Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Detecting Healthcare Phishing Through Official Domain Grounding and Redirect Chain Analysis

Momina Mir
Institute of Space Technology
Muhammad Bilal Rehman
bilal.rehman@numl.edu.pk
National University of Modern Languages
Shakir Ullah
Harbin Engineering University
Waris Ali
National University of Modern Languages

Research Article

Keywords:

Posted Date: November 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11200039/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Detecting Healthcare Phishing Through Official Domain Grounding and Redirect Chain Analysis

Momina Mir[1], Muhammad Bilal Rehman[2*], Shakir Ullah[3], Waris Ali[2]

[1]Department of Electrical Engineering, Institute of Space Technology, Pakistan.
[2*]Department of Software Engineering, National University of Modern Languages, Islamabad, Pakistan.
[3]Harbin Engineering University, Harbin, China.

*Corresponding author(s). E-mail(s): bilal.rehman@numl.edu.pk;
Contributing authors: momina.mir@ist.edu.pk; engr_shakir@hrbeu.edu.cn; wali@numl.edu.pk;

Abstract

Healthcare phishing impersonates a named institution. It often arrives as a genuine Bitly, SendGrid or Microsoft link and reaches a lookalike lab-result or hospital-portal page only after one or more redirects. We ask whether grounding detection in a registry of official hospital domains helps, build a detector to test the question, and measure the answer under a protocol designed to let the claim fail. The detector uses a registry of 67 healthcare brands and their 79 official hostnames, computes features over the resolved redirect chain, not the first hop, and returns a brand-constrained counterfactual evidence card. We find that a simple official-domain rule, not a gradient-boosted model trained on 20 000 URLs, carries the impersonation signal: on 457 lookalikes the rule recalls 0.978 against the model's 0.569, and on combination-TLD lookalikes, which place an official second-level domain under a different top-level domain, the model recalls 0.000 while the rule recalls at least 0.952, a class of evasion to which a learned classifier is blind. Official-domain false positives, measured out-of-fold by leave-one-brand-out, are 0 of 79 (Wilson 95% confidence interval 0.000 to 0.046); removing the same brand from the registry raises them to 56 of 79, so registry coverage, not model capacity, is the binding constraint. Redirect-chain features detect wrapper-then-unknown chains: of 200 such chains, 198 carry no brand claim and the mismatch feature alone recalls 0.010. Correcting a reserved-TLD confound in our own synthetic slice lowers lookalike recall from 1.000 to 0.569. We release the registry, slices and harness.

Keywords: phishing URL detection, healthcare cybersecurity, official-domain allow-list, brand impersonation, redirect chains, lookalike domains, out-of-fold evaluation

1 Introduction

Healthcare organisations are a frequent phishing target because one credential can give access to lab results, insurance records, or a hospital portal [1–4]. Most published work addresses a narrower problem: classify one URL string using lexical or content features [5–9]. That problem is close to saturated on public benchmarks, where boosting on URL-string tables reports 97 to 99% accuracy [9–11]. The operational threat in hospitals is different. A message *claims* Aga Khan

lab results, Sehat Card eligibility, or MyChart, passes through a trusted wrapper such as `bit.ly` or `sendgrid.net`, and then reaches a lookalike host [12–15].

This paper is a measurement study. We ask whether grounding a detector in an explicit registry of official hospital domains improves detection of that threat, we build the detector needed to test the question, and we measure where grounding works and where it does not. The evaluation protocol is designed to let the claims fail: false positives on official domains are measured out-of-fold, every learned result is scored beside a rule that uses no classifier, and a confound we introduced ourselves is corrected and reported rather than absorbed.

The headline result is negative for machine learning and positive for deployment. A rule that consults the official-domain registry, with no model attached, detects brand impersonation better than a gradient-boosted model trained on 20 000 PhiUSIIL URLs plus synthetic augmentation. On one impersonation family the model recalls nothing at all while the rule recalls almost everything. Impersonation is a lookup, not a statistical regularity, and a model that observes it through four summary columns on a corpus of mostly non-healthcare URLs learns it poorly. We report this as the finding rather than as a caveat, because it tells a defender what to deploy.

We also state at the outset what this evaluation cannot settle. Four of the five evaluation slices are constructed from the brand registry and then scored by features that consult the same registry. Leave-one-brand-out removes the training leak, but it does not remove that construction circularity; only an external corpus with impersonation labels [16, 17] or a live feed can. The synthetic lookalike and wrapped slices are adversarial stress tests rather than samples of live traffic, and the held-out healthcare slice has ten phishing URLs, of which the discriminative content is nineteen rows. Sections 3 and 5 scope each of these before the results that depend on them.

Contributions

- **An evaluation protocol for brand-grounded detection that resists circular evaluation.** Official domains are training negatives as well as an evaluation set, so we measure their false-positive rate out-of-fold by leave-one-brand-out; a reserved TLD in our own synthetic slice acted as a label proxy, so we remove it and re-measure; and every learned result is reported beside a rule-only detector and single-feature and majority-class baselines (Section 5.2).
- **An out-of-fold measurement of the coverage limits of an official-domain allow-list.** False positives on official hospital domains are 0 of 79 when the brand is in the registry (Wilson 95% CI $[0.000, 0.046]$) and 56 of 79 when it is not (Section 5.6). Registry coverage, not model capacity, is the binding constraint. We are not aware of an equivalent measurement in the brand-impersonation literature [4, 13–15].
- **The finding that a learned model is blind to the cheapest evasion the rule catches.** On combination-TLD lookalikes the trained model recalls 0.000 while the rule-only detector recalls at least 0.952, and across all 457 lookalikes the rule beats the model 0.978 to 0.569 (Section 5.5). This is an argument against URL classifiers that receive brand evidence only as input features.
- **A released, reproducible registry and harness.** 67 brands, 79 official hostnames, 24 trusted wrappers, five evaluation slices, the detector, the evaluation harness, and invariant tests, with pinned dependencies and bit-for-bit deterministic results.

Section 2 places the work against fifty cited studies, sixteen of them from 2025 and 2026. Section 3 states the threat model. Section 4 describes the detector and its implementation. Section 5 gives the protocol and the measurements. Section 6 discusses deployment.

2 Related work

We cite fifty studies, sixteen of them from 2025 and 2026. Table 1 summarises what they cover.

2.1 Lexical and learning-based URL classifiers

CANTINA and CANTINA+ combined term-frequency heuristics with search-engine and PageRank evidence [19, 23]. Later work replaced hand rules with URL-only feature vectors of

Table 1 What the literature covers versus the detector evaluated here. A "yes" means the theme is central in at least one cited work, not that the problem is solved.

Theme	Lex. ML	Health 2025/26	Imp. / XAI	This work
URL-string features	yes [8, 9]	yes [2]	n/a	baseline
Official hospital list	no	lexicon [4]	logos [18]	explicit list
Redirect-chain feats.	rare [19]	flag [3]	no	hops ≤ 5
Lookalike / homograph	no	no	yes [13, 14]	yes
SHAP/LIME ranks	yes [20]	yes [2, 21]	n/a	not shown
Brand counterfactual	no	no	theory [22]	evidence card
Official-domain FP slice	no	no	no	measured out-of-fold
Rule-vs-model baseline	no	no	no	measured
Coverage limit measured	no	no	no	56/79 off-registry

length, special characters and tokens [5, 7, 24, 25], and Sahingoz *et al.* showed that word and character n-grams separate phishing from legitimate URLs without visiting the page [8]. Parallel lines used hyperlink structure [26], search-engine consistency [27], URL ranking [28], and ensembles of classical learners [29–31]. URLNet learned character- and word-level embeddings for malicious URLs [32], and hybrid URL-plus-HTML networks followed [33].

Two patterns dominate recent work. Authors re-train boosting models, including LightGBM, on PhiUSIIL or similar lexical tables and report 97 to 99% accuracy [9–11, 20], sometimes with genetic-algorithm feature selection [20]. Transformer and federated variants such as FeUDA-BERT and mobile super-learners still consume the URL string or its tokens rather than the resolved hop list [34, 35]. PhiUSIIL is the standard corpus and baseline, with 235 795 URLs, 54 features, incremental learning and a similarity index [9]; independent replications confirm the lexical task is easy [10], and StealthPhisher extends the same line into a hybrid deep model [36]. None of these systems is organised around an official hospital brand list or a redirect chain.

2.2 Datasets, feeds, and evaluation pitfalls

ISCX-URL2016 introduced a multi-class URL corpus with lexical features only [7], and Mohammad *et al.* catalogued feature families that later UCI datasets reused [5, 6]. Live feeds (PhishTank, OpenPhish, Google Safe Browsing) are unbalanced, delayed, and overlapping [37]. Skula and Kresnakova argue that many published sets are neither balanced nor leakage-safe [38]. PhreshPhish adds leakage control and deployment-realistic base rates over hundreds of thousands of paired URL and HTML records [17]; PhishVN adds time stamps and impersonation-scenario labels, which is closer to our threat model [16]; PhishingWebCollector automates feed collection [39]. Two consequences follow: there is no public Pakistani hospital phishing set, so any such slice must be constructed transparently [17, 38]; and high accuracy on PhiUSIIL is not evidence of progress on impersonation, shorteners, or official-domain false positives [9, 10].

2.3 Lookalikes, typosquatting, and homographs

Agten *et al.* measured typosquatting over seven months and found lookalike registration against popular brands at scale [13]; Spaulding *et al.* surveyed the defences then available [40]. Homograph attacks remain effective: PunyVis visualises them [41], Yazdani *et al.* detect suspicious IDN homographs from active DNS [42], VizCheck adds visual domain-name checks [43], and PCDF studies Cyrillic homograph forgery as an attack [14]. Kurnaz *et al.* add large language models to a typosquatting framework [44], and Jarczewski, Białczak, and Mazurczyk compare impersonation target-recognition methods including Phishpedia-style brand identification [15].

These papers treat the host as a forged brand, and they supply the five impersonation families in Section 3. None uses a hospital-specific official-domain list as a hard constraint, and none reports what happens to a brand the system has not been given, which is the measurement in Section 5.6.

VisualPhishNet detects screenshot similarity to branded pages [18], which is complementary but requires rendering the landing page.

2.4 Shorteners, redirects, and trusted-host wrapping

Bell *et al.* measured Twitter's `t.co` wrapper and showed that a trusted shortener does not establish that a link is safe [12], and showed that public feeds disagree and lag [37]. CANTINA+ listed redirection as a feature family [19]; hybrid HTML models sometimes count redirects [33]; Mayo *et al.* use temporally dynamic extraction with LIME and SHAP without using the hop sequence as an input in its own right [21]. Immanuvel Arockiasamy lists "redirect chains" among heuristic flags while training an LSTM on PhiUSIIL character strings [3, 9]: a heuristic flag rather than chain-aware detection, with no recorded hop list, no wrapper-then-unknown feature, and no evaluation slice of shortener-wrapped URLs.

2.5 Explanations

Ribeiro *et al.* introduced Local Interpretable Model-agnostic Explanations (LIME) [45], and Lundberg and Lee unified additive methods as SHapley Additive exPlanations (SHAP) [46]. Phishing papers use both to rank lexical features [2, 20, 21]. Wachter, Mittelstadt, and Russell argued that the useful explanation is a counterfactual: the smallest change that would flip the decision [22]. Here that counterfactual must be brand-constrained. "Would be allowed if the final domain were `hospitals.aku.edu`" is actionable; "would be allowed if URL length dropped by twelve characters" is not. No paper in this review produces that form of explanation for healthcare URLs.

2.6 Healthcare-specific work, 2025 and 2026

Four works address healthcare directly. Karapiperis *et al.* attach LIME and SHAP to a random-forest URL filter on PhishTank and OpenPhish, reach accuracy 0.985, and state that their data are generic and that no healthcare-specific URL set was used [2]. Elongha and Liu detect phishing *email* aimed at practitioners with a domain-specific ensemble, where the object is the message rather than the redirect path of an embedded link [1]. Immanuvel Arockiasamy places LSTM and GNN URL models in a telehealth DevSecOps setting, again on PhiUSIIL [3]. Feretzakis *et al.* present PhishCluster-Health: semantic embeddings with streaming clustering, a curated seed lexicon of medical and insurer brand terms, and campaign-level recall [4]. That lexicon is the closest published analogue to our registry, but the task is campaign clustering with SHAP on a proxy random forest, with no URL-level brand-mismatch features, no hop expansion, and no counterfactual evidence card. Adjacent work improves generic detectors without a hospital grounding [11, 14, 15, 35], and human-behaviour automation [31] and visual similarity [18, 43] are complementary.

2.7 Position of this work

To the best of our knowledge, no published detector simultaneously maintains an explicit official-domain list for named healthcare institutions, classifies the resolved redirect chain rather than the first-hop string, and returns a fixed-template evidence card whose counterfactual is an official hospital domain. More importantly for a measurement study, we are not aware of prior work that reports the coverage limit of a brand-grounded detector, or that scores a learned phishing model against the rule its own features encode. Those two comparisons are what Sections 5.4 and 5.6 contribute, and both of them constrain how such a system should be deployed.

3 Threat model

3.1 Attacker goal and capabilities

The attacker sends a message to a member of hospital staff or to a patient, and wants a click and then a credential for a patient portal, a laboratory result system, or an insurance account [1, 2, 4]. The message asserts an institutional identity: it claims to be Aga Khan lab results, Sehat Card eligibility, NADRA verification, or MyChart.

We assume the attacker can do all of the following, each of which is documented in the literature reviewed in Section 2.

1. **Register a lookalike domain.** Lookalike registration against popular brands is industrial in

scale [13, 40]. We model five families, which are the impersonation patterns the typosquatting and homograph literature documents:

- *extra letter*: `chughtaislab.com` for `chughtailab.com`;
- *hyphenation*: `sehat-card-gov`;
- *homoglyph*: a Cyrillic or otherwise confusable character, or `o`→`0` [14, 41, 42];
- *brand-as-subdomain*: `hospitals.aku.edu.login.example`, where the official name appears in a subdomain of an attacker-controlled registrable domain;
- *combination TLD*: the official second-level domain under a different top-level domain, such as `chughtai.net` for `chughtailab.com`.

2. **Route through a trusted wrapper.** The attacker can send a genuine `bit.ly`, `sendgrid.net`, Safe Links, or cloud-file URL whose first hop is a hostname the recipient has been trained to trust. A trusted shortener does not establish that a link is safe [12].
3. **Control the redirect chain.** The attacker chooses the number of hops, can place the brand token in a path or query rather than in the host, and can serve a block or an error to a resolver it does not like.
4. **Choose a disposable TLD** and register at negligible cost.

We assume the attacker does *not* control the official domains in the registry, and does not hold a certificate for them. Compromise of a genuine hospital domain is out of scope: no URL-level allow-list can detect a message that really does come from the institution's own host.

3.2 Defender position

The defender scores a URL at mail-gateway or inbox time, with optional surrounding text such as a subject line or page title. The decision must be available before the user clicks, which excludes any method that requires the landing page to be rendered.

In scope: the URL string, the resolved redirect chain of at most five hops, the brand identity the URL and its surrounding text claim, and a curated registry of official domains for the institutions the deploying organisation cares about.

Out of scope: page content and screenshots, which would require rendering the landing page [18]; email headers, SPF and DKIM, which are a separate control; attachments; and pixel-level rendering of homographs, which we treat only as a string-level confusable substitution.

3.3 Deployment assumption

The registry is curated by the deploying institution. This assumption is not free, and Section 5.6 measures its cost: a brand absent from the registry is flagged as phishing in 56 of 79 cases. The detector is therefore a per-institution tool, in which a hospital enumerates its own domains, its vendors, and the peers it exchanges mail with. It is not a drop-in classifier for arbitrary healthcare traffic.

3.4 Why the synthetic slices are constructed, not sampled

Three of the five evaluation slices are constructed rather than sampled from a live feed. This is a deliberate choice about what can be measured, and it is a limitation of external validity, so we state both halves.

A random sample of a live feed cannot guarantee coverage of the five impersonation families above; a 300-URL OpenPhish snapshot taken on 2 September 2026 contained no URL that passed our healthcare filter at all. Constructing the slice lets us measure recall *per family*, which is what produced the combination-TLD result in Section 5.5; a pooled figure over live traffic would have hidden it. Every synthetic lookalike uses the reserved TLD `.invalid` (RFC 2606) so that no host can resolve and no pattern corresponds to a registrable domain, and the wrapped slice uses fabricated hop lists rather than live redirects.

The cost is that these slices measure the detector against the threat model rather than against observed traffic. They establish that a family is or is not detected; they do not establish how often that family occurs in the wild. Section 5.10 states which claims this limits.

4 Design and implementation

The detector exists to test whether official-domain grounding helps. It is not intended to improve lexical URL classification, and repeating that

method, with or without genetic-algorithm feature selection [20], is not a contribution. Three constraints follow from the threat model. Decisions are grounded in an explicit official-domain list for named healthcare institutions, rather than in a campaign lexicon [4] or a screenshot-similarity model that must render the landing page [18]. The detector scores the redirect path, not the first-hop string, because a trusted wrapper does not establish that a URL is safe [12, 19]. The user-facing explanation is a Wachter-style counterfactual constrained to that official list [22], not a SHAP ranking of hyphen count [2, 21, 46]. LightGBM combines these features [47]; it is not part of the claim.

4.1 Pipeline

Figure 1 shows the pipeline. A URL enters with optional surrounding text. It is matched against the official brand registry and expanded to at most five redirect hops. Four feature groups (lexical, host, brand, and chain) are concatenated. LightGBM returns class probabilities; if $\max(p_{\text{phish}}, p_{\text{benign}}) < \tau$ the URL is sent to review rather than assigned a binary label [48, 49]. Every scored URL also yields a five-field evidence card. No stage visits the landing page with a JavaScript renderer, submits a form, or downloads more than 8 kB of response body.

4.2 Official brand registry

We implement brand grounding as an *allow-list*, not as a learned logo detector [18] and not as the medical-term lexicon used by PhishCluster-Health [4]. The registry contains 67 healthcare brands with 79 unique official hostnames, compiled from official hospital, laboratory, ministry, insurer, pharmacy, and patient-portal pages. Fifty-seven brands are Pakistani (Aga Khan University Hospital, Shaukat Khanum, Liaquat National, Indus, Chughtai Lab, Sehat Card, NADRA, and others); the rest are portal vendors and laboratories that are commonly impersonated (Epic MyChart, Oracle Health/Cerner, and related hosts). Combined Military Hospital was omitted because no single official public patient domain could be verified. Microsoft, Google, and WhatsApp are not healthcare brands; their wrapper hostnames are in a separate list.

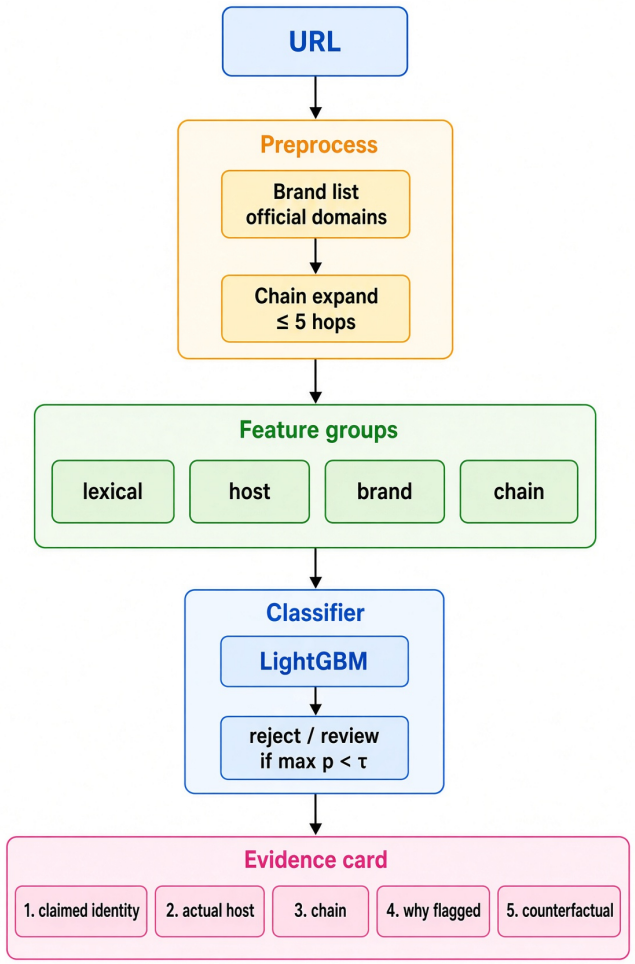


Fig. 1 Detector pipeline. A URL is grounded against the official brand list and expanded to at most five redirect hops. Four feature groups feed LightGBM; predictions with $\max p < \tau$ are sent to review. The output is a five-field evidence card whose counterfactual is constrained to an official hospital domain.

A companion list of 24 trusted wrappers covers shorteners (`bit.ly`, `t.co`, `tinyurl.com`), mail infrastructure (`sendgrid.net`, Safe Links), cloud-file hosts, forms, and messaging (`wa.me`, `t.me`). A wrapper hostname is infrastructure. It is not treated as evidence that a message is from brand X [12].

Claimed identity is recovered from distinctive tokens in the URL, the hop URLs, and the optional text, after a stopword list removes generic tokens such as `hospital`, `lab`, and `portal`. If brand X is claimed, the final registrable domain

(eTLD+1) is a *mismatch* unless it is in X's official list or is a verified subdomain of one of those domains. If the final host is itself a trusted wrapper, the claim is unresolved and is scored as a mismatch until a later hop reaches an official domain. The lookalike literature measures this relation but does not encode it as a hospital allow-list [13, 15, 40].

4.3 Redirect-chain expansion

A detector that sees only the first hop will label many current healthcare phishing URLs benign [12, 37]. CANTINA+ listed redirection among its feature families [19]; hybrid HTML models sometimes count redirects [33]; the 2025 telehealth sketch lists "redirect chains" as a heuristic while training an LSTM on PhiUSIIL character strings [3, 9]. We compute features over the hop sequence itself.

Resolution follows at most five hops. Each hop uses HTTP HEAD first; GET is issued only when HEAD is refused or the `Location` header is missing. Redirects are recorded rather than auto-followed by the client. JavaScript is never executed, forms are never submitted, and GET bodies are capped at 8 kB with a five-second timeout. These limits are for safety rather than accuracy, and they are why this work does not claim VisualPhishNet-style screenshot matching [18]. A failed or blocked resolve is kept as a feature rather than dropping the URL: `redirect_blocked` on HTTP 403/451, `shortener_unresolved` when a shortener never leaves its own host, and `chain_unavailable` when no hop list exists.

PhiUSIIL publishes URL strings and lexical columns, not resolved hops [9]. On that corpus we set `chain_unavailable` = 1 and zero the remaining chain flags, so no row carries hop evidence that does not exist. Wrapped and lookalike evaluation uses *dry* hop lists, precomputed or synthetic, rather than live crawls of phishing landing pages.

Two chain flags encode the redirect patterns of Section 3. `wrapper-then-unknown` is 1 when a trusted wrapper appears and a later hop is neither a wrapper nor an official healthcare host. `brand-in-path` is 1 when a distinctive token of the claimed brand is in a path or query but not in the final eTLD+1, as in `/mychart` on an unrelated host. Lexical length features do not encode this pattern.

Table 2 Feature groups used by the detector. Lexical and host features are the saturated baseline; brand and chain features encode the threat model of Section 3.

Group	#	What it encodes
Lexical	8	URL length, digits, hyphens, entropy, suspicious TLD, subdomain depth, IP host, HTTPS
Host	2	`IsHTTPS`, `IsDomainIP` (PhiUSIIL columns, else derived)
Brand	4	Token overlap, Levenshtein to official host, mismatch, claim-not-in-list
Chain	6	Hop count, wrapper-then-unknown, brand-in-path, blocked, unresolved, unavailable

4.4 Feature groups

Table 2 lists the twenty features. The lexical group is a URL-string baseline of the kind used by Sahingoz, Mohammad, and the PhiUSIIL feature family [5, 7–9]: length, digit and hyphen counts, Shannon entropy, a suspicious-TLD flag, subdomain depth, IP-literal host, and HTTPS. We keep it so that ablations can measure what brand and chain features add, and we do not search it with a metaheuristic [20]. The host group copies PhiUSIIL's `IsHTTPS` and `IsDomainIP` when those columns exist and otherwise derives them from the URL.

The brand group encodes the allow-list. `token-overlap` scores distinctive-token overlap with the registry. `Levenshtein-to-official` is the edit distance from the final host to the nearest official hostname [50], the quantity that typosquatting and homograph papers treat as the attack [13, 14, 42]. `mismatch` and `claim-not-in-list` implement the allow-list rule of Section 4.2. The chain group implements Section 4.3. The four groups concatenate to a 20-dimensional vector. We train four nested ablations: lexical; lexical+host; lexical+host+brand; and the full vector.

4.5 Classifier and reject option

We use LightGBM with 200 trees, 31 leaves, learning rate 0.08, row and column subsampling 0.8, and a minimum of 20 instances per leaf [47]. Gradient boosting is the strongest published learner on PhiUSIIL-style tables [9–11, 20]; it is a baseline choice, not a novelty claim. Labels are inverted

from the PhiUSIIL convention so that $y{=}1$ is phishing.

A hospital IT queue cannot act on every low-confidence score. Chow's reject option, and the later statistical treatment by Herbei and Wegkamp, let a classifier abstain when the posterior is not peaked [48, 49]. We output

$$\hat{y} = \begin{cases} \text{review}, & \max(p_{\text{phish}}, p_{\text{benign}}) < \tau, \\ \text{phish}, & p_{\text{phish}} \geq p_{\text{benign}} \text{ and } \max p \geq \tau, \\ \text{benign}, & \text{otherwise.} \end{cases}$$

The default operating point is $\tau{=}0.65$. It is a review budget, not a claim of optimality; a sweep of τ belongs in the evaluation rather than in the definition of the model. SHAP and LIME are not computed for the user-facing path [45, 46].

4.6 Evidence-style explanations

Karapiperis *et al.* attach LIME and SHAP to a lexical URL classifier motivated by hospital networks, and state that their data are generic [2]. Mayo *et al.* explain temporally dynamic URL features with the same tools [21]. Those explanations rank surface statistics. They do not report that a message claimed AKUH and ended on a host that is not AKUH's.

Following Wachter, Mittelstadt, and Russell, the useful explanation is a counterfactual: the smallest change that would flip the decision [22]. Here that change must be *brand-constrained*. The card has five fields: (1) claimed identity, (2) actual host, with eTLD+1, (3) the hop sequence, (4) why the URL was flagged, in prose built from the brand and chain flags rather than from SHAP ranks, and (5) the counterfactual "Would be allowed if the final domain were ⟨official domain of the claimed brand⟩." If no brand is claimed, the nearest official domain by token overlap is used. An official host with a matching claim produces an empty reason list and a no-change counterfactual.

4.7 Implementation

The detector is 1 500 lines of Python. Public-suffix resolution uses a packaged snapshot with network lookup disabled, so eTLD+1 extraction is deterministic and makes no request at scoring time. Edit distance uses a compiled Levenshtein implementation.

Offline scoring costs 6.2 ms per URL for feature assembly and 0.26 ms for model inference on one core of a consumer laptop, or 6.5 ms per URL end to end: about 11 core-minutes per 100 000 URLs. The rule-only path of Section 5.4, which generates a full evidence card, costs 13.2 ms per URL. These figures exclude live redirect resolution, which is network-bound; its worst case is a design parameter rather than a measurement, at most five hops with a five-second timeout each.

The operational cost that matters is not compute. It is registry maintenance: 67 brands and 79 hostnames must be curated and kept current by the deploying institution, and Section 5.6 measures what happens when a brand is missing.

5 Evaluation

5.1 Protocol

PhiUSIIL has no timestamp, so a time-based split is impossible [9, 17, 38]. We draw a stratified sample of 20 000 URLs from the full corpus (seed 42), hold out every URL that appears in the healthcare slice so that slice cannot leak into training, and then apply a stratified 80/20 split. Chain flags on this sample are unavailable, as described in Section 4.3.

Five evaluation sets test the threat model rather than another 99% on PhiUSIIL [16, 17].

- **PhiUSIIL holdout**, 4 000 URLs, for the feature ablation.
- **Official domains**, 79 constructed `https://`⟨official domain⟩`/` URLs, for false-positive stress. Pages were not fetched. These rows are also training negatives, which Section 5.2 addresses.
- **Synthetic lookalikes**, 457 patterns on the reserved `.invalid` TLD across the five families of Section 3.
- **Synthetic wrapped chains**, 200 dry hop lists through Bitly, SendGrid and Safe-Links-style hosts.
- **Held-out healthcare slice**, 82 PhiUSIIL URLs (10 phishing, 72 legitimate) selected by token and official-domain match rather than by substring search, never used in training.

The healthcare slice is not homogeneous, and we decompose it here rather than after reporting a pooled number. Sixty-three of the 82 rows were

selected *because* their hostname is on the allow-list that the brand features consult, and all 63 are legitimate: they are benign by construction. Fifty-eight of those are `nhs.uk` subdomains matched from a single brand entry, so a slice motivated by Pakistani healthcare is in practice dominated by one British institution. The discriminative content is the remaining 19 rows, which contain all ten phishing URLs. We report both the pooled figure and that decomposition in Section 5.7.

On review-capable predictions we report accuracy, precision, recall, and F_1 on the non-review subset, plus review rate and official-domain false-positive rate; ROC-AUC uses p_{phish} over all rows. Precision, recall and F_1 are reported as undefined, not as zero, when the relevant class is absent from a slice. Square brackets are 95% percentile-bootstrap intervals over evaluation rows; the official-domain count uses a Wilson interval, which unlike a bootstrap of an all-zero vector does not collapse to a point. Over seeds $\{42, \ldots, 46\}$ the full model's holdout accuracy has standard deviation 0.0002, so the gaps below are not seed noise. Metrics other than ROC-AUC are computed at τ=0.65 [48].

5.2 Guarding against circular evaluation

Three features of this design would make the reported numbers unfalsifiable. Each is addressed by a protocol rather than by a caveat.

The official domains are training negatives. Without the 79 official URLs in training, brand-token overlap is observed only on phishing synthetics and the false-positive rate on real hospital hosts is high; with them, reporting the false-positive rate on the same 79 rows is in-sample. We therefore report it *out-of-fold* by leave-one-brand-out (LOBO). In LOBO-B, each brand's rows leave the training set while the brand stays in the registry, which tests whether the detector generalises to a listed hospital's URLs that it was not trained on. In LOBO-A the brand is also removed from the registry, which is the situation of any hospital not among the 67, and which measures the coverage dependence of an allow-list. Domains listed by two brands, such as `aku.edu`, leave training whenever either owner is held out. Features are extracted once under the full registry and cached, so the 67 folds cost 67 model fits rather than 67 feature passes. In LOBO-A only the held-out brand's own evaluation rows are re-featurised under the reduced registry. This favours the model, so the reported degradation is a lower bound.

A reserved TLD can act as a label proxy. All 457 synthetic lookalikes use `.invalid` so that no host can resolve. While `invalid` was also in the suspicious-TLD list, and no benign row used that TLD, one feature separated the slice perfectly. Because `.invalid` cannot occur in live traffic it was never a phishing signal, and it is excluded from the suspicious-TLD list. As a further bound, lookalike metrics are recomputed with `suspicious_tld` forced to zero, and recall is reported per impersonation family rather than pooled. This correction lowers our own previously measured lookalike recall from 1.000 to 0.569.

A rule may be doing the work attributed to the classifier. Every learned result is reported beside a rule-only detector that reads the evidence card's own `flagged` verdict (the deployed allow-list and chain rules, with no model), and beside single-feature and majority-class baselines that show what accuracy each slice yields with no discrimination. Adjacent ablations are compared with McNemar's test on paired correctness, excluding rows that either system sends to review. Feature groups are ablated leave-one-out as well as nested, because nested ordering confounds attribution. Training variance is reported over five seeds.

5.3 PhiUSIIL holdout: brand features lift, host and chain do not

Table 3(a) and Figure 2(a) show the nested ablation. Lexical features alone reach accuracy 0.973 and F_1 0.968. The two host columns change nothing. The four brand features raise accuracy to 0.989 and F_1 to 0.988, and cut the review rate from 0.022 to 0.007. The full 20-dimensional model is indistinguishable from lexical+host+brand.

Leave-one-group-out with McNemar tests attributes the change to groups. Dropping the brand group costs accuracy 0.989→0.974 over 74 discordant rows (p=3.4 × 10^{-11}). Dropping the host group produces *one* discordant row (p=1.0), and dropping the chain group produces *none*. For the chain group this was predicted in

Section 4.3: PhiUSIIL publishes no resolved hops, so `chain_unavailable`=1 on every row [9]. For the host group it is a negative result: `IsHTTPS` and `IsDomainIP` are redundant with the lexical columns that encode the same facts.

Within the brand group the lift has a single source. Added to lexical+host one at a time, `min_levenshtein_official` alone reaches accuracy 0.989, the entire brand-group gain, while `brand_token_overlap` reaches 0.974, and `mismatch` and `claim-not-in-list` reach 0.972, which is no gain. On a generic URL dump the useful brand signal is *near-miss edit distance to an official hostname* [13, 50], not the allow-list rule. The allow-list rule contributes on the hard slices of Section 5.5, not here. This is a narrower claim than "brand grounding helps", and it differs from a genetic-algorithm-on-tokens method [20].

5.4 The rule beats the model on impersonation

A detector organised around an allow-list must show what its classifier adds to the allow-list. We score a rule-only baseline that reads the `flagged` verdict from the evidence-card generator, so it cannot drift from the deployed rules, and it involves no learning.

On the impersonation slice the rule-only detector beats the trained model, and the difference is significant (Table 4). The model is better on wrapped chains and on healthcare recall, but neither of those differences is significant: one discordant row and seven discordant rows respectively. The single significant comparison in the table therefore runs against the classifier.

We claim from this that the allow-list and the chain rules are what detect impersonation, and that the classifier's role is calibration and integration (one score with a principled abstention) rather than impersonation detection. A defender who wants impersonation detection alone can deploy the rule, at 13.2 ms per URL and with no training data.

Two further baselines calibrate the slices. On the pooled healthcare slice a constant "benign" classifier scores accuracy 0.878 against the full model's 0.963. On the official-benign slice, which contains no phishing rows, a constant "benign" classifier also scores accuracy 1.000 and a false-positive rate of 0.000. That slice bounds false positives; on its own it is not evidence of discrimination, and we do not report it as a standalone accuracy.

5.5 Official domains, lookalikes, and wrappers

Figure 2(b) re-scores the same four saved bundles on the hard slices; this section reports the full model on each in turn. Table 5 re-measures the official-domain false-positive rate out-of-fold: each of the 67 brands' rows are removed from training in turn and scored by a model that never saw them.

The out-of-fold count is also zero. Holding each brand's rows out of training produces no false positive on official hospital, laboratory, ministry or portal-vendor domains, and the upper bound of the interval is 4.6% rather than zero. This is the allow-list rule, measured out-of-sample.

On the 457 synthetic lookalikes the corrected recall is **0.569** [0.523, 0.616], not the 1.000 obtained before `.invalid` was removed from the suspicious-TLD list. Re-scoring with `suspicious_tld` forced to zero reproduces 0.569 exactly, which confirms that the TLD accounted for the whole difference. The pooled figure also conceals the per-family result, which is the more useful one. Model recall is 0.919 [0.855, 0.984] on homoglyphs [41, 42], 0.793 on hyphenated brands, 0.537 on extra-letter typos, 0.250 on brand-as-subdomain, and **0.000 on combination-TLD lookalikes**. The rule-only detector recalls at least 0.952 in every one of those five families.

The combination-TLD result is the sharpest finding in this evaluation. Placing an official second-level domain under a different top-level domain, such as `chughtai.net` for `chughtailab.com`, is among the cheapest impersonations available to an attacker: it requires one registration and no typo, and it leaves the brand string intact and correctly spelled. That last property is why the model misses it. Every lexical cue looks normal, the edit distance to an official hostname is small but so is the edit distance for many benign hosts, and the model was trained mostly on generic PhiUSIIL strings in which no such relation exists. The rule catches it because the rule does not need the pattern to be statistically frequent: it checks whether the final registrable domain is in the claimed brand's list, and `chughtai.net` is

Table 3 Ablation on the PhiUSIIL 4 000-URL holdout (τ=0.65, LightGBM, metrics on non-review predictions) and full-model evaluation on hard slices. Brackets are 95% bootstrap intervals. Brand features provide the PhiUSIIL lift; host and chain features produce no measurable change on that corpus. Lookalike and wrapped slices are synthetic; the healthcare slice is constructed and eval-only. Official-domain false positives are reported out-of-fold in Table 5.

(a) PhiUSIIL holdout, n=4000

Feature set	n	#feat.	Acc.	Prec.	Rec.	F_1	AUC	Review
lexical	4000	8	0.973	0.991	0.946	0.968	0.987	0.023
lexical+host	4000	10	0.972	0.990	0.944	0.966	0.987	0.022
lexical+host+brand	4000	14	0.989	0.994	0.981	0.988	0.995	0.007
full (+chain)	4000	20	0.989	0.995	0.981	0.988	0.996	0.005

(b) Full model on hard slices

Slice	n	#feat.	Acc.	Prec.	Rec.	F_1	AUC	Review
official benign	79	20	1.000	n/a	n/a	n/a	n/a	0.000
synthetic lookalikes	457	20	0.569	1.000	0.569	0.726	n/a	0.039
synthetic wrapped	200	20	1.000	1.000	1.000	1.000	n/a	0.000
healthcare (eval-only)	82	20	0.963	0.769	1.000	0.870	0.997	0.000

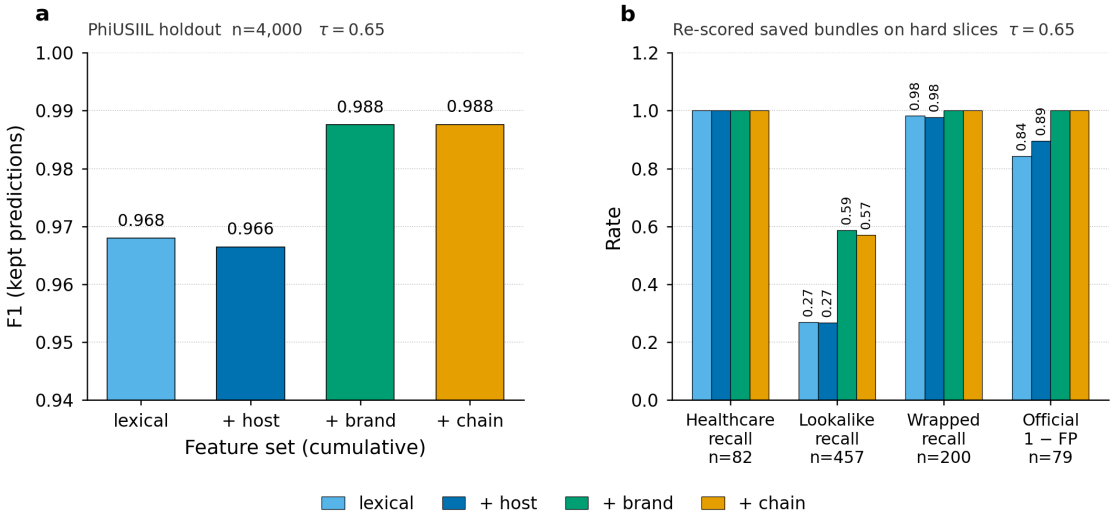


Fig. 2 Ablation. **(a)** F_1 on the PhiUSIIL holdout (n=4000, τ=0.65) as feature groups are added: lexical 0.968, +host 0.966, +brand 0.988, +chain 0.988. Brand features produce the lift; chain is flat because that corpus has no resolved hops. **(b)** The same four saved bundles re-scored on the hard slices. Official-domain 1−FP rises from 0.843 (lexical) and 0.895 (+host) to 1.000 once brand features are added, and review on that slice falls from 0.354 to 0.000. Lookalike recall rises from 0.268 to 0.587 with brand features and is 0.569 for the full model; chain features cost a little recall here by adding abstention on a slice with no real wrappers. Wrapped recall rises from 0.983 to 1.000. Healthcare recall is 1.000 for every ablation (ten phishing URLs). Lookalike figures are the corrected ones, measured after `.invalid` was removed from the suspicious-TLD list.

not. A learned URL classifier that receives brand evidence only as input features can therefore be completely blind to an evasion that a lookup detects deterministically.

On the 200 wrapped chains (dry hops through Bitly, SendGrid and Safe-Links-style hosts, not live crawls [12]) the full model recalls 200/200 with review rate 0. This slice is the one place where the chain group works, and the attribution matters: 198 of the 200 rows carry no claimed brand, and the `mismatch` feature alone recalls 0.010. Wrapped recall is therefore not a brand-feature result. It is `wrapper_before_unknown` firing on a trusted wrapper followed by an unlisted host, which is the pattern Section 4.3 targets and which PhiUSIIL cannot represent [16, 17].

Table 4 Full model versus the rule-only detector (allow-list plus chain rules, no classifier). Recall unless noted. McNemar excludes rows either system sends to review.

Slice	Full model	Rules only	McNemar
Lookalikes (457)	0.569	**0.978**	$p \approx 0$, 184 rows rules-only correct
Wrapped (200)	**1.000**	0.995	p=1.0, 1 discordant
Healthcare (82)	**1.000**	0.700	p=1.0, 7 discordant (4/3)
Healthcare acc.	0.963	0.951	n/a

Table 5 Official-domain false positives under three protocols. LOBO-B holds the brand's rows out of training; LOBO-A additionally removes the brand from the registry, which is the situation of any hospital not among the 67. Intervals are Wilson 95%.

Protocol	FP	Rate	95% CI
In-sample (rows in training)	0/79	0.000	n/a
LOBO-B (out-of-fold, brand listed)	0/79	0.000	[0.000, 0.046]
LOBO-A (brand not listed)	56/79	0.737	[0.601, 0.831]

5.6 Coverage: what happens off the registry

LOBO-A removes the held-out brand from the registry as well as from training, so its official domains are unknown at scoring time. False positives on official hospital domains rise from 0/79 to **56/79** (0.737, Table 5). A hospital that is not on the registry is flagged as phishing roughly three times in four.

This is the operating envelope of an allow-list, and it is a property rather than a defect: a system whose decision rule is "the claimed brand's official domain" cannot recognise a brand it has never been given. Two consequences follow. The detector is deployable per institution, as stated in Section 3, and not as a general classifier for arbitrary healthcare traffic. And the engineering target is registry coverage, not model capacity. We are not aware of a brand-impersonation system that reports an equivalent measurement [4, 13–15].

5.7 Held-out healthcare slice

Pooled, the full model reaches accuracy 0.963, precision 0.769, recall 1.000 and review rate 0 on the 82-URL slice. On the 19 discriminative rows identified in Section 5.1 it reaches accuracy 0.947 [0.842, 1.000], recall 1.000 (10/10) and false-positive rate 0.111 [0.000, 0.364]. Recall of 10/10 is genuine and every ablation achieves it; the interval on the false-positive rate is what ten positives and nine negatives support.

Two of the three false positives are classifier errors rather than allow-list errors. The three are `genomicseducation.hee.nhs.uk` (p_{phish}=0.93), `palliativecareguidelines.scot.nhs.uk` (p=0.97), and `heidelberg-university-hospital.com` (p=0.78). The first two are verified subdomains of `nhs.uk`, which is in the registry: the mismatch feature is zero for both, and the rule-only detector accepts them, making one false positive on this slice against the model's three. LightGBM overrides its own brand evidence on long, deeply nested, hyphenated hostnames, where the lexical group dominates. This is the same disagreement as Section 5.4, observed on real hosts rather than synthetic ones. It is an argument for constraining the classifier with the allow-list rather than supplying the rule as one of twenty columns.

OpenPhish's 300-URL snapshot from 2 September 2026 contained no URL passing the same tight filter, so that feed is not a substitute. Karapiperis *et al.* noted that generic feeds are not a healthcare URL corpus [2]; this slice is small by design, and we report it as a decomposition rather than as a single number.

5.8 Operating point

Figure 3 sweeps $\tau \in \{0.50, \ldots, 0.90\}$ of the full model on the same 4 000-URL holdout. At τ=0.50 nothing is reviewed (recall 0.979, FP 0.0057). At the published point τ=0.65 the review rate is 0.005, recall 0.981, missed-phishing rate 0.019 and FP 0.0040. Raising τ to 0.90 gives a missed-phishing rate of 0.015 and FP 0.0023 at a review rate of 0.030. The curve is close to flat between 0.60 and 0.85, so the default is a review budget consistent with Chow's reject option [48, 49] rather than a tuned optimum. The healthcare slice at the same τ has review rate 0 and recall 1.0.

5.9 Evidence cards

Table 6 shows the output for a lookalike that claims Aga Khan lab results and for the official `hospitals.aku.edu` host. The flagged card names the claimed brand, the off-list host, the suspicious TLD, and the counterfactual "Would be

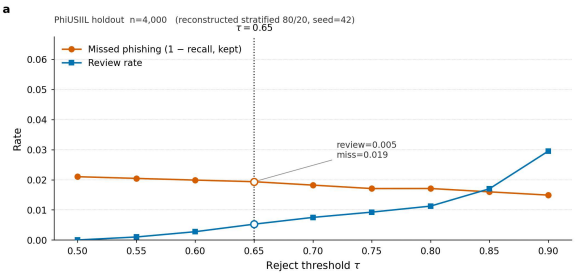


Fig. 3 Reject threshold τ on the PhiUSIIL holdout (n=4000). At τ=0.65 the review rate is 0.005 and kept-set recall is 0.981. Missed-phishing rate is 1−recall among non-review predictions.

allowed if the final domain were `aku.edu`" [22]. The official card has no reasons. Ten generated cards (seven impersonations, two official hosts, one extra-letter Chughtai typo) were checked against that intent.

One of the ten is a miss that we keep in the record. `https://chughtaislab.com/e-report` (extra letter relative to official `chughtailab.com`) is a near-miss that the evidence-card rules describe as a Levenshtein lookalike, but LightGBM scores it benign. The card and the classifier can disagree; we report both, and Section 5.5 shows the disagreement is systematic rather than anecdotal, since the rules outperform the model on this family. A wrapped Bitly example (`bit.ly/akuh-lab` → `aku-reports.tk`) is scored phish with both wrapper-then-unknown and brand-mismatch reasons. SHAP is not computed on any of these cards [2, 21, 46].

The card is actionable by construction: its repair field names an official domain, which is a check a recipient can perform. We report no human evaluation, and we therefore do not claim that the card is more actionable than a SHAP ranking. The audit worksheet is prepared and its rater columns are empty.

5.10 What the numbers do not say

Five limits follow from the design, three of them quantified above.

First, chain-aware detection is unevaluated on live traffic. The chain group works on the wrapped slice and produces no measurable change elsewhere, but every hop list in this work is synthetic and dry [9, 12]. The measurement that would settle it, the fraction of live healthcare phishing that passes through a wrapper, is longitudinal and cannot be back-filled from a frozen corpus.

Second, the detector depends on registry coverage, and Section 5.6 quantifies that dependence: 0.737 false positives on hospitals off the registry.

Third, the healthcare slice has ten phishing URLs and nineteen discriminative rows. Its intervals are wide, and it cannot rank this system against 2025 healthcare papers trained on generic feeds [2, 3].

Fourth, every evaluation slice except those nineteen rows is constructed from the brand registry and then scored by features that consult the same registry. Leave-one-brand-out removes the training leak; it does not remove that construction circularity. Only an external corpus with impersonation labels [16, 17] or a live feed can.

Table 6 Evidence cards, verbatim detector output. **(a)** a lookalike claiming Aga Khan University Hospital; **(b)** the official host. The counterfactual is a brand-constrained repair naming an official domain, not a SHAP attribution. `[?]` marks a hop whose status was not resolved.

(a) Flagged	
URL	`https://familyhifazat-aku.tk/lab-results`
Claimed identity	Aga Khan University Hospital
Actual host	`familyhifazat-aku.tk` (eTLD+1 `familyhifazat-aku.tk`)
Chain	`familyhifazat-aku.tk [?]`
Why flagged	Claimed identity 'Aga Khan University Hospital' but final host '`familyhifazat-aku.tk`' is not in the official domain list (expected e.g. `aku.edu`). Suspicious TLD (`tk`).
Counterfactual	Would be allowed if the final domain were `aku.edu`.
Decision	**phish** (p_{phish}=1.000)
(b) Not flagged	
URL	`https://hospitals.aku.edu/`
Claimed identity	Aga Khan University Hospital
Actual host	`hospitals.aku.edu` (eTLD+1 `aku.edu`)
Chain	`hospitals.aku.edu [200]`
Why flagged	Not flagged: final host '`hospitals.aku.edu`' is an official domain for 'Aga Khan University Hospital'.
Counterfactual	No change needed; final domain `aku.edu` is official for Aga Khan University Hospital.
Decision	**benign** (p_{phish}=0.000)

Fifth, the per-family recalls are measured on constructed `.invalid` patterns rather than on live campaigns [13, 14]. They establish that the model is blind to the combination-TLD family and that the rule is not. They do not establish how often that family is used in the wild, and we make no claim about its prevalence.

6 Discussion

6.1 Why impersonation resists a URL classifier

A rule that consults an official-domain registry, with no classifier attached, detects brand impersonation better than a gradient-boosted model trained on 20 000 PhiUSIIL URLs plus synthetic augmentation (Table 4). This does not argue against learning. It indicates which component holds the information.

The relation "this host is not any official host of the brand this URL claims" is a lookup between two strings, one of which comes from a curated list. It is not a statistical regularity in URL surface form. A model that observes the relation through four summary columns, on a corpus in which almost no row involves a healthcare brand at all, has little to learn it from. The combination-TLD result in Section 5.5 is the clean case: the impersonating host is correctly spelled, of ordinary length, and lexically unremarkable, so every feature the model relies on reads as benign, and recall is 0.000. The rule does not need the pattern to be frequent, and recalls at least 0.952.

The general lesson for phishing detection is that identity claims should be enforced as constraints rather than supplied as features. Section 5.7 shows the same failure on real hosts: LightGBM overrides a zero mismatch feature on two legitimate `nhs.uk` subdomains because their hostnames are long and hyphenated, and produces three false positives where the rule produces one. In both directions the model degrades a correct allow-list decision by averaging it against lexical evidence.

6.2 Why this is not just an allow-list

The obvious objection is that an allow-list is not a research contribution. Three of our measurements answer it.

We measure when the allow-list suffices, and it is not always. The rule recalls 0.700 on the healthcare slice against the model's 1.000, and 0.995 on wrapped chains, so a defender who deploys the rule alone loses recall on real PhiUSIIL phishing URLs that carry no brand claim. The combination of rule and model is what reaches 1.000 on both.

We measure when it fails, and the failure is severe and previously unquantified: 56 of 79 official hospital domains are flagged when their brand is off the registry (Section 5.6). An allow-list is only as good as its coverage, and this paper puts a number on the cost of a gap. That number, not the allow-list itself, is what a deploying institution needs.

We measure that the learned alternative is worse on the cheapest attack. A reader who concludes "just use a model" would ship a detector with 0.000 recall on combination-TLD impersonation.

6.3 Deployment

Scoring costs 6.5 ms per URL on one core, about 11 core-minutes per 100 000 URLs, so compute is not the constraint at inbox scale. Live redirect resolution is network-bound and bounded by design at five hops with a five-second timeout each; a deployment that cannot afford that latency inline can resolve chains asynchronously and score the first hop synchronously, at the cost of the chain features that Section 5.5 shows are what detect wrapper-then-unknown chains.

The real cost is registry maintenance. Sixty-seven brands and 79 hostnames must be curated and kept current, and the LOBO-A measurement says what a stale or incomplete registry costs. This suggests an operating discipline: track registry coverage as the primary health metric, alert on claimed brands that resolve to no registry entry, and treat an unrecognised brand as a request for curation rather than as a detection.

The reject option supports this. At τ=0.65 the review rate on the PhiUSIIL holdout is 0.005, so a queue of low-confidence URLs is small enough to inspect, and the evidence card gives a reviewer the claimed identity, the actual host, and the official domain that would have been accepted.

6.4 Two findings that constrain the design

Within the brand group, the PhiUSIIL lift comes entirely from near-miss edit distance to an official hostname rather than from the mismatch rule (Section 5.3). Edit distance transfers to a corpus of mostly non-healthcare URLs; the allow-list rule is deployment-specific and shows its value only on the hard slices. A system that must work without a curated registry should keep the distance feature and expect nothing from the rule.

On the wrapped slice, where 198 of 200 rows carry no claimed brand, detection comes from the chain group, and the mismatch feature alone recalls 0.010. That slice is the only place in this work where redirect-awareness does something measurable, and it is synthetic. We treat this as evidence that the chain features are correctly specified, not as evidence about live traffic.

Data availability statement. The brand registry, all evaluation slices, the detector, the evaluation harness and the figure scripts are released with the paper. Numbers reproduce deterministically: `src/train.py` regenerates the ablation and the saved model bundles bit-for-bit at seed 42, `src/run_evaluation.py` regenerates the leave-one-brand-out, baseline, stratified and interval results, and `tests/test_invariants.py` asserts the properties the claims rest on, including the filter-leakage cases and the slice row counts. Pinned dependency versions are given in `requirements.txt`; LightGBM and a compiled Levenshtein implementation are both required, because the code falls back to different implementations when they are absent.

PhiUSIIL is redistributed by the UCI Machine Learning Repository under its own terms [9] and is not included; `data/README.md` records the retrieval URL, the download date and the label convention, including the inversion applied on load. The OpenPhish snapshot is a live community feed and is not a labelled train/test split.

7 Conclusion

We asked whether grounding healthcare phishing detection in an explicit registry of official hospital domains helps, built a detector to test the question, and measured the answer under a protocol designed to let the claim fail.

Grounding helps, and we can say where. Out-of-fold, brand grounding produces no false positive on 79 official hospital domains (Wilson 95% CI $[0.000, 0.046]$) where a lexical model false-positives at 0.157. Redirect-chain features, not brand features, are what detect wrapper-then-unknown chains: on the wrapped slice 198 of 200 rows carry no brand claim and the mismatch feature alone recalls 0.010. On a generic corpus the brand-group lift reduces to one feature, near-miss edit distance to an official hostname.

Grounding is best applied as a rule. A lookup against the registry, with no classifier, recalls 0.978 on 457 impersonation patterns against the trained model's 0.569, and recalls at least 0.952 on every impersonation family including the combination-TLD family, on which the model recalls 0.000. Identity claims should be enforced as constraints rather than supplied to a classifier as features.

The binding constraint is coverage, not capacity. A hospital absent from the registry is flagged as phishing in 56 of 79 cases, so the system is a per-institution tool and registry curation is its primary operational cost. We are not aware of a prior brand-impersonation system that reports this measurement.

Four claims this evaluation does not support: that the classifier beats the rule on impersonation; that chain-aware detection works on live traffic, since every hop list here is synthetic; that the evidence card is more actionable than a SHAP ranking, which needs a rater study; and any recall figure for hospitals outside the registry. The first is a finding rather than a gap. The remaining three are addressable with a longitudinal feed collection, a rater study on the prepared audit sheet, and evaluation on an external corpus with impersonation labels [16, 17].

Funding. The authors received no funding for this research.

References

[1] Elongha, G., Liu, M.: Detecting phishing emails targeting healthcare practitioners: a domain-specific ensemble approach using diverse datasets? Issues In Information Systems (2025) https://doi.org/10.48009/4_iis_2025_131

[2] Karapiperis, D., Feretzakis, G., Kalles, D., Mitropoulos, S.: Explainable ai for url threat detection in healthcare cybersecurity: A case study using lime and shap. Studies in Health Technology and Informatics (2025) https://doi.org/10.3233/shti250712

[3] Immanuvel Arockiasamy, J.M.: Securing telehealth with state-of-theart machine learning: A devsecops framework for real-time phishing detection. Health Informatics - An International Journal **14**(2), 01–15 (2025) https://doi.org/10.5121/hiij.2025.14201

[4] Feretzakis, G., Karapiperis, D., Mitropoulos, S.: Real-time phishing campaign detection for healthcare organizations: An explainable ai approach using semantic clustering. Studies in Health Technology and Informatics (2026) https://doi.org/10.3233/shti260852

[5] Mohammad, R.M., Thabtah, F., McCluskey, L.: Intelligent rule-based phishing websites classification. IET Information Security **8**(3), 153–160 (2014) https://doi.org/10.1049/iet-ifs.2013.0202

[6] Mohammad, R.M., Thabtah, F., McCluskey, L.: Tutorial and critical analysis of phishing websites methods. Computer Science Review **17**, 1–24 (2015) https://doi.org/10.1016/j.cosrev.2015.04.001

[7] Mamun, M.S.I., Rathore, M.A., Lashkari, A.H., Stakhanova, N., Ghorbani, A.A.: Detecting malicious urls using lexical analysis. Lecture Notes in Computer Science, 467–482 (2016) https://doi.org/10.1007/978-3-319-46298-1_30

[8] Sahingoz, O.K., Buber, E., Demir, O., Diri, B.: Machine learning based phishing detection from urls. Expert Systems with Applications **117**, 345–357 (2019) https://doi.org/10.1016/j.eswa.2018.09.029

[9] Prasad, A., Chandra, S.: Phiusiil: A diverse security profile empowered phishing url detection framework based on similarity index and incremental learning. Computers & Security **136**, 103545 (2024) https://doi.org/10.1016/j.cose.2023.103545

[10] Rawla, A., Singh, S., Daniyal, M., Dubey, P.: Detection of phishing attacks in phiusiil dataset using deep learning. Procedia Computer Science **259**, 543–552 (2025) https://doi.org/10.1016/j.procs.2025.04.003

[11] Bhamaryuda, R., Putrada, A.G., Dawani, F.: Lightgbm with class weights for solving imbalance in nlp-based url phishing detection. In: 2026 14th International Conference on Information and Communication Technology (ICoICT) (2026). https://doi.org/10.1109/icoict71118.2026.11664766

[12] Bell, S., Komisarczuk, P.: Measuring the effectiveness of twitter's url shortener (t.co) at protecting users from phishing and malware attacks. In: ACSW '20: Australasian Computer Science Week 2020 (2020). https://doi.org/10.1145/3373017.3373019

[13] Agten, P., Joosen, W., Piessens, F., Nikiforakis, N.: Seven months' worth of mistakes: A longitudinal study of typosquatting abuse. In: Network and Distributed System Security Symposium (2015). https://doi.org/10.14722/ndss.2015.23058

[14] Kumar, R., Basak, P., Patni, J.C.: Pcdf: Phishing attack using cyrillic homograph domain forgery. In: 2025 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS) (2025). https://doi.org/10.1109/icbds67396.2025.11376673

[15] Jarczewski, M., Białczak, P., Mazurczyk, W.: Phishing website impersonation: Comparative analysis of detection and target recognition methods. Applied Sciences **16**(2), 640 (2026) https://doi.org/10.3390/app16020640

[16] Vu, T.N.: Phishvn: A time-stamped vietnamese url phishing dataset with impersonation-scenario labels and confidence tiers. Data in Brief, 113195 (2026) https://doi.org/10.1016/j.dib.2026.113195

[17] Dalton, T., Gowda, H., Rao, G., Pargi, S., Hadj Khodabakhshi, A., Rombs, J.,

Jou, S., Marwah, M.: Phreshphish: A real-world, high-quality, large-scale phishing website dataset and benchmark. arXiv preprint arXiv:2507.10854 (2025) 2507.10854

[18] Abdelnabi, S., Krombholz, K., Fritz, M.: Visualphishnet: Zero-day phishing website detection by visual similarity. In: CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security (2020). https://doi.org/10.1145/3372297.3417233

[19] Xiang, G., Hong, J., Rose, C.P., Cranor, L.: Cantina+. ACM Transactions on Information and System Security **14**(2), 1–28 (2011) https://doi.org/10.1145/2019599.2019606

[20] Kocyigit, E., Korkmaz, M., Sahingoz, O.K., Diri, B.: Enhanced feature selection using genetic algorithm for machine-learning-based phishing url detection. Applied Sciences **14**(14), 6081 (2024) https://doi.org/10.3390/app14146081

[21] Mayo, C., Tchuindjang, M., Brohi, S., Ersotelos, N.: A temporally dynamic feature-extraction framework for phishing detection with lime and shap explanations. Future Internet **18**(2), 101 (2026) https://doi.org/10.3390/fi18020101

[22] Wachter, S., Mittelstadt, B., Russell, C.: Counterfactual explanations without opening the black box: Automated decisions and the gdpr. SSRN Electronic Journal (2017) https://doi.org/10.2139/ssrn.3063289

[23] Zhang, Y., Hong, J.I., Cranor, L.F.: Cantina. In: WWW'07: 16th International World Wide Web Conference (2007). https://doi.org/10.1145/1242572.1242659

[24] Buber, E., Dırı, B., Sahingoz, O.K.: Detecting phishing attacks from url by using nlp techniques. In: 2017 International Conference on Computer Science and Engineering (UBMK) (2017). https://doi.org/10.1109/ubmk.2017.8093406

[25] Jain, A.K., Gupta, B.B.: Towards detection of phishing websites on client-side using machine learning based approach. Telecommunication Systems **68**(4), 687–700 (2018) https://doi.org/10.1007/s11235-017-0414-0

[26] Jain, A.K., Gupta, B.B.: A machine learning based approach for phishing detection using hyperlinks information. Journal of Ambient Intelligence and Humanized Computing **10**(5), 2015–2028 (2019) https://doi.org/10.1007/s12652-018-0798-z

[27] Rao, R.S., Pais, A.R.: Jail-phish: An improved search engine based phishing detection system. Computers & Security **83**, 246–267 (2019) https://doi.org/10.1016/j.cose.2019.02.011

[28] Rao, R.S., Vaishnavi, T., Pais, A.R.: Catchphish: detection of phishing websites by inspecting urls. Journal of Ambient Intelligence and Humanized Computing **11**(2), 813–825 (2020) https://doi.org/10.1007/s12652-019-01311-4

[29] Rao, R.S., Pais, A.R.: Detection of phishing websites using an efficient feature-based machine learning framework. Neural Computing and Applications **31**(8), 3851–3873 (2019) https://doi.org/10.1007/s00521-017-3305-0

[30] Korkmaz, M., Sahingoz, O.K., Diri, B.: Detection of phishing websites by using machine learning-based url analysis. In: 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (2020). https://doi.org/10.1109/icccnt49239.2020.9225561

[31] Srinivasa Rao, R., Pais, A.R.: Detecting phishing websites using automation of human behavior. In: ASIA CCS '17: ACM Asia Conference on Computer and Communications Security (2017). https://doi.org/10.1145/3055186.3055188

[32] Le, H., Pham, Q., Sahoo, D., Hoi, S.C.H.: Urlnet: Learning a url representation with deep learning for malicious url detection. arXiv preprint arXiv:1802.03162 (2018) 1802.03162

[33] Korkmaz, M., Kocyigit, E., Sahingoz, O.K., Diri, B.: A hybrid phishing detection system using deep learning-based url and content analysis. Elektronika ir Elektrotechnika **28**(5), 80–89 (2022) https://doi.org/10.5755/j02.eie.31197

[34] Rashid, F., Doyle, B., Seneviratne, S.: Feuda-bert: Federated url domain-aware bert for phishing url detection. In: 2025 IEEE 50th Conference on Local Computer Networks (LCN) (2025). https://doi.org/10.1109/lcn65610.2025.11146376

[35] Rao, R.S., Kondaiah, C., Pais, A.R., Lee, B.: A hybrid super learner ensemble for phishing detection on mobile devices. Scientific Reports **15**(1) (2025) https://doi.org/10.1038/s41598-025-02009-8

[36] Prasad, A., Yadav, V., Solanki, C., Goswami, H., Jha, T., Nagal, D.: Stealthphisher: A defensive framework against phishing attack using hybrid deep learning and genai. Expert Systems with Applications **299**, 130205 (2026) https://doi.org/10.1016/j.eswa.2025.130205

[37] Bell, S., Komisarczuk, P.: An analysis of phishing blacklists: Google safe browsing, openphish, and phishtank. In: ACSW '20: Australasian Computer Science Week 2020 (2020). https://doi.org/10.1145/3373017.3373020

[38] Skula, I., Kvet, M.: A framework for preparing a balanced and comprehensive phishing dataset. IEEE Access **12**, 53610–53622 (2024) https://doi.org/10.1109/access.2024.3387437

[39] Frąszczak, D., Frąszczak, E.: Phishingwebcollector: Async python library for automated phishing feed collection. SoftwareX **33**, 102463 (2026) https://doi.org/10.1016/j.softx.2025.102463

[40] Spaulding, J., Upadhyaya, S., Mohaisen, A.: The landscape of domain name typosquatting: Techniques and countermeasures. In: 2016 11th International Conference on Availability, Reliability and Security (ARES) (2016). https://doi.org/10.1109/ARES.2016.84

[41] Fouss, B., Ross, D.M., Wollaber, A.B., Gomez, S.R.: Punyvis: A visual analytics approach for identifying homograph phishing attacks. In: 2019 IEEE Symposium on Visualization for Cyber Security (VizSec) (2019). https://doi.org/10.1109/vizsec48167.2019.9161590

[42] Yazdani, R., Toorn, O., Sperotto, A.: A case of identity: Detection of suspicious idn homograph domains using active dns measurements. In: 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW) (2020). https://doi.org/10.1109/eurospw51379.2020.00082

[43] Zouahi, H., Talhi, C., Boudar, O.: Vizcheck: Enhancing phishing attack detection through visual domain name homograph analysis. In: 2024 7th Conference on Cloud and Internet of Things (CIoT) (2024). https://doi.org/10.1109/ciot63799.2024.10757049

[44] Kurnaz, D., Sarı, I.K., Zeydan, E.: Multilayer algorithmic framework with llm integration for typosquatting and phishing domain analysis. In: 2026 29th Conference on Innovation in Clouds, Internet and Networks (ICIN) (2026). https://doi.org/10.1109/icin69025.2026.11481803

[45] Ribeiro, M.T., Singh, S., Guestrin, C.: "why should i trust you?". In: KDD '16: The 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2016). https://doi.org/10.1145/2939672.2939778

[46] Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems (2017). https://arxiv.org/abs/1705.07874

[47] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y.: Lightgbm: A highly efficient gradient boosting decision tree. In: Advances in Neural Information Processing Systems, vol. 30 (2017)

[48] Chow, C.K.: On optimum recognition error and reject tradeoff. IEEE Transactions on Information Theory **16**(1), 41–46 (1970) https://doi.org/10.1109/TIT.1970.1054406

[49] Herbei, R., Wegkamp, M.H.: Classification with reject option. Canadian Journal of Statistics **34**(4), 709–721 (2006) https://doi.org/10.1002/cjs.5550340410

[50] Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady **10**(8), 707–710 (1966)