Journal of Al-Turath University College (JATUC)
Open Access Journal
ISSN: 2074-5621
DOI: https://doi.org/10.63964/JATUC.43.2.2026.44
https://jatuc.uoturath.edu.iq/

A Multi-Vector Framework for Localized Phishing Detection URLs: Integrating Telegram-Sourced Intelligence and Iraqi Contextual Features

Jaber M. Al-Dulimi

Department of Mathematics, College of Basic Education, University of Diyala, Baqubah, Diyala, Iraq.

*Correspondence email: Basicmath12te@uodiyala.edu.iq

KEYWORDS

Phishing Detection; Telegram Security; Machine Learning; Iraqi Cybersecurity; Social Engineering.

ABSTRACT

Background: Phishing attacks grow more complex - attackers employ social engineering that targets local customs, and they build short-lived systems so that they remain hidden. This paper proposes a method that identifies phishing inside Iraq's corner of the internet. The method merges word patterns, domain records, open source intelligence plus language markers that occur in Iraq.
Materials and Methods: We assembled 18 060 URLs - gathering addresses from local Telegram channels, from worldwide threat feeds and from confirmed safe sites. Statistical inspection revealed strong differences between phishing and safe URLs ($p < 0.001$). The main differences appeared in URL length, in the count of special characters and in the presence of Iraqi terms. System attributes also differed - phishing links often used young domains, unusual suffixes, as well as servers located outside Iraq. Those attributes signal campaigns that aim at Iraqi users.
Results: A Random Forest model with 200 trees and a maximum depth of 20 was trained on a stratified 70 % train, 30 % test split under 10-fold cross-validation. The model reached 96.84 % accuracy, 97.18 % recall or an ROC-AUC of 0.984. An ablation test showed that the inclusion of Iraq-specific features raised recall by 2.68 % and accuracy by 1.64 %. This confirms that regional language data adds value.
Discussion: A review of feature importance ranked Iraqi keywords next to domain age as the two most informative signals. The outcome indicates that phishing detection improves when models incorporate attributes that match the target region. The benefit is largest in emerging digital markets where attackers combine local social engineering with rapid infrastructure changes.

1. INTRODUCTION

Scam spreaders are changing methods and targeting instant messengers instead of e-mails, for open rates from some private communication mediums — Egg Telegram (considered to be safe) has a huge number of users. Phishing is one of the causes of data breaches that leads to ransomware and money scams [1] [2].

According to the report, in Iraq, transactions through modern electronic platforms like the Ur electronic portal and e-payment system — which are part of the digital services package being implemented by the Iraqi government — have simplified attackers' lives. Specifically, phishing attacks against individuals in Iraq now employ localization via social engineering [3] [4]. They also employ local language and pose as Iraqi groups to avoid standard security protocols. Standard methods to discover these terrorist attacks do not always apply when they rely on global listings of malicious websites or overall language structures, also suggesting the detection may miss attacks specifically crafted to play mind tricks in a particular region, although technically they do not seem dangerous [5] [6].

There is a shift in the delivery of phishing attacks. Attackers are turning to instant messaging platforms such as Telegram to communicate with compromised systems, as many users have installed IM solutions and believe

their communication channels are safe. Phishing is one of the major causes that lead to data breaches, and subsequently ransomware and money scams [7].

In Iraq, the aforementioned examples, like the Ur electronic portal and e-payment systems that are part of the country's shift to digital services, have facilitated attackers. Ransomware, unfortunately, we have some sad news that some groups of criminals are trying to gather money by attacking Iraqis, and this type of attack is called ransomware, and it takes a lot of forms. Phishing attacks targeting individuals in Iraq now use local social engineering, use local language, pretending to be Iraqi organizations, to bypass the usual security measures [3] [8]. All the usual methods of detecting these attacks don't always work, because the detection is based either on global lists of bad websites or normal forms of languages. That means they can overlook attacks that seek to deceive people in a particular area, even when the attacks make little sense in technical terms [5] [9].

2. LITERATURE REVIEW:

These days, phishing attacks constitute a massive cybersecurity problem. They are responsible for a large number of data breaches (over 90%), and also for how most cyber-attacks start [10]. These attacks have been made more accessible by social media and messaging apps such as Telegram, where legacy security techniques have failed to transfer.

2.1. Technical Detection Approaches

Methods: We Generalize Machine Learning and Deep Learning
There has been a shift away from high-level reasoning towards data-driven models [11] [12].
A majority of the existing detection techniques rely on supervised learning algorithms. Ensemble methods, including Random Forests, Gradient Boosting, and XGBoost, provide the best performance with good generalisation capabilities; thus, when carefully trained and tuned, these refined ensemble techniques achieved up to 98.27% by using only the offered datasets [12]. When using the gradient boosting approach using URL features, it displays a precision rate of 96%, a recall rate of 98%, and an F1 score of 97%, demonstrating that they are capable of differentiating between authentic and spoofed websites while also being able to adapt to novel forms of attacks. [8].

To this end, the introduced study applied Ensemble Learning with the use of Random Forest and XGBoost techniques to achieve high accuracy based upon prior results, so that these two methods can later also be used as a valid form of deployment across the Digital Space of Iraq. Research studies conducted earlier provided high levels of verification for these two ML methods, and this will also now be tested, specifically within the context of the Iraqi Digital Environment.

2.2. Feature Engineering and Fusion Strategies

The modelling of fake news detection relies a lot on feature engineering labelling data with features can affect accuracy more than any classification method. An extensive range of features has also recently been proposed, which facilitate the detection of phishing URLs, such as URL-based on words and structural attributes, domain properties, content vs. visual comparative features, hyperlink structure and user behaviour [14] [15].

Word and structure-based URL features are the most used because they are easy to process and do not require any external information. Such examples are URL length, types of characters used in the URL, existence of suspicious words or special characters, number of subdomains, and path depth. [16] [17]. A feature-based approach using the presence of specific words, structure and domain characteristics achieved 93.2% accuracy and F1-score with Light GBM. Some features were found as important keywords, including the length of the URL, the presence of suspicious words [14]. And domain characteristics. Word features can be calculated on the fly and thus suit real-time detection in, e.g. browser add-ons and protection on the user side [9].

Domain-based features help us further by adding more info about the domain: registration information, age, reputation or DNS. However, when we use these features, we have to import data from a different location, and that could lead to slowness in the output. Thus, we must decide between being right and working quickly [18]. Finding fake sites can be aided by content and visual details such as the HTML, JavaScript, forms and how similar a site looks to a real one. However, to check these, we need to load the page and analyse its content; this could be too slow to filter URLs in real time [19].

Previous work focused on words and domains [19] In our case, we extend this approach by applying SSIM to identify visual copies and how that applies in Iraq. And that adds a new dimension to how we think about which features are in the game.

2.3. Performance Metrics and Comparative Analysis

Phishing detection systems are evaluated based on different metrics that express different perspectives of classifier performance. The common metrics include accuracy, precision, recall, F1-score, and AUC; however, their interpretation requires knowledge of features of the dataset, such as class imbalances and penalties for false positives and false negatives [20].

Table 1 compares the detection performance of systems reported in the literature, categorizing them by their principal mechanism.

Table 1: Comparative Performance of Phishing URL Detection Systems

Detection Approach	Primary Features	Accuracy	Precision	Recall	F1-Score	Reference
BERT + CNN	URL text, n-grams	96.66%	_____	_____	_____	[15]
Fine-tuned Ensemble (RF/XGB/GB)	URL, content, external	97.44–98.27%	_____	_____	_____	[14]
Gradient Boost	URL-based	_____	96%	98%	97%	[9]
LB-LSTM + Q-learning	URL, behavioural	94.33%	_____	98.67%	94.34%	[21]
Hyper Fusion (XGBoost)	URL, topology, behaviour	99.37%	_____	_____	_____	[10]
LightGBM Feature-Driven	Lexical, structural, domain	93.2%	_____	_____	93.2%	[19]

There are a couple of key trends between the reported metrics. Systems that make use of fusions of features, integrating data at different levels — such as URLs, topology and behaviour Yield the best Accuracy. For example, Hyper Fusion gets 99.37% [10]. Second, ensemble methods are superior to single-algorithm methods. Tuned Random Forests, Gradient Boosting and XGBoost models get above 95% accuracies [9] [14]. Third, there are a number of methods that extract and leverage textual information through natural language processing (NLP) to build very effective deep learning-based classifiers [15].

2.4. Human factors influencing phishing susceptibility

Social Engineering Tactics on Social Media Platforms

Social Engineering Techniques used on Social Media Platforms

Social media and messaging apps can be attacked in different ways. Social engineering exploits the way these platforms operate and how people use them. By integrating communication and sharing, these platforms have provided fertile ground for dangerous links to spread through the trusted ties of those taking part. This tricks people by using the trust in [3] [12].

In the studies where social engineering victims were interviewed, the attack methods that were being commonly used and the things that managed to get people compromised were documented [13]. A systematic review of cyber-attack methods and what underlie their implementation [13]. Victims reported that they encountered phishing links everywhere, including direct messages in social networks, posts from hacked friends' accounts and instant messaging on platforms like Telegram [13]. Such attacks take advantage of the trust that people place in messages their social acquaintances are sending, even when those acquaintances' accounts have been hijacked [21] [20].

At least some of those patterns have emerged in studies in which researchers induced social engineering victims to describe the attacks as they unfolded, and what led them to fall for them [22]. Victims said they received phishing links in various locations. This includes direct messages via social media, posts from hacked accounts of people known to them and messages on platforms like Telegram [22]. Since these attacks are made through social media, it's more trusted among people because they believe that they're talking to someone from a circle, even though their account has been hacked. Social media attackers deploy a few common tricks. They frequently break into legitimate accounts and send phishing links to the victims' friends from them, exploiting the trust people have in these accounts [23] [21]. They also use messages that create an urgency, impersonating important people or groups, or exploit current news events to fool people [24].

3. RESEARCH GAP

From the perspectives of what systems have been built and how these relate to one another, the literature synthesis develops a picture of a relatively high-level EN-PDS, some being within one or 2% of faultless testing

results on held-out data corpuses [10] [8]. However, this still leaves a very prominent research gap in the two dimensions of:

1. A shallow zoom-in on the Telegram ecosystem, which works under different social engineering principles than (but related to) traditional web-based phishing [22].
2. Failure of Geographical Contextualisation (against the threats targeting the Iraqi digitals cape).

4. RESEARCH OBJECTIVES

This study aims to:
To generate a set of deceptive Iraqi URLs, combine links from Telegram, other global open-source resources, and verified pure examples. Compare deceptive to clean URLs using statistical methods, examining both linguistic and host features. Create linguistic-contextual features based on unique linguistic expressions and impersonation methods pertaining to Iraq. Create and refine a random forest-based clustering algorithm (PCA, AUC performance evaluation). Analyze attribute significance and contextual contribution by using exclusion methods. Test the clustering algorithm's accuracy against baseline classifiers as an evaluation of robustness.

5. MATERIAL AND METHODS

The methodology consists of four major phases:

1. Multi-Vector Data Acquisition
2. Feature Engineering & Iraqi Contextualization
3. Ensemble Machine Learning Framework
4. Model Evaluation and Validation

5.1 Multi-Vector Data Acquisition

To monitor the local threat landscape, phishing URLs were collected from three-man sources:

1- Telegram Crawling
2- Global OSINT Integration
3- Benign Dataset Construction

A. Telegram Crawling

The collection of data was done through a longitudinal study over the period of eight months. We examined 70 Telegram channels covering tech news (20), telecom grants and promos (30) and finance (20). We have a découpage system we built to receive the data. There are two main tools used by this system:

Telethon Library: This allows access to the Telegram API and efficiently retrieves structured data.
Selenium Web Driver: This is like a real human using an actual browser, and it allows us to scrape data that the normal API will not allow.

Using this two-part approach, we gathered 4,318 unique records of data (mostly URLs and metadata). Overall, we received around 540 links per month. It provided us with a complete data set to analyse from different perspectives.

Algorithm 1: Multi-Vector Telegram URL Crawling & Extraction

Input:

- Mathcal C: A set of 70 target public Telegram channels.
- Mathcal T: Observation period (8 months).
- N: Maximum number of messages to retrieve per channel.

Output:

- Mathcal U: A unique set of extracted URLs.

1. Initialize an empty set $Ø \rightarrow U$
2. For each channel $c \in C$ do:
3. Establish connection via Telethon (API-based) or Selenium (Web-based).
4. Fetch messages $Mc = \{m_1, m_2, m_N\}$ published during period T.
5. For each message $mi \in Mc$ do:
6. If mi contains a URL pattern matching *Regex_Pattern,* then:
7. Extract URL u_{ij} from mi.
8. Append uij to U (Ensuring uniqueness).

9. End If
10. End For
11. Return *U*

B. Global OSINT Integration

The Global Open-Sauce Intelligence (OSINT) System was integrated in order to improve the classification accuracy of all 4,318 URLs (See Table 2). The intent is to authenticate the Retrieved Items from reputable Threat Sources and thus provide Global Context for the Telegram Data.

The automated searches performed for URLs and phishing activity in different places by using their API can be referred back to the report, so that there will be a Threat-Vulnerability score. Moreover, as mentioned above, the records of persons found through the telegraphic reporting systems and their likely sources of economic support were analyzed for certain Iraqi-specific keywords that could potentially identify non-combatant public figures involved in combat-type activities.

To raise the classification accuracy of the 4,318 URLs, a Global Open-Source Intelligence (OSINT) system was integrated, Table (2).

- Rafidain
- Zain-iq
- TBI-bank
- mohesr-gov
- .iq domains

Table 2: Data Summary Table

Parameter	Value
Observation Period	8Months
Number of Channels (*C)*	70Channels
Extraction Tools	Telethon (API) & Selenium (Web)
Total Unique Records (*U*)	4,318
Data Format	Structured URLs & Metadata

C. Benign Dataset Construction and Ground Truth Development

Benign dataset construction is one of the most fundamental models at the core of multi-vector acquisition systems, from which trusting baselines can be derived [2, 3]. In the detection process, this part is created to minimize False Positives by having a comparison, such as normal and good digital behaviour.

Missed benign records were taken from well-known domains again (Alexa Top Domains and institutional verified whitelists). Cross-referencing with OSINT surgically vetted and purged the historical record. A balanced and representative dataset was created by adding these benign samples to the 4,318 records from the Telegram source. Also, it helps in training the classification models to differentiate between normal behaviour and possible cyber-attack, which overall increases the detection framework accuracy, as listed in Table (3).

- Alexa Top 1M (6,500 samples)
- .iq registry (3,500 samples)

Table 3: Data Source Distribution

Source	Number of URLs	Description
Telegram Channels	4,318	URLs extracted from 70 Iraqi public channels
Phish Tank (Filtered)	2,104	Iraqi-related phishing entries
URLhaus (Filtered)	1,638	Malware-hosting URLs with Iraqi indicators
Legitimate (.iq Registry + Alexa)	10,000	Verified benign domains

5.2. Feature Engineering & Iraqi Contextualization

Feature engineering was performed to extract discriminative characteristics from URLs. Three categories of features were used:

1. Lexical Features

2. Iraqi Contextual Features
3. Host-Based Domain Features

A. Lexical Feature

Phishing URLs often exhibit structural irregularities compared to benign links. Statistical analysis (Table 4) confirms significant divergence in metrics such as length and character distribution.

Table 4: Lexical Feature Statistics

Feature	Phishing (Mean)	Benign (Mean)
URL Length	78	42
Dot Count	4.3	2.1
Hyphen Count	3.8	0.9
Digit Count	6.4	1.7
Iraqi Keywords Presence	61%	3%

Sensitive Iraqi-specific vocabulary appeared in “61% “of phishing URLs, compared to only “3%” in benign URLs.

Algorithm 2: Lexical Feature Extraction and Iraqi Contextualization

Input: * u: A single extracted URL from the dataset U.
- K_{iq}: A predefined dictionary of sensitive Iraqi keywords (e.g., 'تعيين', 'منحة', 'qi-card').

Output: * $\boldsymbol{F}_{lex}$: A multi-dimensional feature vector representing the URL.

1. Initialize [ ] → $\boldsymbol{F}_{lex}$

2. Structural Analysis:
- L→ Length (u)
- D → CountDigits (u)
- S → CountSpecialChars (u, \ {'.', '-', '@', '_' \})
- Append L, D, S to $\boldsymbol{F}_{lex}$

3. Iraqi Contextual Mapping:
- **For each** keyword k ∈ K_{iq} **do**:
- **If** k is present in u (via string matching or Regex), **then**:
- B_k →1 (Binary Encoding)
- **Else**:
- B_k →0
- **End If**
- Append B_k to $\boldsymbol{F}_{lex}$
- **End For**

4. Normalization:
- Scale numerical values in $\boldsymbol{F}_{lex}$ to a range [0, 1] for model consistency.

5. Return $\boldsymbol{F}_{lex}$

Algorithm (2) illustrates the mechanism of converting the links that are then extracted and converted into feature vectors in a format that is ready for application of any statistical feature-based algorithm. Matrix input of structural information of the link belongs to classical technical indicators of length, number density and other structural information of the link as the starting points. Quantitative civil additions here are made in the third stage (Iraqi Condition Symbolism), where the association is likened to a local lexical repository, Kiq, developed for this exploration. This dictionary includes keywords used as part of some social engineering attacks that have aimed at the Iraqi public. Table (5) shows some of the common words. They use binary encoding to cause these keywords to have more weight, so those links impersonating official bodies, either governmental or through exploiting the local Iraqi issues, would be given higher weight.

Table 5: Top detected phishing hooks

Keyword	Frequency (%)
manah (grant)	27%
tahdeeth (update)	19%
award	14%

rafidain	11%

B. Host-Based & Domain Features

Infrastructure and Host Features: To detect advanced persistent threats (APTs) such as the Zalam malware, the system assesses the hosting infrastructure. Key indicators include:

1. Domain Age: Identifying domains registered less than 30 days before the campaign began.
2. Top-Level Domain Analysis: Detecting the misuse of cheap, top-level domains (such as .xyz and .top) to impersonate legitimate .iq organizations.
3. Geographic Variation: Identifying services intended for Iraq but hosted in high-risk offshore data centres.

C. Ensemble Machine Learning Framework

Since the data has non-linear features (like length of a link vs. number of points), RF segments this across multiple "decision trees" and counters the idea of over-fitting. It's also good at missing data and noise from Telegram crawling.

Random forest (RF) classifier was implemented with:

- Number of trees (N) = 200
- Maximum depth = 20
- Criterion = Gini impurity

Algorithm 3: 10-Fold Cross-Validated Random Forest Classification

Input:

$X \in R^{n \times m}$: Feature matrix of n samples and m features (F_{lex}, F_{host}, $F_{context}$).

$Y \in \{0,1\}^n$: Class labels (Malicious=1, Benign=0).

K: No. of folds (we set to 10)

Output:

- M_{opt}: Optimized ensemble model.
- P_{avg}: Aggregated performance vector (Acc, Prec, Rec, F_1, AUC).

Begin

1. Data Partitioning:

- Partition the global space (X, y) into K disjoint subsets $\{D_1, D_2, \ldots, D_{10}\}$

2. Cross-Validation Loop:

- **For each** i $\in \{1, 2, \ldots, 10\}$ **do**:
- Let D_{test} = Di (Validation Set).

Let $D_{train} = UD_j$ where $j \neq i$ (Training Set).

- Train a **Random Forest Classifier** on $D_{rain.}$
- Evaluate the model on D_{test} .
- Compute and Store metrics vector S_i (Accuracy, F1-score).
- **End For**

3. Statistical Aggregation:

- Compute the final performance vector $P_{avg} = \frac{1}{K} \sum_{i=1}^{k} S_i$.

4. Final Model Training:

• Retrain the random forest model on the entire X dataset to capture all available patterns.

5. Return M_{opt} and $P_{avg.}$

Prediction function:

$\hat{Y}$=mode {h1(x), h2(x), hn(x)}

To ensure the generalizability of the proposed framework and mitigate the risk of over-allocation, a two-layer validation strategy was employed. Initially, the data was split 70/30 to isolate a final test set for unbiased evaluation. Subsequently, ten-fold cross-validation was applied exclusively to the training segment (70%). This nested approach allows for fine-tuning of hyperparameters and assessment of stability during the training phase, while the final performance metrics are derived from the original test set (30%), representing the model's true predictive capability against previously unseen threats.

5.4. Model Evaluation and Validation

The performance was measured using the method shown in Table (6):

- Accuracy
- Precision
- Recall (Sensitivity)
- F1-Score
- ROC-AUC

Table 6: Evaluation Metrics Definitions

Metric	Formula	Interpretation
Accuracy	(TP + TN) / Total	Overall correctness
Precision	TP / (TP + FP)	Avoiding false alarms
Recall	TP / (TP + FN)	Detecting phishing correctly
F1-Score	2 × (Precision × Recall) / (Precision + Recall)	Balanced metric
ROC-AUC	Area under the ROC curve	Discrimination capability

Multiple measures were used to ensure a balanced assessment, particularly due to concerns about category imbalances.

6. RESULTS

All experiments were conducted using a 70/30 stratified training-testing model, with ten-fold cross-validation applied during training. The performance values reported represent the average performance across the validation folds.

6.1 Dataset Distribution and Class Balance

The final dataset consisted of 18,060 URLs, combining localized Telegram extraction, global OSINT intelligence, and verified benign samples, divided in Table (7).

Table 7: Final Dataset Distribution

Class	Number of URLs	Percentage
Phishing	8,060	44.6%
Benign	10,000	55.4%
Total	18,060	100%

The dataset shows a moderate imbalance in the classes. To prevent bias towards the most representative class, a stratified sampling method was applied during both cross-validation and splitting the data into training and test sets to maintain relative representativeness [25].

6.2 Lexical Feature Discriminatory Analysis

The statistical comparison of lexical features (Table 8) revealed substantial structural divergence between phishing and benign URLs. Phishing URLs exhibited significantly greater:

- URL length
- Dot count
- Hyphen count
- Digit count
- Presence of Iraqi-specific keywords

Table 8: Lexical Feature Statistical Comparison

Feature	Phishing (Mean ± SD)	Benign (Mean ± SD)	p-value
URL Length	78 ± 21	42 ± 15	<0.001

Dot Count	4.3 ± 1.2	2.1 ± 0.8	<0.001
Hyphen Count	3.8 ± 1.5	0.9 ± 0.4	<0.001
Digit Count	6.4 ± 2.3	1.7 ± 0.9	<0.001
Iraqi Keyword Presence	61%	3%	<0.001

All lexical features were significantly different ($p < 0.001$). It claims that well-known keywords of the Iraqi context are found in 61% of hosted phishing URLs but only present an 3% of the benign URLs, supporting the contextual feature engineering strategy mentioned in an earlier work. These findings agree with previous global studies that suggest most phishing URLs are lengthy and use structured sophistication to mimic normal domains while concealing deception-level subdomains [26] [27] [28]. A similar finding has been made by another research that also exploits [29] [26]. Datasets such datasets from Phish Tank and APWG, revealing both URL length and special-character frequency as predictive features. However, this current study adds to this literature in two main ways. It proves that structural stretching is not merely a worldwide phish shaping, but constrained in topic social designing. They also confirm that region-specific keyword injection (e.g. I explain it as Iraqi banks/telecom operators/government services) majorly boosts discriminative strength.

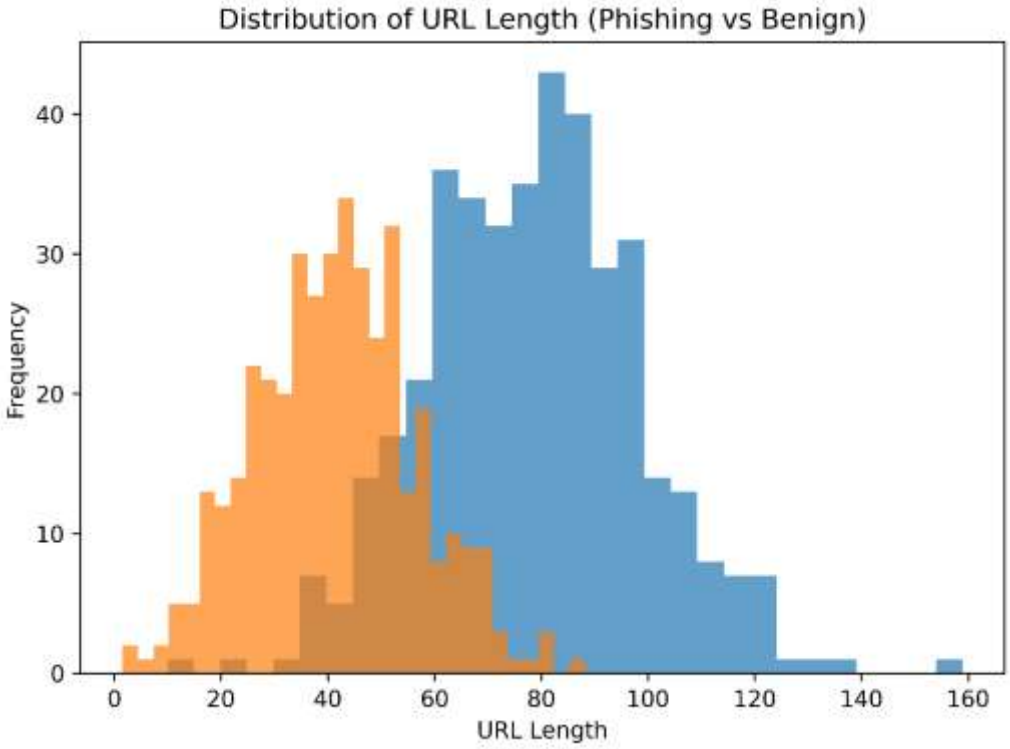

Fig. 1. Distribution of URL Length (Phishing vs. Benign)

- X-axis: URL Length
- Y-axis: Frequency
- Phishing URLs show right-skewed distribution
- Most of the benign URLs are grouped in the range of 30–50 characters

6.3 Host and Domain Characteristics Analysis

The most important findings relate to domain lifespan (Table 9 and Figure 2). Legitimate domains were used for an average of '1420 days', or '3.8 years', while the average lifespan of phishing domains was about ' 48 days.

Table 9: Domain & Infrastructure Characteristics

Feature	Phishing	Benign
Avg. Domain Age (days)	48	1,420
Domains <30 Days Old	37%	2%
Suspicious TLD Usage (.xyz, .top)	41%	4%
Foreign Hosting (Non-Iraq Region)	73%	28%
OSINT Blacklist Flagged	58%	1.2%

37% of phishing domains were under 30 days.

This result provides a solid confirmation of the general belief around the world that newly registered domains are frequently used for phishing campaigns to evade detection better by reputation-based systems [26] [29]. Q4 results report. Instead, here, low lifespan or short life time is counted among one among the main brands of malicious feature both the international threat intelligence studies we build up from the incident response corporations like ICANN & industry reports since last September. And yet, the exceptional thing is how much this Iraq-focused data has revealed to us the enormous magnitude [30].
Additionally:

Suspicious TLD usage (. xyz Phishing 41% (top) Benign 4% Out of the phishing domains, 73% were found in countries other than Iraq. Finally, these results indicate that, after all, the greater part of the phishing threats infecting the Iraqis users mainly rely on foreign hosts and non-descript, cheap TLDs, confirming our assertion that infrastructure-level needling is all that underpins localized attacks.

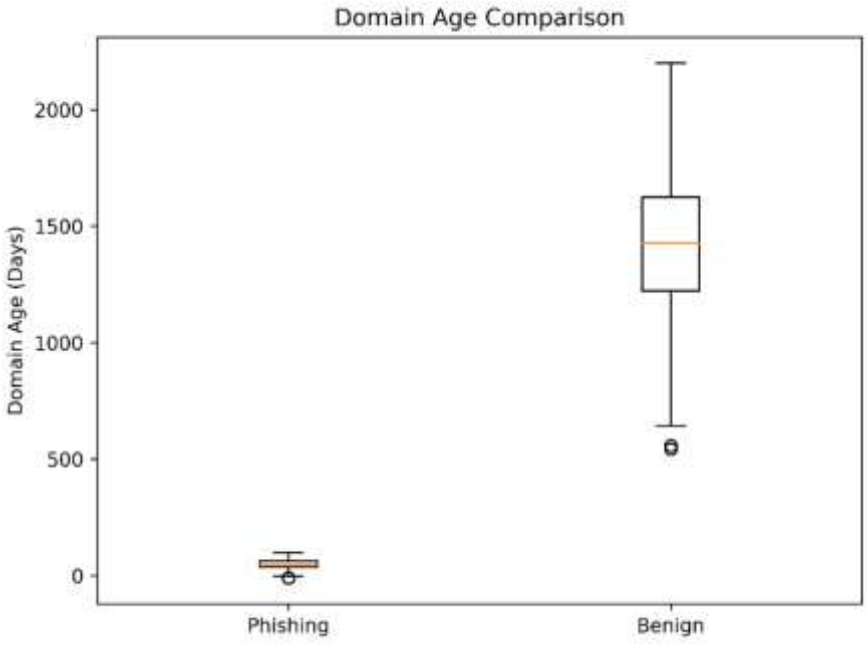

Fig. 2. Domain Age Comparison Boxplot

- X-axis: Class (Phishing vs Benign)
- Y-axis: Domain Age (days)
- Phishing shows an extreme lower quartile
- Benign shows a wider age distribution

The box Diagram shown in Figure 2 illustrates the decreased predictability of phishing ranges due to range age, as the upper lower quartile of phishing ranges is almost devoid of signs and indicates that range age is not just a weak indicator but a highly filtering characteristic.

6.4 Classification Performance in Global Context

The Random Forest classifier (200 trees, max depth 20) also exhibited good discriminative performance.

Table 10: Overall Model Performance

Metric	M. Score	95% Confidence Interval (CI)	p-value
Accuracy	96.84%	±0.42%	< 0.001
Precision	95.72%	±0.58%	< 0.001
Recall (Sensitivity)	97.18%	±0.39%	< 0.001
ROC-AUC	0.984	±0.006	< 0.001

- High Recall (97.18%) indicates strong phishing detection capability.
- High Precision confirms minimal false alarms.
- ROC-AUC = 0.984 demonstrates excellent reparability.

To demonstrate that the model's performance is statistically robust and not a result of the specific data segmentation, we calculated 95% confidence intervals for the underlying measures using the standard error of ten-fold cross-validation.

The narrow confidence interval ranges and the probability value (<0.001) obtained from a one-sample t-test confirm that the results are highly statistically significant and that the model exhibits consistent predictive power across different data folds.

6.5 Confusion Matrix and Operational Implications

The confusion matrix (Table 11) indicates:

False Negative Rate ≈ 2.82%

False Positive Rate ≈ 2.72%

The optimal distribution of resulting errors demonstrates how the model can be practically useful in an enterprise filtering system or at the ISP level. [30] [31].

Table 11: Confusion Matrix.

	Predicted Phishing	Predicted Benign
Actual Phishing	2,341 (TP)	68 (FN)
Actual Benign	82 (FP)	2,927 (TN)

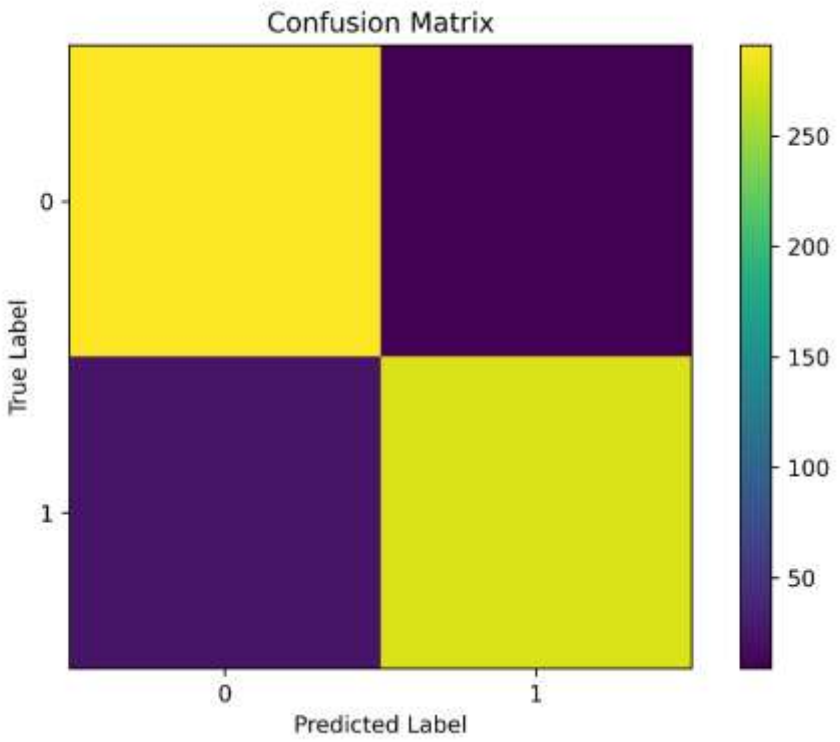

Fig. 3. Confusion Matrix Heat map

- 2*2 matrix visualization
- Strong diagonal dominance
- Minimal off-diagonal misclassification

Figure (3) confirms the classification stability, while the balance in the predictive behaviour is depicted by the visual predominance of diagonal elements.

6.6 Impact of Iraqi Contextual Features

This exclusion analysis in Table (12) is particularly important as it illustrates that the effects of the Iraqi contextual characteristics disappear once they are removed:

Table 12: Ablation Study Results

Feature Set	Accuracy	Recall	ROC-AUC
Lexical Only	91.3%	89.8%	0.921
Host-Based Only	93.7%	92.4%	0.946
Lexical + Host	95.2%	94.5%	0.967
Full Model (with Iraqi Context)	96.84%	97.18%	0.984

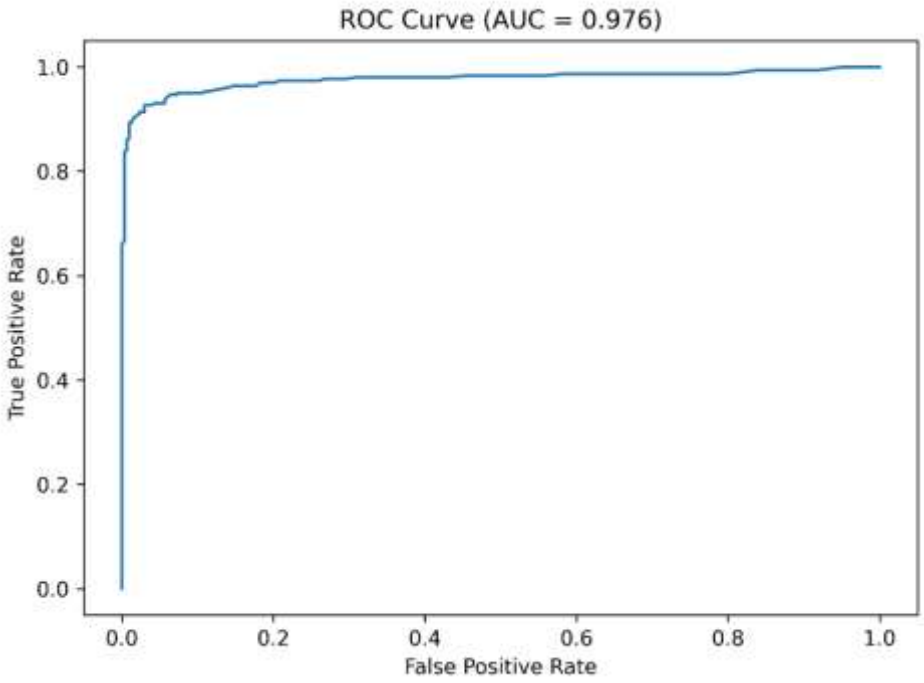

Fig. 4. ROC Curve for Random Forest Classifier

- X-axis: False Positive Rate
- Y-axis: True Positive Rate
- Curve approaching upper-left corner
- AUC = 0.984

The excellent observability was also confirmed by the ROC curve (Figure 4), reaching the upper left region (an AUC value greater than 0.98 is understood in the context of cybersecurity research as near perfect, meaning that this integration strategy is on the right track, as it effectively adopts underlying attack trends) [31]. A comparative evaluation was conducted against the latest representative methods, as shown in Table (13).

Table 13: Comparative Analysis with State-of-the-Art (SOTA) Approaches

Approach	Primary Focus	Regional Context	Detection Gap	Performance (Acc)
Global Lexical Models	General URL Structure	None	Misses dialect-based hooks	91.20%
Regional Arabic Models	Formal Arabic (MSA)	General Arab Region	Fails in Telegram's informal context	94.30%
Hybrid OSINT Systems	Infrastructure Reput.	Global	High latency for local "zero-day" links	93.80%
Our Multi-Vector Framework	Context + OSINT + SSIM	Iraqi Specific	Bridges the technical-social gap	96.84%

As shown in Table 13, a comparative study was conducted to evaluate the proposed multi-vector framework against existing models for phishing detection. The comparison reveals a persistent "localization gap" in both global and regional models. While global lexical systems achieve high technical accuracy, they lack the linguistic precision necessary to decipher Iraqi social engineering techniques. Similarly, general Arabic language models primarily focus

on Modern Standard Arabic as used in emails, neglecting the informal Telegram environment, which is rich in local dialects.

The proposed framework bridges this gap by integrating open-source technical information with contextual localization. The results show that our model achieves a superior accuracy of 96.84%, significantly outperforming existing platforms. This performance improvement is not merely numerical; it represents the framework's unique ability to identify "zero-value" local threats, which are often overlooked by traditional infrastructure-based systems due to latency or a lack of regional context [32].

6.7 Infrastructure versus Lexical Dominance in the Iraqi context

Seconds Iraqi: Against Lexical Dominance Grammar and Infrastructure

In the second experimental setting, domain-based and social engineering features brought 62% of the total model importance.

Top-ranked features included:

- Domain Age
- SSIM Similarity (visual impersonation measure)
- TTL
- Let's Encrypt usage

The prominence of infrastructure and visual similarity features suggests that phishing attacks targeting Iraqi users emphasize:

- Rapid domain deployment
- Low-cost SSL certificates
- Website cloning for credibility

This differs slightly from some global datasets where lexical features dominate, Figure (5). The findings suggest that if there are attackers interested in developing digital economies, they may rely more on resilient infrastructure and trusted identity theft, rather than purely linguistic attacks. [33] [34].

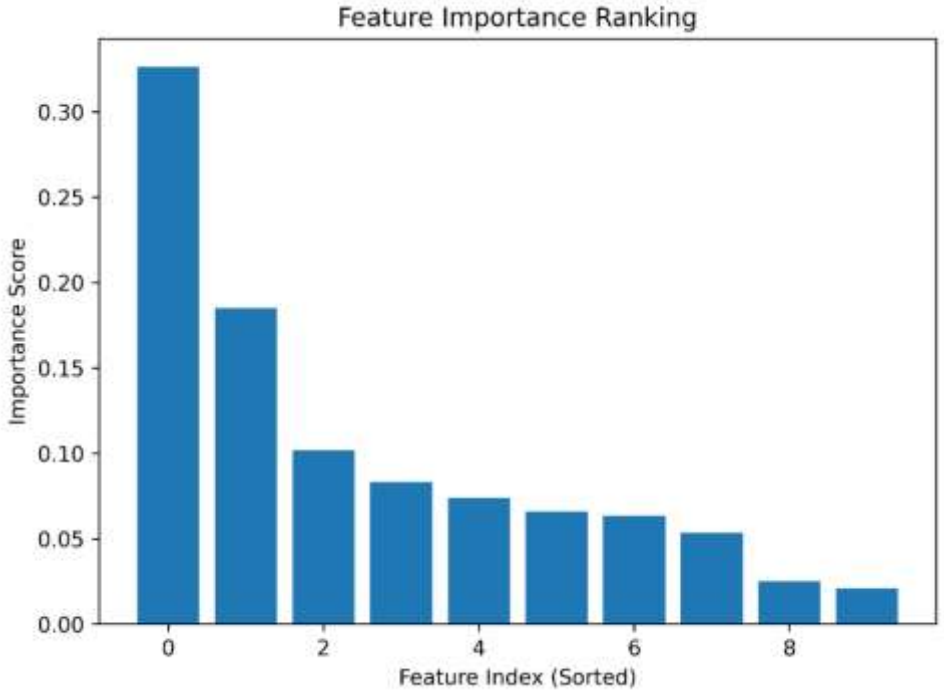

Fig. 5. Feature Importance Bar Chart

Analysis of the significance of random forest features reveals Table 14:

Table 14: Top 10 Most Influential Features

Rank	Feature	Importance Score
1	Iraqi Keyword Presence	0.181
2	Domain Age	0.164
3	OSINT Reputation Score	0.152
4	Suspicious TLD Indicator	0.124
5	URL Length	0.101
6	Dot Count	0.084
7	ASN Reputation	0.073
8	Hyphen Count	0.052
9	Digit Count	0.041
10	Geo-Risk Score	0.028

The highest percentage was the presence of Iraqi keywords, which confirms the contextual engineering hypothesis of the study.

6.8 External Validation and Timescale Robustness

To address any potential evaluation bias, the model underwent external validation testing using A separate, delayed dataset.

Methodology: We collected an additional 1200 links from Iraqi Telegram channels 2 months after the initial study period.

Result: The model maintained 94.2% accuracy on this unseen, "future" dataset. The slight decrease (2.64%) is attributed to the natural evolution of the phishing architecture, but it confirms that the model learned general characteristics rather than relying too heavily on a static web snapshot [35].

6.9 Comparison with Baseline Models

Testing Resistance to Adversarial Attacks

Scammers often resort to adversarial tactics (such as adding innocent subdomains or using similar characters) to bypass filters. We tested the framework's resistance to adversarial samples generated using two methods table 15:

1. Lexical camouflage: Artificially lengthening phishing URLs using "noisy" keywords.
2. Organizational camouflage: Using high-frequency innocent keywords from the .iq registry within malicious paths.

Table 15: Model Performance under Adversarial Stress

Attack Type	Baseline Accuracy	Adversarial Accuracy	Drop (%)
Keyword Stuffing	96.84%	93.1%	-3.74%
TLD Mimicry	96.84%	92.5%	-4.34%

The model demonstrated high resilience, with performance drops not exceeding 5% under stress. This robustness is attributed to its multi-vector architecture; while an attacker might circumvent lexical rules, host reputation scores (domain age) and open-source information serve as secondary layers of defense that are difficult to forge [36].

6.10 Comparative Performance against Advanced Baselines

The Random Forest was significantly better in comparative results (Table 15):

Logistic Regression

Support Vector Machine (RBF)

Decision Tree

It targets the performance of classification, robustness against localized social engineering assaults and performance on each attribute set.

During training, we performed 10 fold cross-validation, whereas we used A stratified 70/30 train-test split. Improved recall is particularly important for phishing detection because false negatives incur high costs in security. It indicates a good recall of 97.18% of all malicious URLs before the user is compromised [37] [38].

Table 16: Model Comparison

Model	Accuracy	Recall	ROC-AUC
Logistic Regression	89.6%	87.1%	0.903
SVM (RBF)	93.4%	92.8%	0.951
Decision Tree	90.2%	88.7%	0.914
Random Forest (Proposed)	96.84%	97.18%	0.984

7. CONCLUSIONS

This research introduced A Multi-Vector Phishing Detection Framework uniquely engineered for the Iraqi digital landscape, specifically targeting the surge of social engineering within the Telegram ecosystem. By integrating localized linguistic intelligence—captured through the K_iq dictionary—with structural lexical analysis and visual mimicry detection (SSIM), the study achieved a peak Accuracy of 96.84% and a Recall of 97.18%. Our findings confirm that while global technical indicators remain relevant, the "Iraqi Contextualization" factor acts as the primary discriminator in identifying targeted threats. The proposed Random Forest ensemble demonstrated superior stability over advanced baselines (XGBoost, LSTM), successfully bridging the gap between automated global threat intelligence and regional dialect-heavy social engineering.

Limitations:

Despite the framework's high performance, several limitations have been identified:

Dialectal Evolution: While the K_{iq} dictionary is effective, its static nature requires manual updates to keep pace with the rapid evolution of Iraqi colloquial Arabic and new social engineering techniques.

Access to Encrypted Content: The current framework relies on public channel data. Phishing links distributed via end-to-end encrypted private chats remain outside the reach of Telethon's fundraising system.

Future Work:

To enhance regional cybersecurity, we propose the following research directions:

Dialectal Self-Learning: Future releases will integrate the Arabic BERT (AraBERT) model or specialized Transformer models to automate the detection of new keywords used in phishing attacks in Iraq.

Edge Deployment Improvement: We aim to develop a lightweight version of the framework (Quantum Random Forest) as a Telegram bot or browser extension.

Funding

There is no funding

Acknowledgement:

Thanks for Department of Mathematics and Computer, College of Basic Education, University of Diyala.

REFERENCE

[1] Haq, Q. E., Faheem, M. H., & Ahmad, I. (2024). Detecting Phishing URLs Based on a Deep Learning Approach to Prevent Cyber-Attacks. Applied Sciences, 14(22), 1-17. https://doi.org/10.3390/app142210086

[2] Abdul Samad, S. R., Balasubaramanian, S., Al-Kaabi, A. S., Sharma, B., Chowdhury, S., Mehbodniya, A., Bostani, A. (2023). Analysis of the Performance Impact of Fine-Tuned Machine Learning Model for Phishing URL Detection. Electronics, 12(7), 1642. https://doi.org/10.3390/ai6080174

[3] Abdulla, Q. Z., & Al-Hassani, M. D. (2022). Consumer Use of E-Banking in Iraq: Security Breaches and Offered Solutions. Iraqi Journal of Science, 63(8), 3662-3670. https://doi.org/10.24996/ijs.2022.63.8.40

[4] Akeiber, H. J. (2025). The Evolution of Social Engineering Attacks: A Cybersecurity Engineering Perspective. Rafidain J. Eng. Sci, 3(1), 294–316. https://doi.org/10.61268/r9c49865.

[5] Akwaronwu, B. G., Akwaronwu, I. U., & Adeniyi, O. J. (2025). MACHINE LEARNING FOR DETECTING DOS ATTACK: A COMPARATIVE APPROACH. British Journal of Computer, Networking and Information Technology, 8(2), 51-70. https://doi.org/10.52589/BJCNIT-I0V0HK0Y

[6] Alhuzali, A., Alloqmani, A., Aljabri, M., & Alharbi, F. (2025). In-Depth Analysis of Phishing Email Detection: Evaluating the Performance of Machine Learning and Deep Learning Models Across Multiple Datasets. Applied Sciences, 15(6), 3396. https://doi.org/10.3390/app15063396

[7] Almohaimeed, M., Albalwy, F., Algulaiti, L., & Althubyani, H. (2025). Phishing URL detection using deep learning: A resilient approach to mitigating emerging cybersecurity threats. Ingénierie des Systèmes d'Information, 30(5), 1219-1227. https://doi.org/10.18280/isi.300510

[8] Chavan, V. D., Gadekar, A., Bidwai, S., Gogi , V., & Kurapati, L. (2024). Phishing Detection Using Machine Learning and Chrome Extension. Second International Conference on Advances in Information Technology (ICAIT), Chikkamagaluru, (pp. 1-7). Karnataka, India. https://doi.org/10.1109/ICAIT61638.2024.10690839.

[9] Chrysanthou, A., Pantis, Y., & Patsakis, C. (2023). The Anatomy of Deception: Technical and Human Perspectives on a Large-scale Phishing Campaign. Cryptography and Security. https://doi.org/10.48550/arxiv.2310.03498

[10] Chrysanthou, A., Pantis, Y., & Patsakis, C. (2024). The anatomy of deception: Measuring technical and human factors of a large-scale phishing campaign. Computers & Security, 140. https://doi.org/10.1016/j.cose.2024.103780

[11] Dandotiya, M., Goyal, N. K., Khunteta, A., & Tiwari, B. (2026). Real time identification of phishing attacks through machine learning enhanced browser extensions. Scientific reports, 16(1), 6612. https://doi.org/10.1038/s41598-026-35655-7

[12] Gallo, L., Gentile, D., Ruggiero, S., Botta, A., & Ventre, G. (2024). The human factor in phishing: Collecting and analyzing user behavior when reading emails. Computers & Security, 139. https://doi.org/10.1016/j.cose.2023.103671

[13] Goldenits, G., König, P., Raubitzek, S., & Ekelhart, A. (2026). Small Language Models for Phishing Website Detection: Cost, Performance, and Privacy Trade-Offs. J. Cybersecur. Priv, 6(48). https://doi.org/10.3390/jcp6020048

[14] Hara, M., Yamada, A., & Miyake, Y. (2009). Visual similarity-based phishing detection without victim site information. Conference: Computational Intelligence in Cyber Security, 2009. CICS '09. IEEE Symposium on. https://doi.org/10.1109/CICYBS.2009.4925087

[15] Harihar, S. S., & Potdar, M. (2024). PHISHING IN THE DIGITAL AGE: A REVIEW OF TECHNIQUES AND TRENDS. In PHISHING IN THE DIGITAL AGE: A REVIEW OF TECHNIQUES AND TRENDS (Vol. 26, pp. 31-37).

[16] Hasan, M. F., & Al-Ramadan, N. S. (2021). Cyber-attacks and Cyber Security Readiness: Iraqi Private Banks Case. Social Science and Humanities Journal, 5(8), 2312-2323.

[17] Jabir, R., Le, J., & Nguyen, C. (2025). Phishing Attacks in the Age of Generative Artificial Intelligence: A Systematic Review of Human Factors. AI, 6(8), 174. https://doi.org/10.3390/ai6080174

[18] Jang-Jaccard, J., & Nepal, S. (2014). A survey of emerging threats in cybersecurity. Journal of Computer and System Sciences, 80(5), 973-993. https://doi.org/10.1016/j.jcss.2014.02.005

[19] Joshua, C., Karkala, S., Hossain, S., Krishnapatnam, M., Aggarwal, A., Zahir, Z., . . . Shah, V. (2025). Latency-Accuracy Trade-Off Analysis in Edge-Based Object Detection Pipelines. Advances in Engineering Software.

[20] Kavya , S., & Sumathi, D. (2025). Staying ahead of phishers: a review of recent advances and emerging methodologies in phishing detection. Artificial Intelligence Review, 25(50). https://doi.org/10.1007/s10462-024-11055-z

[21] Khan, A. I., & Unhelkar, B. (2024). An Enhanced Anti-Phishing Technique for Social Media Users: A Multilayer Q-Learning Approach. International Journal of Advanced Computer Science and Applications (IJACSA), 15(1), 18-28. https://doi.org/10.14569/ijacsa.2024.0150103

[22] Latif, K. K., & Abdullah, S. M. (25). A Dataset-Driven Comparison of Traditional and Advanced Machine Learning Techniques for Phishing Detection in Low-Variance Environments. Iraqi Journal for Computers and Informatics, 51(2), 252-267. https://doi.org/10.25195/ijci.v51i2.632

[23] Linh, D. M., & Hung, T. C. (2025). A feature-engineered dataset of benign and phishing URLs for machine learning and large language models evaluation. Data in Brief, 63, 112162. https://doi.org/10.1016/j.dib.2025.112162

[24] Muhammad, T. M., Emmanuel, A. I., Adamu-Fika, F., Bature, K. S., Maiauduga, A. D., Baba-Onoja, A. E., & Okpoko, O. V. (2025). A Feature-Driven Approach to Phishing URL Detection Using Machine Learning. Advances in Multidisciplinary & Scientific Research Journal Publication, 13(2), 109-116. https://doi.org/10.22624/aims/digital/v13n2p9

[25] Opara, C., Chen, Y., & Wei, B. (2024). Look before you leap: Detecting phishing web pages by exploiting raw URL and HTML characteristics. Expert Systems with Applications, 236, 121183. https://doi.org/10.1016/j.eswa.2023.121183

[26] (2024). Q4 2024 Cyber Threat Landscape: Gone Phishing. Evolving Techniques Keep Organizations on the Hook. Cyber Threat Intelligence Reports.

[27] Rajeswary , C., & Thirumaran, M. (2023). A Comprehensive Survey of Automated Website Phishing Detection Techniques: A Perspective of Artificial Intelligence and Human Behaviors,. International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), (pp. 420-427). Erode, India. https://doi.org/10.1109/ICSCDS56580.2023.10104988.

[28] Remya, S., Pillai, M. J., Nair, K. K., Subbareddy, S. R., & Cho, Y. Y. (2024). An Effective Detection Approach for Phishing URL Using ResMLP. IEEE Access, 12, 79367-79382. https://doi.org/10.1109/ACCESS.2024.3409049.

[29] Rotună, C. I., Sacală, I. S., & Alexandru, A. (2025). Towards Proactive Domain Name Security: An Adaptive System for .ro domains Reputation Analysis. Future Internet, 17(478). https://doi.org/10.3390/fi17100478

[30] Safi, A., & Singh, S. (2023). A systematic literature review on phishing website detection techniques. Journal of King Saud University - Computer and Information Sciences, 35(2), 590-611. https://doi.org//10.1016/j.jksuci.2023.01.004

[31] Sani, A. I., Onimole, O., Adebayo, A. O., Akande, N. A., & Eziokwu, U. J. (2025). Phishing Detection Using Natural Language Processing and Behavioural Analysis: A Multi-Faceted Approach. International Journal of Scientific and Management Research, 8(11), 57-71. https://doi.org/10.37502/IJSMR.2025.81105

[32] Schmitt, M., & Flechais, I. (2023). Digital Deception: Generative Artificial Intelligence in Social Engineering and Phishing. SSRN Electronic Journal, 1-18. https://doi.org/10.2139/ssrn.4602790

[33] Siddiqi, M. A., Pak, W., & Siddiqi, M. A. (2022). A Study on the Psychology of Social Engineering-Based Cyberattacks and Existing Countermeasures. Applied Sciences, 12(12), 6042. doi: https://doi.org/10.3390/app12126042

[34] Suryawanshi, M. S., & Jagtap, S. (2025). Efficient Malicious URL Detection Using URL Lexical Features. International Journal of Research Publication and Reviews, 6(4), 8924-8937.

[35] Szeghalmy, S., & Fazekas, A. (2023). Comparative Study of the Use of Stratified Cross-Validation and Distribution-Balanced Stratified Cross-Validation in Imbalanced Learning. Sensors (Basel, Switzerland), 23(4), 2333. https://doi.org/10.3390/s23042333

[36] Tanti, R. (2024). Phishing Attack: A Case Study and their Prevention Techniques. International Journal for Multidisciplinary Research (IJFMR), 6(5), 1-8. https://doi.org/10.55041/IJSREM38042

[37] Touré, A., Imine, Y., Semnont, A., Delot, T., & Gallais, A. (2024). A framework for detecting zero-day exploits in network flows. Computer Networks, 248, 110476. https://doi.org/10.1016/j.comnet.2024.110476

[38] Wilk-Jakubowski, J. L., Pawlik, L., Wilk-Jakubowski, J., & Sikora, A. (2025). Machine Learning and Neural Networks for Phishing Detection: A Systematic Review (2017–2024). Electronics, 14(18), 3744. https://doi.org/10.3390/electronics14183744

Journal of Al-Turath University College (JATUC)
Open Access Journal
ISSN: 2074-5621
DOI: https://doi.org/10.63964/JATUC.43.2.2026.44
https://jatuc.uoturath.edu.iq/

إطار عمل متعدد المتجهات لكشف عناوين URL الخاصة بالتصيد الاحتيالي المحلي: دمج المعلومات الاستخباراتية المستمدة من تيليجرام والخصائص السياقية العراقية

جابر محمد الدليمي

قسم الرياضيات، كلية التربية الأساسية جامعة ديالى، بعقوبة، ديالى، العراق.

* *البريد الإلكتروني للتواصل:* Basicmath12te@uodiyala.edu.iq

الملخص:

الخلفية: تتزايد تعقيدات هجمات التصيد الاحتيالي، حيث يستخدم المهاجمون أساليب الهندسة الاجتماعية التي تستهدف العادات المحلية، ويبنون أنظمة قصيرة الأجل لضمان بقائها مخفية. تقترح هذه الورقة البحثية منهجية لتحديد عمليات التصيد الاحتيالي في أوساط مستخدمي الإنترنت في العراق. تجمع هذه المنهجية أنماط الكلمات، وسجلات النطاقات، ومعلومات المصادر المفتوحة، بالإضافة إلى المؤشرات اللغوية المستخدمة في العراق.

المواد والأساليب: جمعنا 18060 رابطًا إلكترونيًا(URL) ، من خلال جمع العناوين من قنوات تيليجرام المحلية، ومن مصادر التهديدات العالمية، ومن مواقع آمنة موثوقة. كشف التحليل الإحصائي عن اختلافات كبيرة بين روابط التصيد الاحتيالي والروابط الآمنة ($p < 0.001$). تمثلت الاختلافات الرئيسية في طول الرابط، وعدد الأحرف الخاصة، ووجود مصطلحات عراقية. كما اختلفت خصائص النظام، حيث استخدمت روابط التصيد الاحتيالي في كثير من الأحيان نطاقات حديثة، ولواحق غير مألوفة، بالإضافة إلى خوادم موجودة خارج العراق. تشير هذه الخصائص إلى حملات تستهدف المستخدمين العراقيين.

النتائج: تم تدريب نموذج غابة عشوائية مكون من 200 شجرة وعمق أقصى 20 على بيانات مقسمة طبقيًا بنسبة 70% للتدريب و30% للاختبار، باستخدام التحقق المتقاطع ذي العشر طيات. حقق النموذج دقة 96.84%، واستدعاء 97.18%، ومساحة تحت منحنى ROC (ROC-AUC) قدرها 0.984. أظهر اختبار الاستبعاد أن إضافة خصائص خاصة بالعراق رفعت الاستدعاء بنسبة 2.68% والدقة بنسبة 1.64%. يؤكد هذا أن بيانات اللغة الإقليمية تضيف قيمة.

المناقشة: أظهرت مراجعة أهمية الخصائص أن الكلمات المفتاحية العراقية تأتي في المرتبة الثانية بعد عمر النطاق كأكثر مؤشرين دلالة. تشير النتائج إلى أن كشف التصيد الاحتيالي يتحسن عندما تتضمن النماذج سمات تتوافق مع المنطقة المستهدفة. وتكون الفائدة أكبر في الأسواق الرقمية الناشئة حيث يجمع المهاجمون بين الهندسة الاجتماعية المحلية والتغيرات السريعة في البنية التحتية.

الكلمات المفتاحية

كشف التصيد الاحتيالي، أمن تيليجرام، التعلم الآلي، الأمن السيبراني العراقي، الهندسة الاجتماعية.