CHANNEL ASYMMETRY IN MULTI-COMPONENT PHISHING DETECTION: A CROSS-VALIDATED COMPARISON OF URL, HTML AND EMAIL-TEXT CLASSIFIERS

Nazokat Mirtakhirova
Danish Shehzad
Department of Computer Science, New Uzbekistan University Tashkent, Uzbekistan
260674@newuu.uz
d.shehzad@newuu.uz
DOI:https://doi.org/10.5281/zenodo.23250433

Abstract—Multi-component phishing detectors are normally reported as a single headline accuracy, which conceals how differently the individual channels behave. This paper re-examines a three-component lightweight framework — a DistilBERT classifier over URLs, a Random Forest over thirteen structural HTML features, and a second DistilBERT over email body text — and asks not how well the framework performs but how unequally its channels perform. The three channels receive 99.87 %, 93.81 % and 96.79 % accuracy under the same five-fold cross-validation procedure. The spread in point estimates, however, is the less important part of the story: the standard deviation of accuracy across folds ranges by a factor of 15.7 from the URL to the email channel, and the standard deviation of recall by a factor of 37.5. Both content channels also exhibit high precision above recall, meaning the errors are in missed phishing rather than false alarms, and that the imbalance is largest (1.27 pp) in the email channel that faces the human victim. We argue that the per-channel error rate alone is not enough for designing a fusion layer and the dispersion across folds and the direction of precision–recall gap should be reported together.

Keywords—phishing detection; DistilBERT; Random Forest; cross-validation; channel fusion; model reliability

I. INTRODUCTION

Phishing is still one of the most prevalent attack vectors against individuals and organisations, and it does not come through one medium. A campaign might have a plausible sender, a persuasive message body, a deceptive HTML landing page and a clever URL. Detection frameworks have adapted by becoming multi-component: separate classifiers are trained on different signals and their verdicts combined.

But reporting practice hasn't changed to keep pace with that change. A multi-component system is usually summarised by one accuracy figure, or by a list of per component accuracies presented as if they were interchangeable. We start with the observation that they are not. Two classifiers that both report 95 % accuracy, for example, may differ strikingly in the stability of that number in terms of resampling of the data and whether their residual errors are false alarms or missed attacks. For a security system, these two properties matter far more than the first decimal place of the headline number.

This paper takes a three-component lightweight framework and looks at it in this light. We deliberately chose a relatively low cost (two DistilBERT models and one Random Forest) framework because our deployment target is a resource-constrained mail gateway and not a GPU

© Ushbu material Creative Commons Attribution 4.0 International (CC BY 4.0) litsenziyasi asosida tarqatiladi.

cluster. We have studied all three components in the same five-fold cross validation protocol, and this makes the dispersion statistics comparable. We make three observations. First, the point estimates are all six percentage points in accuracy for URLs, i.e. 99.87 % in URLs and 93.81 % in HTML structure. In terms of error rates, the gap is much larger. The HTML channel makes 47.6 times as many errors as the URL channel. Second, the fold-to-fold dispersion is also very unequal. The 95 % confidence interval on accuracy is 0.20 percentage points wide for URLs and 3.13 percentage points wide for email text, a factor of 15.7. Third, both content channels are accurate over recall, so their errors are mostly on the phishing class; the gap reaches 1.27 percentage points in the email channel.

Taken together, these results suggest a design rule: the fusion layer should weigh component verdicts by their reliability as opposed to their mean accuracy, and the channel that is most challenging to learn is the one that is the main source of the attacks that reach a human reader.

II. Related Work

Data from URL-based detection has been the longest quantitative record. Bahnsen et al. [1] trained an LSTM on two million URLs and achieved 98.7 % accuracy; Le et al. [2] introduced URLNet and applied character-level and word-level convolutions to five million URLs and achieved 97.2 % accuracy. Haynes et al. [3] moved the task to lightweight transformers suitable for mobile deployment. The main characteristic of this line of work is high and stable accuracy with our URL results achieving that. Structural and content-based detection results show clearly lower numbers. Aljofey et al. [4] extracted character-level TF-IDF features from the HTML and plaintext of 60,000 pages and reached 96.25 % with XGBoost; Opare et al. [5] applied deep learning to HTML and reported 93 %. Fang et al. [6] combined recurrent and convolutional layers with attention over email content. Transformer-based email classification has been explored by Jamal et al. [7], who combined RoBERTa and DistilBERT at substantial GPU cost.

What the literature does not provide is a controlled comparison of these channels under one protocol. The figures are drawn from different papers, different datasets and different splitting strategies, so the apparent ordering-URL easiest, HTML and email harder-cannot be separated from methodological variation. Nor is dispersion usually reported at all: none of the papers cited above indicate a standard deviation across folds. The contribution of the present paper is to fix the evaluation protocol and to treat dispersion as a first-class result, rather than a missing one.

III. Experimental Setup

A. Channels and data

Each channel is trained on the dataset appropriate to its signal. The URL channel uses LegitPhish (2025), reduced from 101,000 raw samples to 99,888 after removal of unlabelled rows, duplicates and malformed strings, and normalised to lowercase with protocol and www prefixes stripped; the resulting class split is 62,508 legitimate to 37,380 phishing. The HTML channel uses CIC-Trap4Phish (2025), which supplies thirteen pre-extracted structural features over a balanced 9,999 / 9,999 split and therefore needs no resampling. The email-text channel uses the phishing email corpus released by zefang-liu (2024), 18,634 messages after cleaning, split 11,322 legitimate to 7,328 phishing, itself an aggregation of Enron, Nazario and SpamAssassin material.

Class imbalance was handled identically in the two text channels, by weighted cross-entropy with weights derived from the class frequencies (0.799 / 1.336 for URLs, 0.824 / 1.273 for email).

© Ushbu material Creative Commons Attribution 4.0 International (CC BY 4.0) litsenziyasi asosida tarqatiladi.

B. Models

DistilBERT [8] was chosen for both text channels over BERT [9] and RoBERTa because it retains approximately 97 % of BERT's language understanding at 40 % fewer parameters and roughly 60 % faster inference, which is the governing constraint for gateway deployment. The URL model uses a maximum sequence length of 128 tokens, batch size 32, learning rate 2e-5, and three epochs. The email model uses 256 tokens, reflecting the greater length of message bodies, with dropout and early stopping; training halted after nine epochs with the loss falling from 0.5444 to 0.2011. The HTML channel uses a Random Forest [10] of 100 trees on the thirteen supplied features; no feature scaling is required and no explicit regularisation was added, since bagging and feature subsampling already constrain variance.

C. Protocol and derived statistics

All three channels were evaluated by five-fold cross-validation, each fold serving once as the test partition. We report the mean and the standard deviation of accuracy, precision, recall and F1 across folds. From these we derive three quantities that the raw means do not expose: the error rate 1 − accuracy, which linearises differences near the ceiling; the 95 % confidence interval of the mean, computed as $\bar{x} \pm t(4) \cdot s/\sqrt{5}$ with $t(4) = 2.776$; and the precision–recall gap, whose sign indicates whether residual errors are false alarms or missed attacks.

D. Computational framing

The choice of components is driven by the deployment point. A mail gateway processing a continuous stream of messages is under a latency budget in the tens of milliseconds per item, and usually without a dedicated accelerator. BERT-base is about 110 million parameters and consumes at least half a gigabyte of memory and hundreds of milliseconds per inference on commodity hardware so it is beyond the budget for high volume triage. The distilled version and the forest were chosen for the reasons of accuracy, not for accuracy. That matters for the interpretation of the results below: the comparison is between channels on a fixed and low computational budget and a larger model might narrow the gaps we report. Whether it would also narrow the dispersion gaps is an open question, since variance is the result of the heterogeneity of the data as much as of model capacity.

IV. Results

A. Per-channel performance

Table I gives the cross-validated means and standard deviations. Fig. 1 plots them with error bars.

TABLE I. Cross-validated performance of the three channels (mean ± SD, 5 folds)

Channel	Acc.	Prec.	Rec.	F1
URL (DistilBERT)	.9987	.9984	.9981	.9983
	±.0008	±.0018	±.0010	±.0011
HTML (Random Forest)	.9381	.9411	.9347	.9379
	±.0027	±.0049	±.0046	±.0027
Email text (DistilBERT)	.9679	.9653	.9526	.9585

© Ushbu material Creative Commons Attribution 4.0 International (CC BY 4.0) litsenziyasi asosida tarqatiladi.

Channel	Acc.	Prec.	Rec.	F1
	±.0126	±.0067	±.0358	±.0174

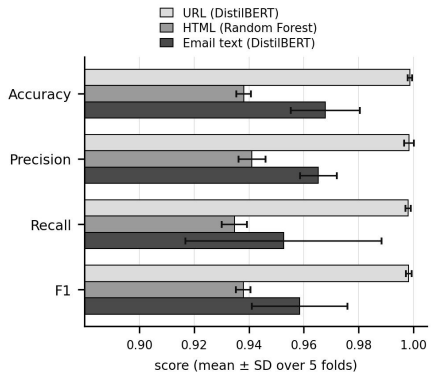


Fig. 1. Cross-validated scores by channel. Error bars are one standard deviation across the five folds.

The ordering is unambiguous and the URL channel is close to saturation. Read as accuracies the three figures look comparable; read as error rates they do not. The URL channel errs on 0.13 % of samples, the email channel on 3.21 % and the HTML channel on 6.19 % — ratios of 24.7 and 47.6 against the URL baseline.

B. Dispersion across folds

The more informative difference is in stability. Table II reports the derived statistics.

TABLE II. DERIVED RELIABILITY STATISTICS BY CHANNEL

Channel	Err. %	Miss %	95 % CI (acc.)	CI width
URL	0.13	0.19	.9977–.9997	.0020
HTML	6.19	6.53	.9347–.9415	.0067
Email text	3.21	4.74	.9523–.9835	.0313

The confidence intervals do not overlap, so the three channels are separated at the conventional level despite only five folds. Their widths, however, differ by far more than their centres: the email interval is 15.7 times wider than the URL interval and 4.7 times wider than the HTML interval. The single largest dispersion figure in the whole evaluation is the standard deviation of email recall, 0.0358, which is 35.8 times the corresponding URL value and gives a coefficient of variation of 3.76 % against 0.10 %. In practical terms, the email classifier's sensitivity to phishing moved between roughly 92 % and 99 % depending on which fold it was tested on.

C. Direction of the residual error

Accuracy is symmetric between the two error types; a security system is not. Table III decomposes the gap.

TABLE III. PRECISION–RECALL GAP AND ITS SECURITY READING

Channel	P − R (pp)	CV of R	Dominant error
URL	+0.03	0.10 %	balanced

© Ushbu material Creative Commons Attribution 4.0 International (CC BY 4.0) litsenziyasi asosida tarqatiladi.

Channel	P − R (pp)	CV of R	Dominant error
HTML	+0.64	0.49 %	missed phishing
Email text	+1.27	3.76 %	missed phishing, unstably

Both content channels are conservative: when they do raise an alarm they are usually right, but they let a measurable share of phishing through. The URL channel is effectively balanced. The ordering of the gap follows the ordering of difficulty, and its largest value coincides with its least stable estimate.

D. What the URL model attends to

The quantitative pattern invites a qualitative check: is the URL channel easy because the signal is genuinely lexical, or because the dataset is separable for some incidental reason? To probe this we applied LIME [11], which perturbs the input and observes the change in the model's output to attribute the decision to individual input fragments.

For the constructed probe secure-paypal-login.phishing-site.com/steal/credentials the model returned a phishing confidence of 1.00, and the attribution concentrated on four substrings: credentials, login, secure and paypal. Three of them are generic lures and the fourth is a brand token appearing outside its registrable domain. The decision is based on exactly the same features a human analyst would see and validate the lexical reading of the channel in Section V-A. This same observation has a caution. A classifier based on brand tokens and lure words is good against the attack patterns in its training distribution and weak against URLs that don't have them at all: a compromised legitimate domain, a shortened link or a brand new name with no lexical cue. The near-zero variance we find across folds is a comment about the homogeneity of the dataset, not on robustness in the wild. That distinction is made clear by attribution and accuracy because reporting attribution makes it apparent; accuracy alone does not.

V. Discussion

A. Why the channels differ

A URL is a short, highly constrained string, and phishing URLs leave surface traces — brand substrings, misleading subdomains, unusual path tokens — that a subword model can learn from a hundred thousand examples. The decision is close to lexical pattern matching, and the near-saturated, low-variance result is what one should expect. Email body text is the opposite: the signal is persuasive language, urgency, impersonation and pretext, which overlaps heavily with legitimate business correspondence. The decision boundary is semantic, the class defining features are spread over whole messages and only 18,634 examples are sufficient for that problem. HTML structure remains between the two with only thirteen engineered features available to the classifier. The dispersion pattern is very clear: if the decision is lexical, folds are similar and there is little variance; if the decision is semantic and the corpus is heterogeneous (three source corpora merged), the composition of a fold substantially changes the problem.

B. Consequences for the fusion layer

The decision boundary is semantic, the class defining features are spread over whole messages and only 18,634 examples are sufficient for that problem. HTML structure remains between the two and only thirteen engineered features are available to the classifier. The

© Ushbu material Creative Commons Attribution 4.0 International (CC BY 4.0) litsenziyasi asosida tarqatiladi.

dispersion pattern is very clear: if the decision is lexical, folds are similar and there is little variance. If the decision is semantic and the corpus is heterogeneous (three source corpora merged), the composition of a fold substantially changes the problem.

C. The security asymmetry

The practical concern is that the channel ordering is inverse to the channel's proximity to the victim. URLs are the easiest to classify and filter by other means; email text is the hardest to classify and is precisely the surface on which social-engineering attacks succeed. An attacker who embeds a legitimate or newly registered URL in a persuasive message is routed to the weakest and least stable part of the system. A headline accuracy calculated over a mixture of the three channels would report this configuration as near-perfect.

D. Implications for reporting practice

The larger point is that multi-component results are published. Of the works we examined in Section II, none report a dispersion statistic and most report a single accuracy per model. Our figures show that two components of the same framework can differ by more than an order of magnitude in the stability of that number but appear only a few percentage points apart. So a reader comparing published systems on headline accuracy is therefore comparing quantities whose uncertainty is unknown and demonstrably unequal.

We would propose three additions to the standard results table, none of which costs extra experimentation once cross-validation is already being run: the standard deviation across folds for every reported metric; the error rate rather than, or alongside, the accuracy, since differences near the ceiling are otherwise compressed into invisibility; and the signed precision–recall gap, which states in one number whether a detector fails by crying wolf or by staying silent. The first two cost a column each; the third costs a subtraction.

VI. Limitations

The comparison is between channels as instantiated here, not between channels in the abstract: each component is trained on its own dataset, so dataset difficulty and channel difficulty are confounded. Separating them would require a single corpus annotated on all three signals, which to our knowledge does not exist. Five folds give only four degrees of freedom, so the confidence intervals are wide and the dispersion estimates are themselves imprecise; ten-fold or repeated cross-validation would tighten them. Finally, the fusion strategies proposed in Section V-B are motivated by the measurements but have not yet been implemented or evaluated, and no end-to-end latency figures are reported, so the deployment claim rests on DistilBERT's published efficiency rather than on measurement in our own setting.

VII. Conclusion

Under one cross-validation protocol, the three channels of a lightweight phishing framework achieve 99.87 %, 96.79 % and 93.81 % accuracy, but differ by a factor of 15.7 in the width of their confidence intervals and by 37.5 in the variability of recall. Both content channels miss phishing more often than they raise false alarms, and in email text. This per-component accuracy table provides an optimistic picture of a multi-component detector. We recommend that dispersion across folds and the signed precision-recall gap be reported as standard, and fusion weights derived from interval lower bounds rather than point estimates. We will implement and test such a reliability-weighted fusion layer next.

© Ushbu material Creative Commons Attribution 4.0 International (CC BY 4.0) litsenziyasi asosida tarqatiladi.

REFERENCES

1. A. C. Bahnsen, E. C. Bohorquez, S. Villegas, J. Vargas, and F. A. González, "Classifying phishing URLs using recurrent neural networks," in Proc. APWG Symp. Electronic Crime Research (eCrime), 2017, pp. 1–18.
2. H. Le, Q. Pham, D. Sahoo, and S. C. H. Hoi, "URLNet: Learning a URL representation with deep learning for malicious URL detection," arXiv:1802.03162, 2018.
3. K. Haynes, H. Shirazi, and I. Ray, "Lightweight URL-based phishing detection using natural language processing transformers for mobile devices," Procedia Computer Science, vol. 191, pp. 127–134, 2021.
4. A. Aljofey et al., "An effective detection approach for phishing websites using URL and HTML features," Scientific Reports, vol. 12, 8842, 2022.
5. E. Opare et al., "HTML-based phishing detection using deep learning," 2024.
6. Y. Fang, C. Zhang, C. Huang, L. Liu, and Y. Yang, "Phishing email detection using improved RCNN model with multilevel vectors and attention mechanism," IEEE Access, vol. 7, pp. 56329–56340, 2019.
7. S. Jamal et al., "Improved phishing email detection using transformer models," 2024.
8. V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," arXiv:1910.01108, 2019.
9. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
10. L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
11. M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?' Explaining the predictions of any classifier," in Proc. ACM SIGKDD, 2016, pp. 1135–1144.

© Ushbu material Creative Commons Attribution 4.0 International (CC BY 4.0) litsenziyasi asosida tarqatiladi.