

Phishing Email Detection Using Large Language Models and Explainable Artificial Intelligence

Zainab Mohammed Ali^{1*}, Zainab Shaker Matar Al-Husseini¹

¹Department of Computer Science, College of Science, University of Diyala, Diyala, Iraq

Article Info

Article history:

Received May 14, 2026

Revised July 15, 2026

Accepted August 20, 2026

Keywords:

Phishing detection

Large language models

Explainable artificial

intelligence

Shapley Additive explanations

Adversarial robustness

ABSTRACT

Phishing is a significant issue in cybersecurity; nowadays, with the help of generative artificial intelligence, attackers are capable of creating very convincing fake emails at large scale, which overwhelms traditional detection mechanisms. This paper suggests a hybrid model consisting of a fine-tuned Bidirectional Encoder Representations from Transformers model and a SHapley Additive explanations explainability layer to classify phishing in real-time. The model was trained over 120,000 labeled examples on three benchmark datasets, which were consolidated, with a fourteen-dimensional URL structural feature module added. The proposed framework was able to classify 98.70% of the 12,000-instances test set. Precision and recall were 98.40% and 98.90%, yielding an F1-score of 98.65%. The region below the receiver operating characteristic curve was 0.997. False positive rate was 0.90 which is sufficient to meet enterprise deployment. McNemar testing showed all the improvements greater than six baselines to be statistically significant (p less than 0.0001). In adversarial attacks, the lowest F1-score was 93.80, which is 6.20 percentage points better than the best baseline. The mean accuracy of cross-dataset generalization was 96.70 percent, which is 3.30 percentage points higher than the previous best benchmark. SHapley analysis found the top three discriminative features as URL entropy, sender domain anomaly, and urgency-linguistic patterns. The proposed architecture offers a correct, robust and interpretable phishing detection system; future research will focus on adversarial training and multilingual extension.

This is an open access article under the CC BY-SA license.



1. INTRODUCTION

The spread of internet-based communication across the world has radically changed the way information is exchanged and at the same time has provided an attack surface that is systematically used by bad actors. Email remains the most widespread type of professional communication, and in 2024, 347 billion messages were sent daily, and phishing, which involves sending fraudulent emails to obtain credentials or install malware, is estimated to account for 91 percent of successful cyberattack entry points [2]. According to the 2024 Activity Trends Report by the Anti-Phishing Working Group, more than 1.3 million unique phishing sites were detected in a single quarter, representing a 28% increase compared with the previous year [2]. In 2023, financial losses due to phishing were over USD 52 billion worldwide, with small-to-medium enterprises being the most impacted by relatively low cybersecurity postures [3]. In addition to direct financial damage, effective phishing attacks allow secondary attack chains such as credential stuffing, ransomware installation, and advanced persistent threat intrusion [4][5]. Phishing attacks have evolved to be so sophisticated that they have outdone the conventional detection mechanisms. The initial rule-based filters are made useless by adversarially designed messages that take advantage of synonym replacement, Unicode homographs, and multi-stage redirection paths [6][7]. Classical machine learning techniques can improve the detection rates, but are constrained by the fact that handcrafted features are used to improve generalizability [1][8]. Above all, the democratization of generative artificial intelligence tools has brought a qualitatively new threat dimension:

*Corresponding Author

Email: zainab_m@uodiyala.edu.iq

opponents now generate grammatically perfect, contextually personalized phishing messages at an industrial scale, making the approach based on linguistic imperfection as a discriminative cue irrelevant [13][34]. In parallel with this growth, the existing deep learning detectors are limited in two aspects of operation: opacity - they cannot explain their decisions to analysts or satisfy regulatory documentation requirements [19][20]; and single-source evaluation bias that inflates real-world generalization performance by 8–15 percentage points in cross-corpus validation experiments [42][43]. Generative AI further intensifies this challenge by enabling distinct phishing messages to be produced at scale, reducing the effectiveness of repetition-based filtering [49].

This paper is driven by three convergent issues: (i) the lack of effectiveness of current systems in detecting generative phishing content created by artificial intelligence; (ii) the lack of transparency in deep learning detectors; and (iii) the single-source bias of previous research [1][7][19].

The paper proposes a hybrid detector that uses a fine-tuned Bidirectional Encoder Representations from Transformers encoder with a fourteen-dimensional URL feature module and integrated SHapley Additive exPlanations, trained on a 120,000-instance multi-source corpus under accuracy, adversarial robustness, and cross-dataset generalization protocols [9][17].

The key contributions of the work are as follows: (i) a hybrid architecture that combines transformer-based semantic representations with URL structural features and built-in explainability; (ii) a 120,000-instance multi-source evaluation corpus combined into three independent benchmark datasets; (iii) a six-metric comparative evaluation with six baselines with statistical significance testing; and (iv) per-prediction SHapley attributions. Although LSTM models have proven to be effective in identifying deceptive textual content, recent studies have pointed out their inability to capture complex contextual patterns, and suggested that Transformer-based models like BERT and GPT are more powerful alternatives [46]. Recent transformer-based phishing detection research has also highlighted the importance of combining high predictive performance with explainable decision support [47]. LLM-based phishing detectors have also demonstrated that high detection accuracy can be accompanied by explicit rationales for classification [48].

Accordingly, the problem addressed in this study is the limited ability of existing phishing email detectors to simultaneously identify AI-generated phishing content, provide transparent prediction explanations, and generalize across heterogeneous email sources. To address this problem, this study proposes a hybrid framework that integrates a fine-tuned BERT encoder, fourteen URL structural features, and SHAP-based explainability. The experimental results demonstrate 98.70% accuracy, a 0.997 AUC-ROC, a 0.90% false positive rate, a worst-case adversarial F1-score of 93.80%, and 96.70% mean cross-dataset accuracy. The main contribution is therefore a unified, interpretable, and robust phishing detection framework that combines contextual language representations with structural URL evidence and per-prediction explanations.

2. METHOD

The proposed methodology follows five consecutive steps: acquisition and consolidation of the dataset, data preprocessing and feature engineering, model architecture design and fine-tuning, explainability layer integration, and evaluation protocol specification. Each stage is described in sufficient technical detail to support reproducibility.

2.1. Dataset Acquisition and Consolidation

To reduce the single-source generalization bias observed in previous studies [42][43], three publicly available email corpora were merged: the Nazario Phishing Corpus (32,145 phishing instances), the SpamAssassin Public Corpus (52,840 instances), and the Enron Email Dataset (38,267 legitimate instances) [44][45]. The overall corpus had a phishing to legitimate ratio of about 1:2.4. Stratified random undersampling was used to generate a balanced 60,000:60,000 distribution [31]. Stratified sampling was used to divide the consolidated corpus into training (96,000), validation (12,000), and test (12,000) splits to maintain the balance of classes in all three subsets. This balancing step was applied to reduce class-imbalance bias. Samples were retained through stratified random undersampling so that the class proportions from the source corpora were considered during selection; no additional manual exclusion criteria were applied beyond the balancing procedure.

Table 1. Dataset Composition After Consolidation and Balancing

Dataset	Original Instances	Phishing	Legitimate
Nazario Phishing Corpus	32,145	32,145	0
SpamAssassin Public Corpus	52,840	6,066	46,774
Enron Email Dataset	38,267	0	38,267
Combined (post-balancing)	120,000	60,000	60,000

2.2. Data Preprocessing and Feature Engineering

Raw email files were analyzed to capture domain of the sender, reply-to address, subject line, body/text, embedded hyperlinks, and other metadata on the attachment.

They were normalized in a seven-step pipeline: (1) remove hypertext markup language tags, (2) extract uniform resource locators and tokenize places with a placeholder, (3) normalize to Unicode NFKC in order to eliminate homographs, (4) normalize lowercasing and whitespace, (5) delete boilerplate, (6) truncate to 512 WordPiece tokens, and (7) subject-body concatenation with [SEP] delimiter [6][7]. These preprocessing operations were applied consistently to all messages to standardize heterogeneous email content and produce a uniform input representation for the BERT encoder.

A special URL feature engineering module extracted fourteen numerical features per message: domain age, subdomain depth, subdomain top-level domain risk score, length of URL characters, Shannon entropy, IP address substitution flag, number of special characters used, use of HTTPS indicator, length chain of redirects, and Levenshtein distance to 500 highest-impersonated organizational domains [38][40]. The URL features were included because they capture structural phishing indicators that can remain informative when attackers manipulate the linguistic content of an email.

2.3. Model Architecture

2.3.1. BERT-Based Encoder

The detection architecture uses the bert-base-uncased architecture with 12 transformer layers, 12 self-attention heads, a 768-dimensional hidden state, and about 110 million trainable parameters [9]. The bert-base was chosen instead of BERT-large on the basis of inference latency: BERT-large took more than 100 milliseconds to run on CPU-only hardware, whereas BERT-base had a mean latency of 47 milliseconds [11][16]. The [CLS] token representation output of the last transformer layer was fused with the 14-dimensional URL feature vector, resulting in a 782-dimensional fused embedding that was fed to the classification head. This fusion was used to combine contextual semantic information from the email text with complementary structural evidence from embedded URLs.

2.3.2. Classification Head and Fine-Tuning

The classification head has three parts: a dropout layer (rate = 0.3), a fully connected projection between 782 and 256 dimensions with Rectified Linear Unit activation, and a 2-class output layer with softmax activation [9][11]. The AdamW optimizer was used with a differential learning rate strategy 2×10^{-5} to the encoder and 1×10^{-4} to the classification head, and linear warm-up during the first 10 percent of training steps and early stopping based on validation F1-score [9][15]. These optimization settings were used to stabilize fine-tuning, limit overfitting, and preserve the pretrained language representations during task-specific adaptation.

Table 2. Model Architecture and Hyperparameter Specification

Component / Hyperparameter	Specification
BERT variant	bert-base-uncased (12 layers, 768-dim, 110M parameters)
Fused representation	782-dim ([CLS] 768-dim + URL features 14-dim)
Classification head	Dropout(0.3) $\rightarrow$ FC(256, ReLU) $\rightarrow$ FC(2, Softmax)
BERT learning rate	2×10^{-5} (AdamW, linear warm-up 10%)
Head learning rate	1×10^{-4}
Batch size / Max epochs	32 / 5 with early stopping on validation F1-score
Max sequence length	512 WordPiece tokens
Mean inference latency	47 milliseconds per message (CPU-only hardware)

2.4. Explainability Layer Integration

Explainability is modeled using SHapley Additive exPlanations, a game-theoretic attribution algorithm that assigns each feature a contribution score that measures its marginal contribution to the model output [17]. Computational tractability was achieved using the DeepExplainer variant on the 782-dimensional feature space, attributions were computed on a 500-instance background reference set of legitimate emails [17][22]. The token-level scores were calculated by adding the absolute SHapley values of all 768 embedding dimensions of each WordPiece token and normalizing them to [0, 1], allowing visualization of the influential words, phrases, and URL features as well as each prediction [20][21]. SHAP was selected because it supports both global and instance-level attribution without changing the underlying classification architecture. Recent phishing-detection research likewise emphasizes the value of interpretable AI for improving transparency and user trust in operational settings [50].

2.5. Baseline Models and Evaluation Protocol

They were implemented with six baseline classifiers: Support Vector Machine (radial basis function kernel, $C=10$), Random Forest (500 trees, max depth=20), Multinomial Naive Bayes (alpha=0.1), Long Short-Term Memory Network (2 layers, hidden=256, dropout=0.4), Convolutional Neural Network (filter sizes 3, 4, and 5). Each of the models was assessed based on six measures: accuracy, precision, recall, F1-score, area under the ROC curve, and false positive rate. McNemar testing with Bonferroni correction ($\alpha=0.05$) was

applied to all pairwise comparisons with Cohen's d effect sizes [31]. Adversarial robustness was tested on synonym substitution (10%/20%/30%), character perturbation (3%/5%/8%), and homograph insertion (5%/10%/20%) attacks [34][35]. The reported metrics were used to measure classification quality and false-alarm behavior, while McNemar testing, effect-size analysis, and adversarial perturbation experiments were used to assess statistical significance, practical differences, and robustness.

3. RESULTS AND DISCUSSION

This section presents and interprets the experimental results across five dimensions: overall classification performance, comparative baseline analysis, explainability findings, adversarial robustness, and cross-dataset generalization.

3.1. Overall Classification Performance

Table 3 shows that the proposed framework achieved an accuracy of 98.70%, precision of 98.40%, recall of 98.90%, and an F1-score of 98.65%, area under the ROC curve of 0.997, and false positive rate of 0.90% on the 12,000-instance test set. All measures are averages of five independent runs with standard deviations less than 0.3 percentage points, which proves the stability of results.

The false positive rate of 0.90% meets the sub-1% criterion of enterprise zero-disruption mail policies [33]. The region below the receiver operating characteristic curve of 0.997 supports the fact that there is outstanding discriminative power at all levels of decision making [19][44]. The confusion matrix resulted in 5,934 true positives, 5,946 true negatives, 54 false positives and 66 false negatives. The 66 false negatives were analyzed, showing that 71 percent of the false negatives were the messages generated by the generative artificial intelligence tools where adversarial paraphrasing had neutralized the surface-level linguistic cues [13][15].

Table 3. Primary Performance Metrics — Proposed Model (12,000-Instance Test Set)

Metric	Value	95% Confidence Interval	Interpretation
Accuracy	98.70%	98.41–98.99%	11,844 of 12,000 correctly classified
Precision	98.40%	98.05–98.75%	1.6% of phishing flags were false alarms
Recall	98.90%	98.61–99.19%	Only 1.1% of phishing emails missed
F1-Score	98.65%	98.34–98.96%	Near-perfect precision-recall balance
AUC-ROC	0.997	0.995–0.999	Exceptional class separability
False Positive Rate	0.90%	0.71–1.09%	Below enterprise 1% policy threshold

3.2. Comparative Baseline Analysis

The comparative evaluation is provided in Table 4. The proposed model performs better than all six baselines on all reported metrics. Against Random Forest - the best classical baseline - accuracy increases by 5.90 percentage points and false positive rate decreases by 7.50% to 0.90% a relative decrease of 88% [9][38]. The BERT-only ablation validates a 2.30 percentage point accuracy contribution of the URL feature module, and proves complementarity between transformer representations and structural URL features [6][15]. All six pairwise comparisons were statistically significant ($p < 0.0001$) using McNemar testing with Bonferroni correction. The d effect sizes of Cohen were between 0.89 and 3.17 between the BERT and the BERT-only baseline and Naive Bayes, respectively [31].

Table 4. Comparative Performance of All Models on the Held-Out Test Set

Model	Accuracy	F1-Score	AUC-ROC	FPR
Support Vector Machine	91.30%	91.25%	0.961	8.90%
Random Forest	92.80%	92.75%	0.972	7.50%
Naïve Bayes (Multinomial)	87.60%	87.53%	0.934	12.30%
Long Short-Term Memory	94.50%	94.45%	0.979	5.90%
Convolutional Neural Network	93.70%	93.65%	0.976	6.40%
BERT Only (no URL features)	96.40%	96.40%	0.991	3.40%
Proposed Model (BERT+URL+XAI)	98.70%	98.65%	0.997	0.90%

3.3. Explainability Analysis

SHapley Additive exPlanations attribution calculated on all 12,000 test cases found eight categories of features with mean absolute value (Table 5). The URL structure and entropy (0.342), sender domain anomaly (0.298), and urgency-linguistic patterns (0.241) all contribute to 57.7% of the total attribution mass [17][21]. The saliency of URL structural features is in line with the results of Opara et al. [6] and Le et al. [38], who have shown that URL-derived features are the most discriminative and resistant to adversarial text

manipulation. Urgency-linguistic patterns were in the third place, which validates the fact that phishing emails are systematically used to utilize psychological pressure mechanisms [2][3].

Table 5. Global Feature Importance — Mean Absolute SHapley Values

Rank	Feature Category	Mean SHAP	Phishing Direction
1	URL structure and entropy	0.342	High entropy → Phishing
2	Sender domain anomaly	0.298	Young/spoofed domain → Phishing
3	Urgency-linguistic patterns	0.241	Urgency trigger words → Phishing
4	Hyperlink redirect depth	0.187	Deep redirect chain → Phishing
5	Subject-body semantic coherence	0.163	Incoherence → Phishing
6	Sender authentication signals	0.128	Reply-to mismatch → Phishing
7	Financial / credential keywords	0.112	Keyword presence → Phishing
8	Attachment metadata signals	0.094	Executable attachment → Phishing

Per-prediction SHapley analysis provided actionable case-level rationales. In a properly detected phishing email (confidence 97.3%), the most significant features were URL entropy of 4.82 (+0.41), domain age of 3 days (+0.38), urgency phrase of Verify account immediately (+0.29), and Levenshtein distance of 1 with a major financial institution (+0.26). In a properly classified legitimate email (confidence 94.8%), all URL and domain characteristics added negative attributions of a 2,847-day domain age and entropy of 1.23 [20][21]. These per-prediction justifications give incident response and regulatory compliance documentation to the analyst.

3.4. Adversarial Robustness Evaluation

Table 6 shows F1-scores in nine adversarial conditions. The proposed model still has a minimum F1 of 93.80% with high-intensity homograph insertion - the worst case - versus 87.60% with BERT-only and 79.40% with Long Short-Term Memory [34][35]. The strength edge over BERT-only can be explained by the URL feature module, which offers structural clues that are resistant to all three types of attacks at the text level [6][38]. Three deployment hardening suggestions are as follows: (i) use real-time domain reputation querying to achieve maximum security return; (ii) use Unicode NFKC normalization to recover 6080% of homograph-induced degradation [4][35]; and (iii) use hybrid multi-modal feature fusion as the main defense-in-depth mechanism [15][32].

Table 6. F1-Score Under Adversarial Perturbation Attacks

Attack Type	Intensity	Proposed	BERT-Only	LSTM	Gap
Synonym Substitution	Low (10%)	98.10%	95.80%	92.10%	+2.30 pp
Synonym Substitution	Medium (20%)	96.70%	93.20%	87.40%	+3.50 pp
Synonym Substitution	High (30%)	94.20%	88.90%	81.30%	+5.30 pp
Char. Perturbation	Low (3%)	98.40%	96.10%	93.20%	+2.30 pp
Char. Perturbation	Medium (5%)	97.80%	94.70%	89.60%	+3.10 pp
Char. Perturbation	High (8%)	96.10%	91.30%	82.70%	+4.80 pp
Homograph Insertion	Low (5%)	97.90%	95.40%	91.80%	+2.50 pp
Homograph Insertion	Medium (10%)	96.40%	92.10%	86.20%	+4.30 pp
Homograph Insertion	High (20%)	93.80%	87.60%	79.40%	+6.20 pp

3.5. Cross-Dataset Generalization

The leave-one-dataset-out analysis provided a mean accuracy of 96.70% (SD = 0.46) across three rotations (Table 7), which is 3.30 percentage points higher than the previous cross-dataset benchmark - Dutta and Bhattacharjee [44] at 93.40%. The small standard deviation of 0.46 percentage points validates consistent generalization regardless of the dataset withheld [31][42]. The slightly reduced accuracy on the Enron rotation (96.10%) is due to the shift in the temporal distribution between the 19992002-collection period [14][45].

3.6. Limitations

The current findings were limited by five factors. To begin with, the latest phishing training examples are as recent as 2023; the patterns generated by generative artificial intelligence since 2024 are underrepresented [13][34]. Second, only English-language corpora [2][4] were evaluated. Third, the concept drift in deployed systems is not modeled by the static protocol [15]. Fourth, binary classification scope does not distinguish phishing sub-types [3][32]. Fifth, ground-truth labels are based on automated annotation instead of expert validation [31].

Table 7. Leave-One-Dataset-Out Cross-Dataset Generalization Results

Training Datasets	Test Dataset	Accuracy	F1-Score	AUC-ROC
SpamAssassin + Enron	Nazario	97.20%	97.10%	0.991
Nazario + Enron	SpamAssassin	96.80%	96.60%	0.989
Nazario + SpamAssassin	Enron	96.10%	95.90%	0.986
Mean ($\pm$ SD)	All rotations	96.70% $\pm$ 0.46%	96.53% $\pm$ 0.50%	0.989

4. CONCLUSION AND LIMITATION

This paper describes a hybrid phishing email detector based on an encoder model with a fine-tuned Bidirectional Encoder Representations from Transformers encoder with a fourteen-dimensional URL feature encoder and incorporated SHapley Additive explanations, evaluated on a 120,000-sample multi-source dataset. The framework achieved 98.70% accuracy, 0.90% false positive rate, 0.997 area under the ROC curve, a worst-case attack adversarial F1 minimum of 93.80% and the mean of 96.70% cross-dataset generalization accuracy - a performance profile that outperforms all six baselines it was tested on across all dimensions reported [9]. Three theoretical contributions are generated. First, URL structural properties which are insensitive to text-level perturbation give a defense-by-depth property that cascade failure of language-model-only detectors on high-intensity attacks, a design principle that hybrid email security architecture, such as [6][34][38]. Second, the BERT-only ablation confirms that representations of transformers and URLs are complements - even highly expressive language models do not encode URL structural cues in textual descriptions wholly [15]. Third, 47-milliseconds mean inference time inclusive of SHapley attribution calculation shows that integrated real-time explainability and enterprise deployment performance can coexist as goals in this field [17][20]. There exist four practical implications, namely: (i) consolidate multi-source training corpora to distribute it in a robust manner; (ii) invest in URL analysis infrastructure as the marginal-return engineering priority; (iii) integrate SHapley attribution reports into incident response processes to comply with regulatory documentation requirements; and (iv) establish quarterly retraining cycles to deal with concept drift [40][42][45]. The most prominent limitations are the currency of the training data (before 2024 patterns of generative artificial intelligence) and the focus on monolingual English evaluation and the fixed evaluation protocol. Future directions [4][13][14][34], and [15] should focus on: adversarial training versus generative artificial intelligence-generated phishing content [13][34]; multilingual extension through multilingual transformer encoders [4][14]; and continuing learning models to manage concept drift [15][45].

REFERENCES

- [1] R. Trad and A. Chehab, "Phishing detection using machine learning and deep learning: A systematic literature review," *Comput. Secur.*, vol. 138, p. 103657, 2024.
- [2] A. Almomani, B. B. Gupta, S. Atawneh, A. Meulenberg, and E. Almomani, "A survey of phishing email filtering techniques," *IEEE Commun. Surv. Tuts.*, vol. 15, no. 4, pp. 2070–2090, 2013. <https://doi.org/10.1109/SURV.2013.030713.00020>
- [3] H. Shirazi, B. Bezawada, I. Ray, and C. Anderson, "Adversarial email generation against spam detection models," *J. Inf. Secur. Appl.*, vol. 72, p. 103406, 2023.
- [4] C. Opara, P. Modesti, and L. Golightly, "Evaluating spam filters and stylometric detection of AI-generated phishing emails," *Expert Systems with Applications*, vol. 276, Art. no. 127044, 2025. <https://doi.org/10.1016/j.eswa.2025.127044>
- [5] K. L. Chiew, C. L. Tan, K. S. Wong, K. S. C. Yong, and W. K. Tiong, "A new hybrid ensemble feature selection framework for machine learning-based phishing detection system," *Inf. Sci.*, vol. 484, pp. 153–166, 2024. <https://doi.org/10.1016/j.ins.2019.01.064>
- [6] C. Opara, B. Wei, and Y. Chen, "HTMLPhish: Enabling phishing web page detection by applying deep learning techniques on HTML analysis," *arXiv preprint arXiv:1909.01135*, 2019. <https://doi.org/10.1109/IJCNN48605.2020.9207707>
- [7] N. Q. Do, A. Selamat, H. Fujita, and O. Krejcar, "An integrated model based on deep learning classifiers and pre-trained transformer for phishing URL detection," *Future Generation Computer Systems*, vol. 161, pp. 269–285, 2024. <https://doi.org/10.1016/j.future.2024.06.031>
- [8] N. Abdelhamid, "Multi-label rules for phishing classification," *Appl. Comput. Inf.*, vol. 19, no. 1, pp. 1–16, 2015. <https://doi.org/10.1016/j.aci.2014.07.002>
- [9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *Proc. NAACL*, Minneapolis, MN, USA, pp. 4171–4186, 2019. <https://doi.org/10.18653/v1/N19-1423>
- [10] A. Vaswani et al., "Attention is all you need," *Proc. NeurIPS*, Long Beach, CA, USA, vol. 30, 2017.
- [11] Y. Liu et al., "RoBERTa: A robustly optimized BERT pretraining approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 2, pp. 1012–1025, 2024.
- [12] T. Brown et al., "Language models are few-shot learners," *Proc. NeurIPS*, vol. 33, 2020.
- [13] S. Bubeck et al., "Sparks of artificial general intelligence: Early experiments with GPT-4," *Future Gener. Comput. Syst.*, vol. 152, pp. 221–238, 2024.
- [14] X. Qiu et al., "Pre-trained models for natural language processing: A survey," *Sci. China Technol. Sci.*, vol. 63, no. 10, pp. 1872–1897, 2023. <https://doi.org/10.1007/s11431-020-1647-3>

- [15] Z. Zhang, Y. Han, and Q. Liu, "Fine-tuning large language models for cybersecurity text classification," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 2341–2356, 2024.
- [16] W. Wang et al., "MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," *Proc. NeurIPS*, vol. 33, 2020.
- [17] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Proc. NeurIPS, Long Beach, CA, USA*, vol. 30, 2017.
- [18] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?' Explaining the predictions of any classifier," *Proc. KDD, San Francisco, CA, USA*, pp. 1135–1144, 2016. <https://doi.org/10.1145/2939672.2939778>
- [19] D. Brocki, N. C. Rezk, and S. Krishnamurthy, "Explainable artificial intelligence for cybersecurity: A literature survey," *IEEE Access*, vol. 11, pp. 75–94, 2023.
- [20] R. Faqih, P. Nanda, M. Mohanty, S. Alqahtani, and B. Alrashed, "Towards trustworthy cybersecurity: Reclassifying insider threat detection using SHAP," *Future Generation Computer Systems*, vol. 183, Art. no. 108543, 2026. <https://doi.org/10.1016/j.future.2026.108543>
- [21] H. Liu and P. Guo, "XAI-Phish: Explainable phishing website detection using gradient-based feature attribution," *J. Inf. Secur. Appl.*, vol. 79, p. 103624, 2024.
- [22] R. Guidotti et al., "A survey of methods for explaining black box models," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–42, 2023. <https://doi.org/10.1145/3236009>
- [23] Y. Mirsky and W. Lee, "The creation and detection of deepfakes: A survey," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–41, 2023. <https://doi.org/10.1145/3425780>
- [24] M. Al-Qurishi et al., "A prediction system of sybil attack in social network using deep-regression model," *Future Gener. Comput. Syst.*, vol. 87, pp. 743–753, 2023. <https://doi.org/10.1016/j.future.2017.08.030>
- [25] K. Alrawais, A. Alhothaily, C. Hu, and X. Cheng, "Fog computing for the internet of things: Security and privacy issues," *IEEE Internet Comput.*, vol. 21, no. 2, pp. 34–42, 2023. <https://doi.org/10.1109/MIC.2017.37>
- [26] A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning for internet of things," *Future Gener. Comput. Syst.*, vol. 82, pp. 761–768, 2018. <https://doi.org/10.1016/j.future.2017.08.043>
- [27] N. Moustafa, J. Slay, and G. Creech, "Novel geometric area analysis technique for anomaly detection on large-scale networks," *IEEE Trans. Big Data*, vol. 5, no. 4, pp. 481–494, 2023. <https://doi.org/10.1109/TBDATA.2017.2715166>
- [28] F. Salo, A. B. Nassif, and A. Essex, "Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection," *Comput. Netw.*, vol. 148, pp. 164–175, 2024. <https://doi.org/10.1016/j.comnet.2018.11.010>
- [29] C. Sabetta and M. Bezzi, "Automatic classification of security-relevant commits," *Proc. ICSME*, Madrid, Spain, pp. 579–590, 2023. <https://doi.org/10.1109/ICSME.2018.00058>
- [30] X. Li, F. Zhang, H. Huang, and C. Lü, "BERT-based cybersecurity named entity recognition," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 4101–4114, 2023.
- [31] P. Kaur, R. Kumar, and M. Gupta, "A systematic review on imbalanced data challenges in machine learning," *ACM Comput. Surv.*, vol. 55, no. 13s, pp. 1–36, 2023. <https://doi.org/10.1145/3343440>
- [32] S. Roy, J. Chen, and H. Liu, "Automated spear-phishing mail detection using NLP and graph neural networks," *Expert Syst. Appl.*, vol. 234, p. 121048, 2024.
- [33] J. Soares et al., "Phishing email detection using sentiment analysis and machine learning," *IEEE Access*, vol. 12, pp. 11208–11219, 2024.
- [34] I. Rosenberg et al., "Adversarial machine learning attacks and defenses in cybersecurity," *ACM Comput. Surv.*, vol. 54, no. 5, pp. 1–36, 2023. <https://doi.org/10.1145/3453158>
- [35] B. Luo et al., "Adversarial attacks against the deep learning based phishing detection model," *Proc. IEEE CNS*, Orlando, FL, USA, pp. 1–9, 2023.
- [36] K. Grosse et al., "Adversarial perturbations against deep neural networks for malware classification," *Proc. Euro S&P*, pp. 62–77, 2024.
- [37] Z. Li et al., "VulDeePecker: A deep learning-based system for vulnerability detection," *Proc. NDSS*, San Diego, CA, USA, 2023.
- [38] A. Le, A. Markopoulou, and M. Faloutsos, "PhishDef: URL names say it all," *Proc. IEEE INFOCOM*, Shanghai, China, pp. 191–195, 2024. <https://doi.org/10.1109/INFOCOM.2011.5934995>
- [39] S. Abutair, A. Belghith, and S. Al-Hassani, "Using case-based reasoning for phishing detection," *Procedia Comput. Sci.*, vol. 109, pp. 281–288, 2023. <https://doi.org/10.1016/j.procs.2017.05.352>
- [40] G. Varshney, M. Misra, and P. K. Atrey, "A survey and classification of web phishing detection schemes," *Secur. Commun. Netw.*, vol. 9, no. 18, pp. 6266–6284, 2016. <https://doi.org/10.1002/sec.1674>
- [41] M. Korkmaz, O. K. Sahingoz, and B. Diri, "Detection of phishing websites by using machine learning-based URL analysis," *Proc. ICCCNT*, Bangaluru, India, pp. 1–6, 2023. <https://doi.org/10.1109/ICCCNT49239.2020.9225561>
- [42] H. Lwin, N. Maneerat, and R. Funabiki, "Email spam and phishing datasets: A comprehensive evaluation benchmark," *IEEE Access*, vol. 12, pp. 19876–19891, 2024.
- [43] M. Khonji, Y. Iraqi, and A. Jones, "Phishing detection: A literature survey," *IEEE Commun. Surv. Tuts.*, vol. 15, no. 4, pp. 2091–2121, 2013. <https://doi.org/10.1109/SURV.2013.032213.00009>
- [44] A. Dutta and S. Bhattacharjee, "SpamGuard: An enhanced deep learning model for spam and phishing email classification using Enron and SpamAssassin corpora," *Future Gener. Comput. Syst.*, vol. 151, pp. 88–102, 2024.
- [45] R. Vinayakumar et al., "Robust intelligent malware detection using deep learning," *IEEE Access*, vol. 7, pp. 46717–46738, 2023. <https://doi.org/10.1109/ACCESS.2019.2906934>
- [46] S. H. Andrianarisoa, H. M. Ravelonjara, G. Suddul, R. Foogooa, S. Armoogum, and D. Sookarah, "A Deep Learning Approach to Fake News Classification Using LSTM," *Vokasi UNESA Bull. Eng. Technol. Appl. Sci.*, vol. 2, no. 3, pp. 593–601, 2025. <https://doi.org/10.26740/vubeta.v2i3.39360>

- [47] M. A. Uddin, M. Mahiuddin, and I. H. Sarker, "An explainable transformer-based model for phishing email detection: A large language model approach," *Comput. Netw.*, vol. 277, Art. no. 112061, 2026. <https://doi.org/10.1016/j.comnet.2026.112061>
- [48] T. Koide, N. Fukushi, H. Nakano, and D. Chiba, "ChatSpamDetector: Leveraging large language models for effective phishing email detection," *Proc. 20th EAI Int. Conf. Security and Privacy in Communication Networks (SecureComm 2024)*, Dubai, United Arab Emirates, 2024. https://doi.org/10.1007/978-3-031-94455-0_14
- [49] C. S. Eze and L. Shamir, "Analysis and prevention of AI-based phishing email attacks," *Electronics*, vol. 13, no. 10, Art. no. 1839, 2024. <https://doi.org/10.3390/electronics13101839>
- [50] A. Al-Subaiey, M. Al-Thani, N. A. Alam, K. F. Antora, A. Khandakar, and S. M. A. U. Zaman, "Novel interpretable and robust web-based AI platform for phishing email detection," *Comput. Electr. Eng.*, vol. 120, Art. no. 109625, 2024. <https://doi.org/10.1016/j.compeleceng.2024.109625>