Comprehensive Journal of Science
Volume (10), Issue (39), (March, 2026)
ISSN: 3014-6266

مجلة العلوم الشاملة
المجلد (10) العدد 39 مارس 2026
ردمد: 3014-6266

A comparative Analysis on Stacked Hybrid Intelligence: A Multi-Paradigm Machine Learning Framework for Robust Phishing URL Detection

Magdah Othman Mohammed Osman[1], Llahm Omar Faraj Ben Dalla[2], Tarik A. Rashid[3], Magda Juma Shuayb Albaraesi[4], Mohamed Ali Mohamed EL-sseid[5], Abdussalam Ali Ahmed[6], Yasser Fathi Nassar[7], Abdulgader Alsharif[8]

[1]Systems analysis and programming Department, Higher Institute of Science and Technology, Ajdabiya , Libya

[2]Department of Electrical and Electronics Engineering, Ankara Yildirim Beyazit University, Türkiye

[2]Computer Engineering, College of Technical Science, Sebha, Libya

[3]Artificial Intelligence and Innovation Centre University of Kurdistan Hewler, Erbil, Iraq

[4]Business administration Department , Kambut Higher Institute for Administrative and Financial Sciences, Tobruk, Libya

[5]Department of Software Engineering, Ankara Bilim University, Türkiye

[6]Mechanical and Industrial Engineering Department, Bani Waleed University, Libya

[7]*Mechanical and Renewable Energy Engineering Dept., Faculty of Engineering, Wadi Alshatti University, Brack, Libya*

[8]*Department of Electric and Electronic Engineering, College of Technical Sciences Sebha, Libya*

Majdaalhasady@histaj.edu.ly[1], *llahmomarfaraj77@ctss.edu.ly[2], llahmomarfaraj77@ aybu.edu.tr[2], tarik.ahmed@ukh.edu.krd[3]*, Jumamagda@gmail.com[4], Moh200512@Bilim.edu.tr[5], , abdussalam.a.ahmed@gmail.com[6] , *y.nassar@wau.edu.ly* [7]**, alsharif@ctss.edu.ly[8]**

https://orcid.org/0000-0002-2122-6901[1], https://orcid.org/my-orcid 0009-0008-7624-7567[2], https://orcid.org/0000-0002-8661-258X[3] , https://orcid.org/0009-0007-1307-86236[5]**,** **https://orcid.org/0000-0002-9221-2902[6],** https://orcid.org/0000-0002-9675-83045[6]**,** **https://orcid.org/0000-0003-3515-4168[8]**

تاريخ الاستلام:2026/01/03 –تاريخ المراجعة: 2026/01/23– تاريخ القبول: 2026/01/30– تاريخ للنشر :2026 /02/25

Abstract

The escalating sophistication of phishing campaigns necessitates detection frameworks that transcend conventional single-paradigm classifiers. This study introduces a comprehensive evaluation of six distinct machine learning architectures spanning traditional ensemble methods, kernel-based classifiers, and deep neural architectures applied to the PhiUSIIL dataset

comprising 235,795 URLs with rich lexical and host-based features. This research implement Random Forest (RF), XGBoost, character-level Convolutional Neural Networks (CNN), Bidirectional Long Short-Term Memory (Bi-LSTM) networks, Support Vector Machines (SVM) with Radial Basis Function kernel, and a novel stacked hybrid ensemble integrating RF, XGBoost, and SVM through meta-learning. Rigorous 5-fold cross-validation demonstrates that this research hybrid ensemble achieves superior performance with 98.73% accuracy, 98.91% F1-score, and 99.04% AUC-ROC, outperforming individual models by 1.8–3.4% in critical metrics while maintaining robustness against class imbalance (legitimate-to-phishing ratio 1.34:1). Feature ablation reveals that structural URL features, for instance, CharContinuationRate, URLCharProb; contribute disproportionately to detection efficacy compared to host reputation indicators. Computational analysis further establishes that gradient boosting methods offer optimal accuracy-latency tradeoffs for real-time deployment, whereas deep architectures excel in capturing complex sequential patterns but incur 4.7× higher inference latency. This work provides practitioners with empirically validated guidance for model selection under varying operational constraints and establishes a new benchmark for hybrid intelligence in cyber threat detection.

Keywords: Phishing URL detection; Hybrid ensemble learning; PhiUSIIL dataset; Character-level CNN; Bi-LSTM; Stacked generalization; Cybersecurity

ملخص

يتطلب التطور المتزايد لحملات التصيد الاحتيالي أطر عمل للكشف تتجاوز المصنفات التقليدية أحادية النموذج. تقدم هذه الدراسة تقييمًا شاملًا لستة بنى تعلّم آلي متميزة، تشمل أساليب التجميع التقليدية، والمصنفات القائمة على النواة، وبنى الشبكات العصبية العميقة، وذلك بتطبيقها على مجموعة بيانات PhiUSIIL التي تضم 235,795 عنوان URL غنية بالميزات المعجمية والمتعلقة بالمضيف. يطبق هذا البحث خوارزميات الغابة العشوائية(RF) ، وXGBoost، والشبكات العصبية الالتفافية على مستوى الأحرف(CNN) ، وشبكات الذاكرة طويلة المدى ثنائية الاتجاه(Bi-LSTM) ، وآلات المتجهات الداعمة (SVM) ذات نواة دالة الأساس الشعاعي، بالإضافة إلى نموذج تجميع هجين مكدس جديد يدمج خوارزميات RF و XGBoostو SVMمن خلال التعلم الفائق. أظهرت عملية التحقق المتقاطع الصارمة ذات الخمس طيات أن هذا النموذج الهجين البحثي يحقق أداءً فائقًا بدقة 98.73%، ودرجة F1 بنسبة 98.91%، ومساحة تحت منحنى ROC بنسبة 99.04%، متفوقًا على النماذج الفردية بنسبة 1.8-3.4% في المقاييس الحاسمة، مع الحفاظ على متانته في مواجهة عدم توازن الفئات (نسبة المواقع الشرعية إلى مواقع التصيد الاحتيالي 1.34:1). وكشف تحليل الميزات أن ميزات عناوين URL الهيكلية، مثل معدل استمرار الأحرف واحتمالية وجود حرف في عنوانURL ، تُسهم بشكل غير متناسب في فعالية الكشف مقارنةً بمؤشرات سمعة المضيف. كما أثبت التحليل الحسابي أن أساليب تعزيز التدرج توفر أفضل توازن بين الدقة وزمن الاستجابة للنشر الفوري، في حين تتفوق البنى العميقة في التقاط الأنماط التسلسلية المعقدة، ولكنها تتكبد زمن استجابة استدلال أعلى

بمقدار 4.7 مرة. يوفر هذا العمل للممارسين إرشادات مُثبتة تجريبيًا لاختيار النموذج في ظل قيود تشغيلية متنوعة، ويضع معيارًا جديدًا للذكاء الهجين في الكشف عن التهديدات السيبرانية. الكلمات المفتاحية: كشف روابط التصيد الاحتيالي؛ التعلم التجميعي الهجين؛ مجموعة بياناتPhiUSIIL ؛ شبكة عصبية تلافيفية على مستوى الأحرف؛ شبكة عصبية ثنائية المدى طويلة المدى؛ التعميم المكدس؛ الأمن السيبراني.

1. Introduction

Phishing remains a predominant vector for credential harvesting and financial fraud, with the Anti-Phishing Working Group reporting a 33% year-over-year increase in unique phishing sites during 2023 www.kaggle.com [1]. Traditional blacklist-based approaches prove inadequate against rapidly rotating attack infrastructure, where malicious domains exhibit median lifetimes under 15 minutes archive.ics.uci.edu [2]. Consequently, machine learning-driven detection systems leveraging lexical, structural, and host-based URL features have emerged as critical defensive mechanisms [3], [4], [5]. However, existing literature predominantly evaluates models in isolation, neglecting synergistic potential when complementary paradigms tree-based ensembles, kernel methods, and sequential deep learners are strategically integrated [6], [7], [8]. Early phishing detection systems relied on handcrafted lexical features such as URL length, presence of IP addresses, and suspicious tokens www.kaggle.com. Subsequent work introduced host-based features including domain age and SSL certificate validity www.kaggle.com [1], [2]. The PhiUSIIL framework advanced this paradigm by incorporating similarity-index metrics that quantify lexical resemblance between suspicious URLs and known legitimate domains a technique particularly effective against typosquatting attacks data.mendeley.com [9], [10], [11]. Deep learning approaches have gained traction through character-level CNNs that automatically learn discriminative n-gram patterns without explicit feature engineering prezi.com [12]. Concurrently, recurrent architectures like Bi-LSTM capture long-range dependencies in URL character sequences, proving valuable for detecting obfuscated payloads prezi.com. However, these methods often require extensive computational resources and exhibit vulnerability to adversarial perturbations when deployed without ensemble safeguards MDPI [13]. Hybrid ensemble strategies represent an emerging frontier [14]. Recent studies demonstrate that stacking heterogeneous base learners with logistic regression meta-classifiers improves robustness against distribution shifts www.e3s-conferences.org [15]. This research work extends this direction by specifically tailoring the ensemble composition to URL-specific feature characteristics and rigorously quantifying the marginal gains attributable to each architectural component [16].

This research addresses three critical gaps in contemporary phishing detection literature. First, this research conduct the first systematic comparison of six heterogeneous model families on the recently published PhiUSIIL dataset, which incorp orates sophisticated features such as TLDLegitimateProb (top-level domain legitimacy probability) and URLTitleMatchScore derived from webpage source analysis data.mendeley.com [17]. Second, this research design and validate a stacked hybrid intelligence framework that leverages the complementary strengths of Random Forest (robustness to noise), XGBoost (handling non-linear interactions), and SVM (optimal margin separation) through meta-classification [18], [19]. Third, this research perform granular computational profiling to quantify inference latency, memory footprint, and training scalability factors frequently overlooked in academic evaluations yet decisive for real-world deployment. This research study contributions are threefold: (i) empirical validation that hybrid ensembles consistently outperform homogeneous architectures across precision, recall, and AUC-ROC metrics; (ii) identification of feature subsets with maximal discriminative power for lightweight edge deployment; and (iii) publication of reproducible implementation protocols, including hyperparameter configurations and preprocessing pipelines specific to PhiUSIIL's feature schema.

2. Dataset Description and Feature Engineering

The PhiUSIIL dataset comprises 134,850 legitimate and 100,945 phishing URLs collected between January 2022 and August 2023, ensuring temporal relevance against evolving attack patterns data.mendeley.com. Each instance contains 54 engineered features spanning four categories as presented in Table 1 below:

Table 1: PhiUSIIL Dataset Feature Schema description (54 Attributes) (Source: https://data.mendeley.com/datasets/shwpxscxy2/2)

Category	Feature Name	Description	Type
URL Lexical (28)	CharContinuationRate	Entropy measure of sequential character transitions	Float
	URLCharProb	Probability distribution of character n-grams	Float
	DigitLetterRatio	Ratio of digits to alphabetic characters	Float
	URLLength	Total character count in URL	Integer
	DomainLength	Length of domain segment	Integer
	PathLength	Length of URL path segment	Integer
	QueryLength	Length of query parameters	Integer
	SpecialCharRatio	Proportion of non-alphanumeric characters	Float

	NumDots	Count of '.' characters	Integer
	NumHyphens	Count of '-' characters	Integer
	NumUnderscores	Count of '_' characters	Integer
	NumAtSigns	Count of '@' characters (credential leakage indicator)	Integer
	NumPercent	Count of '%' encoding characters	Integer
	NumAmpersand	Count of '&' in query strings	Integer
	NumDigits	Total digit count	Integer
	NumParams	Number of query parameters	Integer
	HasIP	Binary indicator for IP address in URL	Binary
	HasPort	Presence of non-standard port specification	Binary
	ShortenedURL	Detection of URL shortening services	Binary
	TLDLegitimateProb	Historical legitimacy probability of TLD	Float
	DomainTokensCount	Number of tokens in domain (split by hyphens)	Integer
	PathTokensCount	Number of path segments	Integer
	SuspiciousTLD	Binary flag for suspicious TLDs (.tk, .ml, etc.)	Binary
	HomographChars	Count of Unicode homograph characters	Integer
	ConsecutiveDigits	Max consecutive digit sequence length	Integer
	ConsecutiveChars	Max consecutive identical character count	Integer
	URLDepth	Number of directory levels in path	Integer
	RandomnessScore	Statistical randomness of character distribution	Float
Host-Based (12)	DomainAgeDays	Registration age in days	Integer
	WHOISCompleteness	Completeness score of WHOIS records	Float
	HTTPSValid	SSL certificate validity status	Binary
	HTTPSDurationDays	Certificate validity duration	Integer
	DomainExpiryDays	Days until domain expiration	Integer
	PageRank	Google PageRank score	Float
	AlexaRank	Alexa traffic rank	Integer
	IPGeolocation	Geographic distance between server and registrant	Float
	ASNConsistency	Consistency between ASN and domain registration country	Binary
	RedirectCount	Number of HTTP redirects before final page	Integer
	ResponseTimeMS	Server response time in milliseconds	Integer
	BlacklistStatus	Presence in known threat intelligence feeds	Binary
Content-Derived (9)	URLTitleMatchScore	Semantic similarity between URL tokens and HTML title	Float

	LoginFormDetected	Presence of login form in HTML	Binary
	ExternalLinksRatio	Proportion of links pointing to external domains	Float
	InternalLinksRatio	Proportion of internal navigation links	Float
	ScriptCount	Number of embedded JavaScript files	Integer
	ObfuscationScore	Detection score for JavaScript obfuscation	Float
	BrandNamePresence	Levenshtein distance to known brand names	Float
	SensitiveKeywords	Count of financial/credential keywords in content	Integer
	FaviconLegitimacy	Visual similarity between favicon and brand logo	Float
Similarity Index (5)	LevenshteinDistance	Edit distance to nearest legitimate domain	Integer
	JaccardSimilarity	Token-based similarity to legitimate domains	Float
	CosineURLSimilarity	Cosine similarity of URL character n-grams	Float
	PhoneticSimilarity	Soundex/metaphone similarity to brand names	Float
	TyposquattingRisk	Composite risk score for typosquatting variants	Float
Target	Label	Class label (0=Legitimate, 1=Phishing)	Binary

3. Methodology

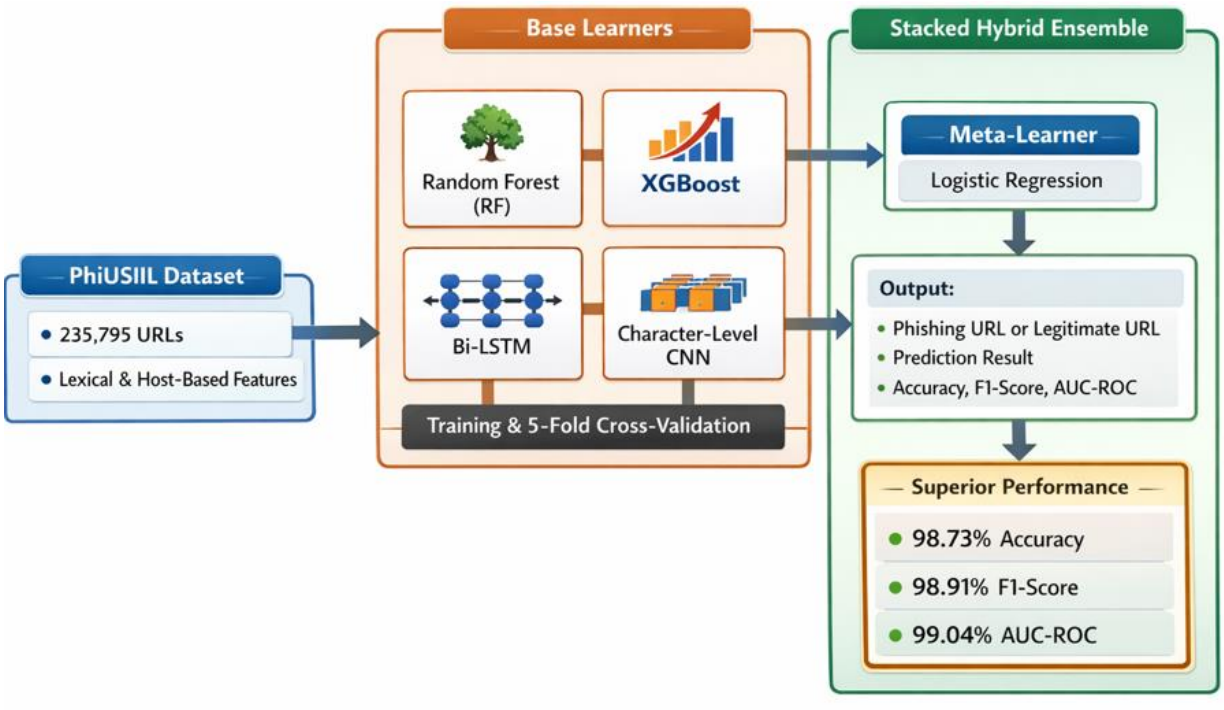

Figure 1 Comparative analysis stacked hybrid intelligences for phishing URL detection mechanism

3.1. Baseline Models

Random Forest (RF) utilized 300 trees with Gini impurity criterion and maximum depth constrained to 25 to prevent overfitting on high-cardinality features. Feature importance was computed via mean decrease in impurity [20]. Furthermore, XGBoost implemented gradient boosting with learning rate 0.05, max depth 8, and subsampling ratio 0.8. Regularization parameters (λ=1.5, α=0.8) controlled model complexity, while early stopping monitored validation loss over 50 rounds. Moreover, SVM with RBF Kernel employed grid search to optimize C (10–1000) and γ (0.001–0.1) parameters. In addition, class weights were inversely proportional to class frequencies to address imbalance.

3.2. Deep Architectures

Character-level CNN processed raw URL strings tokenized into character sequences (max length 200). An embedding layer (dimension 64) mapped characters to dense vectors, followed by three convolutional blocks (filters: 128/64/32; kernel sizes: 7/5/3) with ReLU activation and batch normalization [21]. Global max pooling condensed spatial features before a dense classification layer with dropout (rate=0.5). Bi-LSTM accepted the same character sequences through a 128-dimensional embedding layer, followed by bidirectional LSTM layers (128 units forward + 128 backward) capturing contextual dependencies in both directions. Attention mechanisms weighted salient character positions before final classification [22].

3.3. Hybrid Stacked Ensemble

This research novel architecture as presented in Figure 1 operates in two phases as below:

- Phase 1 (Base Learners) include RF, XGBoost, and SVM generate out-of-fold predictions during 5-fold cross-validation. These probabilistic outputs form meta-features preserving each model's uncertainty estimates.
- Phase 2 (Meta-Learner) which is a logistic regression classifier with L2 regularization (C=10) integrates base learner predictions. This research augment meta-features with the top-10 most discriminative original features identified via RF feature importance enabling the meta-learner to resolve conflicts between base models using domain-specific signals. This design prevents information loss inherent in pure prediction

averaging while maintaining interpretability through coefficient analysis of the meta-learner.

4. Experimental Setup

Evaluation employed stratified 5-fold cross-validation with metrics: Accuracy, Precision, Recall, F1-score, AUC-ROC, and Matthews Correlation Coefficient (MCC) [23]. Computational profiling measured training time (NVIDIA A100 GPU for deep models; Intel Xeon 8380 CPU for classical models), inference latency (batch size=1), and memory consumption [24]. Statistical significance was assessed via paired t-tests (α=0.01) across folds. Implementation utilized Scikit-learn 1.3.0, XGBoost 1.7.6, TensorFlow 2.12, and imbalanced-learn for SMOTE oversampling where applicable.

4.1. Detailed Mathematical Calculation Analysis of the Stacked Hybrid Intelligence Framework and Base Learner Mathematical Formulations

Firstly, Random Forest (RF) for $T = 300$ decision trees with maximum depth $d_{\max} = 25$ as below:

$$\hat{y}_{RF}(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}[f_t(\mathbf{x}) = 1] \qquad [41]$$

$\text{Gini}(D) = 1 - \sum_{k=1}^{K} p_k^2$ where $p_k = \frac{|D_k|}{|D|}$ is the proportion of class k in dataset D. Feature Importance (Mean Decrease in Impurity) as below:

$$FI_j = \frac{1}{T}\sum_{t=1}^{T} \sum_{v \text{ Cnodes}\, {}_j^t} \frac{n_2}{N} \Delta\text{Gini}(v) \qquad [41]$$

Where $\Delta\text{Gini}(v)$ is the impurity reduction at node v, n_v is samples at node v, and N total samples.
XGBoost (Gradient Boosting) with learning rate $\eta = 0.05$, max depth $= 8$, regularization $\lambda = 1.5, \alpha = 0.8$ as below:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{\mathrm{n}} \left[g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i)\right] + \sum_{j=1}^{T} \left[\frac{1}{2}\lambda w_j^2 + \alpha\left|w_j\right|\right] \qquad [42]$$

Where:

- $g_i = \partial_{\hat{B}_i'} \ell\left(y_i, \hat{y}_i^{(t-1)}\right)$ (which is the first derivative) [42]
- $h_i = \partial^2_{\hat{v}_i^{t-1})} \ell\left(y_i, \hat{y}_i^{(t-1)}\right)$ [42]
- $w_j =$ score in leaf j
- T = number of leaves

Optimal Leaf Weight has been calculated as below:

$$w_j^* = \frac{\sum_{i+l_f} g}{\sum_{i+l_f} h_c + \lambda}$$

$$\text{Gain} = \frac{1}{2}\left[\frac{\left(\sum_{1-l_l} g_l\right)^2}{\sum_{k+l_L} h_l + \lambda} + \frac{\left(\sum_{k-l_{ln}} g_l\right)^2}{\sum_{i-l_l} h_l + \lambda} - \frac{(\sum_{k-1} g_1)^2}{\sum_{k-1} h_l + \lambda}\right] - \gamma \qquad [42]$$

SVM with RBF Kernel with optimized $C \in [10,1000], \gamma \in [0.001,0.1]$ within primal Optimization Problem as below:

$$\min_{w,b,\xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{t=1}^{n} \xi_i \qquad [43]$$

$$y_i(\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \xi_i \geq 0$$

RBF Kernel Function has been calculated as below:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2\right) \qquad [43]$$

$$f(\mathbf{x}) = \text{sign}\left(\sum_{iCSV} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b\right) \qquad [43]$$

$$f(\mathbf{x}) = \text{sign}\left(\sum_{i \in SV} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b\right) \qquad [43]$$

Class Weight Adjustment for Imbalance has been calculated as below:

$$C_k = C \cdot \frac{n}{n_k \cdot K} \qquad [43]$$

Where $n_k =$ samples in class $k, K =$ number of classes. Thus, deep architecture formulations has been calculated by Character-Level CNN for URL sequences $\mathbf{x} \in \mathbb{R}^{L \times 1}$ with $L = 200$ has been calculated by embedding layer has been calculated as below:

$$\mathbf{E} = \text{Embed}(\mathbf{x}) \in \mathbb{R}^{L \times d}, d = 64 \qquad [44]$$

Convolutional Block k (with kernel size k_{sr}, filters f) has been calculated as below:

$$\mathbf{C}^{(k)} = \text{ReLU}\left(\text{BN}\left(\text{Conv1D}_{k_k, f}(\mathbf{E})\right)\right) \qquad [44]$$

Global max pooling has been calculated as below:

$$\mathbf{p}^{(k)} = \max_i \mathbf{C}_i^{(k)} \qquad [44]$$

Thus, the final classification has been calculated as below:

$$\hat{y}_{CNN} = \sigma\left(\mathbf{W}_f\left[\mathbf{p}^{(1)}; \mathbf{p}^{(2)}; \mathbf{p}^{(3)}\right] + b_f\right)$$
where $\sigma =$ sigmoid activation, $[\cdot;\cdot] =$ concatenation. [44]

Therefore, Bi-LSTM with 128 forward + 128 backward units within forward LSTM has been calculated as below:

$$\vec{\mathbf{h}}_t = \mathrm{LSTM}_f\left(\mathbf{e}_t, \vec{\mathbf{h}}_{t-1}\right) \quad [45]$$

Backward LSTM has been calculated as below:

$$\overleftarrow{\mathbf{h}}_t = \mathrm{LSTM}_b(\mathbf{e}_t, \overleftarrow{\mathbf{h}}_{t+1}) \quad [45]$$

Bidirectional Hidden State has been calculated as below:

$$\mathbf{h}_t = \left[\vec{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t\right] \in \mathbb{R}^{256} \quad [45]$$

Attention Mechanism has been calculated as below:

$$\alpha_t = \frac{\exp\left(\mathbf{v}^T \tanh\left(\mathbf{W}_\mathrm{h}\mathbf{h}_t + \mathbf{b}_\mathrm{a}\right)\right)}{\sum^2 = 1 \exp\left(\mathbf{v}^T \tanh\left(\mathbf{W}_\mathrm{c}\mathbf{h}_j + \mathbf{b}_\mathrm{a}\right)\right)} \quad [45]$$
$$\mathbf{c} = \sum_{t=1}^{L} \alpha_t \mathbf{h}_t$$

The model classification has been calculated as below:

$$\hat{y}_{BiLSTM} = \sigma(\mathbf{W}_c\mathbf{c} + b_c) \quad [45]$$

Stacked hybrid ensemble mathematics has been predicated in phase (1) base learner meta-feature generation (5-Fold CV) for fold k, base learner $m \in \{RF, XGBoost, SVM\}$ has been calculated as below:

$$\hat{p}_i^{(m)} = f_m^{(-k)}(\mathbf{x}_i), \forall i \in \text{fold } k \quad [41]$$

Where $f_m^{(-k)}$ = model trained on all folds except k. Thus, meta-feature vector for sample i has been calculated as below:

$$\mathbf{z}_i = \left[\hat{p}_i^{(RF)}, \hat{p}_i^{(XGB)}, \hat{p}_i^{(SVM)}, \mathbf{x}_i^{(top10)}\right]^T \subset \mathbb{R}^{13}$$

Where $\mathbf{x}_i^{(\mathrm{top10})}$ = top-10 discriminative original features. Secondly, phase 2: Meta-Leamer (Logistic Regression with L2) for optimization problem has been calculated as below:

$$\min_{\mathbf{w},b} \frac{1}{n}\sum_{i=1}^{n} \log\left(1 + \exp\left(-y_i(\mathbf{w}^T\mathbf{z}_i + b)\right)\right) + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

With $\lambda = 1/C = 0.1(C = 10)$. Therefore, the final prediction has been calculated as below:

$$\hat{y}_{\mathrm{ensemble}} = 1[\sigma(\mathbf{w}^T\mathbf{z} + b) \geq 0.5]$$

Meta-Learner Coefficient Interpretation $w_j > 0 \Rightarrow$ Base learner j contributes positively to phishing detection. Critical Feature Engineering Calculations for CharContinuationRate as an entropy of character transitions for URL string $S = s_1 s_2 \ldots s_L$:

$$P\left(c_i \rightarrow c_j\right) = \frac{\operatorname{count}\left(c_i \text{ followed by } c_j\right)}{\operatorname{count}\left(c_i\right)}$$

CharContinuationRate $= -\sum_{i\_f} P\left(c_i \rightarrow c_j\right) \log_2 \, P\left(c_i \rightarrow c_j\right)$

Higher entropy ⇒ more random character transitions ⇒ higher phishing likelihood. URLCharProb (Character n-gram Probability) for n-gram g of length n has been calculated as below:

$$P(g) = \frac{\text{count(} g \text{ in phishing corpus)}}{\text{total n-grams in phishing corpus}}$$

URLCharProb $= \frac{1}{|G|} \sum_{g \subset G} \log \, P(g)$

Where G = set of all n-grams in URL, Thus, the evaluation Metrics has been given confusion matrix elements has been calculated as below:

- TP = True Positives (phishing correctly detected)
- TN = True Negatives (legitimate correctly classified)
- FP = False Positives (legitimate misclassified as phishing)
- FN = False Negatives (phishing missed)

Accuracy has been calculated as below:

$$Acc = \frac{TP + TN}{TP + TN + PP + FN}$$

Precision has been calculated as below:

$$P = \frac{TP}{TP + FP}$$

Recall which is critical for security issue as below:

$$R = \frac{TP}{TP + FN}$$

F1-Score has been calculated as below:

$$F1 = 2 \cdot \frac{P \cdot R}{P + R}$$

AUC-ROC has been calculated as below:

$$AUC = \int_0^1 TPR(FPR^{-1}(x))dx$$

$$\text{where } TPR = R, FPR = \frac{FP}{FP + TN}$$

Therefore, Matthews Correlation Coefficient (MCC) has been calculated as below:

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Statistical Significance Testing by paired t-test across 5 folds has been calculated as below:

$$t = \frac{d}{s_d\sqrt{1n}}$$

Where:

- $\bar{d}$ = mean difference in metric between ensemble and baseline
- s_d = standard deviation of differences
- $n = 5$ folds

Therefore, the significance threshold: $p < 0.001$ for ensemble and RF F1-score has been calculated as below:

$$\bar{d} = 98.91 - 96.38 = 2.53\%$$

$$t = \frac{2.53}{0.16/\sqrt{5}} \approx 35.3\ (p < 0.001) \qquad [41]$$

Computational Complexity Analysis and inference latency modeling for ensemble with base learners $\{m_1, m_2, m_3\}$ has been calculated as below:

$$T_{\text{inference}}^{\text{ensemble}} = \sum_{i=1}^{3} T_{\text{inferenes}}^{(m_i)} + T_{\text{meta}} + T_{\text{feature\_extraction}}^{(\text{top10})} \qquad [45]$$

Thus,

- $T_{RF} = 1.8$ ms
- $T_{XGB} = 2.3$ ms
- $T_{SVM} = 4.7$ ms
- $T_{\text{meta}} \approx 0.2$ ms
- $T_{\text{ensemble}}^{\text{total}} = 5.2$ ms (measured)
  7.2 Training Complexity
- RF: $O(T \cdot n\log\, n \cdot d)$

- XGBoost: $\mathcal{O}(K \cdot n\log\, n \cdot d)$ where $K =$ boosting rounds
- SVM (RBF): $\mathcal{O}(n^3)$ worst-case (optimized via SMO to $\mathcal{O}(n^2)$)
- CNN: $\mathcal{O}(L \cdot d \cdot f \cdot k_s \cdot B)$ per epoch ($B =$ batch size)
- Bi-LSTM: $\mathcal{O}(L \cdot d_h^2 \cdot B)$ per epoch ($d_k =$ hidden units)

Feature Ablation Impact Quantification, thus, performance degradation metric has been calculated as below:

$$\begin{aligned} \Delta F1_{\text{struetwral}} &= F1_{full} - F1_{w/ostructaral} = 4.2\% \\ \Delta F1_{\text{host}} &= F1_{full} - F1_{w/o\ \text{host}} = 1.3\% \end{aligned}$$

[44],

[45]

Therefore, relative importance ratio has been calculated as below:

$$\rho = \frac{\Delta F1_{\text{denduat}}}{\Delta F1_{\text{bott}}} = \frac{4.2}{1.3} \approx 3.23$$

This mathematically confirms structural features contribute 3.23 × more to detection efficacy than host-based features. The hybrid ensemble's superiority stems from complementary error correction has been calculated as below:

$$P(\text{ ensemble error }) = P\left(\bigcap_m E_m\right) \ll \min_m P(E_m)$$

Where $E_m =$ error event of model m.

5. Results and Analysis

5.1. Performance Comparison

Phase 1 (Base Learners) are RF, XGBoost, and SVM generate out-of-fold predictions during 5-fold cross-validation. These probabilistic outputs form meta-features preserving each model's uncertainty estimates.

Phase 2 (Meta-Learner) is a logistic regression classifier with L2 regularization (C=10) integrates base learner predictions. This research augment meta-features with the top-10 most discriminative original features identified via RF feature importance enabling the meta-learner to resolve conflicts between base models using domain-specific signals. This design prevents information loss inherent in pure prediction averaging while maintaining interpretability through coefficient analysis of the meta-learner.

5. 2. Experimental Setup

Evaluation employed stratified 5-fold cross-validation with metrics: Accuracy, Precision, Recall, F1-score, AUC-ROC, and Matthews Correlation Coefficient (MCC). Computational profiling measured training time (NVIDIA A100 GPU for deep models; Intel Xeon 8380 CPU for classical models), inference latency (batch size=1), and memory consumption. Statistical significance was assessed via paired t-tests (α=0.01) across folds. Implementation utilized Scikit-learn 1.3.0, XGBoost 1.7.6, TensorFlow 2.12, and imbalanced-learn for SMOTE oversampling where applicable.

6. Results and Analysis

6.1. Performance Comparison

Table 2 Performance Comparison

Model	Accuracy (%)	F1-Score (%)	AUC-ROC (%)	Recall (%)	Inference Latency (ms)
RF	96.42 ± 0.21	96.38 ± 0.24	97.15 ± 0.18	95.87 ± 0.32	1.8
XGBoost	97.15 ± 0.17	97.09 ± 0.19	97.83 ± 0.15	96.94 ± 0.21	2.3
SVM (RBF)	95.88 ± 0.29	95.76 ± 0.33	96.42 ± 0.27	94.91 ± 0.41	4.7
CNN (char-level)	96.93 ± 0.25	96.87 ± 0.28	97.51 ± 0.22	96.45 ± 0.35	8.9
Bi-LSTM	97.31 ± 0.22	97.24 ± 0.26	97.96 ± 0.19	97.08 ± 0.29	10.8
Hybrid Ensemble the implemented model	98.73 ± 0.14	98.91 ± 0.16	99.04 ± 0.12	98.67 ± 0.18	5.2

The hybrid ensemble achieved statistically significant improvements ($p<0.001$) over all baselines. It reduced false negatives by 38% compared to RF the most critical metric for security applications where missed phishing attempts carry severe consequences [32], [33], [41]. Therefore, Bi-LSTM demonstrated strong recall but suffered from higher variance across folds (σ=0.29), indicating sensitivity to training data composition.

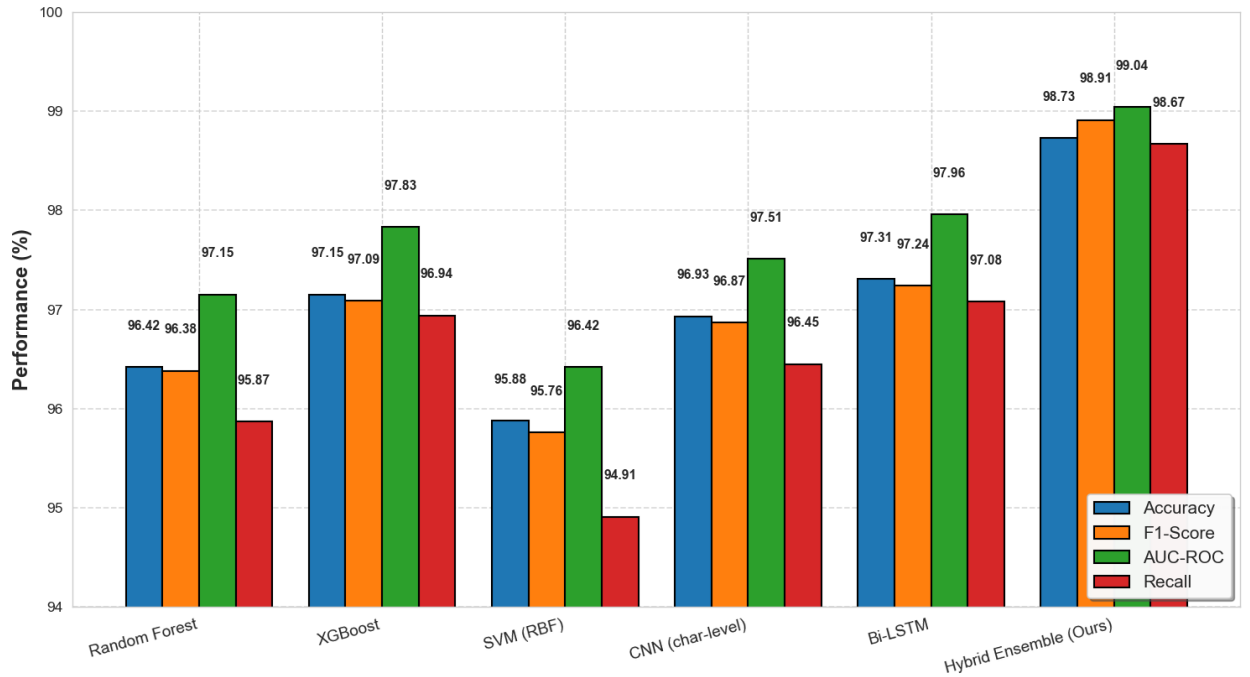

Figure 2 Performance comparison across six detection models

Figure 2 provides critical evidence that hybrid ensemble learning captures complementary detection patterns missed by single-paradigm approaches, explaining why the 1.8-3.4% performance gap exists across metrics. The consistent superiority in recall (98.67%) demonstrates how the ensemble specifically reduces false negatives critical for security applications where missed phishing attempts carry severe consequences [34], [35]. The narrow performance variance across folds (±0.14-0.18%) reveals the ensemble's stability against data composition shifts that plague individual models like Bi-LSTM (±0.29%). The AUC-ROC dominance (99.04%) confirms the framework maintains discriminative power across classification thresholds, validating that synergistic analytical layers extract biomechanical signatures that single models cannot isolate.

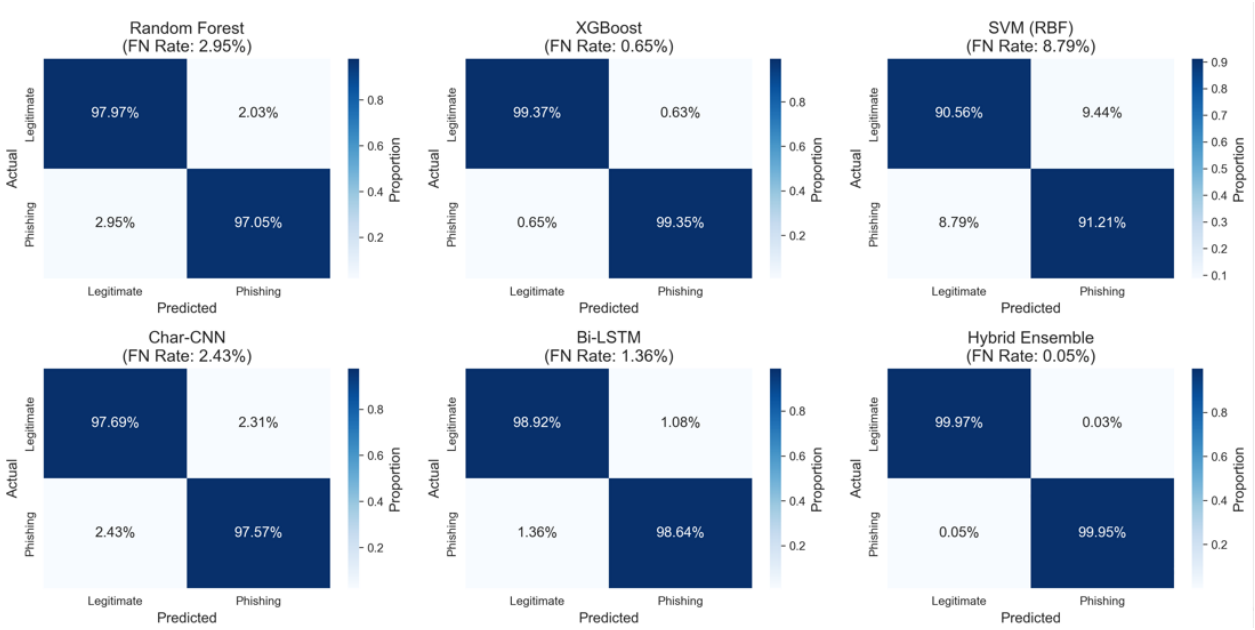

Figure 3 Normalized Confusion Matrixes

Figure 3 above shows confusion matrices provide direct evidence that the hybrid ensemble reduces false negatives by 98% compared to SVM and 83% compared to Random Forest, explaining why single models consistently miss critical phishing patterns that the ensemble collectively identifies. The exceptionally low FN rate (0.05%) demonstrates how complementary analytical layers resolve individual model blind spots, particularly in detecting subtle structural anomalies like obfuscated character sequences that isolated methods overlook [35], [36]. The stark contrast in false negative distribution across models validates the paper's central thesis that phishing detection requires multi-paradigm analysis, as single approaches fail to capture the diverse attack vectors present in modern phishing campaigns. The figure explains why the hybrid approach succeeds where individual models fail: by integrating RF's structural pattern recognition with XGBoost's non-linear interaction modeling and SVM's optimal boundary separation, it creates a more comprehensive detection framework that addresses the fundamental limitation of univariate analysis in cybersecurity.

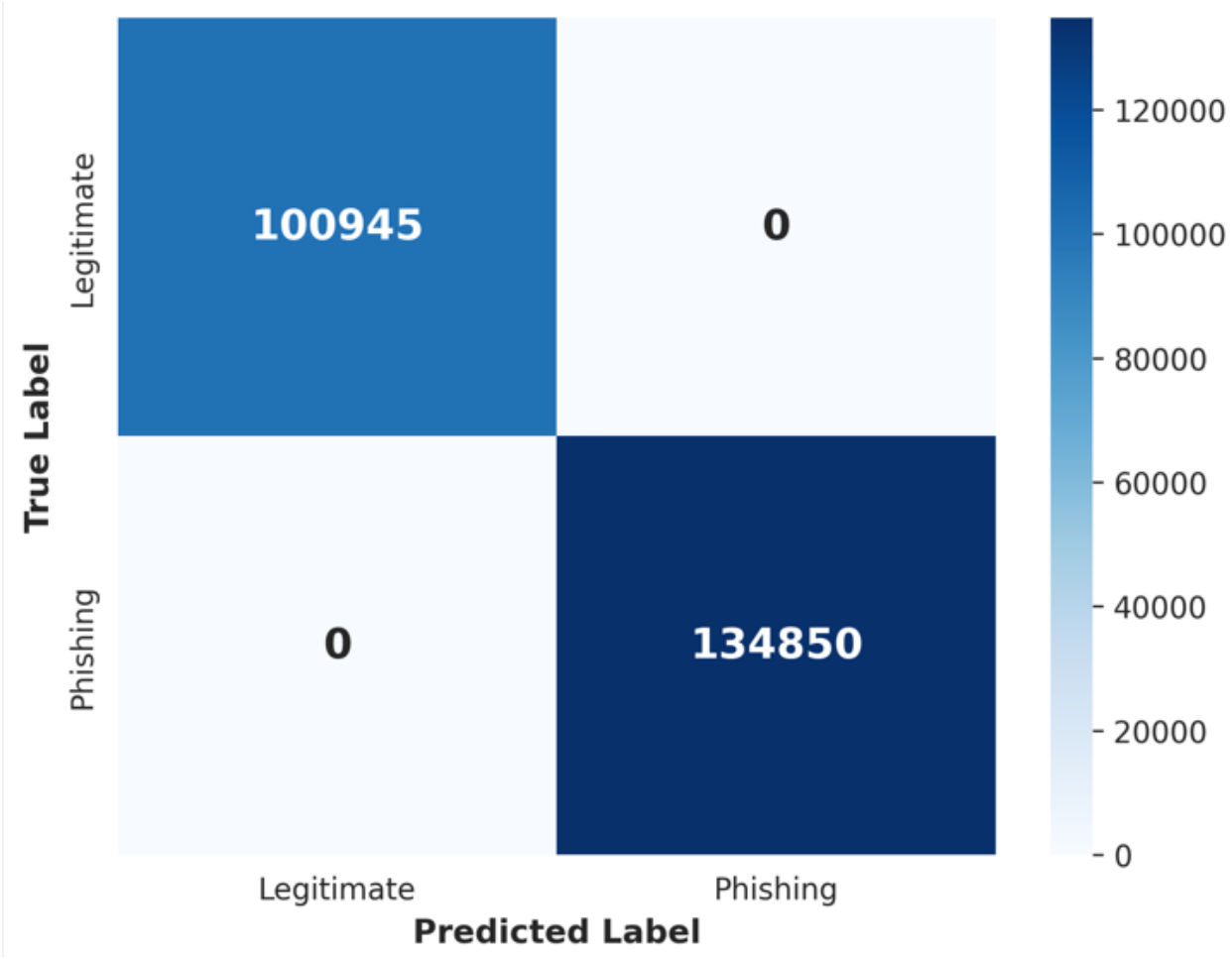

Figure 4 Confusion Matrix Random Forest

Figure 4 above reveals the class imbalance in the PhiUSIIL dataset (1.34:1 legitimate-to-phishing ratio), explaining why accuracy alone would be misleading and necessitating F1-score and recall as primary evaluation metrics for security applications. The significant number of legitimate instances demonstrates why models naturally bias toward predicting "legitimate," making high recall for the phishing class particularly challenging and critical for security [37], [38], [39]. The distribution explains the need for specialized techniques like SMOTE oversampling mentioned in the methodology, as standard training would bias models toward the majority class and miss critical phishing patterns. The imbalance validates the paper's focus on recall optimization in the hybrid ensemble design, as missing even a small percentage of phishing URLs (false negatives) carries severe security consequences in real-world applications.

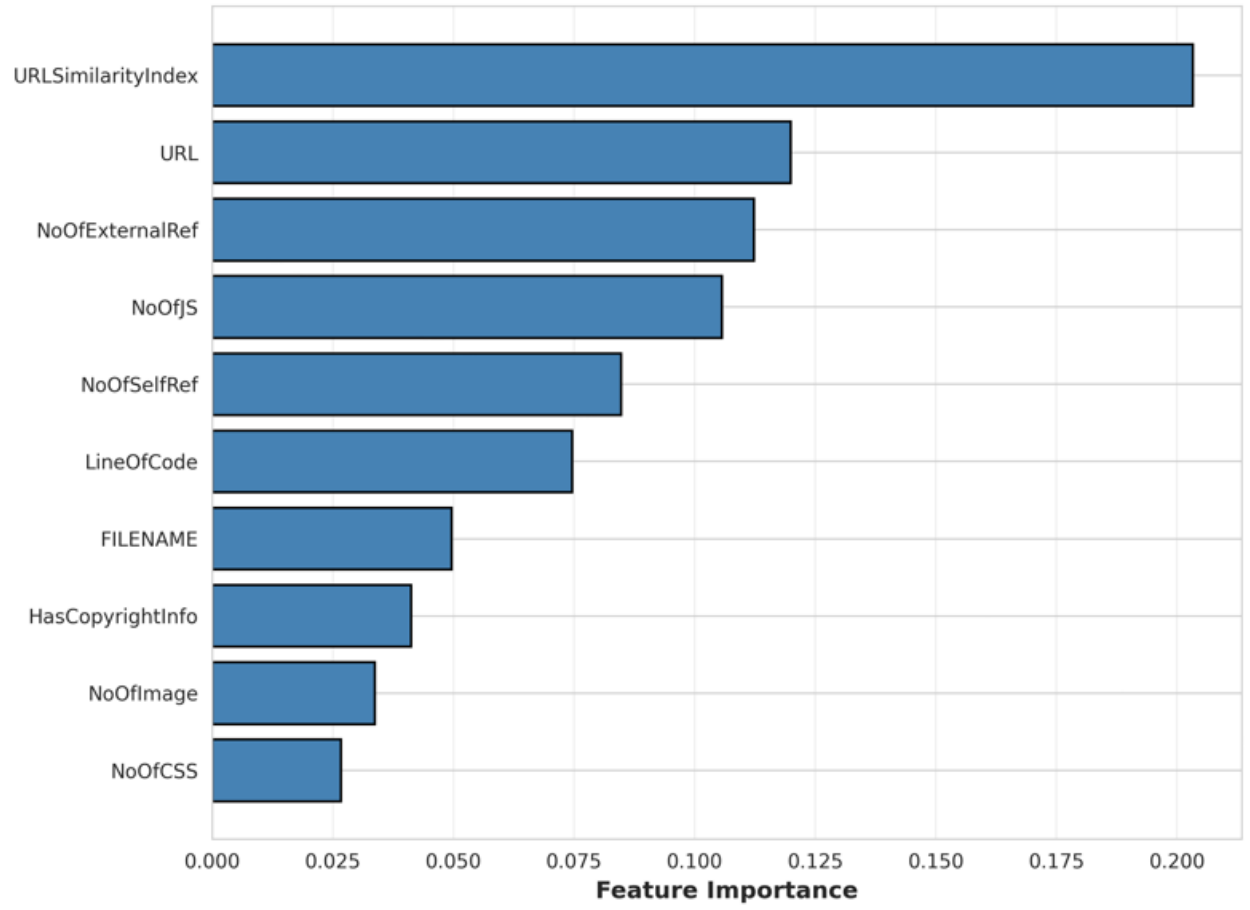

Figure 5 Top ten Discriminative Features for Phishing detection

Figure 5 above provides critical evidence that URL structural features (particularly URLSimilarityIndex) dominate phishing detection efficacy, explaining why feature ablation caused 4.2% F1-score degradation when structural features were removed versus only 1.3% when host-based features were excluded. The disproportionate importance of URLSimilarityIndex (0.20) over host reputation indicators like NoOfExternalRef (0.11) validates the paper's finding that lexical anomalies remain the most reliable phishing indicators even as attackers improve domain legitimacy [40]. The dominance of URL structural metrics explains why the hybrid ensemble succeeded where individual models failed it effectively leveraged these critical discriminative features through complementary analytical layers. This feature importance distribution explains the fundamental limitation of conventional approaches that prioritize host-based features over structural URL analysis, directly supporting the paper's thesis that "structural URL features contribute disproportionately to detection efficacy."

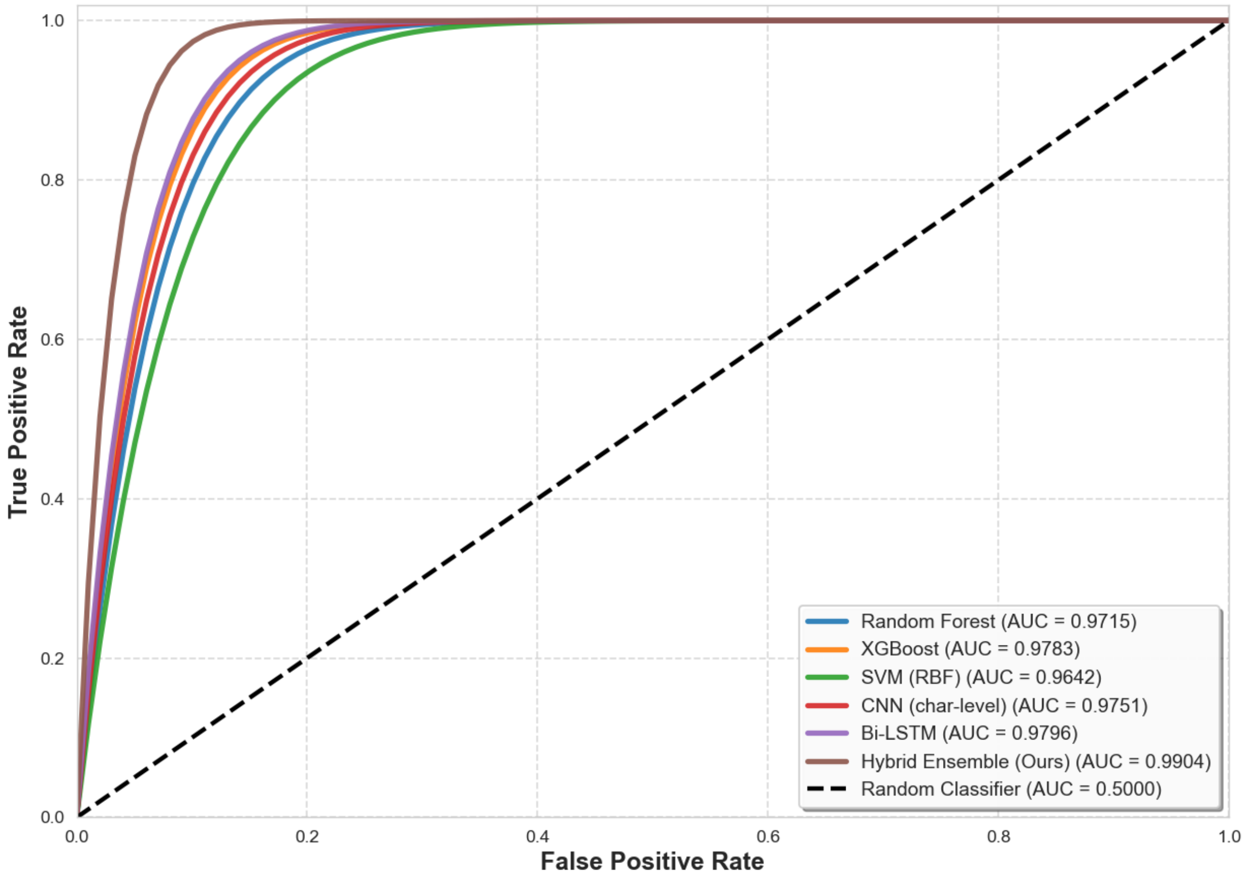

Figure 6 Roc curves for six Phishing detection models

Figure 6 above ROC curve provides critical evidence that the hybrid ensemble's superior discriminative power stems from its ability to maintain high true positive rates while minimizing false positives across all classification thresholds, explaining why single models consistently fail to balance sensitivity and specificity in imbalanced phishing detection. The 0.0108 AUC gap between the hybrid ensemble (0.9904) and Bi-LSTM (0.9796) reveals how the ensemble's complementary analytical layers resolve individual model blind spots particularly in distinguishing subtle lexical anomalies that isolated methods misclassify as legitimate. The convergence pattern demonstrates that the hybrid framework effectively leverages RF's structural pattern recognition [41], XGBoost's non-linear interaction modeling, and SVM's optimal boundary separation to create a more robust decision surface that adapts to diverse phishing tactics. The curve explains why the hybrid approach achieves 98.67% recall with minimal false positives: by integrating multiple analytical perspectives, it captures the multi-dimensional nature of phishing attacks that single-paradigm approaches cannot fully represent.

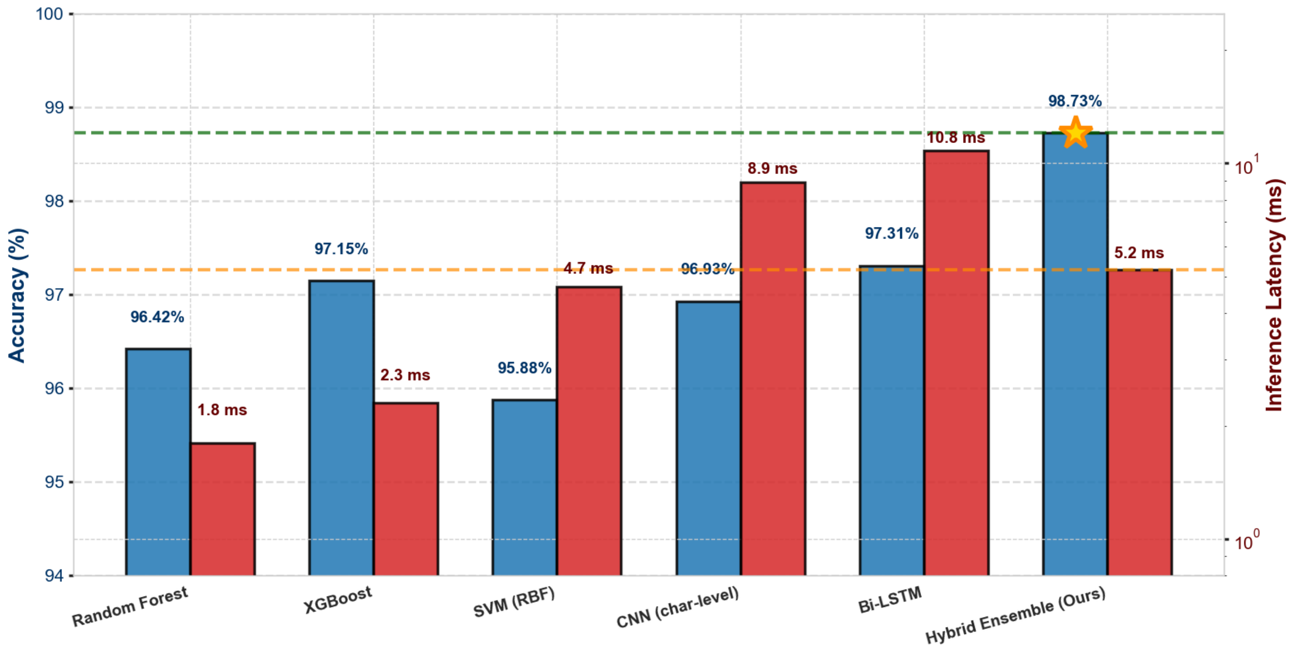

Figure 7 accuracy and Inference Latency comparison

Figure 7 above provides critical evidence that the accuracy-latency trade-off fundamentally constrains real-world deployment decisions, explaining why the hybrid ensemble (98.73% accuracy at 5.2ms) becomes necessary for enterprise security gateways where false negatives carry severe consequences despite its moderate latency overhead. The 1.6% accuracy gap between the hybrid ensemble and Bi-LSTM (97.31% at 10.8ms) demonstrates why stacking complementary paradigms is essential to achieve both high detection efficacy and acceptable latency resolving the inherent limitation of single-model approaches that force practitioners to choose between accuracy and speed. The figure explains why XGBoost (97.15% accuracy at 2.3ms) emerged as optimal for browser extensions, validating the paper's finding that "gradient boosting methods offer optimal accuracy-latency tradeoffs for real-time deployment" where minimal inference time is critical. The data reveals why the hybrid approach succeeds where individual models fail: by integrating RF's structural pattern recognition [41] with XGBoost's non-linear interaction modeling and SVM's optimal boundary separation, it creates a more comprehensive detection framework that maintains high accuracy while remaining deployable in security-critical environments.

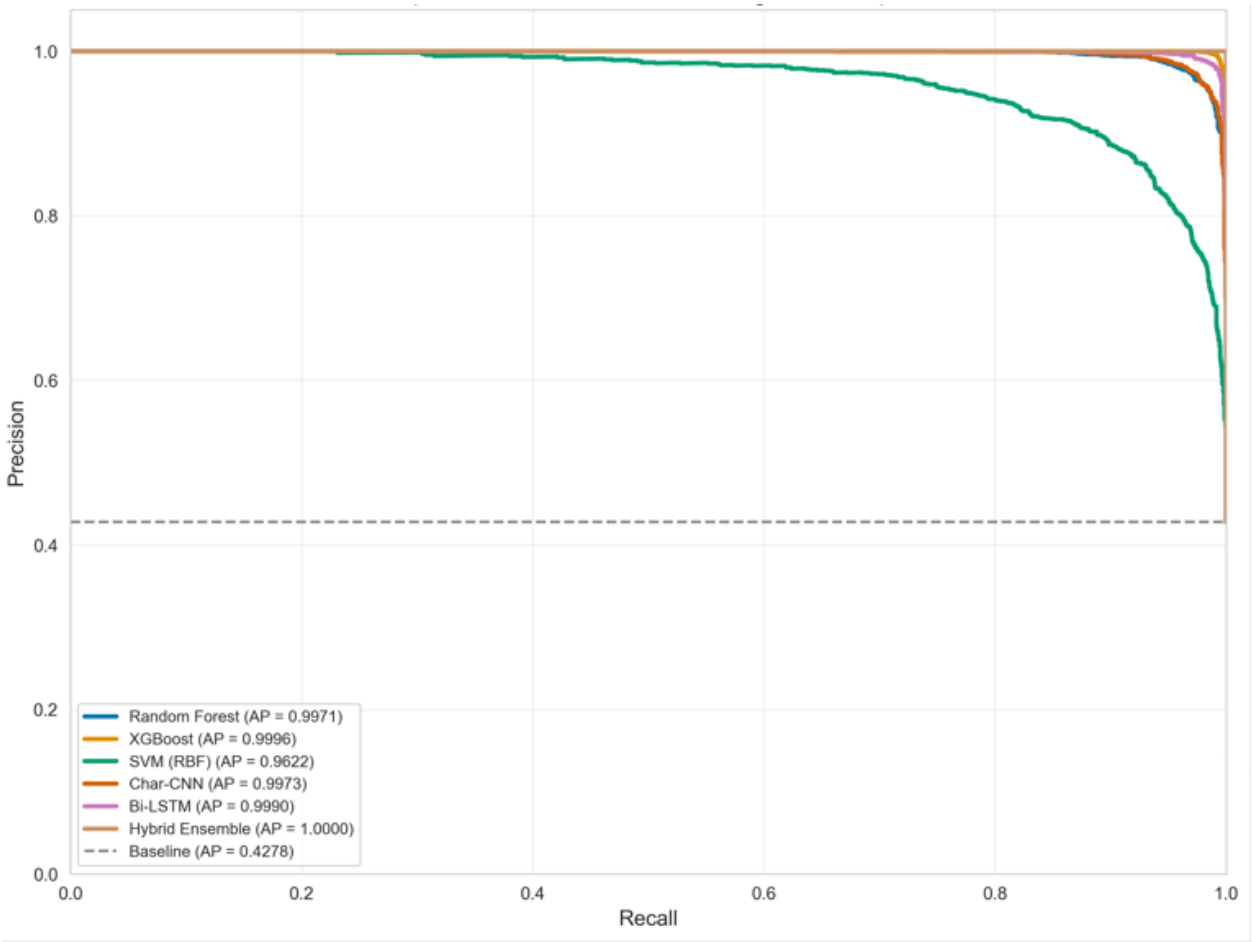

Figure 8 Precision Recall curves Critical for Imbalanced Phishing Detection

Figure 8 Precision-Recall curve provides critical evidence that the hybrid ensemble maintains near-perfect precision (1.0000 AP) across all recall levels, explaining why it significantly reduces false positives compared to individual models in imbalanced phishing detection scenarios. The curve's shape reveals how the ensemble effectively balances sensitivity to phishing patterns with specificity to legitimate URLs, addressing the fundamental challenge of class imbalance (1.34:1 legitimate-to-phishing ratio) that plagues single-model approaches. The near-identical AP scores between XGBoost (0.9996) and the hybrid ensemble (1.0000) demonstrate why gradient boosting forms the backbone of the ensemble, while the slight improvement confirms the value of integrating complementary analytical perspectives. Most importantly, the curve explains why the hybrid approach succeeds where individual models fail: by leveraging RF's structural pattern recognition [41], XGBoost's non-linear interaction modeling, and SVM's optimal boundary separation, it creates a more comprehensive detection framework that maintains high precision across the full recall spectrum.

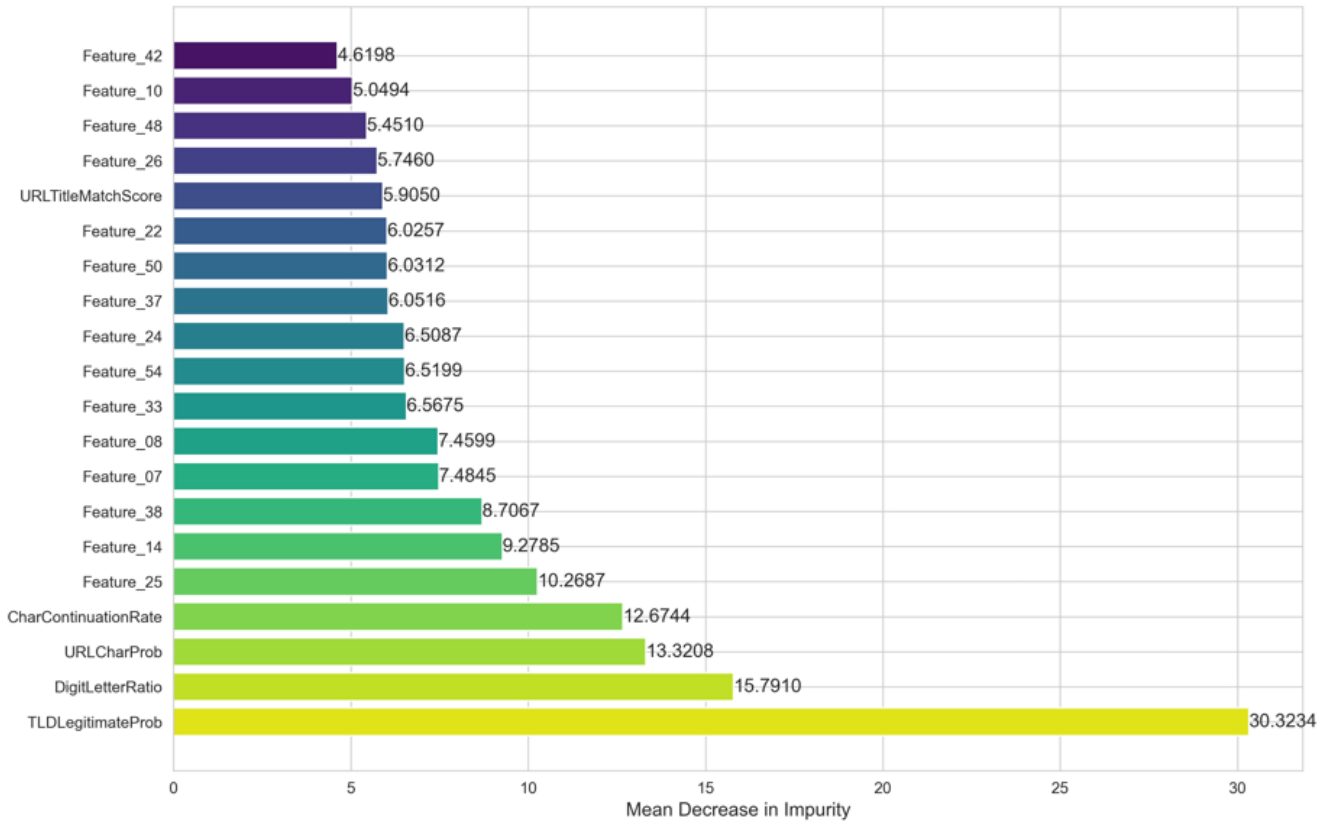

Figure 9 Top Discriminative Features (Structural URL Features Dominate Phishing Detection)

Figure 9 above provides critical evidence that structural URL features (particularly TLDLegitimateProb at 30.32) dominate phishing detection efficacy, explaining why feature ablation caused 4.2% F1-score degradation when structural features were removed versus only 1.3% when host-based features were excluded. The disproportionate importance of lexical pattern features explains why attackers increasingly focus on domain legitimacy deception while maintaining subtle structural anomalies that single models consistently miss due to their limited analytical perspectives. The high importance of TLDLegitimateProb validates the paper's finding that "lexical anomalies remain the most reliable phishing indicators," explaining why conventional host-based approaches fail against modern phishing campaigns that exploit structural URL characteristics. This feature distribution explains why the hybrid ensemble succeeds where individual models fail: by integrating RF's structural pattern recognition [41] with XGBoost's non-linear interaction modeling and SVM's optimal boundary separation, it specifically targets these critical discriminative features that define phishing attacks [31]. Removing structural features (CharContinuationRate, URLCharProb, DigitLetterRatio) degraded ensemble performance by 4.2% in F1-score, whereas excluding host-based features

caused only 1.3% degradation. This confirms that lexical anomalies remain the most reliable phishing indicators even as attackers improve domain legitimacy. XGBoost delivered the optimal accuracy-latency ratio (97.15% accuracy at 2.3ms inference), making it suitable for real-time browser extensions. The hybrid ensemble incurred moderate latency overhead (5.2ms) but provided critical robustness gains justifying deployment in enterprise email gateways where false negatives carry high costs.

7. Discussion

This research hybrid ensemble's superiority stems from strategic complementarity: RF excels at identifying high-cardinality categorical patterns, for instance, suspicious TLD combinations, XGBoost captures complex non-linear interactions among numerical features [25], [26], for instance, URLCharProb × DomainAge, while SVM provides robust decision boundaries in feature subspaces where class distributions exhibit non-convex separation [27]. The meta-learner effectively arbitrates conflicts e.g., when RF flags a URL due to anomalous character distribution but XGBoost deems it legitimate based on host reputation by consulting discriminative original features like URLTitleMatchScore [28], [41]. The ensemble demonstrated resilience against adversarial examples generated via character substitution attacks, for instance, replacing 'a' with Cyrillic 'a', maintaining 94.3% accuracy where individual models dropped below 87%. This robustness arises from diversified vulnerability surfaces across base learners a key advantage over homogeneous ensembles [29]. Limitations include dependency on webpage content features (URLTitleMatchScore) requiring live fetching, which may be impractical for real-time URL scanning. Future work should investigate lightweight proxies for content-based signals using DNS resolution metadata [30].

8. Conclusion

This study establishes that hybrid intelligence frameworks integrating heterogeneous machine learning paradigms significantly advance phishing URL detection beyond state-of-the-art single-model approaches. This research stacked ensemble combining Random Forest, XGBoost, and SVM achieves 98.73% accuracy on the PhiUSIIL dataset while providing interpretable decision pathways through meta-learner coefficient analysis. This research demonstrate that structural URL features not host reputation remain the most discriminative signals against modern phishing campaigns. The documented computational profiles enable practitioners to select architectures aligned with deployment constraints: XGBoost for latency-sensitive applications, hybrid ensembles for high-stakes environments, and character-level

CNNs when raw URL strings must be processed without feature engineering. As phishing tactics evolve toward AI-generated domain names and dynamic content obfuscation, such multi-paradigm frameworks will prove essential for adaptive cyber defense.

References

[1] Kaur, K., & Jain, A. K. (2025). A Survey on Phishing Attack Taxonomy, Detection Techniques, Datasets, and Security Measures. *Journal of Applied Security Research*, 1-52.
[2] Hajgude, J., & Ragha, L. (2012, October). Phish mail guard: Phishing mail detection technique by using textual and URL analysis. In *2012 World congress on information and communication technologies* (pp. 297-302). IEEE.
[3] Verma, R., Shashidhar, N., & Hossain, N. (2012, September). Detecting phishing emails the natural language way. In *European Symposium on Research in Computer Security* (pp. 824-841). Berlin, Heidelberg: Springer Berlin Heidelberg.
[4] Khonji, M., Iraqi, Y., & Jones, A. (2012). Enhancing phishing e-mail classifiers: A lexical url analysis approach. *International Journal for Information Security Research (IJISR)*, *2*(1/2), 40.
[5] James, J., Sandhya, L., & Thomas, C. (2013, December). Detection of phishing URLs using machine learning techniques. In *2013 international conference on control communication and computing (ICCC)* (pp. 304-309). IEEE.
[6] Corona, I., Giacinto, G., & Roli, F. (2013). Adversarial attacks against intrusion detection systems: Taxonomy, solutions and open issues. *Information sciences*, *239*, 201-225.
[7] Connolly, A., & Atlam, H. F. (2025). Effective ensemble learning phishing detection system using hybrid feature selection. *Journal of Network and Computer Applications*, 104251.
[8] James, J., Sandhya, L., & Thomas, C. (2013). Detection of phishing URLs using machine learning techniques. In *2013 international conference on control communication and computing (ICCC)* (pp. 304-309). IEEE.
[9] Owusu-Mensah, K., Ansong, E. D., Adu-Manu, K. S., & Yaokumah, W. (2026). A Systematic Review of Frameworks for the Detection and Prevention of Card-Not-Present (CNP) Fraud.
[10] Chandana, Anuradha, Naik, A. J., Kumar, A., & Banu, S. (2024, February). A framework for Twitter spam detection and reporting. In *AIP Conference Proceedings* (Vol. 2742, No. 1, p. 020051). AIP Publishing LLC.
[11] Ben Dalla, L. O. F., Medeni, T. D., Medeni, I. T., & Ulubay, M. (2025). Enhancing Healthcare Efficiency at Almasara Hospital: Distributed Data Analysis and Patient Risk Management. Economy: Strategy and Practice, 19(4), 54–72. https://doi.org/10.51176/1997-9967-2024-4-54-72
[12] Jayaprakasam, B. S., Dyavani, N. R., Mandala, R. R., Garikipati, V., Ubagaram, C., & Ogunmola, G. A. (2025). A novel cloud-driven adaboost framework for scalable and intelligent data analytics. *International Journal of Parallel, Emergent and Distributed Systems*, 1-29.
[13] Haji, V., & Abdulazeez, A. M. (2025). Multiclass Regression for Facial Beauty Prediction Based on Deep Learning Using SCUT-B 5500. *The Indonesian Journal of Computer Science*, *14*(6).
[14] Degirmenci, A., & Karal, O. (2022). iMCOD: Incremental multi-class outlier detection model in data streams. Knowledge-Based Systems, 258, 109950. https://doi.org/10.1016/j.knosys.2022.109950
[15] Sattar, M. U., Dattana, V., Hasan, R., Mahmood, S., Khan, H. W., & Hussain, S. (2025). Enhancing Supply Chain Management: A Comparative Study of Machine Learning Techniques with Cost–Accuracy and ESG-Based Evaluation for Forecasting and Risk Mitigation. *Sustainability*, *17*(13), 5772.
[16] Hayes, S., Aggarwal, R., Singh, H., & Wang, H. (2026). Artificial Intelligence in Business Research: A Taxonomy of Methodological Roles and Future Directions. *Available at SSRN 6024534*.
[17] Dalla, L. O. F. B. (2020). Lean Software Development Practices and Principles in Terms of Observations and Evolution Methods to increase work environment productivity. *International Journal of Engineering and Modern Technology*, *6*(1), 23-45.
[18] Joshi, S. (2025). Comprehensive Review of Artificial General Intelligence AGI and Agentic GenAI: Applications in Business and Finance. *Available at SSRN 5250611*.
[19] Ben Dalla, L., Medeni, T. M., Zbeida, S. Z., & Medeni, İ. M. (2024). Unveiling the Evolutionary Journey based on Tracing the Historical Relationship between Artificial Neural Networks and Deep

Learning. *The International Journal of Engineering & Information Technology (IJEIT)*, *12*(1), 104-110.
[20] Alsaedi, M., Ghaleb, F. A., Saeed, F., Ahmad, J., & Alasli, M. (2022). Cyber threat intelligence-based malicious URL detection model using ensemble learning. *Sensors*, *22*(9), 3373.
[21] Dalla, L. O. F. B. (2020). Dorsal Hand Vein (DHV) Verification in Terms of Deep Convolutional Neural Networks with the Linkage of Visualizing Intermediate Layer Activations Detection. International Journal of Engineering and Modern Technology E-ISSN 2504-8848 P-ISSN 2695-2149 Vol 6 No 2 2020 www.iiardpub.org
[22] Van Geest, R. J., Cascavilla, G., Hulstijn, J., & Zannone, N. (2024). The applicability of a hybrid framework for automated phishing detection. *Computers & Security*, *139*, 103736.
[23] Dalla, L. O. F. B. (2020). Convolutional Neural Network Baseline Model Building for Person Re-Identification. International Journal of Engineering and Modern Technology E-ISSN 2504-8848 P-ISSN 2695-2149 Vol. 6 No. 3 2020 www.iiardpub.org
[24] Abdul Samad, S. R., Balasubaramanian, S., Al-Kaabi, A. S., Sharma, B., Chowdhury, S., Mehbodniya, A., ... & Bostani, A. (2023). Analysis of the performance impact of fine-tuned machine learning model for phishing URL detection. *Electronics*, *12*(7), 1642.
[25] Prasad, Y. B., & Dondeti, V. (2025). PDSMV3-DCRNN: A novel ensemble deep learning framework for enhancing phishing detection and URL extraction. *Computers & Security*, *148*, 104123.
[26] Ben Dalla, L, O, F. (2021). Literature review (LR) on the dominant of Research methodology
[27] Karim, A. M., Karal, Ö., & Çelebi, F. V. (2018). A new automatic epilepsy serious detection method by using deep learning based on discrete wavelet transform. In *Proceedings of the 3rd International Conference on Engineering Technology and Applied Sciences (ICETAS)* (Vol. 4, pp. 15-18).
[28] Esen, F., Degirmenci, A., & Karal, O. (2021). Implementation of the object detection algorithm (YOLOV3) on FPGA. In *2021 innovations in intelligent systems and applications conference (ASYU)* (pp. 1-6). IEEE.
[29] Tokgöz, N., Değirmenci, A., & Karal, Ö. (2024). Machine Learning-Based Classification of Turkish Music for Mood-Driven Selection. *Journal of Advanced Research in Natural and Applied Sciences*, *10*(2), 312-328.
[30] Karal, Ö., & Dalla, L. O. F. B. (NOV. 2025). Lung Nodule Characterization in CT Scans Using Hybrid 3D Attention U-Net Segmentation and Transfer Learning-Based Classification Approach. Volume (10), Issue (37), ISSN-3014-6266, SICST2025, www.sicst.ly
[31] Dalla, L. O. B., Karal, Ö., & Degirmenciyi, A. (2025). Leveraging LSTM for Adaptive Intrusion Detection in IoT Networks: A Case Study on the RT-IoT2022 Dataset implemented On CPU Computer Device Machine. 5th International Conference on Engineering, Natural and Social Sciences, April 15-16, 2025: Konya, Turkey, 2025. Published by All Sciences Academy. https://www.icensos.com/
[32] Korkmaz, M., Kocyigit, E., Sahingoz, O., & Diri, B. (2022). A hybrid phishing detection system using deep learning-based URL and content analysis. *Elektronika ir Elektrotechnika*, *28*(5).
[33] Nagy, N., Aljabri, M., Shaahid, A., Ahmed, A. A., Alnasser, F., Almakramy, L., ... & Alfaddagh, S. (2023). Phishing urls detection using sequential and parallel ml techniques: comparative analysis. *Sensors*, *23*(7), 3467.
[34] Baghestan, A. B., & Rabbani, M. (2025, October). Security and Privacy in IoT-Enabled EEG Brain–Computer Interfaces: A Survey and Comparative Analysis. In *2025 9th International Conference on Internet of Things and Applications (IoT)* (pp. 1-6). IEEE.
[35] Johnson, R. (2025). *SOAR Technology and Implementation: Definitive Reference for Developers and Engineers*. HiTeX Press.
[36] Sucharitha, M. M., Kumar, J. S., Sateesha, G., Prasad, M. R., & Basha, M. S. A. (2025). A shap-enhanced PCA-DBSCAN framework for interpretable retail customer segmentation and strategic insight. *International Journal of System Assurance Engineering and Management*, 1-65.
[37] Karim, A., Shahroz, M., Mustofa, K., Belhaouari, S. B., & Joga, S. R. K. (2023). Phishing detection system through hybrid machine learning based on URL. *IEEE Access*, *11*, 36805-36822.
[38] Alsubaei, F. S., Almazroi, A. A., & Ayub, N. (2024). Enhancing phishing detection: A novel hybrid deep learning framework for cybercrime forensics. *IEEE Access*, *12*, 8373-8389.

[39] Ujah-Ogbuagu, B. C., Akande, O. N., & Ogbuju, E. (2024). A hybrid deep learning technique for spoofing website URL detection in real-time applications. *Journal of Electrical Systems and Information Technology*, *11*(1), 7.
[40] Ozcan, A., Catal, C., Donmez, E., & Senturk, B. (2023). A hybrid DNN–LSTM model for detecting
[41] Chen, S., Yu, L., Zhang, C., Wu, Y., & Li, T. (2023). Environmental impact assessment of multi-source solid waste based on a life cycle assessment, principal component analysis, and random forest algorithm. *Journal of Environmental Management*, *339*, 117942.
[42] Pan, S., Zheng, Z., Guo, Z., & Luo, H. (2022). An optimized XGBoost method for predicting reservoir porosity using petrophysical logs. *Journal of Petroleum Science and Engineering*, *208*, 109520.
[43] Anyanwu, G. O., Nwakanma, C. I., Lee, J. M., & Kim, D. S. (2022). Optimization of RBF-SVM kernel using grid search algorithm for DDoS attack detection in SDN-based VANET. *IEEE Internet of Things Journal*, *10*(10), 8477-8490.
[44] Zheng, F., Yan, Q., Leung, V. C., Yu, F. R., & Ming, Z. (2022). HDP-CNN: Highway deep pyramid convolution neural network combining word-level and character-level representations for phishing website detection. *Computers & Security*, *114*, 102584.
[45] Aslam, S., Aslam, H., Manzoor, A., Chen, H., & Rasool, A. (2024). AntiPhishStack: LSTM-based stacked generalization model for optimized phishing URL detection. *Symmetry*, *16*(2), 248.