SYNTIA – Systems and Technologies in Artificial Intelligence Applications.
Volume 1 Issue 2 (June 2026). Page: 79-90.

Content-Based Screening of Gambling and Pornographic Pages in a URL Shortener Using Naive Bayes

Zulfikar Alip Takasihaeng[1], Nouval Trezandy Lapatta[2*], Sri Khaerawati Nur[2]

[1] *Informatics Engineering Study Program, Faculty of Engineering, Universitas Tadulako, Palu, Indonesia.*
[2] *Department of Information Technology, Faculty of Engineering, Universitas Tadulako, Palu, Indonesia.*

**Corresponding author: nouval@untad.ac.id*

Abstract: URL shorteners hide the destination of a link and can be used to spread online gambling and pornographic content. Blacklists catch only known addresses, so a step that reads the destination page before the short link is created is attractive. This study describes such a service and evaluates its classifier, which converts the page title, description, keywords, and body text to TF-IDF features (1–3-grams, 120 features) and assigns safe, gambling, or pornography with Naive Bayes; harmful pages are refused. The dataset had 2,167 pages, an undisclosed share of them template-based synthetic text, split 70:30 with augmentation on the training part only. On 644 test pages accuracy was 93.63% (95% CI 91.5–95.3%), macro F1-score 0.9375, and Cohen's kappa 0.904, yet five-fold cross-validation gave only 0.8335 ± 0.0235, a gap the source does not explain. As a filter, the classifier let 31 of 423 harmful pages pass (7.3%) and blocked 7 of 221 safe pages (3.2%). The safe class comes from English-language sites, so the classifier may partly separate languages and sources instead of content. No baseline, latency measurement, or test on real shortened links was performed. The results support the feasibility of content-aware screening with a lightweight model and define the checks needed before it is trusted as a safeguard.

Keywords : content-based filtering; malicious URL; Naive Bayes; online gambling; URL shortener

Received: March 9, 2026 | Revised: April 22, 2026 | Accepted: May 18, 2026 | Published: June 30, 2026

1. Introduction

A URL shortener replaces a long address with a short code that redirects to the original page. The convenience has a side effect: the person who receives a short link cannot see where it leads. Measurements of shortening services over two years showed that they are used to spread spam, phishing, and malware, and that the services' own countermeasures were limited [1]. In Indonesia, online gambling and pornographic sites are a particular concern, and links to them circulate on social media, often in shortened form. A shortener that refuses to create links to such destinations would remove one distribution channel.

Most defenses against malicious links work on the address. Blacklists block known bad addresses but cannot cover sites that appear faster than lists are updated, and lexical and host-based classifiers learn from the characters of the URL and from registration data to judge sites never seen before [2]. A survey of this literature groups methods into blacklist, heuristic, and learning-based approaches and notes that content features, the text and structure of the destination page, are an additional source of evidence at the cost of fetching the page [3]. Gambling and adult sites are a case in which content is especially informative, because the address of such a site (a changing domain with a brand name) says little, whereas the page text contains characteristic vocabulary. The present study evaluates only the content side.

Text classification with a bag-of-words representation is a mature technique, in which documents become vectors of weighted term frequencies and a learning algorithm assigns a category [4]. Support vector machines are a strong standard for this task because text has many features and few of them are irrelevant [5], while Naive Bayes is attractive for its speed, small memory, and simplicity. A lightweight

SYNTIA – Systems and Technologies in Artificial Intelligence Applications is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

classifier suits a shortener, which must decide within the time of a web request. The central question for a deployed filter, however, is not how a model scores on a collection but how often it lets harmful pages through and how often it blocks legitimate ones, and what the collection allows one to conclude.

This article reports a content-aware shortener that classifies the destination page into safe, online gambling, or pornography with TF-IDF features and Naive Bayes, and evaluates the classifier on a collection of 2,167 pages. We ask four research questions. (RQ1) How accurate is the classifier on held-out pages, with uncertainty? (RQ2) How does it behave as a filter, in terms of harmful pages passed and safe pages blocked? (RQ3) Can the gap between cross-validation and test accuracy be interpreted? (RQ4) What does the construction of the dataset allow us to conclude, given that the classes come from different kinds of sources? The contributions are a description of the system, interval estimates and filter-oriented metrics recomputed from the published confusion matrix, an analysis of the cross-validation gap, an analysis of the source and language confound, and a list of checks and baselines that would make the results suitable for deployment decisions. Section 2 describes the system and the methods, Section 3 reports and discusses results, and Section 4 concludes.

2. Method

2.1. System architecture

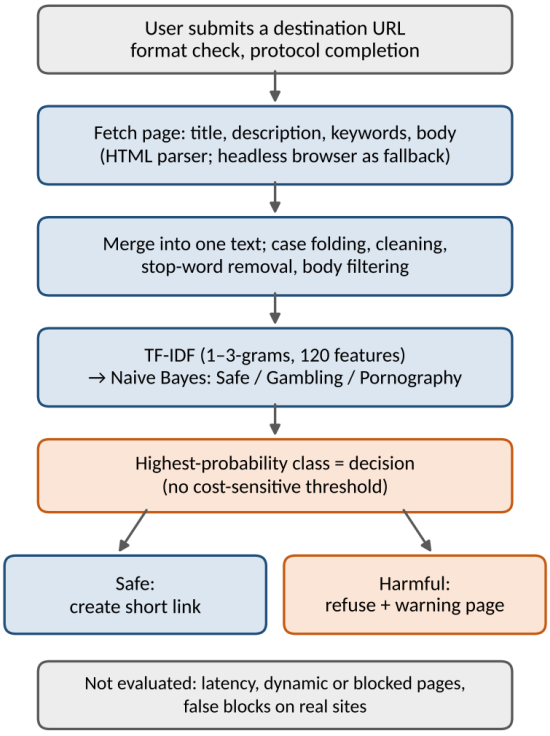


Figure 1. Processing flow of the content-aware shortener.

The service is a web application with a React and TypeScript front end and a Node.js (Express) back end that stores links through an object-relational mapper. When a user submits a URL, the back end validates its format and fetches the page with an HTTP client, extracts the title, description, meta keywords, and body text with an HTML parser, and falls back to a headless browser when ordinary fetching fails on dynamic pages. The fields are merged into one text and sent to a Python service that preprocesses it, computes TF-IDF features, and classifies it (Fig. 1). If the predicted class is gambling or pornography the back end refuses to create the short link and the interface shows the category, the confidence, and the reason; otherwise a unique code is generated, optionally from a user-chosen slug that is checked for duplicates. The classifier is the only decision component described in the source; a keyword list and risk score that the project description mentions are not evaluated.

2.2. Data

The dataset has 2,167 texts in three classes: 735 safe, 734 online gambling, and 698 pornography (Table 1). Each text is the concatenation of the title, description, keywords, and body text of a web page. The harmful classes were collected from pages that contain gambling or adult content and from posts on a social network that promote such content, searched with characteristic keywords. The safe class was collected from educational, technology, programming, science, news, and health sites such as Coursera, edX, NASA, WHO, TechCrunch, and OpenLearn. In addition, synthetic texts were generated from templates by varying keywords and sentence patterns, making up about 15% of the gambling class and 12% of the pornography class; the stratified split placed some of these synthetic texts in the test set, so the test metrics partly reflect template text rather than scraped pages alone. Labels were assigned from the source, the page's dominant keywords, and the match between content and category. Collection ran from March to May 2024 using a Python scraper built on requests and BeautifulSoup, drawing on 40 sites for the gambling class and 35 for the pornography class. We did not store page screenshots or full images; scraping kept only the text fields listed above, and each harmful page was reviewed by one author before labeling rather than rendered for general viewing.

Table 1. Dataset composition.

Class	Texts	Share
Safe	735	33.9%
Online gambling	734	33.9%
Pornography	698	32.2%
Total	2,167	100%

The origin of each class is known only in outline. The classes differ not only in topic but in origin, language, and probably in style: the safe class is described as international educational and news sites, whereas the harmful classes come from sites and posts that target Indonesian users. The source does not state the language of each safe page, but its most heavily weighted terms (Section 3.5) include English function words. A classifier trained on such data can learn to separate sources and languages in addition to, or instead of, topics, a phenomenon known as shortcut learning, in which a model succeeds on a benchmark by a route that does not transfer [6]. Annotation artifacts of this kind are documented in language-inference datasets, where the hypothesis text alone allows above-chance prediction [7].

2.3. Preprocessing and features

Texts were lower-cased, URLs and special characters removed, spaces normalized, and Indonesian stop words removed with the Sastrawi library; digits, very short words, and the excess length of the body text were dropped. Features are TF-IDF weights, in which a term t in document d receives the product of its frequency and an inverse document frequency that down-weights terms occurring in many documents [8]. In the scikit-learn implementation [9], with n documents and document frequency df(t), the weight is

$$w_{t,d} = \mathrm{tf}_{t,d} \cdot \left(\ln \frac{1+n}{1+\mathrm{df}(t)} + 1\right), \qquad \mathbf{x}_d = \frac{\mathbf{w}_d}{\|\mathbf{w}_d\|_2} \tag{1}$$

so each document becomes a unit-length vector. The vocabulary was limited to 120 terms, drawn from unigrams, bigrams, and trigrams, keeping terms that occur in at least 12 documents and in at most 60% of them. A vocabulary of 120 terms is very small: the classifier sees only the 120 most frequent surviving terms, so it behaves much like a keyword matcher, which suits the vocabulary of gambling pages but also makes it sensitive to pages that avoid those terms.

2.4. Classifier and decision rule

Naive Bayes assumes that, given the class, the features are independent. With class prior P(c) and per-class feature weights θ_{cj}, the predicted class of a document x is

$$\hat{c} = \operatorname{argmax}_c \left[\log P(c) + \sum_{j=1}^{V} x_j \log \theta_{cj} \right], \qquad \theta_{cj} = \frac{N_{cj} + \alpha}{N_c + \alpha V} \tag{2}$$

where N_{cj} is the total weight of feature j in documents of class c, N_c the total weight of class c, V = 120 the vocabulary size, and α the smoothing constant [9]. Smoothing prevents a feature unseen in a class from giving a zero probability. Because each document vector has unit length, the class-wise feature totals N_{cj} may be small relative to α = 10, in which case the smoothing flattens the differences between classes; this was not examined, and it is one reason a search over α is among the checks in Section 3.11. The posterior probabilities are obtained by normalizing the scores, and the service reports the highest as the confidence.

The decision rule of the service is to take the class of highest posterior and to refuse the link if it is not safe. The costs of the two kinds of error are, however, not equal. Writing c_{FA} for the cost of letting a harmful page through and c_{FB} for the cost of blocking a safe page, a page should be blocked when the posterior of the harmful classes, $P_h = 1 - P(\text{safe} \mid x)$, satisfies

$$P_h \cdot c_{FA} > (1 - P_h) \cdot c_{FB} \quad \Leftrightarrow \quad P_h > \tau = \frac{c_{FB}}{c_{FA} + c_{FB}} \tag{3}$$

The highest-posterior rule corresponds to equal costs (τ = 0.5 when the safe class is compared with the two harmful classes combined) and is not tuned to the service's objective. If a missed harmful page costs five times as much as a false block, for example, the threshold falls to τ = 1/6 ≈ 0.17.

2.5. Augmentation and validation

The collection was split into 70% training and 30% test data with stratification before augmentation, so that augmented variants of a test document cannot enter the training set. Two augmentations were applied to the training data only. Cross-contamination inserts fragments of text from another category into a document, and noise injection randomly deletes or shuffles words, modifies word forms, truncates the text, and inserts ambiguous words typical of other classes. Random deletion, swapping, and similar edits are known as easy data augmentation for text classification [10]. Because the test set is not augmented, the training distribution is deliberately harder and noisier than the test distribution, which matters for the interpretation of cross-validation in Section 3.4.

Five-fold stratified cross-validation was run on the training data to judge the stability of the model. How the augmented documents were assigned to folds is not described; if variants of one original document fall into different folds, information leaks between folds and the estimate is optimistic, and if scoring includes augmented documents, the estimate describes a harder distribution than the test set. Selecting the model and evaluating it on the same data would bias the estimate [11], [12], and the parameters here were fixed in advance and not tuned on the test set. We set the TF-IDF vocabulary size and the Naive Bayes smoothing constant before running the five-fold split and left them unchanged afterward, so no parameter choice in this study used the test results.

2.6. Evaluation

Accuracy, and the precision, recall, and F1-score of each class, are reported with the macro average. Cohen's kappa and the multiclass Matthews correlation coefficient (MCC) summarize the confusion matrix with correction for chance agreement. Proportions carry Wilson 95% intervals, which are accurate for proportions that are near 0 or 1 or based on a few hundred cases [13]. Because the service is a filter, the three classes are also collapsed into harmful (gambling or pornography) and safe: the miss rate is the share of harmful pages predicted safe, and the false-block rate is the share of safe pages predicted harmful. The share of blocks that are correct (precision of the harmful decision) depends on the prevalence π of harmful pages among submitted links through Bayes' rule,

$$\mathrm{PPV}(\pi) = \frac{\mathrm{Se}\cdot\pi}{\mathrm{Se}\cdot\pi + \mathrm{FPR}\cdot(1-\pi)} \tag{4}$$

where Se is the sensitivity to harmful pages and FPR the false-block rate. All metrics were recomputed from the published confusion matrix (Fig. 2), which holds 644 test pages; the source states that 30% of 2,167 is about 651 pages, and the difference of seven pages matches a cleaning step the source mentions in passing: seven duplicate or empty-body pages were dropped from the test split after the 30% draw, leaving 644. Per-page predictions were not saved, so paired tests against other models are not possible; the checks in Section 3.11 would add them [14], [15].

3. Results and Discussion

3.1. Overall test performance

On the test set the classifier made 603 correct predictions out of 644, an accuracy of 93.63% with a 95% interval of 91.5–95.3%. The macro-averaged precision, recall, and F1-score are 0.941, 0.937, and 0.9375, Cohen's kappa is 0.904, and the multiclass MCC is 0.906 (RQ1). The interval is about four points wide, so a reader should treat the accuracy as somewhere between 91.5% and 95.3% and not as 93.63% exactly. Table 2 gives the per-class results with intervals, and Fig. 2 shows the confusion matrix.

Table 2. Test results per class (n = 644).

Class	Support	Precision (95% CI)	Recall (95% CI)	F1
Safe	221	0.874 (0.826–0.909)	0.968 (0.936–0.985)	0.919
Gambling	220	0.970 (0.936–0.986)	0.877 (0.827–0.914)	0.921
Pornography	203	0.980 (0.950–0.992)	0.966 (0.931–0.983)	0.973
Macro average	644	0.941	0.937	0.938

Recomputed from the confusion matrix in Fig. 2; the values match the thesis table.

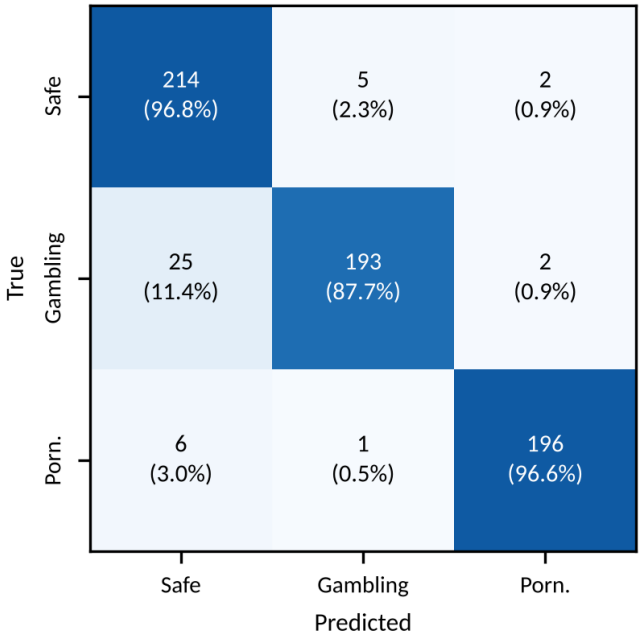


Figure 2. Confusion matrix on the test set (counts and row percentages).

Pornography is recognized best (F1-score 0.973). The safe class has high recall (0.968) but low precision (0.874), and online gambling shows the reverse pattern (precision 0.970, recall 0.877). This is the signature of one dominant error: gambling pages predicted as safe (25 of 220, 11.4%; CI 7.8–16.2%). Together with 6 pornography pages predicted safe, 31 harmful pages were taken for safe, and they make up 12.7% (CI 9.1–17.4%) of all 245 pages predicted safe in this test set.

3.2. The classifier as a filter

For a shortener, the consequential errors are those that cross the safe/harmful line (Table 3, RQ2). Of 423 harmful pages, 31 were predicted safe, a miss rate of 7.3% (CI 5.2–10.2%); of 221 safe pages, 7 were predicted harmful, a false-block rate of 3.2% (CI 1.5–6.4%). Errors between the two harmful classes (3 pages, for example gambling predicted as pornography) do not change the decision to block. In other words, roughly one harmful link in fourteen would obtain a short link, and about one safe link in thirty-two would be refused. Whether this is acceptable depends on costs that the source does not state; for a service whose purpose is prevention, a miss rate of 7.3% is considerable, and the threshold analysis of Eq. (3) shows a way to lower it at the price of more false blocks.

Table 3. Filter view: harmful versus safe (n = 644).

Quantity	Count	Rate (95% CI)
Harmful pages passed as safe (miss)	31 / 423	7.3% (5.2–10.2)
of which gambling	25 / 220	11.4% (7.8–16.2)
of which pornography	6 / 203	3.0% (1.4–6.3)
Safe pages blocked (false block)	7 / 221	3.2% (1.5–6.4)
Sensitivity to harmful pages	392 / 423	92.7% (89.8–94.8)
Harmful pages among predicted safe	31 / 245	12.7% (9.1–17.4)

Computed from the confusion matrix in Fig. 2.

The thesis attributes the gambling-to-safe errors to pages that show only a login or registration screen without the typical gambling vocabulary, such as slot, gacor, jackpot, or casino, and to generic words like "login", "akses", and "link" that appear in both classes. This explanation is plausible given the 120-term vocabulary: a page without any of the retained gambling terms leaves the model with only the prior and generic terms. It was not tested, for example by inspecting the 25 misclassified pages or by counting how many contain none of the top terms. Such pages are also those that matter most in practice, since operators who wish to avoid filters have a reason to publish pages with little text.

3.3. Error structure and class balance

The classes are nearly balanced in the dataset (Table 1), and the test set is balanced too (221, 220, and 203 pages), which makes accuracy more meaningful than in imbalanced problems but does not represent traffic to a shortener, where harmful destinations are probably a small minority of submitted links. Accuracy on a balanced collection is therefore a measure of separability and not an estimate of performance in use. The share of blocks that are correct in the test set, 98.2%, rests on a harmful prevalence of 65.7%; Section 3.9 shows how it falls at realistic prevalence.

3.4. Cross-validation versus test accuracy

Five-fold cross-validation on the training data gave accuracies of 0.837, 0.877, 0.813, 0.827, and 0.813, with a mean of 0.8335 and a standard deviation of 0.0235 (Fig. 3; the standard deviation is the population value, and the sample value is 0.0263). The test accuracy of 0.9363 is 0.103 higher than the cross-validation mean, about 4.4 standard deviations. The thesis calls the difference "about 10%" and "acceptable", and describes the test set as having characteristics "more suited to the patterns learned", and its abstract describes performance as stable (RQ3). A gap of this size is not what chance variation between folds would produce, and a test set that is more suited to the learned patterns is a statement that needs a reason. The gap does not show that the model is unstable, but it does show that cross-validation and test measure different things.

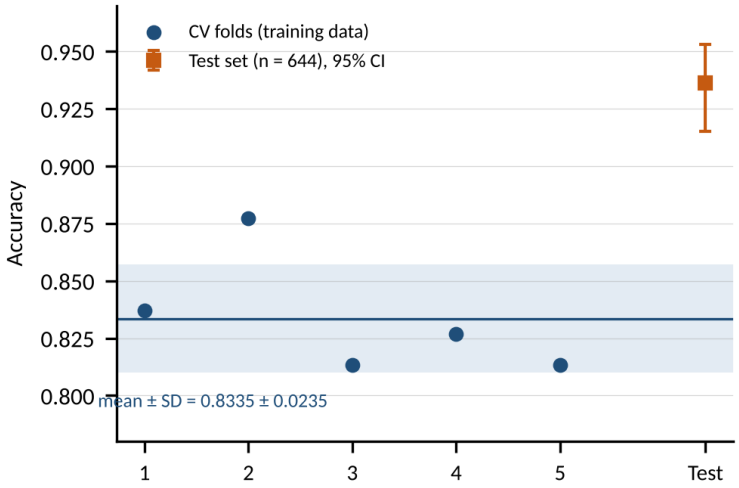


Figure 3. Cross-validation folds and test accuracy with its 95% interval.

Several explanations are consistent with the figures. (a) Augmentation was applied before cross-validation, so the folds contain noisy documents that are harder than the original test pages; cross-validation then measures robustness to noise and the test measures accuracy on clean pages. (b) The test pages are easier because of how the split was made, for instance because pages of the same site or near-duplicate synthetic texts fall on both sides. (c) The test set contains synthetic template texts, which are easy to classify. The source supports none of the three over the others; they can be told apart by scoring cross-validation on un-augmented documents only, by splitting on site, and by reporting accuracy separately for real and synthetic pages. Reporting only the standard deviation across folds also understates uncertainty, because the folds share training data and no unbiased estimator of the variance exists [16].

3.5. Terms that drive the decision

Table 4 lists the five highest-weighted terms of each class. The gambling terms (vip, slot, maxwin, terpercaya, gacor) are exactly the vocabulary of gambling promotion and show that the classifier has learned meaningful cues. The pornography terms (mature, asli, hd, viral, full) describe media offers and are less specific. The safe-class list, in contrast, contains two English function words, "for" and "and", at the top, and the others (tools, tutorial, edukasi) are generic of educational pages.

Table 4. Highest TF-IDF weights per class (from the thesis).

Rank	Safe	Gambling	Pornography
1	for (0.138)	vip (0.198)	mature (0.153)
2	and (0.128)	slot (0.194)	asli (0.147)
3	tools (0.113)	maxwin (0.189)	hd (0.140)
4	tutorial (0.111)	terpercaya (0.180)	viral (0.135)
5	edukasi (0.109)	gacor (0.179)	full (0.134)

English function words are not removed by an Indonesian stop-word list, and they occur in English pages and are rare in Indonesian ones. That two of the five strongest cues of the safe class are such words suggests that part of the separation of safe from harmful pages is a separation of English from Indonesian text, a property of the sources and not of the content. This has two consequences. First, accuracy measured on pages drawn from the same sources overstates accuracy on safe Indonesian pages, such as a local news site, a shop, or a school page, which are what a shortener in Indonesia would mostly see. Second, an operator could pass a gambling page as safe by adding English text. The thesis lists the English-language safe sites among its data sources but does not discuss this, and the ranking by weight is not the same as the model's decision weights, so the interpretation is a hypothesis and not a finding. A direct test is to translate or replace the safe class with Indonesian pages from comparable domains and to retrain.

3.6. How much could synthetic data inflate the score?

The share of synthetic texts in the test set is unknown, but its possible effect can be bounded. If a fraction s of the test pages were template-generated and all were classified correctly, the accuracy on the remaining real pages would be (0.9363 − s)/(1 − s) (Table 5). With a tenth of the test pages synthetic, the real-page accuracy would be 92.9%; with a quarter, 91.5%; with half, 87.3%. The bound assumes the worst case of perfect scores on synthetic pages, so it is an upper limit on the inflation and not an estimate. It shows why the proportion must be reported: the headline accuracy is compatible with a real-page accuracy several points lower.

Table 5. Accuracy on real pages if a share of the test pages were synthetic and all correct.

Synthetic share of test set	0%	10%	25%	50%
Accuracy on real pages (lower bound)	93.6%	92.9%	91.5%	87.3%

Computed as (0.9363 − s)/(1 − s); assumes perfect accuracy on synthetic pages.

3.7. What each result supports

The accuracy and the per-class figures support the feasibility of separating the three classes in this collection with a very small vocabulary. They do not support a claim of generalization to real shortened links, because the evidence for that claim, a test on pages from sources outside the collection, is absent. The same holds for the stability claim: the cross-validation range of 0.81–0.88 is wide, and the gap to the test accuracy is unexplained.

3.8. Probabilities and calibration

The interface shows a confidence score, and the thesis illustrates the mechanism with a page whose posterior probabilities were 0.08 for safe, 0.89 for gambling, and 0.03 for pornography. Posteriors from Naive Bayes are known to be poorly calibrated, because the independence assumption makes the model overconfident when features are correlated, and the scores tend to be pushed toward 0 and 1 [17]. Modern classifiers also often show a gap between confidence and accuracy [18]. A confidence of 0.89 should therefore not be read as an 89% chance that the page is gambling. Before the confidence is displayed to users or used for the threshold of Eq. (3), a reliability diagram on held-out pages should be drawn, and a calibration step such as isotonic or logistic recalibration should be considered.

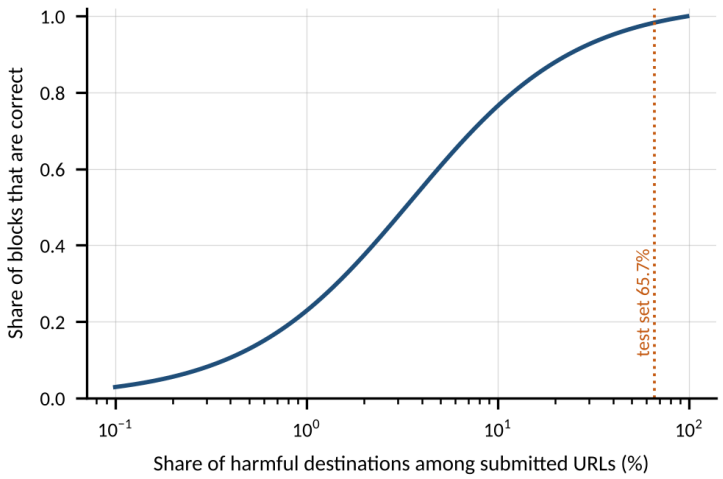


Figure 4. Share of correct blocks as a function of the prevalence of harmful pages.

3.9. Prevalence and the share of correct blocks

Eq. (4) with a sensitivity of 0.927 and a false-block rate of 0.032 gives the share of blocks that are correct at different prevalence (Fig. 4). At the test prevalence of 65.7% it is 98.2%, but if 10% of submitted links were harmful it would be 76.5%, at 2% it would be 37.4%, and at 0.5% only 12.8%. At low prevalence most blocks would be refusals of legitimate links, and the service would need an appeal path or a human

review of blocked links. The prevalence of harmful destinations among links submitted to a particular shortener is not known and could be measured by reviewing a random sample of submissions.

3.10. System testing and what was not measured

The system was tested functionally: nine features were exercised and all succeeded. These tests show that the pipeline works, not how well. The source shows two blocked examples (Fig. 5) but reports no systematic negative tests, with counts of known harmful pages that were refused, and no measurement of time. The claims of real-time operation and fast computation are therefore not supported by data. A fetch of an external page dominates the time of such a service, and the page may be slow, large, blocked, or dynamic; the thesis acknowledges that sites with anti-scraping protection or incomplete HTML give limited metadata, which degrades classification. End-to-end latency, the share of pages whose fetch fails, and the decision taken in that case (allow or refuse) are essential design parameters. The source reports none of the three: no latency figure, no fetch-failure rate, and no stated default for a failed fetch.

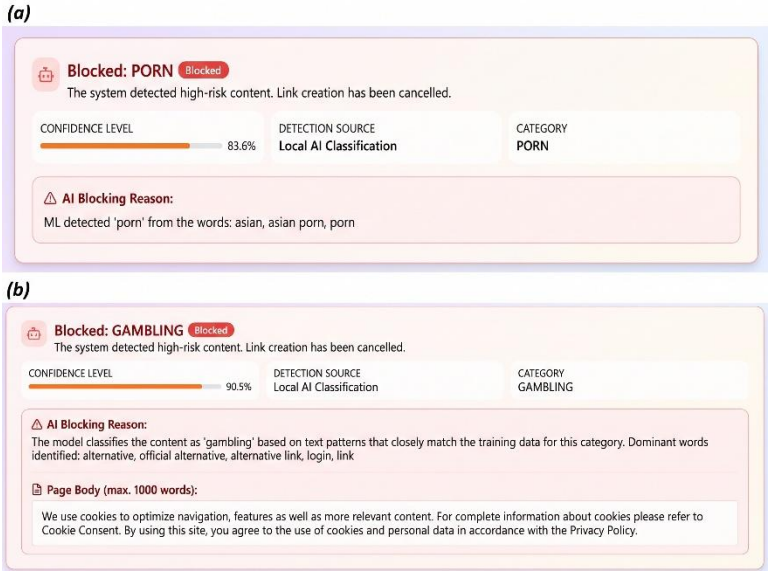


Figure 5. Blocking screens of the service (interface text in Indonesian): (a) a page blocked as pornography with confidence 83.6%, source "local classification", and the terms that triggered the decision; (b) a page blocked as gambling with confidence 90.5%, the dominant terms, and the body text that was fetched.

The two screens in Fig. 5 show what a user sees when a link is refused: the category, a confidence percentage, the detection source, and the terms that drove the decision together with the fetched page text. This lets a refused user see why and a reviewer check the evidence. In the gambling example the dominant terms are "alternatif", "link alternatif", "login", and "link", and the fetched body text is a cookie-consent notice with no gambling vocabulary, which agrees with the explanation of Section 3.2 that gambling pages with little text are the weak point. These are two cases chosen for illustration, not a sample. They suggest a design opportunity: the displayed terms could be logged, and blocks that rest only on generic terms could be sent to review (Section 3.12).

3.11. Threats to validity, checks, and baselines

Table 6 lists the main threats and a check for each. The source confound and the synthetic share could bias accuracy upward by an unknown amount, the cross-validation gap is unexplained, and the absence of baselines leaves the contribution of the classifier undetermined. A very small vocabulary and strong smoothing make the model close to a keyword matcher, so a plain keyword rule is the baseline that matters most; if the rule performs as well, the machine-learning component adds little. Support vector machines are a standard strong baseline for text [5], and Indonesian pretrained encoders [19]–[21] are the natural upper reference, although they are heavier.

Table 6. Threats to validity and checks.

Threat	Check
Safe class from English-language international sites	Add Indonesian safe pages of comparable domains; retrain and test
Unknown synthetic share, possibly in the test set	Report accuracy on real pages only; remove synthetic texts from test
Pages of one site on both sides of the split	Group split by domain; deduplicate
Unexplained CV–test gap	CV on un-augmented documents; augmentation inside folds [22]
No baseline	Keyword rule, logistic regression, linear SVM, Complement NB, IndoBERT
$\alpha = 10$ and 120 features untuned	Search over α, vocabulary size, n-gram range [23]
Probabilities uncalibrated	Reliability diagram; recalibration
No test on real shortened links	Review a random sample of live submissions
Obfuscation, cloaking, dynamic pages	Adversarial tests; fetch-failure policy

Baselines (a keyword rule, logistic regression, a linear SVM, Complement Naive Bayes with tuned α, and a fine-tuned Indonesian encoder) would turn the study into a comparison. With the present test size, a paired comparison on the same 644 pages by the exact McNemar test [14], [15] could detect a difference only if the models disagree on enough pages; for a net difference of 20 pages the exact test is significant at the 5% level only if the models disagree on about 90 pages or fewer, and the sample size needed for a paired comparison depends on that proportion of discordant pages [24]. Per-page predictions should therefore be stored and the test set enlarged. A larger test set also narrows intervals: to estimate a miss rate near 7% with a half-width of one point needs about 2,500 harmful test pages by the usual normal approximation.

3.12. Deployment design

A classifier is one component of a decision procedure, and the procedure must say what happens in every case, including those the classifier does not handle well. The simplest policy, refusing links whose predicted class is harmful, ignores three situations: pages that could not be fetched, pages whose text is too short to classify, and pages with an uncertain posterior. A design with a review band uses two thresholds instead of one: pages with a high harmful posterior are blocked, pages with a low one are allowed, and pages in between are put in a queue for human review or held with a notice. The width of the band is set from the review capacity and from the cost ratio of Eq. (3), and the proportion of pages that fall in it can be read from the held-out posteriors once they are calibrated (Section 3.8).

One further case is a property of shorteners that content-aware screening at creation time cannot address: a link may be created while the destination is harmless and later redirected to harmful content. Measurements of short-URL services found such links among the abuse they observed [1]. A periodic re-scan of stored links, or a scan at redirect time with caching, would close the gap at a computational cost that would also need to be measured. Audit sampling of links that were allowed is the only way to estimate the miss rate in operation, because the test-set rate of 7.3% was obtained on a balanced collection and cannot be transferred to live traffic.

3.13. Ethics, policy, and misuse

Three ethical points deserve attention. First, the dataset contains adult and gambling content collected by scraping, so the authors should document the collection, the handling of explicit material, and compliance with the terms of the sites and platforms. Second, blocking is a policy decision that affects users: a false block of a legitimate link has a cost, and the service needs a transparent reason, an appeal path, and a retention policy for submitted URLs, which may reveal what users intend to share. Third, a published filter invites evasion: operators can change vocabulary, hide text in images, or serve different content to crawlers and to people, which is called cloaking, so the classifier should be seen as one layer among several, together with blacklists, user reports, and the address-based methods described in the literature [1]–[3].

4. Conclusion

A shortener that reads the destination page and classifies it with TF-IDF and Naive Bayes separated safe, gambling, and pornographic pages with 93.63% accuracy (95% CI 91.5–95.3%) on 644 test pages, which shows that content-aware screening is feasible with a very light model. As a filter, however, it let 7.3% of harmful pages through, mainly gambling pages without typical vocabulary, and blocked 3.2% of safe pages. Cross-validation on the training data gave 0.8335, about four standard deviations below the test accuracy, and the source does not explain the gap. Because the safe class comes from English-language international sites, and an unknown share of the data is synthetic, the scores may reflect source and language as much as content. No baseline, latency, or real-link test was performed, and the confidence shown to users is uncalibrated. The next steps are Indonesian safe pages from comparable domains, a split by site, a keyword and SVM baseline, a cost-based threshold with calibrated probabilities, and a measurement of latency and false blocks on live submissions. We do not compare accuracy with other published systems, because their datasets, classes, and protocols differ and the present collection is not public.

AI Disclosure Statement

The authors used Generative AI and AI-Assisted Technologies throughout the preparation of this manuscript: to draft and revise portions of the text, to write and run the scripts behind the experiments and statistical analyses, to generate the figures included in the revision, and to format the document from the authors' own research data and source code. The authors verified the scripts and their outputs, reviewed the resulting text, cross-checked every reference against its original source, and accept full responsibility for the content of this publication.

Author Contributions Statement

Zulfikar Alip Takasihaeng: Conceptualization, Methodology, Software, Investigation, Formal Analysis, Data Curation, Visualization, Writing – Original Draft.
Nouval Trezandy Lapatta: Supervision, Validation, Writing – Review and Editing.
Sri Khaerawati Nur: Resources, Project Administration, Writing – Review and Editing.

References

[1] F. Maggi, A. Frossi, S. Zanero, G. Stringhini, B. Stone-Gross, C. Kruegel, and G. Vigna, "Two years of short URLs internet measurement: Security threats and countermeasures," in *Proc. 22nd Int. Conf. World Wide Web (WWW)*, Rio de Janeiro, Brazil, 2013, pp. 861–872.

[2] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: Learning to detect malicious web sites from suspicious URLs," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, Paris, France, 2009, pp. 1245–1254, doi: 10.1145/1557019.1557153.

[3] D. Sahoo, C. Liu, and S. C. H. Hoi, "Malicious URL detection using machine learning: A survey," arXiv:1701.07179, 2017.

[4] F. Sebastiani, "Machine learning in automated text categorization," *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1–47, Mar. 2002, doi: 10.1145/505282.505283.

[5] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Machine Learning: ECML-98*, Lecture Notes in Computer Science. Berlin, Germany: Springer, 1998, pp. 137–142, doi: 10.1007/BFb0026683.

[6] R. Geirhos et al., "Shortcut learning in deep neural networks," *Nat. Mach. Intell.*, vol. 2, no. 11, pp. 665–673, Nov. 2020, doi: 10.1038/s42256-020-00257-z.

[7] S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. Bowman, and N. A. Smith, "Annotation artifacts in natural language inference data," in *Proc. 2018 Conf. North Amer. Chapter Assoc. Comput. Linguist.: Human Lang. Technol.*, vol. 2, 2018, pp. 107–112, doi: 10.18653/v1/N18-2017.

[8] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manag.*, vol. 24, no. 5, pp. 513–523, 1988, doi: 10.1016/0306-4573(88)90021-0.

[9] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[10] J. Wei and K. Zou, "EDA: Easy data augmentation techniques for boosting performance on text classification tasks," in *Proc. 2019 Conf. Empirical Methods Nat. Lang. Process. 9th Int. Joint Conf. Nat. Lang. Process. (EMNLP-IJCNLP)*, 2019, arXiv:1901.11196.

[11] G. C. Cawley and N. L. C. Talbot, "On over-fitting in model selection and subsequent selection bias in performance evaluation," *J. Mach. Learn. Res.*, vol. 11, pp. 2079–2107, 2010.

[12] S. Varma and R. Simon, "Bias in error estimation when using cross-validation for model selection," *BMC Bioinformatics*, vol. 7, Art. no. 91, 2006, doi: 10.1186/1471-2105-7-91.

[13] L. D. Brown, T. T. Cai, and A. DasGupta, "Interval estimation for a binomial proportion," *Stat. Sci.*, vol. 16, no. 2, pp. 101–133, May 2001, doi: 10.1214/ss/1009213286.

[14] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, Jun. 1947, doi: 10.1007/BF02295996.

[15] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Comput.*, vol. 10, no. 7, pp. 1895–1923, Oct. 1998, doi: 10.1162/089976698300017197.

[16] Y. Bengio and Y. Grandvalet, "No unbiased estimator of the variance of K-fold cross-validation," *J. Mach. Learn. Res.*, vol. 5, pp. 1089–1105, 2004.

[17] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proc. 22nd Int. Conf. Mach. Learn. (ICML)*, 2005, pp. 625–632, doi: 10.1145/1102351.1102430.

[18] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, 2017, arXiv:1706.04599.

[19] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A benchmark dataset and pre-trained language model for Indonesian NLP," in *Proc. 28th Int. Conf. Comput. Linguist. (COLING)*, 2020, pp. 757–770, doi: 10.18653/v1/2020.coling-main.66.

[20] B. Wilie et al., "IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding," in *Proc. 1st Conf. Asia-Pacific Chapter Assoc. Comput. Linguist. and 10th Int. Joint Conf. Natural Lang. Process.*, 2020, pp. 843–857, doi: 10.18653/v1/2020.aacl-main.85.

[21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguist.: Human Lang. Technol. (NAACL-HLT)*, vol. 1, Minneapolis, MN, USA, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.

[22] S. Kaufman, S. Rosset, C. Perlich, and O. Stitelman, "Leakage in data mining: Formulation, detection, and avoidance," *ACM Trans. Knowl. Discov. Data*, vol. 6, no. 4, pp. 1–21, Dec. 2012, doi: 10.1145/2382577.2382579.

[23] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.

[24] R. J. Connor, "Sample size for testing differences in proportions for the paired-sample design," *Biometrics*, vol. 43, no. 1, p. 207, Mar. 1987, doi: 10.2307/2531961.