Enhancing Phishing URL Detection Using a Two-Level Rule-Based Framework Combining Lexical and RDAP Registration Features

Wan Afifie Aliff Bin Wan Abdullah[1], Zulkiflee Bin Muslim[2*], Haniza Nahar[3], Radzi Motsidi[4]

[1]ADTEC JTM Kampus Selandar, 77500 Selandar, Melaka

[2]Faculty of Artificial Intelligence and Cyber Security, Universiti Teknikal Malaysia Melaka (UTeM), Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

[3]Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM), Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

[4]School of Technical Foundation and Diploma Studies, Universiti Teknikal Malaysia Melaka (UTeM), Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

***Corresponding author**

DOI: https://dx.doi.org/10.47772/IJRISS.2026.100800025

Received: 14 August 2026; Accepted: 19 August 2026; Published: 25 August 2026

ABSTRACT

Phishing remains one of the most persistent cyber threats, and almost every campaign ultimately depends on a deceptive Uniform Resource Locator (URL). Existing defences face a structural trade-off: blacklists are reactive and cannot cover newly registered domains during the zero-hour window, while machine-learning detectors, although accurate, are opaque, feature-hungry, and often depend on page content or full DNS telemetry that many organisations cannot collect. This study proposes and evaluates a lightweight, fully interpretable two-level rule-based framework that fuses lexical URL features with domain registration evidence retrieved through the Registration Data Access Protocol (RDAP). Level 1 scores each URL using five transparent lexical rules derived from training-set distributions of domain length, number of dots, number of hyphens, number of digits, and URL entropy. Level 2 applies three RDAP rules covering domain age, days to expiry, and a missing-registration-data flag, targeting the young, short-lived, and poorly documented domains that characterise phishing infrastructure. The two levels are combined through logical OR and AND decision fusion and evaluated on a balanced, held-out set of 400 URLs drawn from a curated corpus of 800. Level 1 achieved 95.50% accuracy (precision 0.9789, recall 0.9300); Level 2 achieved perfect recall (1.0000) at 0.8969 precision; OR fusion preserved perfect recall; and AND fusion delivered the best overall result at 96.50% accuracy with perfect precision, zero false positives, and a Matthews Correlation Coefficient of 0.9323. A confusion-matrix decomposition further shows that the false-positive sets of the two levels are completely disjoint, confirming that lexical and registration evidence fail independently. Exploiting this, a cascaded implementation of AND fusion reproduces identical decisions while issuing RDAP queries for only 47.5% of URLs, a 52.5% reduction in external lookups.

Keywords: Phishing detection, URL lexical features, RDAP, rule-based classification, decision fusion, interpretable security

INTRODUCTION

The migration of commerce, education, and public services to online platforms has made phishing one of the most durable categories of cybercrime. Attackers exploit ordinary communication channels such as electronic mail, instant messaging, and web interfaces to persuade users to disclose credentials or financial details, and the resulting messages are frequently indistinguishable from genuine corporate correspondence [1], [3]. Campaign themes track current events closely: during the COVID-19 period attackers pivoted within weeks to

pandemic-related lures that exploited public anxiety and uncertainty [2]. Routine actions that users reasonably assume to be safe, such as signing in to an account or following a link in a notification, therefore carry real risk.

Phishing has also become substantially more targeted. Attackers now mine personal information, organisational roles, and publicly available records so that each message appears contextually relevant to its recipient [4]. Spear-phishing and whaling campaigns aimed at specific employees or executives amplify organisational impact, because a single successful compromise can expose financial assets, confidential records, and critical systems [3]. The consequences extend beyond immediate monetary loss to identity theft, operational disruption, and reputational damage [5], as well as measurable psychological harm and erosion of trust in digital services among victims [6]. Recent work further shows that large language models materially reduce the cost of producing fluent, individualised lures at scale, which shifts the economics of the attack in the adversary's favour [7].

At the infrastructure layer, the naming system itself has become part of the attack. Fast-flux and name-server flux networks allow malicious sites to remain reachable while continuously changing their technical footprint [8], and domain generation algorithms together with look-alike registrations permit the rapid creation of large numbers of deceptive domains that closely resemble legitimate services [9]. Phishing is consequently not only a problem of deceptive content but a structural problem concerning how domains are registered, provisioned, and perceived.

Operationally, however, the vast majority of deployed defences still rest on blacklists, simple URL heuristics, and content inspection at the mail or web layer. These mechanisms are effective against known threats but are inherently reactive: a malicious URL must be observed, reported, and verified before it can be blocked, leaving a zero-hour interval during which the campaign operates unimpeded [11], [12]. Machine-learning and deep-learning detectors close part of this gap and routinely report accuracies above 95% [10], [32], [35], but they typically require large labelled corpora, dozens of engineered features, page content or full DNS telemetry, and produce decisions that practitioners cannot easily audit. Reliability studies show that such detectors also degrade under distribution shift and adversarial URL manipulation [39], which makes their opacity an operational liability rather than a purely academic concern.

Two comparatively inexpensive sources of evidence remain underexploited in combination. The first is the lexical structure of the URL itself, which is available at zero marginal cost and requires no network interaction. The second is the registration record of the domain behind the URL. The Registration Data Access Protocol (RDAP) is the modern, structured successor to WHOIS and returns registration and expiry events in machine-readable JSON, with standardised extensions for search and reverse search [21], [22], [23]. Large-scale measurement confirms that newly registered domains are disproportionately implicated in phishing [25], making registration lifecycle a genuinely discriminative signal. Yet the literature has largely treated registration metadata as one undifferentiated block within broad "network features", rather than as a small set of first-class lifecycle indicators that can be reasoned about explicitly.

This study addresses that gap. It designs, implements, and evaluates a two-level rule-based framework in which Level 1 operates purely on lexical URL properties and Level 2 operates purely on RDAP-derived lifecycle properties, with the two decisions combined through explicit logical fusion. The framework deliberately forgoes learned models in favour of thresholds that an analyst can read, audit, and adjust. The specific contributions are as follows.

- A compact, fully interpretable two-level detection framework requiring only a URL string and a single RDAP lookup, with no page retrieval, no TLS inspection, and no passive DNS telemetry.
- An empirical evaluation of each level in isolation and of two fusion strategies on a balanced held-out test set, reported with accuracy, precision, recall, F1-score, and an extended diagnostic panel comprising specificity, false-positive rate, negative predictive value, balanced accuracy, Youden's J, and the Matthews Correlation Coefficient.

- A confusion-matrix decomposition establishing that the false-positive sets produced by the lexical and RDAP levels are completely disjoint, which quantifies the complementarity of the two evidence types rather than merely asserting it.

- A cascaded deployment variant that reproduces the decisions of AND fusion exactly while reducing external RDAP queries by 52.5%, together with explicit guidance on which operating point suits monitoring versus automated blocking.

The remainder of the paper is organised as follows. The next section reviews the phishing detection literature and positions the research gap. The methodology section describes the dataset, feature definitions, rule derivation, fusion logic, and evaluation protocol. The results section reports feature separation, rule sets, and detection performance. The discussion interprets these findings, compares them with published approaches, and derives deployment guidance, before the paper concludes.

LITERATURE REVIEW

Blacklists and Heuristic Filtering

URL blacklists such as PhishTank, Google Safe Browsing, and the Anti-Phishing Working Group database remain the first line of defence in browsers, mail gateways, and security platforms. Their appeal is operational: they return a binary verdict at negligible computational cost, which suits environments processing millions of URLs, and they double as a labelling source for research datasets [13]. Modern implementations are layered, consulting a local cache for speed and escalating to online verification when the local check is inconclusive [12].

The limitation is structural rather than incidental. A blacklist can only block what has already been observed, reported, and verified, so every campaign enjoys a zero-hour window before listing [11]. Adversaries widen that window deliberately. Cloaking infrastructure serves benign content to crawlers and security scanners while delivering the phishing page to real victims [15], delaying discovery for the duration of the campaign. A large-scale analysis of 1.577 billion URLs across 95 scanners found substantial and inconsistent lead and lag effects between vendors, with correlated and noisy reporting that undermines naive threshold or majority-vote aggregation [14].

Heuristic filters were introduced to cover the same gap without requiring prior observation, flagging unusual domain tokens, redirect chains, and registration anomalies [13]. They too are bounded by their own patterns: token repositioning, homoglyph substitution, and structural rewriting evade fixed heuristics [12], while aggressive thresholds generate false positives on legitimate but unusual URLs. The consensus in this literature is that neither mechanism suffices alone and that detection must move earlier, toward infrastructure-level indicators observable before a phishing page is rendered.

Content-Based and Learning-Based Detection

Content-based filtering analyses message text or rendered page appearance. Deep-learning classifiers over email text report accuracies around 93% and outperform header heuristics alone [18], [19], while visual approaches compare rendered logos, dominant colours, and layout against trusted references and resist source-code obfuscation because they operate after rendering [20].

These methods degrade sharply under contemporary evasion. Client-side cloaking, in which JavaScript assembles the phishing page only inside the victim's browser, defeats static HTML analysis outright, and hosting on reputable cloud providers neutralises the domain-reputation signals on which many content pipelines depend [17]. Automated cloning compounds the problem: a study of 13,394 samples identified 8,566 cloned phishing pages produced by seven distinct cloning techniques, none of which was detected by the security vendors monitored over a seven-day observation window [16]. Content-based detection also carries an unavoidable cost, since the page must be fetched and rendered before a verdict is possible.

URL-only machine learning avoids that cost. Comparative studies consistently report strong results for ensemble methods, with Random Forest reaching 97% accuracy against Decision Tree at 96% and k-Nearest Neighbours at 94% on an 11,055-URL corpus [33], 98.80% on PhishTank and 97.87% on the UCI dataset [34], and above 99% for Extra Trees and boosting variants when cross-validation is applied [35]. Genetic-algorithm feature selection reduced a 73-feature space to 44 while cutting inference time by 36.62% [36], and genetic optimisation of decision-forest composition reached detection accuracies between 97.26% and 98.37% across three UCI corpora [41], and a recent IJRISS study combining genetic hyperparameter search with k-fold cross-validation on the 235,795-record PhiUSIIL corpus reported 100% accuracy for Random Forest and Decision Tree and 99.87% for k-Nearest Neighbours [37].

Three caveats temper these figures. First, they are obtained on heterogeneous corpora with differing class balances and feature definitions, so cross-study comparison is unreliable; feature engineering choices alone materially move reported performance [38]. Second, near-perfect scores on a single benchmark invite concern about leakage and overfitting, and robustness analyses show that such detectors lose accuracy under distribution shift and deliberate URL perturbation [39]. Third, the resulting models are difficult to audit, which has motivated a distinct line of work on explainable feature selection for phishing detection [40]. Interpretability is therefore not a stylistic preference but a recognised requirement.

Feature Design, DNS Taxonomies, and Registration Data

Across recent surveys, feature representation influences detection outcomes at least as strongly as algorithm choice [10]. Lexical features are the most widely adopted group because they are extractable without external services and impose no latency; carefully selected lexical sets support URL-only detectors that are accurate without being complex [11]. Domain length, subdomain depth, digit and special-character counts, and randomness measures recur as informative indicators, with entropy over the second-level domain gaining importance as campaigns adopt higher-entropy labels to defeat simple string matching [10].

DNS-oriented taxonomies organise features into lexical, network-behavioural, and host or registration groups. Lexical descriptors help expose algorithmically generated names [29] and are treated as an evolving rather than fixed set, since adversaries adapt naming strategies over time [27]. Decision-tree frameworks demonstrate measurable gains when lexical cues are combined with other DNS-derived signals [28]. Network-behavioural features such as query volume, temporal beaconing, and resolution patterns add context but require passive DNS collection [27], [29], and ensemble detectors built on such signals must additionally be engineered to absorb concept drift as domain behaviour evolves [26]. More advanced designs fuse character-level domain representations with domain-to-IP graph structure through attention mechanisms [30], or model DNS activity as time series to expose tunnelling and exfiltration [31]. These approaches are powerful but presuppose packet-level logs or passive DNS feeds that many organisations cannot obtain for privacy, cost, or architectural reasons.

Registration metadata sits between these extremes. Domain age is repeatedly identified as a strong indicator because newly registered and short-lived domains correlate with spam and phishing campaigns [10], [27], and large-scale measurement of newly registered domains confirms their disproportionate role in phishing infrastructure [25]. RDAP provides this evidence in structured, machine-readable form and is explicitly recommended as the successor to legacy WHOIS for abuse investigation [21], with the protocol family continuing to mature through standardised search extensions [22], [23], [24]. Despite this, registration attributes are usually absorbed into broad network feature blocks rather than modelled as explicit lifecycle indicators, and their standalone discriminative power is rarely reported.

Research Gap

The literature therefore leaves a specific space unoccupied. High-performing detectors rely on rich telemetry, large feature sets, and learned models whose decisions resist audit. Lexical-only detectors are inexpensive but discard the registration lifecycle entirely. Registration-aware work treats RDAP as one input among dozens rather than as a compact, independently evaluable evidence channel. Almost no published work quantifies how much can be achieved using only a small, deliberately chosen set of lexical features together with a minimal

set of RDAP lifecycle indicators, combined through explicit logical rules whose behaviour a security analyst can inspect and modify.

This study occupies that space. It restricts itself to eight features across two evidence types, derives every threshold transparently from training-set distributions, and reports not only aggregate accuracy but the structure of the errors each level makes, so that the complementarity between lexical and registration evidence can be measured rather than assumed.

METHODOLOGY

Framework Overview

The proposed framework classifies a URL through two independent rule engines whose outputs are combined by an explicit fusion operator, as shown in Figure 1. Level 1 examines only the URL string and requires no network interaction. Level 2 examines only the RDAP registration record of the corresponding registered domain. Neither engine consults the other, so each produces an interpretable verdict that can be inspected in isolation, and the fusion operator is the single point at which the two evidence types interact. This separation is deliberate: it permits the contribution of each evidence channel to be measured independently and allows the operating point of the deployed system to be changed by substituting the fusion operator alone, without retraining or redesigning either engine.

The study follows a three-phase experimental design. Phase one constructs and partitions the dataset. Phase two extracts features and derives rules using only the training partition. Phase three evaluates the fixed rule sets and fusion operators on the held-out test partition. No test-partition information was used at any point during rule derivation.

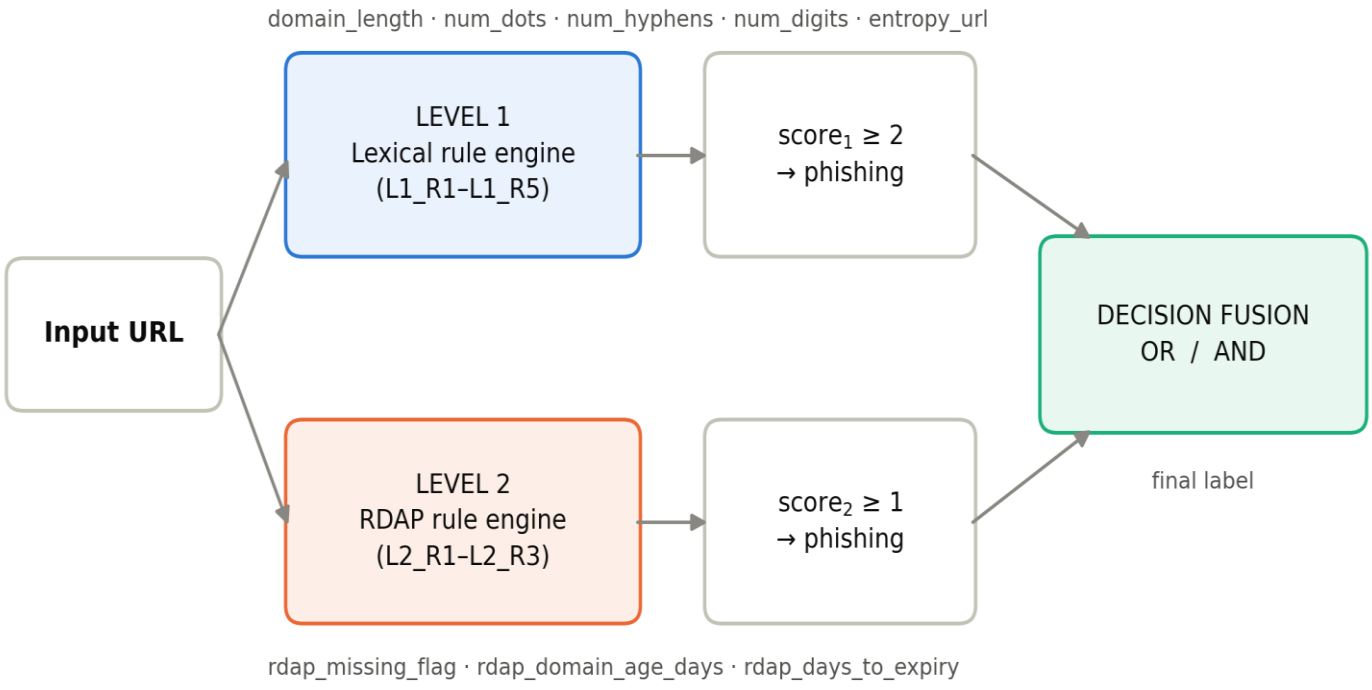

Figure 1: Architecture of the proposed two-level rule-based phishing URL detection framework.

Dataset Construction

A curated offline corpus of 800 URLs was assembled, comprising 400 phishing and 400 legitimate instances. Phishing URLs were drawn from public phishing reporting platforms that publish verified indicators of compromise, retaining only entries marked as active or recently observed. Legitimate URLs were drawn from research-oriented popularity rankings and other trusted benign sources, retaining only well-known, general-purpose services. Cleaning removed duplicates, malformed entries, URLs appearing in both classes, and URLs

resolving to error pages or parked domains. Labelling was source-based: entries surviving cleaning from phishing feeds were labelled phishing, and entries from popularity rankings were labelled legitimate.

Each retained domain was then queried against public RDAP services over HTTPS, and the structured responses were stored locally. Because collection was performed once and the corpus subsequently frozen, every experiment in this study operates on identical data, and the RDAP-derived temporal features are anchored to a single fixed collection date.

The corpus was partitioned by stratified random split into a training subset of 400 URLs (200 phishing, 200 legitimate) and a test subset of 400 URLs (200 phishing, 200 legitimate). Class balance was preserved in both partitions so that accuracy remains an interpretable summary statistic and no metric is inflated by prior skew. The training subset was used exclusively for descriptive analysis and threshold selection; the test subset was reserved for final evaluation.

Feature Definitions

Eight features were extracted per URL, five lexical and three RDAP-derived, as defined in Table 1. The set was kept deliberately small so that every rule remains individually explicable and so that the marginal contribution of each evidence type is not obscured by dozens of correlated attributes.

Lexical features are computed after parsing the URL to isolate the hostname and identify the registered domain. Counting features capture the segmentation and obfuscation typical of impersonating domains, while URL entropy quantifies character randomness and captures algorithmically generated or deliberately scrambled labels that counting features alone would miss.

RDAP features are derived from the creation and expiry events in the registration record. Domain age in days measures elapsed time since registration; days to expiry measures remaining registration validity; and the missing flag records whether either date is absent from the response. The third feature is important in its own right, because incomplete registration disclosure is itself informative rather than merely inconvenient: treating a missing record as a neutral value would discard evidence about the registrar and top-level domain the attacker selected.

Feature	Level	Type	Definition
domain_length	1	Integer	Total number of characters in the registered domain string.
num_dots	1	Integer	Number of dot characters in the domain, representing subdomain depth.
num_hyphens	1	Integer	Number of hyphen characters in the domain.
num_digits	1	Integer	Number of digit characters in the domain.
entropy_url	1	Real	Shannon entropy of the URL string, estimating character randomness.
rdap_domain_age_days	2	Integer	Days elapsed since the RDAP registration (creation) event.
rdap_days_to_expiry	2	Integer	Days remaining until the RDAP expiry event.
rdap_missing_flag	2	Binary	Set to 1 when the registration or expiry date is absent from the RDAP response.

Table 1: Feature definitions used by the two-level framework.

Rule Derivation

Thresholds were selected by comparing the class-conditional distribution of each feature on the training subset and placing the cut point in the region where phishing instances begin to outnumber legitimate ones. No optimiser was applied and no threshold was tuned against the test partition, so the rule sets are reproducible from the reported descriptive statistics alone.

Level 1 uses additive scoring rather than a conjunction of conditions. Each of the five lexical rules contributes one point when satisfied, and a URL is classified as phishing when the resulting score reaches two. Requiring corroboration from at least two independent lexical signals prevents a single incidental property, such as one hyphen in an otherwise ordinary domain, from producing a false alarm, while preserving the ability to flag URLs that are individually unremarkable on every dimension but jointly anomalous.

Level 2 also accumulates points but applies a lower firing threshold of one, reflecting the different character ofregistration evidence. A domain registered days before observation, or one whose registrar declines to publish creation and expiry dates, is anomalous on its own terms rather than as part of a pattern, so demanding corroboration would suppress precisely the signal the level exists to capture. The third Level 2 rule is a deliberate conjunction of youth and short remaining validity, which describes the disposable registration profile typical of short-lived campaign infrastructure.

Decision Fusion

Let d1 and d2 denote the binary verdicts of Level 1 and Level 2, where 1 indicates phishing. Two fusion operators were evaluated. OR fusion assigns the phishing label when d1 = 1 or d2 = 1, maximising the probability that an attack is caught at the cost of admitting every false positive raised by either engine. AND fusion assigns the phishing label only when d1 = 1 and d2 = 1, requiring independent corroboration from both evidence types and therefore suppressing any false positive not produced by both engines simultaneously.

Both operators were fixed before evaluation and applied uniformly across the test set. Together with the two single-level configurations, this yields four operating points, which the results section evaluates on identical data.

Evaluation Metrics

All configurations were evaluated on the held-out test set using the confusion matrix, where true positives (TP) are phishing URLs correctly identified, true negatives (TN) are legitimate URLs correctly identified, false positives (FP) are legitimate URLs incorrectly flagged, and false negatives (FN) are phishing URLs missed. Accuracy, precision, recall, and F1-score are defined conventionally as (TP + TN) / (TP + TN + FP + FN), TP / (TP + FP), TP / (TP + FN), and the harmonic mean of precision and recall respectively.

Because accuracy and F1-score are insensitive to parts of the confusion matrix that matter operationally, four additional diagnostics are reported. Specificity, TN / (TN + FP), and its complement the false-positive rate quantify the burden imposed on legitimate traffic. Negative predictive value, TN / (TN + FN), expresses how much a benign verdict can be trusted, which is the quantity a triage analyst actually relies upon when deciding not to investigate. Balanced accuracy, the mean of recall and specificity, and Youden's J statistic, recall + specificity − 1, summarise performance symmetrically across classes. The Matthews Correlation Coefficient, defined in Equation (1), is reported because it is the only single-figure summary that incorporates all four confusion-matrix cells and remains informative when a classifier is strongly skewed toward one class.

$$MCC = (TP \times TN - FP \times FN) / \sqrt{((TP + FP)(TP + FN)(TN + FP)(TN + FN))} \quad (1)$$

Metrics were computed for Level 1 alone, Level 2 alone, OR fusion, and AND fusion. Training-set performance is also reported for the two configurations for which it is available, in order to estimate the generalisation gap.

Implementation Environment

Experiments were conducted offline on Ubuntu 24.04.4 LTS using Python 3.12.3 with pandas and NumPy for data handling and metric computation, standard URL parsing and HTTP libraries for domain extraction and RDAP retrieval, and matplotlib for visualisation. Because the corpus, the partition, and the rule thresholds are all fixed, every reported figure is deterministically reproducible.

RESULTS

Lexical Feature Separation

Table 2 reports the class-conditional descriptive statistics for the five lexical features on the training subset, and Figure 2 visualises the corresponding class means.

Feature	Class	Mean	Median	Min	Max
Domain length	Legitimate	11.42	11.00	4	26
Domain length	Phishing	27.23	27.00	14	71
Number of dots	Legitimate	1.01	1.00	1	2
Number of dots	Phishing	2.04	2.00	1	3
Number of hyphens	Legitimate	0.10	0.00	0	2
Number of hyphens	Phishing	1.06	1.00	0	10
Number of digits	Legitimate	0.08	0.00	0	3
Number of digits	Phishing	1.75	3.00	0	25
URL entropy	Legitimate	3.59	3.60	2.85	4.18
URL entropy	Phishing	4.15	4.15	3.59	4.76

Table 2: Descriptive statistics for lexical features (training subset, n = 400).

Separation is pronounced on every dimension. Phishing domains are on average 2.4 times longer than legitimate ones (27.23 against 11.42 characters), carry roughly twice as many dots, and contain an order of magnitude more hyphens and digits. The count-based features show a distinctive asymmetry: legitimate domains cluster tightly at low values, with medians of zero for both hyphens and digits, whereas phishing domains occupy a wide and heavy-tailed range extending to 10 hyphens and 25 digits. This asymmetry is what makes simple upper-bound thresholds effective, because the legitimate class is compact rather than merely lower on average.

Entropy behaves differently and is informative for that reason. The class means differ by only 0.56 nats and the ranges overlap substantially between 3.59 and 4.18, so entropy is a weak discriminator in isolation. Its value lies in independence: a phishing domain that is short and free of separators, and therefore invisible to the four counting rules, may still register elevated randomness. Entropy accordingly functions as a corroborating rather than a primary signal, which is precisely the role the additive scoring scheme assigns to it.

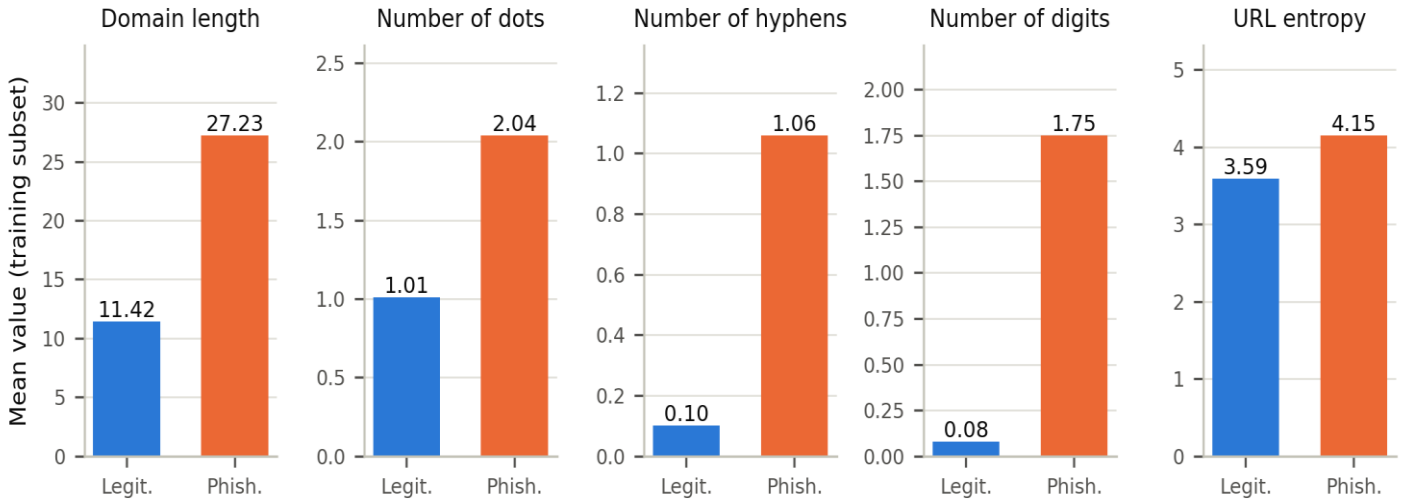

Figure 2: Class-conditional means of the five lexical features on the training subset. Each panel uses its own vertical scale.

RDAP Feature Separation

Table 3 and Figure 3 report the corresponding statistics for the two continuous RDAP features.

Feature	Class	Mean	Median	Min	Max
Domain age (days)	Legitimate	7737.28	7881.50	784	14409
Domain age (days)	Phishing	73.25	32.50	1	150
Days to expiry	Legitimate	635.58	274.00	1	3335
Days to expiry	Phishing	290.75	331.50	214	363

Table 3: Descriptive statistics for RDAP-based features (training subset, n = 400).

Domain age provides the sharpest separation observed anywhere in the study. Legitimate domains average 7,737 days, approximately 21 years, with a minimum of 784 days, while phishing domains average 73 days with a median of 32.5 and a maximum of 150. The two ranges do not overlap at all in this corpus: the youngest legitimate domain is more than five times older than the oldest phishing domain. This single feature is therefore almost perfectly separating on the training data, which explains why Level 2 attains complete recall on the test set.

Days to expiry is weaker but structurally interesting. Legitimate domains span a broad range from 1 to 3,335 days, reflecting heterogeneous renewal practice, whereas phishing domains are confined to a narrow band between 214 and 363 days. The phishing distribution is consistent with single-year registrations purchased shortly before use, and its narrowness rather than its central tendency is what carries information. Because the legitimate range fully contains the phishing range, this feature cannot separate the classes alone, which is why it appears in the rule set only in conjunction with domain age.

The missing-data flag showed the expected asymmetry, with most legitimate domains returning complete registration records and a substantial proportion of phishing domains missing creation dates, expiry dates, or both. This reflects the tendency of phishing campaigns to select registrars and top-level domains with limited disclosure, and it justifies encoding absent data as a positive risk indicator rather than as a neutral value.

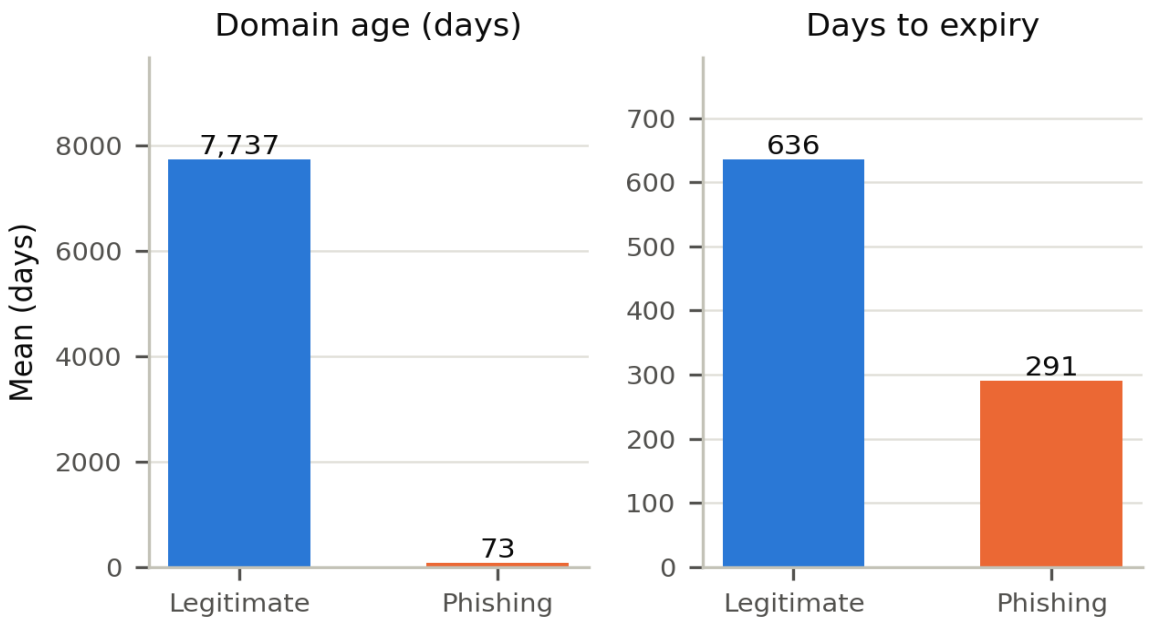

Figure 3: Class-conditional means of the two continuous RDAP features on the training subset.

Derived Rule Sets

Tables 4 and 5 present the rule sets derived from the distributions above.

Rule ID	Feature	Condition	Score
L1_R1	domain_length	domain_length > 20	1
L1_R2	num_dots	num_dots ≥ 2	1
L1_R3	num_hyphens	num_hyphens ≥ 1	1
L1_R4	num_digits	num_digits ≥ 1	1
L1_R5	entropy_url	entropy_url > 4.0	1
L1_FINAL	level1_score	level1_score ≥ 2 → phishing; otherwise legitimate	—

Table 4: Level 1 lexical rule set.

Each threshold sits comfortably inside the observed separation rather than at its boundary. The domain-length cut of 20 characters lies between the legitimate maximum of 26 and the phishing minimum of 14, deliberately positioned so that the rule fires on the bulk of the phishing distribution without being brittle. The domain-age threshold of 365 days is conservative by an order of magnitude: it is more than twice the oldest phishing domain observed (150 days) yet less than half the youngest legitimate domain (784 days), leaving a wide margin on both sides. Thresholds chosen with this much headroom are less likely to require adjustment when the corpus changes, which matters for a framework whose maintenance model is manual revision rather than retraining.

Rule ID	Rule name	Condition	Score
L2_R1	RDAP missing	rdap_missing_flag = 1	1

Rule ID	Rule name	Condition	Score
L2_R2	Young domain	rdap_domain_age_days < 365	1
L2_R3	Young and short expiry	rdap_domain_age_days < 365 AND rdap_days_to_expiry < 180	1
L2_FINAL	Final decision	level2_score ≥ 1 → phishing; otherwise no RDAP flag	—

Table 5: Level 2 RDAP-based rule set.

Detection Performance

Table 6 reports the test-set confusion matrices and primary metrics for all four configurations, and Figure 4 compares them visually.

Configuration	Accuracy	Precision	Recall	F1-score	TN	FP	FN	TP
Level 1 (lexical)	0.955	0.9789	0.93	0.9538	196	4	14	186
Level 2 (RDAP)	0.9425	0.8969	1	0.9456	177	23	0	200
Fusion OR	0.9325	0.8811	1	0.9368	173	27	0	200
Fusion AND	0.965	1	0.93	0.9637	200	0	14	186

Table 6: Test-set performance of Level 1, Level 2, and the two fusion strategies (n = 400; 200 phishing, 200 legitimate).

Each configuration occupies a distinct and predictable position. Level 1 is the stronger standalone engine, reaching 95.50% accuracy with only 4 false positives from 200 legitimate URLs, but it misses 14 phishing URLs. Level 2 inverts this profile: it detects every phishing URL in the test set, yet raises 23 false positives, reducing precision to 0.8969. OR fusion inherits Level 2's perfect recall but accumulates the false positives of both engines, producing the lowest accuracy of the four at 93.25%. AND fusion delivers the best overall result, 96.50% accuracy with perfect precision and not a single false positive across 200 legitimate URLs, while retaining Level 1's recall of 0.9300.

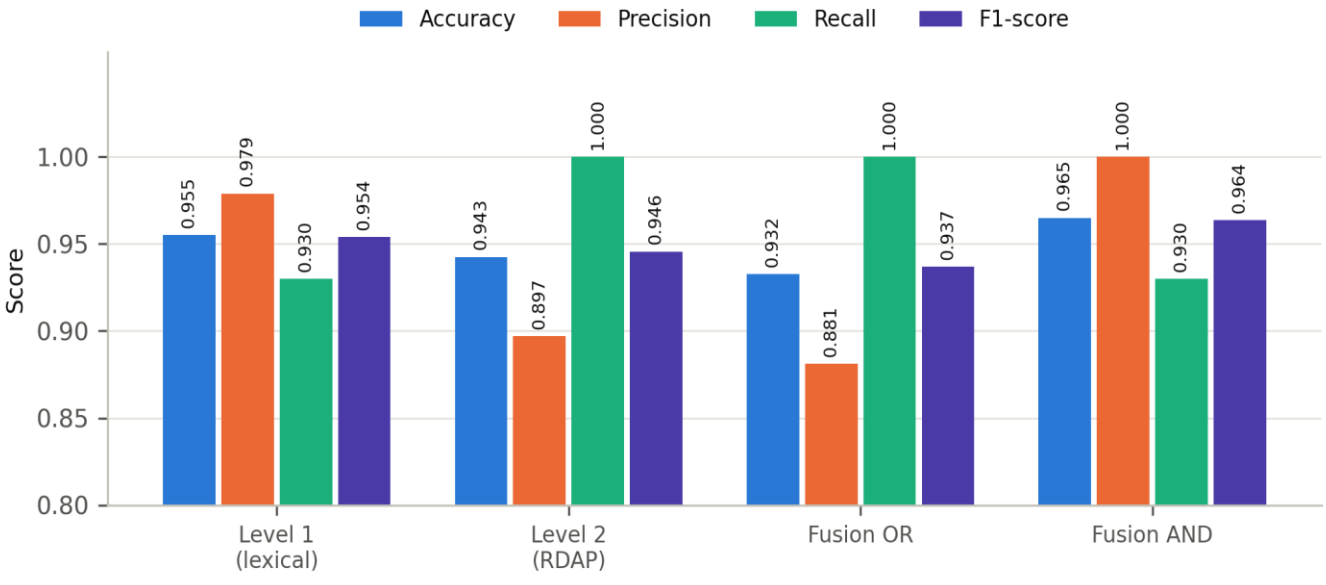

Figure 4: Test-set performance of the four configurations across accuracy, precision, recall, and F1-score.

Extended Diagnostics and Error Structure

Table 7 reports the extended diagnostic panel computed from the same confusion matrices.

Configuration	Specificity	FPR	NPV	Balanced accuracy	Youden's J	MCC
Level 1 (lexical)	0.98	0.02	0.9333	0.955	0.91	0.9111
Level 2 (RDAP)	0.885	0.115	1	0.9425	0.885	0.8909
Fusion OR	0.865	0.135	1	0.9325	0.865	0.873
Fusion AND	1	0	0.9346	0.965	0.93	0.9323

Table 7: Extended diagnostics derived from the test-set confusion matrices.

The extended panel sharpens the comparison in ways the primary metrics obscure. AND fusion achieves the highest Matthews Correlation Coefficient at 0.9323 and the highest Youden's J at 0.9300, confirming that its advantage is genuine rather than an artefact of the balanced class distribution. Its false-positive rate of exactly zero is the operationally decisive figure for any automated blocking deployment, where each false positive denies a user access to a legitimate service.

Negative predictive value exposes a property invisible in Table 6. Both Level 2 and OR fusion achieve an NPV of 1.0000, meaning that a benign verdict from either configuration was correct in every instance. This is the quantity that governs triage: an analyst who dismisses a URL cleared by Level 2 was, on this corpus, never wrong to do so. AND fusion, by contrast, has an NPV of 0.9346, so roughly one benign verdict in fifteen is mistaken. The two configurations therefore answer different operational questions, and the choice between them is not reducible to which has the higher accuracy.

The structure of the errors is more informative still. Level 1 produced 4 false positives and Level 2 produced 23. OR fusion, whose false-positive set is the union of the two, produced exactly 27 — precisely 4 + 23. AND fusion, whose false-positive set is the intersection, produced exactly 0. Since $|A \cup B| = |A| + |B|$ and $|A \cap B| = 0$ hold simultaneously, the two sets are provably disjoint: no legitimate URL in the test set was misclassified by both engines. Lexical anomaly and registration anomaly therefore fail on entirely different legitimate domains. This is a stronger statement than observing that the two levels perform comparably, and it is the reason AND fusion can eliminate false positives entirely rather than merely reducing them.

The same decomposition applies to false negatives with the opposite sign. Level 2 has zero false negatives, so every one of the 14 phishing URLs missed by Level 1 was caught by Level 2. The evidence needed to reach perfect recall was therefore present in the system, and OR fusion recovers it; AND fusion discards it by construction, since a phishing URL invisible to Level 1 cannot survive a conjunction. The 14 missed URLs are consequently not a limitation of the framework as a whole but a consequence of the fusion operator chosen, which is a design decision rather than a performance ceiling.

Figure 5 places the four configurations in precision–recall space against iso-F1 contours. Level 1 and AND fusion cluster in the high-precision region at recall 0.9300, while Level 2 and OR fusion occupy the perfect-recall boundary at lower precision. The four points span an F1 range of only 0.0269, from 0.9368 to 0.9637, but a precision range of 0.1189 and a recall range of 0.0700. Summary statistics alone would therefore make the configurations appear nearly equivalent when their operational behaviour differs substantially.

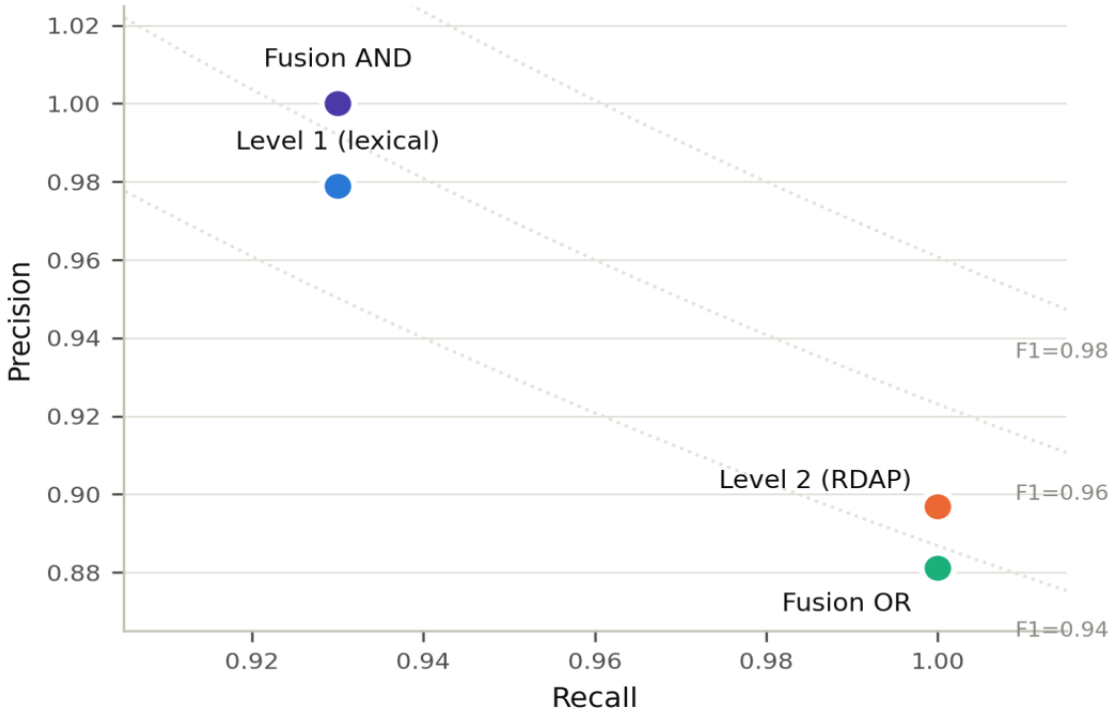

Figure 5: Operating points of the four configurations in precision–recall space, with iso-F1 contours.

Generalisation

Training-set accuracy was 0.9725 for Level 1 and 0.9850 for AND fusion, against test-set values of 0.9550 and 0.9650 respectively. The generalisation gap is therefore 1.75 percentage points for Level 1 and 2.00 points for AND fusion. Level 2 and OR fusion retained perfect recall on both partitions with correspondingly elevated false-positive counts, reproducing their test-set behaviour.

Gaps of this magnitude indicate that the thresholds transfer to unseen data rather than encoding training-set idiosyncrasies. This is the expected consequence of the derivation procedure: five thresholds selected by visual inspection of class distributions have far fewer degrees of freedom than a fitted model, so there is comparatively little capacity to overfit. Low variance is not the same as low bias, however, and the same rigidity that limits overfitting here is what would limit adaptation to a shifted distribution.

DISCUSSION

Complementarity of Lexical and Registration Evidence

The central empirical finding is that the two evidence types fail independently. Their false-positive sets are disjoint, and the phishing URLs missed by Level 1 are entirely covered by Level 2. This is not coincidental. A legitimate domain triggers Level 1 when it happens to be long, segmented, or hyphenated — properties associated with organisational naming conventions and regional service structures, which have nothing to do with registration behaviour. A legitimate domain triggers Level 2 when it is recently registered or when its registrar publishes incomplete records — properties of the registration channel, which have nothing to do with the character composition of the name. The two error modes are generated by unrelated processes, so their independence is a structural property of the feature design rather than a fortunate artefact of this corpus.

This matters for the fusion strategy in a way that mere performance comparison does not capture. Conjunctive fusion suppresses false positives to the extent that the constituent detectors err independently; if both levels made mistakes on the same legitimate domains, AND fusion would inherit those errors rather than eliminating them. The perfect precision observed here is therefore a measurable consequence of evidence-type diversity, and it suggests a design principle for multi-level detection generally: the value of adding a level is determined less by its standalone accuracy than by the degree to which its errors are uncorrelated with those of the existing levels.

Comparison with Published Approaches

Table 8 positions the framework against representative URL-based detectors from the recent literature. Direct numerical comparison would be inappropriate, since these studies use different corpora, class balances, and feature counts, and the present corpus is markedly smaller. The comparison is offered to situate the framework's operating characteristics, not to claim superiority.

Study	Approach	Features	Dataset size	Reported accuracy	Interpretable
Kayode-Ajala [33]	RF / DT / KNN	30	11,055	0.97 (RF)	No
Choudhary et al. [34]	RF and four others	—	10,000 / 11,055	0.9880 (RF)	No
Awasthi & Goel [35]	Ensemble + k-fold	30	2,456 / 11,055	> 0.99	No
Kocyigit et al. [36]	GA feature selection + RF	44 of 73	—	0.9293	No
Chong et al. [37]	KNN / RF / DT + GA	—	235,795	1.0000 (RF, DT)	No
This study	Two-level rules + AND fusion	8	800	0.965	Yes

Table 8: Positioning against representative URL-based phishing detectors. Accuracies are not directly comparable across differing corpora.

Three observations follow. First, the framework attains accuracy within roughly two to three percentage points of learned models that use three to nine times as many features, while remaining fully auditable: every decision reduces to at most eight named conditions with published thresholds. Second, several of the highest reported figures warrant caution. Accuracies of 100% on a single corpus [37], and above 99% on standard benchmarks [35], are difficult to interpret given documented sensitivity of such detectors to distribution shift and adversarial perturbation [39]; the present study's more modest figure on a smaller corpus is at least accompanied by an explicit account of where and why the errors occur. Third, and most importantly, none of the compared systems yields a decision an analyst can inspect. This is the substantive difference between the approaches, and it is the reason explainability has become an active research direction in its own right [40].

The framework's operating profile is also distinctive. Perfect precision with zero false positives is rarely reported in this literature, where headline accuracy is usually the optimisation target. For automated blocking, where a single false positive denies a user a legitimate service and erodes confidence in the control, that property is worth more than a marginal accuracy improvement.

Deployment Guidance and Query Efficiency

The four configurations map onto distinct operational roles. OR fusion, with perfect recall and an NPV of 1.0000, suits monitoring, triage, and threat-hunting contexts in which a missed phishing URL is far costlier than an analyst-reviewed false alarm; its 27 false positives across 400 URLs represent a manageable review load. AND fusion, with perfect precision and a zero false-positive rate, suits automated blocking at a gateway, proxy, or mail security appliance, where an incorrect block is user-visible and expensive. Level 1 alone is appropriate wherever RDAP is unavailable, rate-limited, or too slow, since it needs no network interaction and still reaches 95.50% accuracy. Level 2 alone is best understood as a corroborating engine rather than a standalone detector, given its 11.50% false-positive rate.

The framework also admits an efficiency optimisation that follows directly from the fusion logic. Under AND fusion, a URL rejected by Level 1 is classified legitimate irrespective of Level 2, so its RDAP record is never required. Level 2 can therefore be evaluated lazily, only for URLs that Level 1 has already flagged. Because Level 1 flagged 190 of the 400 test URLs (186 true and 4 false positives), this cascade issues RDAP queries

for 47.5% of traffic, a 52.5% reduction, while producing decisions identical to the parallel implementation. Given that the RDAP lookup is the only component incurring network latency and the only one subject to third-party rate limiting, this halves the dominant operational cost at no cost in accuracy. The same optimisation does not apply to OR fusion, where a Level 1 rejection does not determine the outcome.

LIMITATIONS

Four limitations qualify these findings. First, the corpus of 800 URLs is small and artificially balanced, whereas operational traffic is dominated by legitimate URLs. Under realistic prevalence, precision would fall for any fixed false-positive rate, and Level 2's 11.50% rate in particular would generate a substantially larger absolute alert volume. Accuracy on a balanced corpus should therefore be read as a controlled measurement rather than a deployment forecast.

Second, the phishing sample was drawn from a limited set of public feeds over a bounded period, and the extreme separation in domain age — non-overlapping ranges between the classes — is likely sharper than a broader or longer-running sample would show. Campaigns that compromise established domains rather than registering new ones would defeat the Level 2 age rule entirely, and such campaigns are underrepresented here.

Third, the thresholds are static and manually derived. This is the source of the framework's interpretability and of its resistance to overfitting, but it also means that adaptation requires human revision rather than retraining. An adversary who learns the published thresholds can construct URLs that satisfy at most one Level 1 rule while registering domains through registrars that publish complete records and purchasing multi-year registrations. The framework raises the cost of evasion but does not eliminate it, and this is an inherent property of any published rule set.

Fourth, evaluation was entirely offline. End-to-end latency, RDAP availability and rate-limiting behaviour under load, caching effectiveness, and integration with existing security infrastructure were not measured, and the study includes no user-facing assessment. Since awareness-oriented interventions and technical controls address different parts of the same risk [42], the framework should be understood as one layer within a broader defence rather than as a complete solution.

CONCLUSION

This study designed, implemented, and evaluated a two-level rule-based framework for phishing URL detection that combines five lexical URL features with three RDAP-derived registration lifecycle features, fusing the two verdicts through explicit logical operators. On a balanced held-out test set of 400 URLs, the lexical level achieved 95.50% accuracy with 0.9789 precision, the RDAP level achieved perfect recall at 0.8969 precision, OR fusion preserved perfect recall, and AND fusion delivered the best overall performance at 96.50% accuracy with perfect precision, a zero false-positive rate, and a Matthews Correlation Coefficient of 0.9323. All three research objectives were met: the framework was designed and built, the contribution of each evidence type was measured independently, and all four configurations were quantified across seven complementary metrics.

Beyond the headline figures, the study establishes two results that carry further. The first is that the false-positive sets of the lexical and RDAP levels are completely disjoint, which explains why conjunctive fusion eliminates false positives rather than merely reducing them, and which suggests that error decorrelation is a more useful criterion for adding a detection level than standalone accuracy. The second is that AND fusion admits a cascaded implementation producing identical decisions while issuing RDAP queries for only 47.5% of URLs, halving the framework's only network-dependent cost.

The practical value of the approach lies in what it does not require. It needs no page retrieval, no TLS inspection, no passive DNS telemetry, no labelled training corpus, and no model retraining, which places it within reach of organisations whose telemetry, computational budget, or privacy posture rules out heavier detectors. Every decision reduces to at most eight named conditions with published thresholds that an

administrator can inspect and adjust, and the fusion operator provides a single control for moving between a recall-oriented and a precision-oriented posture without touching either engine.

Future work should proceed along four lines. Evaluation on larger and temporally distributed corpora with realistic class imbalance would establish whether the observed separation persists, particularly for campaigns that abuse established domains rather than registering new ones. Replacing the binary fusion operators with weighted or confidence-based combination would allow the recall of OR fusion and the precision of AND fusion to be traded continuously rather than chosen discretely. Adding a learned layer over the Level 1 and Level 2 outputs, rather than over the raw features, would preserve the interpretability of the underlying evidence while capturing interactions the current logic cannot express. Finally, engineering work on RDAP caching, parallel querying, and fallback to alternative registration sources would address the practical constraints that most affect deployment.

Ethical Approval

This study did not involve human participants, animals, or personally identifiable data, and therefore did not require ethical approval. All phishing indicators were obtained from public reporting platforms that publish them for security research and awareness purposes, and all legitimate URLs were obtained from public popularity rankings. RDAP queries were issued only for domains already present in the collected corpus, and the retrieved registration records were used solely for research purposes.

Conflict of Interest

The authors declare that they have no competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

The feature definitions, rule sets, and complete confusion matrices required to reproduce every result reported in this paper are provided in full within the paper. The curated URL corpus and the associated RDAP records are available from the corresponding author on reasonable request. Deliberately, no ready-to-use collection or exploitation scripts are distributed, in order to avoid facilitating misuse.

REFERENCES

1. Safi, A., & Singh, S. (2023). A systematic literature review on phishing website detection techniques. Journal of King Saud University – Computer and Information Sciences, 35(2). https://doi.org/10.1016/j.jksuci.2023.01.004
2. Al-Qahtani, A. F., & Cresci, S. (2022). The COVID-19 scamdemic: A survey of phishing attacks and their countermeasures during COVID-19. IET Information Security, 16(5). https://doi.org/10.1049/ise2.12073
3. Kaur, B. (2025). Social engineering attacks in the digital age. Theseus. https://www.theseus.fi/handle/10024/896456
4. Muhammad Saeed Liaqat, Mumtaz, G., Rasheed, N., & Mubeen, Z. (2024). Exploring phishing attacks in the AI age: A comprehensive literature review. Journal of Computing & Biomedical Informatics, 7(02).
5. Alnemari, S., & Alshammari, M. (2023). Detecting phishing domains using machine learning. Applied Sciences, 13(8), 4649. https://doi.org/10.3390/app13084649
6. Madupati, B. (2024). Cyber attacks in the remote work era: An analysis of phishing, ransomware, and mitigation strategies. International Journal of Science and Research, 13(9), 703–708. https://doi.org/10.21275/sr24903073257
7. Akter, T. (2025). A taxonomy and multi-layered defense framework for generative AI-powered phishing campaigns on social media platforms. University of Turku. https://www.utupub.fi/handle/10024/182730

8. Nagunwa, T. (2024). Detection of phishing websites hosted in name server flux networks using machine learning. Journal of Computer Science, 20(1), 10–32. https://doi.org/10.3844/jcssp.2024.10.32
9. Sun, X., & Liu, Z. (2023). Domain generation algorithms detection with feature extraction and domain center construction. PLOS ONE, 18(1), e0279866. https://doi.org/10.1371/journal.pone.0279866
10. Aljabri, M., Altamimi, H. S., Albelali, S. A., Al-Harbi, M., Alhuraib, H. T., Alotaibi, N. K., Alahmadi, A. A., Alhaidari, F., Mohammad, R. M. A., & Salah, K. (2022). Detecting malicious URLs using machine learning techniques: Review and research directions. IEEE Access, 10, 121395–121417. https://doi.org/10.1109/access.2022.3222307
11. Jadhav, S. (2023). Phishing website detector using ML. International Journal for Research in Applied Science and Engineering Technology, 11(5), 5509–5514. https://doi.org/10.22214/ijraset.2023.52872
12. Fernando, M., Mahmood, A., Jabed, M., & He, Z. (2025). Phishlex: A real-time machine learning model for zero-day phishing detection by systematizing URL techniques. https://doi.org/10.2139/ssrn.5205908
13. Vyawhare, C. R., Totare, R. Y., Sonawane, P. S., & Deshmukh, P. B. (2022). Machine learning system for malicious website detection: A literature review. International Journal for Research in Applied Science and Engineering Technology, 10(5), 56–61. https://doi.org/10.22214/ijraset.2022.42050
14. Choo, E., Nabeel, M., Kim, D., De Silva, R., Yu, T., & Khalil, I. (2024). A large scale study and classification of VirusTotal reports on phishing and malware URLs. ACM SIGMETRICS Performance Evaluation Review, 52(1), 55–56. https://doi.org/10.1145/3673660.3655042
15. Arun, A., & Abosata, N. (2024). Next generation of phishing attacks using AI powered browsers. arXiv. https://doi.org/10.48550/arxiv.2406.12547
16. Wong, A., Abuadbba, A., Almashor, M., & Kanhere, S. (2022). PhishClone: Measuring the efficacy of cloning evasion attacks. arXiv. https://doi.org/10.48550/arXiv.2209.01582
17. Abuadbba, A., Wang, S., Almashor, M., Ahmed, M. E., Gaire, R., Camtepe, S., & Nepal, S. (2022). Towards web phishing detection limitations and mitigation. arXiv. https://doi.org/10.48550/arXiv.2204.00985
18. Thakur, K., Ali, M. L., Obaidat, M. A., & Kamruzzaman, A. (2023). A systematic review on deep-learning-based phishing email detection. Electronics, 12(21), 4545. https://doi.org/10.3390/electronics12214545
19. Magdy, S., Abouelseoud, Y., & Mikhail, M. (2022). Efficient spam and phishing emails filtering based on deep learning. Computer Networks, 206, 108826. https://doi.org/10.1016/j.comnet.2022.108826
20. Pandey, P., & Mishra, N. (2023). Phish-Sight: A new approach for phishing detection using dominant colors on web pages and machine learning. International Journal of Information Security. https://doi.org/10.1007/s10207-023-00672-4
21. Lübbers, H. (2023). Investigating phishing attacks using the Registration Data Access Protocol (RDAP). CEUR Workshop Proceedings, 3631. https://ceur-ws.org/Vol-3631/paper6.pdf
22. Loffredo, M., & Martinelli, M. (2024). RFC 9536: Registration Data Access Protocol (RDAP) reverse search. RFC Editor. https://www.rfc-editor.org/rfc/rfc9536.pdf
23. Harrison, T., & Singh, J. (2026). RFC 9910: Registration Data Access Protocol (RDAP) Regional Internet Registry (RIR) search. RFC Editor. https://www.rfc-editor.org/rfc/rfc9910.pdf
24. Newton, A. (2026, February 10). The current state of RDAP. APNIC Blog. https://blog.apnic.net/2026/02/10/the-current-state-of-rdap/
25. Agarwal, S., & Vasek, M. (2025). Examining newly registered phishing domains at scale. Workshop on the Economics of Information Security. https://discovery.ucl.ac.uk/id/eprint/10209951/
26. Chiang, P.-H., & Tsai, S.-C. (2024). Detection of malicious domains with concept drift using ensemble learning. IEEE Transactions on Network and Service Management, 21(6), 6796–6809. https://doi.org/10.1109/TNSM.2024.3435516
27. Hassaoui, M., Hanini, M., & El Kafhali, S. (2024). Data science in cybersecurity to detect malware-based domain generation algorithm: Improvement, challenges, and prospects. Journal of Computational and Cognitive Engineering, 3(3), 213–225. https://doi.org/10.47852/bonviewjcce42022875
28. Thein, T. T., Shiraishi, Y., & Morii, M. (2023). Malicious domain detection based on decision tree. IEICE Transactions on Information and Systems, E106.D(9), 1490–1494. https://doi.org/10.1587/transinf.2022ofl0002

29. Hung, N. V., & Shone, N. (2025). Detecting DGA-managed thingbots using DNS traffic. Journal of Control Engineering and Applied Informatics, 27(2). https://doi.org/10.61416/ceai.v27i2.9302
30. Ma, S., Pang, T., Cui, R., & Yang, D. (2024). A malicious domain detection method based on DNS logs. 283–288. https://doi.org/10.1109/icbctis64495.2024.00051
31. Machmeier, S., & Heuveline, V. (2024). Detecting DNS tunnelling and data exfiltration using dynamic time warping. 2024 8th Cyber Security in Networking Conference (CSNet), 83–91. https://doi.org/10.1109/csnet64211.2024.10851475
32. Kibreab Adane, Beyene, B., & Abebe, M. (2023). Single and hybrid-ensemble learning-based phishing website detection: Examining impacts of varied nature datasets and informative feature selection technique. Digital Threats: Research and Practice. https://doi.org/10.1145/3611392
33. Kayode-Ajala, O. (2023). Applying machine learning algorithms for detecting phishing websites: Applications of SVM, KNN, decision trees, and random forests. International Journal of Information and Cybersecurity.
34. Choudhary, T., Mhapankar, S., Bhddha, R., Kharuk, A., & Patil, R. (2023). A machine learning approach for phishing attack detection. Journal of Artificial Intelligence and Technology, 3(3), 108–113. https://doi.org/10.37965/jait.2023.0197
35. Awasthi, A., & Goel, N. (2022). Phishing website prediction using base and ensemble classifier techniques with cross-validation. Cybersecurity, 5(1). https://doi.org/10.1186/s42400-022-00126-9
36. Kocyigit, E., Korkmaz, M., Sahingoz, O. K., & Diri, B. (2024). Enhanced feature selection using genetic algorithm for machine-learning-based phishing URL detection. Applied Sciences, 14(14). https://doi.org/10.3390/app14146081
37. Chong, J. C., Sim, N. Y., Khoh, C. W., & Yi, L. T. (2025). Phishing attack detection on URLs using KNN, RF, DT with GA and k-fold cross validation approach. International Journal of Research and Innovation in Social Science, 9(1), 1623–1641. https://doi.org/10.47772/IJRISS.2025.9010134
38. Ari Kustiawan, Y., & Ghauth, K. I. (2025). Evaluating the impact of feature engineering in phishing URL detection: A comparative study of URL, HTML, and derived features. IEEE Access, 13, 126756–126768. https://doi.org/10.1109/access.2025.3579223
39. Sabir, B., Babar, M. A., Gaire, R., & Abuadbba, A. (2022). Reliability and robustness analysis of machine learning based phishing URL detectors. IEEE Transactions on Dependable and Secure Computing, 1–18. https://doi.org/10.1109/tdsc.2022.3218043
40. Shafin, S. S. (2024). An explainable feature selection framework for web phishing detection with machine learning. Data Science and Management. https://doi.org/10.1016/j.dsm.2024.08.004
41. Balogun, A. O., Adewole, K. S., Raheem, M. O., Akande, O. N., Usman-Hamza, F. E., Mabayoje, M. A., Akintola, A. G., Asaju-Gbolagade, A. W., Jimoh, M. K., Jimoh, R. G., & Adeyemo, V. E. (2021). Optimized decision forest for website phishing detection. Lecture Notes in Networks and Systems, 568–582. https://doi.org/10.1007/978-3-030-90321-3_47
42. Santelices, R. B. (2025). A students' perspective on cybersecurity awareness and education. International Journal of Research and Innovation in Social Science, 9(IIIS), 7976–7988. https://doi.org/10.47772/IJRISS.2025.903SEDU0597