ISBN: 978-628-96613-2-3. ISSN: 2414-6390. Digital Object Identifier: https://dx.doi.org/10.18687/LEIRD2025.1.1.857

Performance of Deep Learning models for phishing detection in controlled environments: A systematic review

Gianmarco Antonio Rivera Carhuapoma[1]; Evelyn Elizabeth Ayala Ñiquen[2]; Carlos David Neyra Rivera[3]
[1,2,3]Universidad Tecnológica del Perú, Perú, *1221127@utp.edu.pe, c26915@utp.edu.pe, c29136@utp.edu.pe*

***Abstract*– Phishing remains one of the most persistent threats in the field of cybersecurity, driving the development of Deep Learning (DL) models to enhance their detection. This Systematic Literature Review aims to identify, categorize, and analyze DL models applied to phishing attack detection in controlled environments. Based on the analysis of recent studies, these models have demonstrated strong performance, particularly in metrics such as Accuracy and F1-score. The review also examines commonly used architectures such as RNNs, CNNs, and hybrid models, the types of attack vectors addressed, and the experimental conditions under which the models were evaluated. However, recurring limitations were identified, including the use of non-representative datasets, a lack of standardization in evaluation metrics and attack vectors, and limited validation in real-world scenarios. This review offers a structured synthesis of the current state of DL-based phishing detection and serves as a reference point for future research aimed at improving model performance and practical applicability.**

Keywords: Phishing, Deep Learning, Detection Performance, Controlled Environment

Desempeño de modelos de Deep Learning en la detección de ataques de phishing en entornos controlados: una revisión sistemática

Gianmarco Antonio Rivera Carhuapoma[1]; Evelyn Elizabeth Ayala Ñiquen[2]; Carlos David Neyra Rivera[3]
[1,2,3]Universidad Tecnológica del Perú, Perú, *1221127@utp.edu.pe, c26915@utp.edu.pe, c29136@utp.edu.pe*

Resumen– ***El phishing continúa siendo una de las amenazas más persistentes en el ámbito de la ciberseguridad, lo que ha impulsado el desarrollo de modelos de Deep Learning (DL) para mejorar su detección. Esta Revisión Sistemática de la Literatura tiene como objetivo identificar, clasificar y analizar los modelos de DL aplicados a la detección de ataques de phishing en entornos controlados. A partir del análisis de estudios recientes, estos modelos demostraron un desempeño sólido, particularmente en métricas como Accuracy y F1-score. Asimismo, se exploraron las arquitecturas más utilizadas como RNN, CNN y modelos híbridos, los vectores de ataque abordados y las condiciones experimentales de evaluación. Sin embargo, se identificaron limitaciones recurrentes como el uso de datos poco representativos, la falta de estandarización en vectores y métricas, y la escasa validación en contextos reales. Esta revisión organiza de forma estructurada la evidencia disponible sobre modelos de DL para detección de phishing y proporciona una base referencial para futuras investigaciones orientadas a mejorar su desempeño y aplicabilidad.***

Palabras clave: Phishing, Deep Learning, Desempeño del modelo, Entornos controlados

I. Introducción

El phishing ha logrado establecerse como una amenaza prioritaria en ciberseguridad, manteniendo su impacto a lo largo del tiempo, destacando por su capacidad para adaptarse y explotar vulnerabilidades humanas y tecnológicas [1]. Este tipo de ataque ha evolucionado en complejidad, abarcando desde correos electrónicos hasta sitios web maliciosos, enlaces en redes sociales y vectores más recientes como servicios en la nube o blockchain [2]. Para enfrentar esta amenaza, se ha explorado ampliamente el uso del Deep Learning (DL), que permite modelar relaciones complejas y detectar patrones anómalos en distintos tipos de entradas, como texto, estructura HTML o características de URLs [3]. Varios estudios han demostrado que los modelos DL superan a los métodos tradicionales de detección en métricas clave como Accuracy y F1-score [4].

Los vectores de ataque más comunes incluyen URLs maliciosas, estructuras web falsas, correos electrónicos fraudulentos y señales contextuales como formularios o scripts incrustados [5]. Esta diversidad ha motivado el desarrollo de modelos DL capaces de adaptarse a múltiples tipos de entrada, generando un panorama de soluciones técnicas altamente especializado [6]. Sin embargo, estos enfoques no son homogéneos. Algunos modelos se enfocan exclusivamente en un vector, como las URLs, mientras que otros integran contenido textual o visual, lo que influye en su capacidad de generalización [7]. Además, muchos estudios se han realizado en condiciones controladas, es decir, con conjuntos de datos preprocesados, balanceados y limitados en variabilidad, lo que ha permitido obtener altos niveles de rendimiento [8]. Por ejemplo, se han reportado modelos que alcanzan valores de F1-score y Precision muy sólidos, aunque estas cifras suelen lograrse bajo configuraciones experimentales específicas [9]. Bajo estas condiciones, la evaluación del rendimiento está fuertemente influenciada por condicionales como el tipo de datos, su limpieza, el equilibrio de clases y el entorno de prueba [10].

Dado el interés creciente por este campo, han surgido revisiones que intentan sintetizar los avances alcanzados, pero muchas de ellas se centran en aspectos aislados como un único vector, una familia de modelos o sin detallar las condiciones de evaluación [11]. Este panorama fragmentado complica la tarea de extraer conclusiones generalizables sobre la efectividad real de los modelos DL aplicados al phishing [12]. A pesar del entusiasmo por el uso del DL en el ámbito de la ciberseguridad, persisten limitaciones metodológicas que afectan la validez de los resultados y su comparabilidad. Entre ellas destaca la utilización no uniforme de métricas de evaluación: algunos estudios priorizan únicamente Accuracy, mientras que otros emplean F1-score, Recall o AUC sin justificar su elección, lo que dificulta establecer comparaciones objetivas entre modelos [13].

Es importante señalar que la calidad y representatividad de los datos utilizados sigue siendo un desafío crítico. La calidad y actualidad de los datos utilizados representa otro reto crítico, ya que muchos conjuntos no reflejan adecuadamente las condiciones modernas de ataque [14], [15]. Otro aspecto problemático es la frecuencia con la que se reportan resultados en entornos altamente controlados, sin una descripción clara sobre la replicabilidad de los experimentos o el impacto de los parámetros de entrenamiento [16]. Si bien algunos trabajos presentan métricas elevadas, estas suelen obtenerse sin validar los modelos en contextos abiertos o con nuevos conjuntos de prueba [17]. También se observa una escasa integración de múltiples vectores de ataque, lo que reduce el potencial de los modelos para ajustarse a nuevos contextos o situaciones reales donde los atacantes combinan técnicas diversas de evasión [18]. Finalmente, aunque existen iniciativas en áreas emergentes como detección de phishing en entornos de computación en la nube (cloud) o blockchain, estas siguen

siendo escasas, lo que expone la necesidad de seguir explorando su eficacia en plataformas modernas [19].

Dado que muchos estudios presentan resultados poco generalizables por evaluarse en condiciones artificiales, se vuelve necesaria una revisión sistemática que identifique y analice los modelos de DL utilizados en la detección de phishing, priorizando aquellos evaluados en entornos controlados, donde las condiciones experimentales están definidas y pueden ser replicadas [20].

El objetivo de esta revisión sistemática es analizar el rendimiento de los modelos de DL aplicados a la detección de ataques de phishing en entornos controlados. Para ello, se examinan las métricas de evaluación utilizadas, las condiciones experimentales en las que fueron aplicados y los factores que inciden en sus resultados. A través de este análisis estructurado, se busca sintetizar la evidencia disponible y facilitar comparaciones objetivas entre enfoques existentes, resaltando los elementos que influyen en su desempeño bajo condiciones reproducibles.

En cuanto a la organización del documento, este estudio ha sido estructurado con base en el enfoque PIOC (Problema, Intervención, Resultado y Contexto), lo que ha permitido formular preguntas de investigación precisas orientadas a evaluar el rendimiento de modelos de DL para detectar ataques de phishing en entornos controlados. La sección 2 detalla la metodología empleada, que incluye las estrategias de búsqueda aplicadas en bases de datos académicas como Scopus y Web of Science, así como el proceso de cribado y selección de artículos a través del protocolo PRISMA. A continuación, la sección 3 presenta los resultados organizados en función de cada componente del esquema PIOC, abordando tanto hallazgos cuantitativos como cualitativos relacionados con los tipos de ataques evaluados, las arquitecturas utilizadas, las métricas de desempeño reportadas y las características de los entornos simulados. Posteriormente, en la sección 4 se desarrolla una discusión interpretativa de estos resultados, poniendo énfasis en la efectividad de los modelos aplicados y los enfoques más recurrentes observados en la literatura. Finalmente, la sección 5 expone las conclusiones del estudio, sintetizando los principales hallazgos identificados a partir del análisis de los artículos revisados.

II. Metodología

A. Estrategia de Búsqueda

La presente revisión sistemática de literatura (RSL) se realizó mediante la estrategia de búsqueda PIOC (Población, Intervención, Resultados (Outputs) y Contexto) (Tabla I). Debido a ello, se formuló la siguiente pregunta de investigación: ¿Cuál es el desempeño de los modelos de DL en la detección de ataques de phishing en entornos controlados? De esta derivaron las subpreguntas por cada elemento PIOC (Tabla II).

TABLA I
Componentes PIOC

Acrónimo	Componente	Descripción
P	Problema	Ataques de tipo phishing
I	Intervención	Modelos de Deep Learning utilizados en la detección de phishing
O	Resultados	Desempeño de los modelos en la detección de phishing
C	Contexto	Entornos de simulación controlada

TABLA II
Preguntas de Investigación

Acrónimo	Pregunta
Pregunta Principal	¿Cuál es el desempeño de los modelos de DL en la detección de ataques de phishing en entornos controlados?
P	¿Qué tipos de ataques de phishing han sido abordados en los estudios revisados?
I	¿Cómo se han diseñado los modelos de DL aplicados a la detección de phishing?
O	¿Cómo ha sido evaluado el desempeño de los modelos en la detección de ataques de phishing?
C	¿Qué características definen los entornos de simulación utilizados en los estudios?

Con el objetivo de asegurar la obtención de publicaciones confiables y pertinentes, se identificaron palabras claves para la búsqueda de información (Tabla III), se realizó la búsqueda en las fuentes de información científica Scopus y Web of Science (WOS), durante el periodo entre marzo y mayo del 2025. Mediante la metodología PIOC se formuló una ecuación de búsqueda (Tabla IV), empleando los operadores lógicos para combinar las palabras clave.

TABLA III
Términos de Búsqueda

Valor	Descripción	Palabras clave	Keywords
P	Ataques Phishing	"phishing" OR "detección de phishing" OR "ataque de phishing" OR "antiphishing" OR "sitios web de phishing" OR "phishing por correo electrónico" OR "detección de fraude"	"phishing" OR "phishing detection" OR "phishing attack" OR "anti-phishing" OR "phishing websites" OR "email phishing" OR "fraud detection"
I	Modelos de Deep Learning	"aprendizaje profundo" OR "red neuronal profunda" OR "CNN" OR "RNN" OR "LSTM" OR "GRU" OR "BERT" OR "transformer" OR "red neuronal" OR "aprendizaje profundo"	"Deep learning" OR "deep neural network" OR "CNN" OR "RNN" OR "LSTM" OR "GRU" OR "BERT" OR "transformer" OR "neural network" OR "Deep learning"
O	Desempeño de los modelos	"clasificación" OR "rendimiento de detección" OR "precisión de detección" OR "falsos positivos" OR "recobrado" OR "f1-	"classification" OR "detection performance" OR "detection accuracy" OR "false positive" OR "recall" OR "f1-score" OR "precision"

		score" OR "precisión"	
C	Entornos de simulación controlada	"Conjunto de datos públicos" OR "conjunto de datos de referencia" OR "entorno controlado" OR "datos simulados" OR "clasificación desbalanceada" OR "conjunto de datos desbalanceado" OR "configuración experimental" OR "datos sintéticos"	"Public dataset" OR "benchmark dataset" OR "controlled environment" OR "simulated data" OR "unbalanced classification" OR "imbalanced dataset" OR "experimental setup" OR "synthetic data"

TABLA IV
ECUACIÓN DE BÚSQUEDA

SCOPUS	Web of Science
(TITLE-ABS-KEY ("phishing" OR "phishing detection" OR "phishing attack" OR "anti-phishing" OR "phishing websites" OR "email phishing" OR "fraud detection") AND TITLE-ABS-KEY ("deep learning" OR "deep neural network" OR "CNN" OR "RNN" OR "LSTM" OR "GRU" OR "BERT" OR "transformer" OR "neural network" OR "Deep-learning") AND TITLE-ABS-KEY ("classification" OR "detection performance" OR "detection accuracy" OR "false positive" OR "recall" OR "f1-score" OR "precision") AND TITLE-ABS-KEY ("public dataset" OR "benchmark dataset" OR "controlled environment" OR "simulated data" OR "unbalanced classification" OR "imbalanced dataset" OR "experimental setup" OR "synthetic data"))	"phishing" OR "phishing detection" OR "phishing attack" OR "anti-phishing" OR "phishing websites" OR "email phishing" OR "fraud detection" (All Fields) AND "deep learning" OR "deep neural network" OR "CNN" OR "RNN" OR "LSTM" OR "GRU" OR "BERT" OR "transformer" OR "neural network" OR "Deep-learning" (All Fields) AND "classification" OR "detection performance" OR "detection accuracy" OR "false positive" OR "recall" OR "f1-score" OR "precision" (All Fields) AND "public dataset" OR "benchmark dataset" OR "controlled environment" OR "simulated data" OR "unbalanced classification" OR "imbalanced dataset" OR "experimental setup" OR "synthetic data" (All Fields)

B. *Proceso de Cribado*

Para garantizar un proceso sistemático y transparente en la selección de artículos, se aplicó la metodología PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) [21]. Esta metodología permitió filtrar los estudios mediante criterios de inclusión y exclusión (Tabla V).

TABLA V
CRITERIOS DE INCLUSIÓN Y EXCLUSIÓN

CRITERIOS DE INCLUSIÓN	CRITERIOS DE EXCLUSIÓN
CI1: Los estudios enfocados a ciberataques de tipo phishing.	CE1: Estudios que no reporten métricas de evaluación de desempeño.
CI2: Los estudios deben aplicar metodologías de DL en la detección de phishing.	CE2: Estudios en idiomas diferentes al inglés y español.
CI3: Investigaciones realizadas en ambientes controlados.	CE3: Estudios diferentes a artículos originales y conference paper.

Se aplicó la metodología PRISMA para la identificación y selección de estudios relevantes. En total, se recuperaron 162 artículos de SCOPUS y 32 artículos de Web of Science (WOS). Se aplico un filtrado mediante la metodología PRISMA, una vez aplicados los criterios inclusión y exclusión definidos, finalmente se obtuvieron 15 artículos (Figura 1).

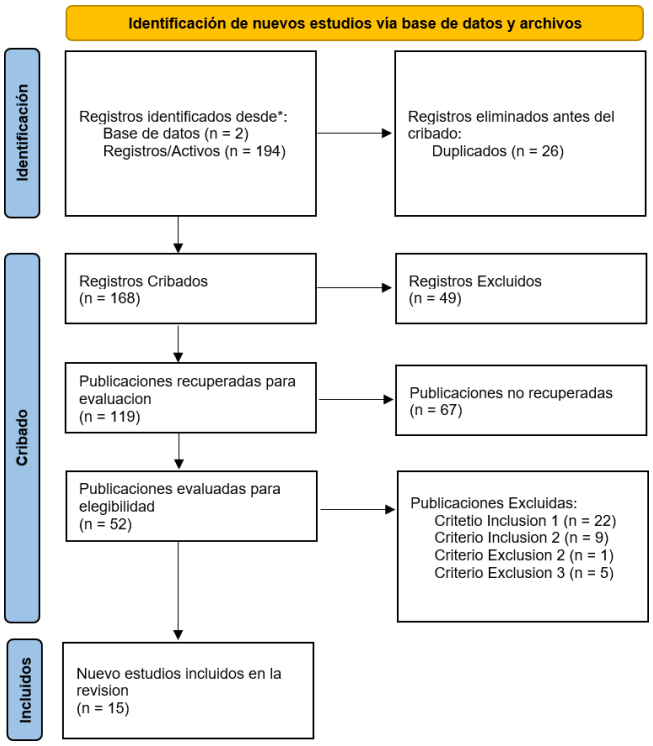

Fig. 1 DIAGRAMA PRISMA

III. RESULTADOS

Esta sección expone los resultados derivados del análisis de los estudios seleccionados. La información se organiza en cinco bloques. Se inicia con una caracterización general de los artículos revisados, seguida de los hallazgos relacionados con el tipo de ataque detectado, los modelos de detección empleados, el desempeño alcanzado y, finalmente, las condiciones en las que fueron evaluados. Esta organización permite ofrecer una visión sistemática y coherente de los elementos clave identificados en la literatura.

A. *Perfil bibliométrico de los estudios seleccionados*

Con el objetivo de contextualizar la literatura analizada, se ha elaborado una tabla bibliométrica que resume aspectos clave de los quince estudios incluidos (Tabla VI). Esta tabla presenta información como los autores, el título del artículo, la revista donde se publicó, el país de origen, el año de publicación y la cantidad de citas registradas hasta la fecha. Se observa que la mayoría de los trabajos fueron publicados en los años 2023 y 2024, los países con mayor representación en la producción científica revisada son India y China. Las citas acumuladas varían considerablemente, destacándose algunos

estudios con alto impacto en la comunidad académica, como el artículo de Aljofey A. [22] con 131 citas, mientras que otros corresponden a publicaciones recientes cuya visibilidad e impacto aún se encuentran en desarrollo.

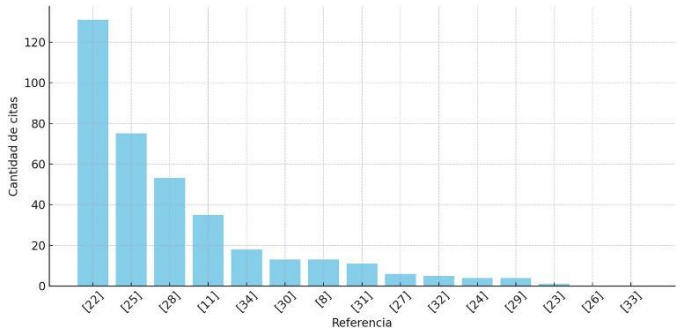

Fig. 2 CANTIDAD DE CITAS POR ESTUDIO INCLUIDO EN LA REVISIÓN

TABLA VI
DATOS BIBLIOMÉTRICOS DE LOS ESTUDIOS SELECCIONADOS

Ref.	Autores	Titulo	Revista	País	Año	N Citas
[23]	Elberri M.A.; Tokeşer Ü.; Rahebi J.; Lopez-Guede J.M.	A cyber defense system against phishing attacks with deep learning game theory and LSTM-CNN with African vulture optimization algorithm (AVOA)	International Journal of Information Security	India	2024	1
[8]	Altwaijry N.; Al-Turaiki I.; Alotaibi R.; Alakeel F.	Advancing Phishing Email Detection: A Comparative Study of Deep Learning Models	Sensors	India	2024	13
[22]	Aljofey A.; Jiang Q.; Qu Q.; Huang M.; Niyigena J.-P.	An effective phishing detection model based on character level convolutional neural network from URL	Electronics (Switzerland)	China	2020	131
[24]	Zaher M.A.; Eldakhly N.M.	Brain Storm Optimization with Long Short Term Memory Enabled Phishing Webpage Classification for Cybersecurity	Journal of Cybersecurity and Information Management	Egipto	2022	4
[11]	Ariyadasa S.; Fernando S.; Fernando S.	Combining Long-Term Recurrent Convolutional and Graph Convolutional Networks to Detect Phishing Sites Using URL and HTML	IEEE Access	Sri Lanka	2022	35
[25]	Zhu E.; Ju Y.; Chen Z.; Liu F.; Fang X.	DTOF-ANN: An Artificial Neural Network phishing detection model based on Decision Tree and Optimal Features	Applied Soft Computing Journal	China	2020	75
[26]	Wolert R.; Rawski M.	Email Phishing Detection with BLSTM and Word Embeddings	International Journal of Electronics and Telecommunications	India	2023	0
[27]	Zonyfar C.; Lee J.-B.; Kim J.-D.	HCNN-LSTM: Hybrid Convolutional Neural Network with Long Short-Term Memory Integrated for Legitimate Web Prediction	Journal of Web Engineering	India	2023	6
[28]	Bountakas P.; Xenakis C.	HELPHED: Hybrid Ensemble Learning PHishing Email Detection	Journal of Network and Computer Applications	India	2023	53
[29]	Alshahrani H.J.; Tarmissi K.; Yafoz A.; Mohamed A.; Motwakel A.; Yaseen I.; Abdelmageed A.A.; Mahzari M.	Improved Fruitf ly Optimization with Stacked Residual Deep Learning Based Email Classification	Intelligent Automation and Soft Computing	China	2023	4
[30]	Brindha R.; Nandagopal S.; Azath H.; Sathana V.; Joshi G.P.; Kim S.W.	Intelligent Deep Learning Based Cybersecurity Phishing Email Detection and Classification	Computers, Materials and Continua	India	2023	13
[31]	Qachfar F.Z.; Verma R.M.; Mukherjee A.	Leveraging Synthetic Data and PU Learning For Phishing Email Detection	CODASPY 2022 - Proceedings of the 12th ACM Conference on Data and Application Security and Privacy	India	2022	11
[32]	Dutta A.K.; Meyyappan T.; Qureshi B.; Alsanea M.; Abulfaraj A.W.; Al Faraj M.M.; Sait A.R.W.	Optimal Deep Belief Network Enabled Cybersecurity Phishing Email Classification	Computer Systems Science and Engineering	India	2023	5

[33]	Mahendru S.; Pandit T.	SecureNet: A Comparative Study of DeBERTa and Large Language Models for Phishing Detection	2024 IEEE 7th International Conference on Big Data and Artificial Intelligence, BDAI 2024	India	2024	0
[34]	Maci A.; Santorsola A.; Coscia A.; Iannacone A.	Unbalanced Web Phishing Classification through Deep Reinforcement Learning	Computers	China	2023	18

B. Tipos de phishing y vectores de entrada utilizados

Los quince estudios revisados abordan diversas formas de ataques de phishing, siendo el phishing por correo electrónico el tipo más frecuente, presente en ocho investigaciones (53.3 %) [23], [25], [27]–[32], mientras que siete estudios (46.7 %) se enfocan en el phishing web [8], [11], [22], [24], [26], [33], [34].

En los estudios centrados en phishing web, los vectores estructurales o sintácticos utilizan componentes como la cadena de la URL, atributos HTML, estructura del DOM, tiempo de carga de las páginas, formularios de ingreso y metainformación visual (por ejemplo, elementos gráficos, íconos o diseños que acompañan el contenido) como representación de entrada [11], [22]–[25], [27], [32], [34]. En contraste, las investigaciones enfocadas en el phishing por correo electrónico optan por vectores de tipo textual o semántico, los cuales elaboran en base a la información presente en los encabezados, enlaces incrustados y cuerpo del mensaje, utilizando técnicas como *embeddings* (representaciones numéricas de texto o URLs en espacios vectoriales), codificación contextual y análisis secuencial de tokens [8], [28]–[30], [32], [33].

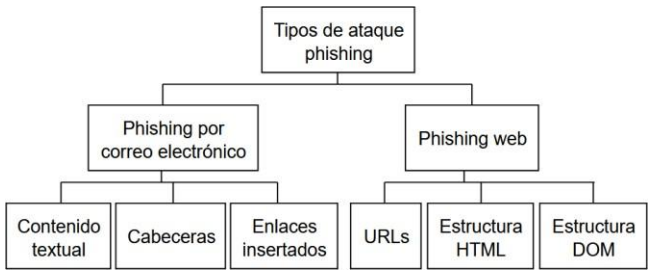

Fig. 3 TIPOS DE PHISHING Y SUS VECTORES

Los vectores de ataque más empleados en los estudios revisados comprenden tanto características estructurales como textuales. Dentro de los enfoques estructurales, la cadena de URL fue el subtipo más utilizado, presente en ocho estudios (53.3 %) [11], [22]–[25], [27], [29], [32]. Le sigue el uso de atributos y estructura HTML, reportado en seis investigaciones (40 %) [11], [23], [24], [26], [27], [29], y la estructura del DOM, empleada en cuatro artículos (26.7 %) [11], [24], [27], [29]. Los formularios de ingreso fueron considerados en tres estudios (20 %) [11], [23], [27], y el tiempo de carga o latencia de páginas se identificó en dos investigaciones (13.3 %) [11], [29]. En cuanto a los vectores textuales, el cuerpo del mensaje fue utilizado en siete estudios (46.7 %) [8], [28]–[33], mientras que los encabezados del correo y los enlaces incrustados aparecieron en cuatro estudios cada uno (26.7 %) [8], [29], [31], [33]. Además, se emplearon representaciones vectoriales como embeddings semánticos en cinco investigaciones (33.3 %) [8], [28], [29], [32], [33], y técnicas de análisis secuencial o por tokens en cuatro artículos (26.7 %) [8], [28], [30], [32].

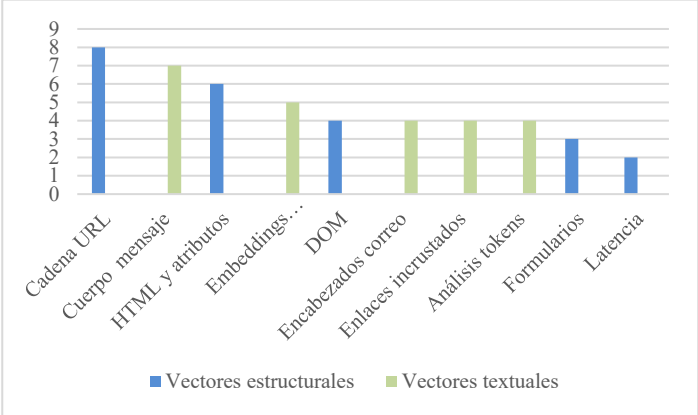

Fig. 4 FRECUENCIA DE USO POR VECTOR DE ATAQUE

C. Arquitecturas de Deep Learning empleadas.

Los estudios analizados implementan una variedad de modelos de DL para detectar ataques de phishing, con predominio de arquitecturas recurrentes y convolucionales. Las redes neuronales recurrentes (RNN) y sus variantes LSTM y BiLSTM fueron las más empleadas, presentes en diez estudios [8], [23], [24]–[27], [30], [32], [34]. Las redes convolucionales (CNN), utilizadas de forma individual o combinadas con RNN, aparecieron en nueve investigaciones [23], [24]–[29], [34]. En cinco estudios se propusieron arquitecturas híbridas CNN-LSTM [23], [25]–[28], y tres trabajos recurrieron a enfoques optimizados con algoritmos bioinspirados como AVOA, BSO o Fruitfly [23], [24], [29]. Modelos como BLSTM se aplicaron en dos estudios [8], [26], mientras que redes neuronales artificiales (ANN) se utilizó en un artículo [25]. Otros enfoques incluyeron redes profundas como Deep Belief Networks (DBN) [32], modelos basados transformers como DeBERTa [33], aprendizaje por refuerzo con Deep Q-Network (DQN) [34], y aprendizaje con datos positivos y no etiquetados (PU learning) [31]. Finalmente, un estudio incorporó técnicas ensemble (técnica que combina varios modelos para mejorar la precisión) para la combinación de modelos [28].

De los quince estudios revisados, diez (66.7 %) implementan variantes de redes neuronales recurrentes (RNN), incluyendo LSTM y BiLSTM [8], [23], [24]–[27], [30], [32], [34], y nueve (60 %) emplean arquitecturas convolucionales (CNN), ya sean utilizadas solas o en conjunto con otras como las RNN [23], [24]–[29], [34]. Cinco estudios (33.3 %) presentan arquitecturas híbridas que combinan CNN con RNN [23], [25]– [28], mientras que tres (20 %) utilizan redes neuronales artificiales (ANN) o perceptrones multicapa [25], [28], [31]. Dos investigaciones (13.3 %) adoptan Deep Belief Networks (DBN) [30], [32], y tres (20 %) incorporan algoritmos de optimización bioinspirados como AVOA, BSO o Fruitfly [23], [24], [29]. Además, se identificó un estudio basado en aprendizaje por refuerzo (6.7 %) [34], uno que aplica PU learning (6.7 %) [31] y otro que implementa un modelo de tipo transformer (DeBERTa, 6.7 %) [33].

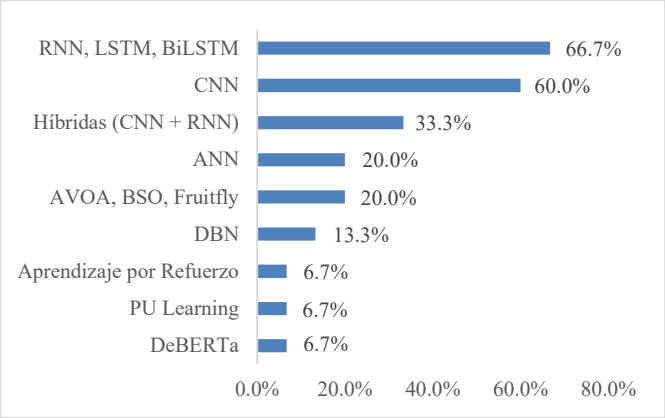

Fig. 5 DISTRIBUCIÓN DE MODELOS DL UTILIZADOS

Los estudios emplean datos diferenciados según el tipo de ataque de phishing abordado. Para phishing web, se utilizan conjuntos compuestos por URLs maliciosas y legítimas, estructuras HTML, y contenido visual de sitios reales o simulados, recolectados de fuentes como PhishTank, OpenPhish, datos de Alexa, UCI Machine Learning Repository, o mediante técnicas de scraping personalizado [11], [23]–[25], [26]–[29], [31], [33]. En el caso de phishing por correo electrónico, se emplean bases de mensajes legítimos y maliciosos, extraídos de fuentes como Enron, Nazario, SpamAssassin, o datasets privados compilados por los autores [8], [30], [32], [34]. Algunas investigaciones utilizan datos generados sintéticamente o amplificados mediante técnicas como PU learning, con el objetivo de mitigar el desbalance o suplir la escasez de ejemplos en clases minoritarias [25], [28], [31]. A pesar de la variedad de fuentes, varios estudios identifican limitaciones significativas, entre ellas la antigüedad de los datos, el sesgo de las colecciones públicas y la baja representatividad de ataques modernos o dinámicos [23]–[24], [26], [27], [30], [32], [33].

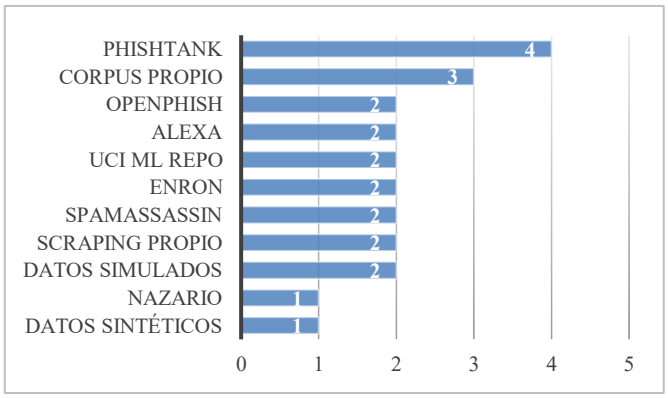

Fig. 6 ORIGEN DE DATOS USADOS PARA EL ENTRENAMIENTO

De los quince estudios revisados, siete (46.7 %) utilizaron exclusivamente datos públicos [8], [23], [24], [29], [31]–[33], seis (40 %) emplearon datasets privados [22], [25]–[28], [34], y dos (13.3 %) combinaron fuentes públicas y privadas [11], [30]. Asimismo, nueve investigaciones (60 %) reportaron limitaciones en los datos utilizados [8], [23]–[24], [26], [27], [30], [32], [33].

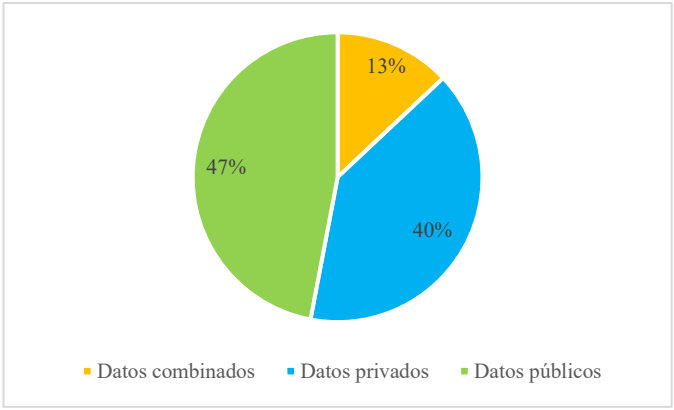

Fig. 7 PROPORCIÓN DE ESTUDIOS SEGÚN EL ORIGEN DE LOS DATOS UTILIZADOS

D. Métricas de evaluación y desempeño de los modelos.

De los quince estudios analizados, la métrica Accuracy fue utilizada en todos los estudios como indicador principal de desempeño, al igual que la métrica f1-score, empleada en los quince artículos revisados [8], [11], [22]–[34]. La métrica Recall también estuvo presente en la totalidad de los estudios, aunque con distintos niveles de profundidad en su interpretación [8], [11], [22]–[34]. Por su parte, la métrica Precision fue reportada en catorce estudios, con excepción del artículo [22], que no la incluyó explícitamente [8], [23], [24]–[34]. La métrica AUC (Área Bajo la Curva ROC) fue utilizada en seis investigaciones, reflejando un interés más específico en el poder discriminativo del modelo [23], [25], [27]–[29], [33]. Finalmente, cinco artículos compararon explícitamente modelos de DL frente a modelos tradicionales de ML

utilizando estas métricas para destacar mejoras de rendimiento [8], [23], [24], [25], [31].

En cuanto a los valores obtenidos, los modelos evaluados mostraron un rendimiento destacado. Por ejemplo, el modelo HELPHED alcanzó un Accuracy de 98.9 %, un F1-score de 98.8 % y un AUC de 0.991 [35], mientras que DTOF-ANN reportó un Accuracy de 98.7 % y un F1-score de 98.4 % [25]. El modelo basado en Fruitfly Optimization logró un accuracy de 98.4 % y un F1-score de 98.3 % [29], mientras que un modelo basado en CNN alcanzó un Accuracy de 98.4 % con una Precision de 98.3 % [27]. Por su parte, el estudio AVOA obtuvo un valor AUC de 0.986 [23], y el modelo con arquitectura DeBERTa reportó un AUC de 0.987 [33].

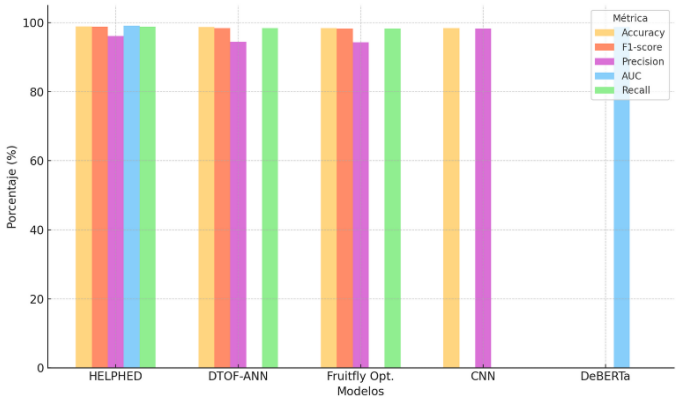

Fig. 8 COMPARACIÓN DEL DESEMPEÑO DE MODELOS DE DETECCIÓN DE PHISHING EN MÉTRICAS CLAVE DE EVALUACIÓN

E. Entornos de simulación y condiciones experimentales.

Los estudios revisados emplearon una variedad de herramientas, entornos para implementar y evaluar sus modelos en condiciones controladas. Las plataformas más recurrentes fueron Python junto con bibliotecas especializadas como TensorFlow y Keras, facilitando la construcción y entrenamiento de arquitecturas de DL [11], [22]–[25], [27]–[29], [31], [33]. Algunos de estos sistemas integraron flujos automatizados que incluían scraping web, normalización de datos y validación cruzada mediante módulos internos [11], [22], [23], [29]. Sin embargo, solo un conjunto limitado de investigaciones justificó explícitamente la validez de sus entornos, argumentando que sus configuraciones emulan patrones reales de ataque, permiten la generalización de resultados o están diseñadas para replicarse en distintos contextos [11], [23], [25], [28], [30], [32].

En los estudios revisados, los entornos de simulación controlados fueron diseñados para replicar el comportamiento operativo de ataques de phishing en condiciones reproducibles. En el caso del phishing web, se implementaron mecanismos de lectura automatizada de sitios mediante técnicas como web scraping, análisis de estructuras HTML, extracción de características desde certificados digitales y procesamiento de URLs legítimas y maliciosas obtenidas de fuentes como PhishTank, OpenPhish, Alexa o el UCI Repository [11], [22]–[25], [27]–[29], [31], [33]. En algunos estudios, estas URLs fueron introducidas en el sistema a través de rutinas que simulaban sesiones de navegación o evaluaciones en lotes. Para los escenarios centrados en phishing por correo electrónico, se configuraron entornos con bandejas de entrada simuladas, sistemas de recepción asincrónica de correos o almacenamiento local de mensajes para su procesamiento posterior, utilizando datasets como Enron, Nazario, SpamAssassin o colecciones privadas [8], [26], [30], [32], [34]. Algunos trabajos incorporaron además técnicas de generación de datos sintéticos o métodos de aprendizaje con ejemplos no etiquetados (PU learning), para representar contextos con clases desbalanceadas o escasa disponibilidad de datos reales [25], [28], [31].

De los quince estudios analizados, trece (86.7 %) implementaron sus modelos en entornos completamente simulados [8], [23]–[29], [31]–[34], mientras que dos estudios (13.3 %) incorporaron parcialmente datos operacionales o condiciones derivadas de infraestructura real [11], [30]. En cuanto al detalle del entorno, once artículos (73.3 %) especificaron características de la simulación, tales como generación automatizada de muestras, parámetros controlados o condiciones de laboratorio [23]–[26], [28], [29], [31]–[33], mientras que los cuatro restantes (26.7 %) no describen claramente las condiciones internas utilizadas [8], [27], [30], [34].

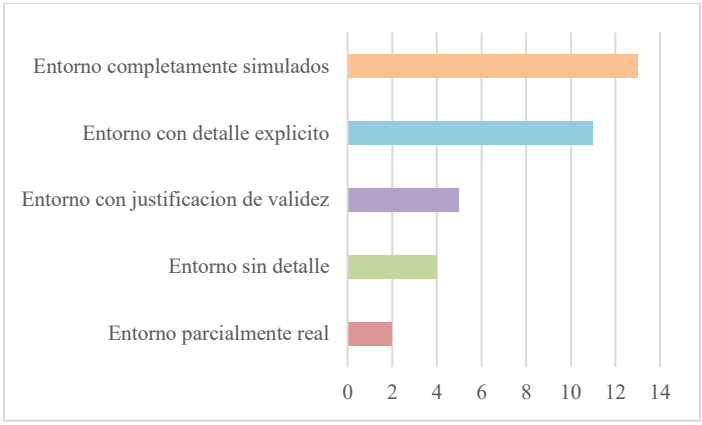

Fig. 9 CRITERIOS DE LOS ENTORNOS DE SIMULACIÓN EN LOS ESTUDIOS REVISADOS

IV. DISCUSIÓN

El análisis comparativo de los estudios revisados muestra que, si bien el phishing por correo electrónico continúa siendo el foco principal de investigación, como señalan Bountakas y Xenakis [28] y Brindha et al. [30], el interés creciente en el phishing web refleja la respuesta de los investigadores hacia la evolución de las tácticas de los atacantes, como destacan Ariyadasa et al. [11] y Aljofey et al. [22]. Este patrón sugiere que los futuros modelos deberán adaptarse a escenarios con

vectores mixtos para mantener su efectividad. En comparación con trabajos que se limitan a un solo tipo de vector, se observa un avance en investigaciones que integran vectores estructurales y textuales en un mismo modelo, lo que permite mejorar la detección en escenarios más complejos y variados [11], [28], lo que podría impulsar la investigación hacia marcos unificados que gestionen distintos tipos de phishing de forma simultánea.

También se identificó una alta variabilidad en la selección de vectores de entrada según el tipo de phishing abordado. Esta inconsistencia metodológica limita la comparación objetiva entre estudios y obstaculiza el desarrollo de marcos generalizables para la detección de ataques. En contraste, autores como Elberri et al. [23] y Mahendru y Pandit [33] subrayan la importancia de desarrollar marcos que permitan seleccionar y evaluar los vectores de forma consistente, facilitando la comparación de resultados y la mejora continua de los modelos frente a ataques cada vez más sofisticados. En conjunto, estos hallazgos sugieren que abordar la diversidad de vectores de phishing podría ser no solo una necesidad técnica, sino también un requisito clave para mantener la efectividad y fortalecer la resiliencia de los sistemas de detección frente a amenazas en constante evolución.

El análisis de los estudios revisados demuestra que las arquitecturas basadas en RNN y CNN continúan siendo la base principal en la detección de phishing, lo que coincide con trabajos como los de Elberri et al. [23] y Ariyadasa et al. [11], quienes destacan la capacidad de estas redes para extraer patrones complejos de texto y estructuras web. En comparación, enfoques híbridos como los presentados por Bountakas y Xenakis [28] y Zonyfar et al. [27] evidencian avances importantes al combinar CNN y LSTM, mejorando la capacidad de los modelos para adaptarse a ataques más variados. También se encontró que, aunque modelos recientes como los propuestos por Mahendru y Pandit [33] basados en transformers, o por Maci et al. [34] con aprendizaje por refuerzo, muestran resultados prometedores, pero su adopción es limitada frente a arquitecturas tradicionales, lo que podría deberse a la complejidad de implementar estos modelos más avanzados y a la falta de estudios que demuestren consistentemente su superioridad en escenarios prácticos. A diferencia de investigaciones que solo usan datos generados sintéticamente, varios estudios utilizan conjuntos públicos como PhishTank [23], mientras que otros recurren a datasets privados o combinan ambas fuentes, lo que refleja un panorama heterogéneo que dificulta la comparación de resultados entre modelos. En contraste, varios autores como Elberri et al. [23] y Brindha et al. [30] advierten sobre el uso de datos desactualizados o poco variados impidiendo que los modelos puedan validarse frente a amenazas recientes, lo que sugiere la necesidad de contar con datasets dinámicos y actualizados que reflejen el panorama actual del phishing. De esta manera, se enfatiza la necesidad de validar los modelos con datos más recientes y diversos, para garantizar su efectividad frente a tácticas de phishing en constante evolución.

Los estudios revisados muestran que métricas como Accuracy, F1-score y Recall son las más comúnmente utilizadas en la evaluación del rendimiento de los modelos de DL en la detección de phishing, lo que refleja un consenso sobre la importancia de medir tanto el acierto general como el equilibrio entre precisión y sensibilidad [23], [25], [28]. En comparación, métricas como el AUC fueron reportadas solo en algunos trabajos [23], [25], [33], lo que evidencia que no todos los estudios priorizan analizar el poder discriminativo de los modelos, limitando la posibilidad de compararlos a profundidad. También se encontró que modelos como el de Bountakas y Xenakis [28] y el DTOF-ANN propuesto por Zhu et al. [25] alcanzaron valores de Accuracy y F1-score superiores al 98 %, lo que demuestra un rendimiento destacado en entornos controlados. A diferencia de lo reportado en estudios que usaron datos más recientes y variados [8], señalando que estos altos valores podrían no trasladarse a escenarios reales debido a la antigüedad de los datasets o su falta de representatividad frente a ataques actuales [23], [22], [33], lo que reafirma la necesidad de interpretar estos resultados según las condiciones experimentales establecidas por las que fueron obtenidos, sin extender sus implicancias a contextos abiertos o reales.

La revisión de los estudios evidencia que la mayoría de los modelos de DL para la detección de phishing fueron evaluados en entornos simulados construidos con herramientas como Python y bibliotecas especializadas, lo que coincide con enfoques reportados por Ariyadasa et al. [11], Zhu et al. [25], Zonyfar et al. [27] y Mahendru y Pandit [33], quienes destacan la flexibilidad de estas plataformas para desarrollar experimentos en condiciones controladas. En comparación, trabajos como los de Ariyadasa et al. [11], Aljofey et al. [22] y Zhu et al. [25] integraron flujos automatizados incluyendo scraping, normalización y validación cruzada que permitieron construir simulaciones más cercanas al comportamiento real de los ataques. Asimismo, estudios como los de Elberri et al. [23], Bountakas y Xenakis [28] y Brindha et al. [30] justificaron la validez de sus entornos, argumentando que sus configuraciones emulan patrones típicos de phishing o facilitan la replicabilidad. En contraste, investigaciones como las de Wolert y Rawski [27] y Maci et al. [34] no detallaron las condiciones internas de sus simulaciones, lo que limita la evaluación de la robustez y reproducibilidad de sus modelos.

Además de la infraestructura técnica, varios estudios también simularon flujos de datos realistas como parte de sus entornos controlados. Para el caso de phishing web, se utilizaron técnicas como análisis de estructuras HTML, lectura automatizada de certificados digitales y procesamiento de URLs legítimas y maliciosas obtenidas de fuentes como

PhishTank o Alexa [11], [22]. Para phishing por correo electrónico, se configuraron bandejas de entrada simuladas o sistemas de procesamiento asincrónico, utilizando datasets públicos como Enron y SpamAssassin [26], [30]. Además, se emplearon técnicas como PU learning o la generación de datos sintéticos para representar escenarios con clases desbalanceadas o escasa disponibilidad de datos reales [25], [28], [31]. Estas estrategias permitieron evaluar los modelos bajo condiciones reproducibles pero alineadas con desafíos operativos reales.

V. Conclusiones

Esta revisión sistemática analizó el rendimiento de modelos de Deep Learning (DL) aplicados a la detección de ataques de phishing en entornos controlados, a partir de 15 estudios seleccionados. Los resultados evidencian que los modelos DL presentan un desempeño destacado en métricas como Accuracy, F1-score y Recall, superando en varios casos el 98 %. Se destacan arquitecturas como RNN, LSTM, CNN y sus variantes híbridas, así como el uso de optimizadores bioinspirados. Respecto a los vectores de entrada, los estudios abordaron tanto phishing web como por correo electrónico, empleando URLs, estructuras HTML, encabezados y cuerpos de mensajes, además de representaciones numéricas como embeddings. No obstante, se identificaron limitaciones metodológicas que afectan la generalización de los resultados: uso de datasets desactualizados, falta de estandarización en métricas y predominio de evaluaciones en entornos completamente simulados. Estos factores limitan la extrapolación a escenarios reales y dificultan comparaciones objetivas entre enfoques. En ese sentido, esta revisión proporciona una base estructurada sobre el estado actual del campo y plantea la necesidad de avanzar hacia modelos validados en contextos operativos, con datos más diversos, técnicas de evaluación consistentes y arquitecturas que integren múltiples vectores de ataque. Fortalecer estos aspectos permitirá mejorar la robustez, adaptabilidad y aplicabilidad de las soluciones DL frente a escenarios de phishing cada vez más complejos y dinámicos.

Referencias

[1] C. Catal, G. Giray, B. Tekinerdogan, S. Kumar, and S. Shukla, "Applications of deep learning for phishing detection: a systematic literature review," Knowl Inf Syst, vol. 64, no. 6, pp. 1457–1500, 2022, doi: 10.1007/s10115-022-01672-x.

[2] H. F. Atlam and O. Oluwatimilehin, "Business Email Compromise Phishing Detection Based on Machine Learning: A Systematic Literature Review," Electronics (Switzerland), vol. 12, no. 1, 2023, doi: 10.3390/electronics12010042.

[3] E. Benavides, W. Fuertes, S. Sanchez, and M. Sanchez, Classification of Phishing Attack Solutions by Employing Deep Learning Techniques: A Systematic Literature Review, vol. 152. 2020. doi: 10.1007/978-981-13-9155-2_5.

[4] M. Somesha and A. R. Pais, "Classification of Phishing Email Using Word Embedding and Machine Learning Techniques," Journal of Cyber Security and Mobility, vol. 11, no. 3, pp. 279–320, 2022, doi: 10.13052/jcsm2245-1439.1131.

[5] B. Wei et al., "A deep-learning-driven light-weight phishing detection sensor," Sensors (Switzerland), vol. 19, no. 19, 2019, doi: 10.3390/s19194258.

[6] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "A Systematic Literature Review on Phishing Email Detection Using Natural Language Processing Techniques," IEEE Access, vol. 10, pp. 65703–65727, 2022, doi: 10.1109/ACCESS.2022.3183083.

[7] A. Safi and S. Singh, "A systematic literature review on phishing website detection techniques," Journal of King Saud University - Computer and Information Sciences, vol. 35, no. 2, pp. 590–611, 2023, doi: 10.1016/j.jksuci.2023.01.004.

[8] N. Altwaijry, I. Al-Turaiki, R. Alotaibi, and F. Alakeel, "Advancing Phishing Email Detection: A Comparative Study of Deep Learning Models," Sensors, vol. 24, no. 7, 2024, doi: 10.3390/s24072077.

[9] M. K. Prabakaran, P. Meenakshi Sundaram, and A. D. Chandrasekar, "An enhanced deep learning-based phishing detection mechanism to effectively identify malicious URLs using variational autoencoders," IET Inf Secur, vol. 17, no. 3, pp. 423–440, 2023, doi: 10.1049/ise2.12106.

[10] A. Mughaid, S. AlZu'bi, A. Hnaif, S. Taamneh, A. Alnajjar, and E. A. Elsoud, "An intelligent cyber security phishing detection system using deep learning techniques," Cluster Comput, vol. 25, no. 6, pp. 3819–3828, 2022, doi: 10.1007/s10586-022-03604-4.

[11] S. Ariyadasa, S. Fernando, and S. Fernando, "Combining Long-Term Recurrent Convolutional and Graph Convolutional Networks to Detect Phishing Sites Using URL and HTML," IEEE Access, vol. 10, pp. 82355–82375, 2022, doi: 10.1109/ACCESS.2022.3196018.

[12] S.-J. Bu and S.-B. Cho, "Deep character-level anomaly detection based on a convolutional autoencoder for zero-day phishing url detection," Electronics (Switzerland), vol. 10, no. 12, 2021, doi: 10.3390/electronics10121492.

[13] O. K. Sahingoz, E. Buber, and E. Kugu, "DEPHIDES: Deep Learning Based Phishing Detection System," IEEE Access, vol. 12, pp. 8052–8070, 2024, doi: 10.1109/ACCESS.2024.3352629.

[14] B. Jha, M. Atre, and A. Rao, "Detecting Cloud-Based Phishing Attacks by Combining Deep Learning Models," in Proceedings - 2022 IEEE 4th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications, TPS-ISA 2022, 2022, pp. 130–139. doi: 10.1109/TPS-ISA56441.2022.00026.

[15] K. Joshi et al., "Machine-Learning Techniques for Predicting Phishing Attacks in Blockchain Networks: A Comparative Study," Algorithms, vol. 16, no. 8, 2023, doi: 10.3390/a16080366.

[16] S. Das Guptta, K. T. Shahriar, H. Alqahtani, D. Alsalman, and I. H. Sarker, "Modeling Hybrid Feature-Based Phishing Websites Detection Using Machine Learning Techniques," Annals of Data Science, vol. 11, no. 1, pp. 217–242, 2024, doi: 10.1007/s40745-022-00379-8.

[17] R. Zieni, L. Massari, and M. C. Calzarossa, "Phishing or Not Phishing? A Survey on the Detection of Phishing Websites," IEEE Access, vol. 11, pp. 18499–18519, 2023, doi: 10.1109/ACCESS.2023.3247135.

[18] F. Rashid, B. Doyle, S. C. Han, and S. Seneviratne, "Phishing URL detection generalisation using Unsupervised Domain Adaptation," Computer Networks, vol. 245, 2024, doi: 10.1016/j.comnet.2024.110398.

[19] M. M. Saeed and Z. A. Aghbari, "Survey on Deep Learning Approaches for Detection of Email Security Threat," Computers, Materials and Continua, vol. 77, no. 1, pp. 325–348, 2023, doi: 10.32604/cmc.2023.036894.

[20] R. J. van Geest, G. Cascavilla, J. Hulstijn, and N. Zannone, "The applicability of a hybrid framework for automated phishing detection," Comput Secur, vol. 139, 2024, doi: 10.1016/j.cose.2024.103736.

[21] M. J. Page et al., "The PRISMA 2020 statement: An updated guideline for reporting systematic reviews," 2021. doi: 10.1136/bmj.n71

[22] A. Aljofey, Q. Jiang, Q. Qu, M. Huang, and J.-P. Niyigena, "An effective phishing detection model based on character level convolutional neural network from URL," Electronics (Switzerland), vol. 9, no. 9, pp. 1–24, 2020, doi: 10.3390/electronics9091514.

[23] M. A. Elberri, Ü. Tokeşer, J. Rahebi, and J. M. Lopez-Guede, "A cyber defense system against phishing attacks with deep learning game theory and LSTM-CNN with African vulture optimization algorithm (AVOA)," Int J Inf Secur, vol. 23, no. 4, pp. 2583–2606, 2024, doi: 10.1007/s10207-024-00851-x.

[24] M. A. Zaher and N. M. Eldakhly, "Brain Storm Optimization with Long Short Term Memory Enabled Phishing Webpage Classification for Cybersecurity," Journal of Cybersecurity and Information Management, vol. 9, no. 2, pp. 20–30, 2022, doi: 10.54216/JCIM.090202.

[25] E. Zhu, Y. Ju, Z. Chen, F. Liu, and X. Fang, "DTOF-ANN: An Artificial

Neural Network phishing detection model based on Decision Tree and Optimal Features," Applied Soft Computing Journal, vol. 95, 2020, doi: 10.1016/j.asoc.2020.106505.
[26] R. Wolert and M. Rawski, "Email Phishing Detection with BLSTM and Word Embeddings," International Journal of Electronics and Telecommunications, vol. 69, no. 3, pp. 485–491, 2023, doi: 10.24425/ijet.2023.146496.
[27] C. Zonyfar, J.-B. Lee, and J.-D. Kim, "HCNN-LSTM: Hybrid Convolutional Neural Network with Long Short-Term Memory Integrated for Legitimate Web Prediction," Journal of Web Engineering, vol. 22, no. 5, pp. 757–782, 2023, doi: 10.13052/jwe1540-9589.2251.
[28] P. Bountakas and C. Xenakis, "HELPHED: Hybrid Ensemble Learning PHishing Email Detection," Journal of Network and Computer Applications, vol. 210, 2023, doi: 10.1016/j.jnca.2022.103545.
[29] H. J. Alshahrani et al., "Improved Fruitf ly Optimization with Stacked Residual Deep Learning Based Email Classification," Intelligent Automation and Soft Computing, vol. 36, no. 3, pp. 3139–3155, 2023, doi: 10.32604/iasc.2023.034841.
[30] R. Brindha, S. Nandagopal, H. Azath, V. Sathana, G. P. Joshi, and S. W. Kim, "Intelligent Deep Learning Based Cybersecurity Phishing Email Detection and Classification," Computers, Materials and Continua, vol. 74, no. 3, pp. 5901–5914, 2023, doi: 10.32604/cmc.2023.030784.
[31] F. Z. Qachfar, R. M. Verma, and A. Mukherjee, "Leveraging Synthetic Data and PU Learning For Phishing Email Detection," in CODASPY 2022 - Proceedings of the 12th ACM Conference on Data and Application Security and Privacy, 2022, pp. 29–40. doi: 10.1145/3508398.3511524.
[32] A. K. Dutta et al., "Optimal Deep Belief Network Enabled Cybersecurity Phishing Email Classification," Computer Systems Science and Engineering, vol. 44, no. 3, pp. 2701–2713, 2023, doi: 10.32604/csse.2023.028984.
[33] S. Mahendru and T. Pandit, "SecureNet: A Comparative Study of DeBERTa and Large Language Models for Phishing Detection," in 2024 IEEE 7th International Conference on Big Data and Artificial Intelligence, BDAI 2024, 2024, pp. 160–169. doi: 10.1109/BDAI62182.2024.10692765.
[34] A. Maci, A. Santorsola, A. Coscia, and A. Iannacone, "Unbalanced Web Phishing Classification through Deep Reinforcement Learning," Computers, vol. 12, no. 6, 2023, doi: 10.3390/computers12060118.