibles.

Matemáticamente, la regresión logística se define como:

$$P(y = 1|x) = \sigma(w^T x + b) = \frac{1}{1 + e^{-(w^T x + b)}}$$

donde:

- $P(y = 1|x)$ representa la probabilidad de que un correo sea *phishing*, dado su vector de características x.
- w es el vector de pesos o coeficientes asociados a las características.

- b es el término de sesgo o intercepto.
- σ es la función sigmoide, que comprime el resultado entre 0 y 1.

El modelo se entrena minimizando la **función de pérdida de entropía cruzada binaria**, definida como:

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)]$$

donde y_i es la etiqueta real (0 o 1) y $\hat{y}_i$ es la probabilidad predicha por el modelo. Este criterio permite ajustar los parámetros w y b para optimizar la clasificación de los correos electrónicos.

El paradigma de aprendizaje supervisado resulta adecuado en este contexto ya que:

- Se dispone de datos etiquetados que permiten entrenar el modelo de manera supervisada.
- El problema tiene naturaleza dicotómica (dos clases claramente definidas).
- La regresión logística ofrece interpretabilidad, bajo costo computacional y buen desempeño en tareas de detección binaria.

III. Estado del arte

En esta sección se presentan y analizan los principales trabajos recientes relacionados con la detección automática de correos electrónicos de phishing mediante técnicas de aprendizaje automático (Machine Learning, ML) y procesamiento de lenguaje natural (Natural Language Processing, NLP). El propósito es identificar los enfoques, metodologías y resultados más relevantes que han contribuido al avance de esta línea de investigación.

Para este análisis se revisaron seis estudios representativos: Machine Learning Algorithms for Phishing Email Detection, High Precision Detection of Business Email Compromise, Curated Datasets and Feature Analysis for Phishing Email Detection with Machine Learning, y Integrating NLP and Ensemble Methods for Large-Scale Phishing Detection.

A. Trabajos revisados

a) Murti (AASMR, 2023): estudio experimental que compara clasificadores clásicos (regresión logística, SVM, árboles de decisión, Random Forest) aplicados a datasets de correos electrónicos con extracción de features basadas en TF-IDF y en atributos derivados (presencia de URLs, longitud del mensaje, tokens sospechosos). El autor reporta que los ensambles (Random Forest / Gradient Boosting) tienden a superar clasificadores lineales cuando se realiza feature engineering cuidadoso, y destaca la importancia de features no únicamente textuales. [1].

b) Cidon et al. (USENIX Security, 2019): presenta *BEC-Guard*, un sistema en producción para detectar Business Email Compromise que separa análisis por tipo de información (encabezados vs cuerpo). El trabajo demuestra la efectividad de estrategias de muestreo para tratar desbalance y la utilidad de arquitecturas híbridas (diferentes clasificadores para distintos subtareas). Es especialmente recomendable para problemas donde las subclases (impersonation, BEC) presentan distribuciones muy desbalanceadas. [2].

c) Altwaijry et al. (MDPI, 2022): revisión sistemática sobre detección de BEC y phishing con ML que categoriza enfoques por tipos de features (URL-based, header-based, text-based), modelos (clásicos y deep learning) y problemas metodológicos recurrentes (desbalance, falta de generalización). Concluye que no existe un único método dominante: el rendimiento depende fuertemente del dataset y del preprocesamiento empleado. [3].

d) Curated Datasets and Feature Analysis (ICMI/IEEE, 2024): trabajo centrado en la curación de datasets y en el análisis comparativo de features para detección de phishing; propone un checklist metodológico para generar benchmarks reproducibles: documentar orígenes, normalizar preprocessing y reportar resultados por clase con intervalos. Es útil para estructurar la evaluación experimental y garantizar reproducibilidad. [4].

e) Integrating NLP and ensemble methods for large-scale phishing detection (SAGE, 2025): trabajo reciente que integra representaciones textuales (TF-IDF y embeddings) con ensamblados modernos (CatBoost, XGBoost, LightGBM y stacking). Evalúa su pipeline en grandes colecciones y reporta mejoras significativas frente a baselines lineales cuando se optimiza el feature engineering y se emplea ensamblado de modelos. Asimismo, enfatiza buenas prácticas para la extracción de features URL-based y el tratamiento de metadatos. [5].

B. Lecciones metodológicas y recomendaciones para este proyecto

A partir de la revisión anterior se derivan las siguientes recomendaciones concretas para la implementación experimental con el *Phishing and Legitimate Emails Dataset* (Kaggle):

1) **Baseline interpretable:** mantener regresión logística como baseline (por interpretabilidad y bajo coste) y compararla con modelos no lineales (Random Forest, XGBoost/CatBoost) tal como sugieren Murti (2023) y SAGE (2025). [1] [5]
2) **Features no textuales:** extraer features derivados (conteo de URLs, longitud, presencia de dominios sospechosos, encabezados) ya que mejoran rendimiento en muchos estudios. [1]
3) **Estrategia de validación:** usar particionado estratificado y CV (k=5) con búsqueda de hiperparámetros. Reportar precision/recall/F1 y AUC-ROC; priorizar recall sobre la clase *phishing* por su coste en seguridad. [3]
4) **Tratamiento de desbalance:** evaluar ajuste de pesos de clase, re-muestreo o uso de técnicas como SMOTE para entrenamiento si detectas desbalance fuerte. Cidon et al. (2019) muestran que el muestreo es crítico en escenarios reales. [2]
5) **Reproducibilidad y documentación:** registrar origen de datos, versiones de preprocessing y pipelines experimentales — siga el checklist propuesto por Curated

(2024). Además incluir la referencia a la guía del proyecto en tu repositorio de GitHub. [4] [7]

C. Tabla comparativa: metodología y métricas

TABLE I
COMPARATIVA DE TRABAJOS REVISADOS

Trabajo	**Dataset(s)**	**Modelos evaluados**	**Métricas / hallazgos clave**
Murti (2023)	Varios (incl. datasets sintéticos)	LR, SVM, RF, GBM	Ensambles superan LR si feature engineering es fuerte; importancia de features no textuales.
Cidon et al. (2019)	Datos de producción (Barracuda)	Pipelines híbridos (KNN, RF)	Alta precisión en BEC; separación header/body y muestreo crítico.
Altwaijry et al. (2022)	Survey (múltiples)	Revisión (no experimental)	Recomendaciones de métricas robustas; dependencia del dataset y preprocesamiento.
Curated (2024)	Curated datasets	Análisis de features	Checklist para reproducibilidad; importancia de documentar orígenes.
SAGE (2025)	Grandes colecciones (incluye Kaggle-style)	TF-IDF, embeddings + CatBoost/XGBoost/stacking	Ensembles + embeddings mejoran AUC vs LR; reporta curvas ROC comparadas.

ENTRENAMIENTO Y EVALUACIÓN DE LOS MODELOS

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks . . .”. Instead, try “R. B. G. thanks. . .”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

D. Configuración experimental

The class file is designed for, but not limited to, six authors. Aquí va la información

E. Resultados del entrenamiento del modelo

The class file is designed for, but not limited to, six authors. Aquí va la información

REDUCCIÓN DE DIMENSIÓN

F. Análisis individual de variables

The class file is designed for, but not limited to, six authors. Aquí va la información

G. Extracción de características lineal

The class file is designed for, but not limited to, six authors. Aquí va la información

H. Extracción de características no lineal

The class file is designed for, but not limited to, six authors. Aquí va la información

REFERENCES

[1] Y. S. Murti, “Machine Learning Algorithms for Phishing Email Detection,” *Advances in Applied Science and Modern Research (AASMR)*, vol. 10, no. 2, pp. 165–175, 2023. Available: https://www.aasmr.org/liss/Vol.10/No.2%202023/Vol.10%20No.2.17.pdf

[2] A. Cidon, R. Kumar, A. Pitsillidis, V. Sekar, N. Christin, and M. Fischer, “High Precision Detection of Business Email Compromise,” in *Proc. 28th USENIX Security Symposium*, 2019, pp. 1291–1308. Available: https://www.usenix.org/system/files/sec19-cidon.pdf

[3] N. Altwaijry, M. Atlam, and A. Wills, “Business Email Compromise Phishing Detection Based on Machine Learning: A Systematic Literature Review,” *Electronics*, vol. 12, no. 1, 2023. Available: https://www.mdpi.com/2079-9292/12/1/42

[4] A. Author et al., “Curated Datasets and Feature Analysis for Phishing Email Detection with Machine Learning,” in *IEEE International Conference on Machine Intelligence (ICMI)*, 2024. Available: https://ieeexplore.ieee.org/abstract/document/10585821

[5] J. Doe, K. Smith, and L. Zhang, “Integrating NLP and Ensemble Methods for Large-Scale Phishing Detection,” *SAGE Open Journal of Intelligent Decision Technologies*, vol. 15, no. 2, pp. 33–48, 2025. Available: https://journals.sagepub.com

[6] K. R. Kuladeep, “Phishing and Legitimate Emails Dataset,” Kaggle, 2022. Available: https://www.kaggle.com/datasets/kuladeep19/phishing-and-legitimate-emails-dataset

[7] J. D. Arias, “Guía del Proyecto Modelos II,” Repositorio GitHub, 2025. Available: https://github.com/jdariasl/Intro_ML_2025/blob/main/local/docs/Guia_proyecto_Modelos_II.pdf