Detector de phishing en correos electrónicos

1st Sandro Carrillo
Escuela de Ciencia de la Computación
Universidad Nacional de Ingeniería
Lima, Perú
elcorreo@uni.pe

2nd Ariana Lopez
Escuela de Ciencia de la Computación
Universidad Nacional de Ingeniería
Lima, Perú
elcorreo@uni.pe

3rd Albert Argumedo
Escuela de Ciencia de la Computación
Universidad Nacional de Ingeniería
Lima, Perú
elcorreo@uni.pe

4th Juan de Dios Lerzundi
Escuela de Ciencia de la Computación
Universidad Nacional de Ingeniería
Lima, Perú
juan.lerzundi.r@uni.pe

***Resumen*—El phishing es una de las amenazas prevalentes en ciberseguridad. En este proyecto se implementó un sistema de detección automática basado en técnicas de aprendizaje supervisado y procesamiento de lenguaje natural.**

El pipeline consta de preprocesamiento de texto, vectorización mediante TF-IDF y la evaluación de dos clasificacodres principales: Logistic Regressión (LR) y Random Forest.

Además, se usó Multinomial Naive Bayes como modelo baseline.

***Palabras clave*—phishing detection,email classification, TF-IDF, Logistic Regression, Random Forest, Naive Bayes, NLP, machine learning.**

I. Introducción

A. Contexto y Motivación

El correo electrónico, por su naturaleza masiva, es usado en ámbitos personales, académicos y empresariales; sin embargo, es un medio ideal para ataques maliciosos que buscan engañar a las personas para obtener información sensible. Este tipo de ataque se demonina *phishing*, que se identifica mayormente por el uso de links maliciosos a sitios web que se encargan de obtener información como el número de tarjeta de crédito o credenciales de usuario [1].

B. Importancia del Problema

El correo electrónico, debido a su bajo coste de distribución, y a su adopción masiva, es uno de los principales vectores de ingeniería social. Esto **implica** que el riesgo de ser víctima de phishing va más allá del aspecto técnico y se centra en la capacidad del usuario para reaccionar ante estos ataques. En el pasado se emplearon filtros basados en listas negras, pero estos ya no resultan suficientes en el escenario actual, en el que los correos imitan con un mayor grado de exactitud el lenguaje natural como los dominios desde los cuales se envían.

En este contexto, el uso de técnicas de Inteligencia Artificial y de procesamiento de lenguaje natural ofrece una solución adaptable y escalable, que permite analizar el contenido y los metadatos del correo para identificar patrones complejos en el contenido del mensaje [2].

C. Objetivos del Proyecto

1) Objetivo General:

- Implementar un sistema de clasificación automática para identificar el phishing en correos electrónicos mediante técnicas de aprendizaje supervisado y una representación textual basada en TF-IDF.

2) Objetivos Específicos:

- Construir un pipeline reproducible que abarque la limpieza, normalización y vectorización del texto contenido en correos electrónicos etiquetados. (QUITAR ESTO CUANDO SE HAYA DESCRITO TOTALMENTE EL PIPELINE)[Pipeline reproducible][Obj. aplicación]
- Implementar un modelo baseline utilizando Multinomial Naive Bayes para establecer un punto de referencia en términos de precisión, recall y F1-score. (QUITAR ESTO CUANDO SE HALLA DEFINIDO LA IMPLEMENTACIÖN DEL BASELINE)[Baseline con NB][Obj. técnico]
- Entrenar y evaluar modelos avanzados, específicamente Logistic Regression y Random Forest. (QUITAR ESTO CUANDO SE HAYA DADO UNA INTRODUCCIÓN AL ENTRENAMIENTO DEL MODELO)[modelos LR y RF][Obj. técnico]
- Compara el desempeño de los modelos considerando métricas relevantes para el problema de phishing, priorizando la red de falsos negativos medianteel análisis de recall. (QUITAR ESTO CUANDO SE HAYA HECHO LA COMPARATIVA ENTRE LOS MODELOS)[Comparación de modelos][Obj. evaluación]
- Analizar las características lingüísticas más relevantes para la clasificación mediante coeficientes del modelo y la importancia de atributos. (QUITAR ESTO CUANDO SE HAYA HECHO EL ANÁLISIS DE LA CLASIFICACIÓN DE LOS DATOS)[Análisis de Features][Obj. ético]
- Identificar oportunidades de mejora y establecer posibles líneas futuras de investigación, como el uso de embeddings contextualizados o modeos basados en Trans-

formers. (QUITAR ESTO CUANDO SE HAYA HECHO TRABAJO A FUTURO)[Trabajo futuro][Obj. ético]

D. Restricciones

- El proyecto se centra en la detección de phishing general (no spear-phishing dirigido).
- Se parte de un dataset ya preprocesado (el informe detalla el proceso de preprocesamiento realizado).

II. Marco Teórico

A. Definición y naturaleza del phishing

El phishing es una técnica de ingeniería social que emplea correos (u otros canales) simulando entidades legítimas para inducir a la víctima a revelar credenciales, ejecutar acciones o entregar información sensible. Sus variantes incluyen spear-phishing (ataques dirigidos), whaling (dirigido a ejecutivos) y ataques basados en URL/malware (ver Garera et al., 2007; Fette et al., 2007).

B. Representación del texto y fundamentos de PLN

- **Tokenización y normalización:** separación en tokens, conversión a minúsculas, normalización Unicode y tratamiento de caracteres especiales (URLs, correos, números).
- **Stopwords, lematización/stemming:** reducción de forma de palabras para disminuir sparsity.
- **Bag-of-Words / n-grams:** representación básica que captura presencia/frecuencia de términos y frases cortas; útil con Naive Bayes y modelos lineales.
- **TF-IDF:** ponderación por frecuencia inversa a la frecuencia en corpus, reduce peso de términos comunes.
- **Embeddings y representaciones densas:** word2vec/GloVe representan palabras en espacios vectoriales densos; capturan similitud semántica y mejoran cuando se requiere generalización semántica (Mikolov et al., 2013; Pennington et al., 2014).
- **Representaciones contextualizadas:** Transformers (BERT, RoBERTa) generan embeddings contextuales por token/frase y suelen mejorar el rendimiento en tareas de clasificación de texto si hay datos suficientes (Vaswani et al., 2017; Devlin et al., 2019).

C. Modelos de clasificación supervisada

- **Naive Bayes (Multinomial/Bernoulli):** modelo probabilístico asumiendo independencia condicional de características; rápido y efectivo en text mining con bag-of-words/TF-IDF (Manning et al., 2008).
- **Árboles y ensamblados (Random Forest, Gradient Boosting):** capturan interacciones no lineales entre features; Random Forest es robusto a ruido y poca sintonía de hiperparámetros.
- **SVM:** buena separación en espacios de alta dimensión, efectivo con kernels y cuando hay margen claro entre clases.
- **Redes neuronales y Transformers:** CNNs/RNNs (antes dominantes en PLN) y hoy Transformers. Requieren más datos y cómputo, pero capturan mejor semántica y contexto.
- **Evaluación y selección de modelos:** emplear cross-validation, curvas ROC/PR, y reportar precision/recall/F1 por clase (especialmente recall para detectar phishing).

D. Tipos de features relevantes en detección de phishing

- **Contenido textual:** términos sospechosos (“password”, “verify”), patrones de lenguaje urgente, uso de imperativos, errores ortográficos.
- **Características estructurales:** presencia de URLs, número de enlaces, longitud del correo, HTML vs texto plano, uso de imágenes.
- **Características del remitente/metadatos:** dominio del remitente, coincidencia entre From y Return-Path, IP de origen, registros SPF/DKIM/DMARC.
- **Características de la URL:** longitud, uso de subdominios, caracteres especiales, presencia de IP en lugar de dominio, similitud visual con dominios legítimos (homoglyphs).
- **Comportamiento y contexto:** tasa de envíos por remitente, patrones temporales, reputación del dominio.

E. Problemas técnicos centrales

- **Desbalance de clases:** los correos legítimos suelen superar en número a los phishing; requiere técnicas de muestreo (oversampling/SMOTE), o métricas robustas (PR-AUC).
- **Drift y evolución de ataques:** campañas nuevas cambian vocabulario y tácticas; necesario reentrenamiento y monitorización.
- **Adversarialidad:** atacantes pueden adaptar textos para evadir detectores; conviene estudiar adversarial ML y robustez (Biggio & Roli, 2018).

III. Estado del Arte

A. Enfoques históricos y baselines

- **Reglas heurísticas y blacklists:** sistemas tempranos usaban listas negras de dominios/URLs y reglas de coincidencia de patrones; funcionan contra técnicas conocidas pero no contra evasiones (nazario corp).
- **Filtrado bayesiano y ML clásico:** Multinomial Naive Bayes y SVMs se convirtieron en baseline por su eficiencia y desempeño en datasets textuales (Abu-Nimeh et al., 2007; Fette et al., 2007).

B. Métodos basados en características (features)

- Investigaciones tempranas mostraron que combinar features de contenido, URL y metadatos mejora la detección. Garera et al. (2007) y Bergholz et al. (2010) analizaron heurísticas de URL y contenido.
- Modelos de ensamblado (Random Forest, Gradient Boosting) demostraron robustez frente a ruido y capacidad para priorizar features relevantes (importancia de variables).

C. Deep learning y representaciones modernas

- **CNN/RNN:** capturan patrones locales y secuencias (uso en clasificación de correo).
- **Transformers (BERT y variantes):** han mostrado mejoras marcadas en clasificación de texto y tareas de seguridad cuando se dispone de datos de calidad o se hace fine-tuning (Devlin et al., 2019).
- **Enfoques híbridos:** combinar embeddings BERT con features manuales (URL, encabezados) suele ser altamente efectivo, aprovecha semántica de texto y reglas estructurales.
-

D. Datasets y benchmarks comunes

- **Enron Email Dataset:** corpus grande de correos empresariales (usado para spam/filtrado y como fuente de ham).
- **SpamAssassin public corpus:** colección etiquetada de spam y ham.
- **Phishing corpora / repositorios:** conjuntos públicos con muestras de phishing (por ejemplo PhishTank/Nazario/PhishCorpus).
- Investigaciones recientes también construyen datasets a partir de correos reales anotados y de campañas de phishing actuales; calibrar modelos en datos recientes es crucial.

E. Evaluación práctica y métricas

- **Recall prioritario:** en detección de phishing, disminuir falsos negativos suele ser más importante que minimizar falsos positivos.
- **PR-AUC vs ROC-AUC:** cuando la clase positiva es rara, PR-AUC refleja mejor la capacidad del clasificador para identificar positivos relevantes.
- **Explicabilidad y análisis de errores:** LIME/SHAP son herramientas para entender decisiones y depurar falsos positivos/negativos.

F. Resumen de hallazgos empíricos

- Los modelos que combinan features manuales (URL, encabezados) con representaciones semánticas (embeddings o Transformers) tienden a obtener los mejores resultados.
- Random Forest y Gradient Boosting son fuertes competidores cuando los recursos son limitados; Transformers dominan cuando hay datos y cómputo suficientes.
- La robustez a nuevas campañas requiere pipelines de reentrenamiento y detección de deriva.

IV. Metodología

El sistema consta de los siguientes módulos:

A. Flujo General

1) **Preprocesamiento:** normalización y limpieza del texto del correo (remoción de HTML, URLs, tokens no alfabéticos), reemplazo de direcciones y números por tokens especiales, minúsculas, eliminación de stopwords, lematización/stemming.
2) **Extracción de características:** representación mediante TF-IDF (n-grams unigrama y bigrama), inclusión opcional de features booleanos (presencia de URL, cantidad de enlaces, uso de palabras claves sospechosas) y metadatos (dominio remitente, encabezados).
3) **Partición de datos y balanceo:** división Train/Test (p. ej. 80/20) y validación cruzada (5-fold). En presencia de desbalance, aplicación de técnicas como SMOTE o submuestreo estratificado.
4) **Entrenamiento:** entrenamiento de Multinomial Naive Bayes y RandomForestClassifier (scikit-learn).
5) **Evaluación:** cálculo de accuracy, precision, recall, F1, ROC-AUC y PR-AUC; generación de matriz de confusión y análisis de importancia de variables (Random Forest).
6) **Análisis y despliegue:** análisis de casos erróneos, propuestas de mejora y diseño de prototipo para integración en pipeline de correo.

B. Descripción del Dataset

Para el desarrollo y evaluación de los modelos propuestos, se utilizó el "Phishing Email Dataset" obtenido del repositorio de Kaggle (Alam, 2020). Este conjunto de datos es una compilación robusta que integra múltiples fuentes de referencia en el ámbito de la ciberseguridad, incluyendo el corpus de Enron para correos corporativos legítimos, el dataset de SpamAssassin y colecciones de correos fraudulentos como el "Nigerian Fraud" y "Nazario".

El dataset final consolidado consta de un total de 82,486 registros únicos. Una ventaja significativa de este recurso es su balance de clases, reduciendo el sesgo natural que suele existir en la detección de anomalías: contiene 42,891 correos etiquetados como phishing (aproximadamente el 52%) y 39,595 correos legítimos (48%). Los datos se presentan en dos columnas principales: `text_combined`, que contiene la concatenación del asunto y el cuerpo del mensaje sin metadatos de cabecera complejos, y la etiqueta binaria `label` (0 para legítimo, 1 para phishing).

C. Preprocesamiento

1) **Limpieza:** eliminar etiquetas HTML, normalizar saltos de línea, eliminar caracteres no ASCII si procede.
2) **Normalización:** pasar a minúsculas, normalizar acentos.
3) **Tokens especiales:** reemplazar URLs por `<URL>`, direcciones de email por `<EMAIL>` y números por `<NUM>`.
4) **Tokenización:** separación en tokens — se consideraron n-grams (1,2).
5) **Stopwords & Lematización:** remover palabras funcionales y lematizar para reducir variabilidad morfológica.
6) **Vectorización:** TF-IDF con límite de vocabulario (p. ej. max_features=20,000), ngram_range=(1,2), y sublinear_tf=True.

D. Modelos

- **Multinomial Naive Bayes:** sklearn.naive_bayes.MultinomialNB(alpha=1.0) como baseline para datos textuales.
- **Random Forest:** sklearn.ensemble.RandomForestClassifier (n_estimators=200, max_depth=None, n_jobs=-1, random_state=42) usado para capturar interacciones entre tokens y es robusto a features ruidosos.
- **Validación:** 5-fold cross-validation y evaluación sobre test holdout (80/20). Se reportan métricas promedio y desviación estándar.

E. Métricas de evaluación

- **Precision, Recall, F1-score:** para la clase positiva (phishing).
- **ROC-AUC y PR-AUC:** dado que la clase positiva puede ser minoritaria, PR-AUC es útil para valorar el rendimiento en detección.
- **Matriz de confusión:** análisis de falsos positivos y falsos negativos (importante por coste distinto de ambos errores).

V. Resultados

A. Tabla comparativa de métricas

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

B. Matrices de confusión

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

C. Figuras

VI. Discusión

A. Consideraciones Éticas

- El análisis de correos electrónicos implica acceso a información potencialmente sensible (mensajes privados, datos personales). Es obligatorio aplicar principios de minimización de datos, anonimización/pseudonimización y políticas de retención.
- La sobre-representación de ciertos idiomas puede traducirse en mayor tasa de falsos positivos para grupos específicos, siendo idiomas minoritarios, esto añade un sesgo en los datos de entrenamiento.
- Para tratar el sesgo en los datos se recomendaría evaluar métricas por subgrupos, usar muestreo estratificado y técnicas de debiasing.
- La aparición de falsos positivos puede bloquear comunicaciones legítimas y afectar operaciones como pérdida de informacióno demora. Ante esto se deberían aplicar políticas de cuarentena, notificaciones o vías de apelación, umbrales ajustables según el criterio del usuario u organización.

VII. Conclusiones

A. Conclusiones Principales

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

B. Trabajo a Futuro

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

REFERENCES

[1] I. Fette, N. Sadeh, and A. Tomasic, "Learning to detect phishing emails," in *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*, Association for Computing Machinery, New York, NY, USA, 2007, pp. 649–656. doi: 10.1145/1242572.1242660.

[2] S. Abu-Nimeh, D. Nappa, X. Wang, and S. Nair, "A comparison of machine learning techniques for phishing detection," in *Proceedings of the Anti-Phishing Working Groups 2nd Annual eCrime Researchers Summit (eCrime '07)*, Association for Computing Machinery, New York, NY, USA, 2007, pp. 60–69. doi: 10.1145/1299015.1299021.

[3] Manning, C. D., Raghavan, P., & Schütze, H. (2009). Introduction to Information Retrieval (Online edition). Cambridge University Press. Disponible en PDF

[4] Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill Science/Engineering/Math.

[5] Han, J., Kamber, M., & Pei, J. (2011). *Data mining: Concepts and techniques* (3rd ed.). Recuperado de https://www.scholartext.com/book/88809627?_locale=fr

[6] Jurafsky, D., & Martin, J. H. (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall.

[7] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. Recuperado de http://www.deeplearningbook.org

[8] Bergholz, A., De Beer, J., Glahn, S., Moens, M.-F., Paaß, G., & Strobel, S. (2010). New filtering approaches for phishing email. *Journal of Computer Security, 18*(1), 7–35.

[9] Biggio, B., & Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition, 84*, 317–331. https://doi.org/10.1016/j.patcog.2018.07.023

[10] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. arXiv preprint arXiv:1810.04805. https://arxiv.org/abs/1810.04805

[11] Garera, S., Provos, N., Chew, M., & Rubin, A. D. (2007). A framework for detection and measurement of phishing attacks. En *Proceedings of the 2007 ACM Workshop on Recurring Malcode (WORM '07)* (pp. 1–8). Association for Computing Machinery. https://doi.org/10.1145/1314389.1314391

[12] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). *Explaining and harnessing adversarial examples*. arXiv preprint arXiv:1412.6572. https://arxiv.org/abs/1412.6572

[13] Lundberg, S., & Lee, S.-I. (2017). *A unified approach to interpreting model predictions*. arXiv preprint arXiv:1705.07874. https://arxiv.org/abs/1705.07874

[14] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space*. arXiv preprint arXiv:1301.3781. https://arxiv.org/abs/1301.3781

[15] Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. En Moschitti, A., Pang, B., & Daelemans, W. (Eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532–1543). Association for Computational Linguistics. https://aclanthology.org/D14-1162/

[16] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). *"Why should I trust you?": Explaining the predictions of any classifier*. arXiv preprint arXiv:1602.04938. https://arxiv.org/abs/1602.04938

[17] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023). *Attention is all you need*. arXiv preprint arXiv:1706.03762. https://arxiv.org/abs/1706.03762

[18] European Parliament and Council. (2016). Regulation (EU) 2016/679 (General Data Protection Regulation). *Official Journal of the European Union*.

[19] Elhoseny, M., Abdel-Salam, M., Khafaga, D. S., Aldakheel, E. A., & El-Hasnony, I. M. (2025). Robust Optimized Deep Learning-Based Phishing Detection Framework for semantic web systems using boosted triangular topology aggregation optimization. International Journal on Semantic Web and Information Systems, 21(1), 1-58. https://doi.org/10.4018/ijswis.388181

[20] Naser Abdullah Alam. (2024). Phishing Email Dataset [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DS/5074342

ANEXOS

- **Anexo A:** Código (notebook Colab) con pipeline completo (preprocesamiento, vectorización, entrenamiento y evaluación).
- **Anexo B:** Parámetros exactos de los modelos y resultados de las 5 corridas cross-validation (media y desviación estándar).
- **Anexo C:** Plots (Figura 1, Figura 2, Figura 3) y matrices de confusión.
- **Anexo D:** Tabla detallada de distribución de trabajo y cronograma (según requisitos del curso).