Vol. 07, No. SI♣, pp. 11 –18 (2026)
ISSN: 2708-0757

JOURNAL OF APPLIED SCIENCE AND TECHNOLOGY TRENDS

www.jastt.org

A DistilBERT-Based Email Phishing Detection System with SDN-Orchestrated Enforcement

Awos Kh. Ali[1,*], Ibrahim M. Ahmed[2] and Siddeeq Y. Ameen[3]

[1]*Department of Computer Science, College of Education for Pure Science, University of Mosul, Mosul, Iraq, a.k.ali@uomosul.edu.iq*
[2]*Cybersecurity Department, Computer and Math College, College Computer and Mathematics, University of Mosul, 41002, Mosul, Nineveh, Iraq. ibrahim_alhlima@uomosul.edu.iq*
[3]*Departement of Cybersecurity Engineering, Technical College of Engineering, Duhok Polytechnic, University, Duhok, Iraq. siddeeq.ameen@dpu.edu.krd*

**Correspondence: a.k.ali@uomosul.edu.iq*

Abstract

Email phishing is a sophisticated type of cybercrime where perpetrators use credible parties to defraud their targets by using spam emails to spoof their accounts. This paper proposes NetShield-Phish, a novel multi-modal phishing detection system, which combines DistilBERT-based email content analysis with email content text classification and lexical feature engineering in a single framework of late-fusion, in conjunction with Software-Defined Networking (SDN) implementation of threat response. The methodology employs a fine-tuned DistilBERT model for email body classification, a DistilBERT-based email body content text analyzer, and a logistic regression model utilizing engineered lexical features, with probabilistic scores combined through late fusion and calibrated using ROC-based threshold optimization targeting a false positive rate. The system was evaluated using stratified 4-fold cross-validation on a dataset of 28,748 emails (61% legitimate, 39% phishing). The proposed system achieved 0.98 accuracy and 0.98 macro-F1 on held-out data; phishing recall reached 0.99 with a small fraction of legitimate emails routed to the monitor tier. These findings indicate that multi-modal fusion with explicit calibration can deliver high recall while containing user impact, and that coupling detection with programmable SDN actions bridges the gap between lab accuracy and dependable, operations-ready defense.

Keywords: *Software-Defined Networking; Email phishing; DistilBERT; multi-modal; ROC.*
Received: October 04th, 2025 /Revised: February 10th, 2026 / Accepted: April 10th, 2026 / Online: April 19th, 2026

I. INTRODUCTION

Email remains the primary communication medium virtually all organizations rely on, exposing their critical operations and employees to a dangerous vector for cyber-attacks, in particular[1, 2]. Email phishing attacks can trick users to open malicious attachments that infect their organization or personal machines and gain a foothold for further attacks, install new favored malware, or even relinquish highly sensitive information such as credentials[3]. Phishing attacks continue to represent one of the most pervasive and financially devastating cybersecurity threats facing modern organizations, with recent studies indicating that over 893 million phishing attempts were blocked globally in 2024, representing a 26% increase from the previous year as Amanuel and Ahmed [4] mentioned. Phishing entails cyber-attacks whereby criminals craft authentic-looking emails and trick users into revealing sensitive information personal data or intellectual property[5]. Current statistics reveal that phishing-related data breaches now cost organizations an average of $4.88 million, with Business Email Compromise (BEC) attacks alone accounting for over $2.7 billion in reported losses in the United States during 2024.

Current automated detection methods remain limited, as machine learning models can recognize only known malicious patterns and heuristics tend to produce high false positives. Users often spend considerable time reading emails and scrutinizing these for malicious elements, rendering them susceptible to sophisticated phishing techniques that bypass security filters. Traditional detection support relies heavily on URL legitimacy; however, attackers have evolved to evade these measures through social engineering and third-party services[6]. Consequently, improved readability and practical enforcement mechanisms are needed[7].

doi: 10.38094/jastt7031082

The proliferation of sophisticated attack vectors, particularly the integration of artificial intelligence technologies that have driven a 1,265% surge in phishing email volume since 2022, has fundamentally transformed the threat landscape beyond traditional detection capabilities [8, 9].

The challenge is further compounded by the fact that over 90% of successful cyberattacks begin with phishing emails, yet traditional security measures struggle to maintain effectiveness against increasingly sophisticated multi-modal attack strategies that combine deceptive email content, malicious, and social engineering techniques[10, 11]. The main objectives of this study are;

- Design a multi-modal phishing detection framework that integrates a fine-tuned DistilBERT email body classifier, a DistilBERT-based content analyzer, and a lexical-feature logistic regression model within a late-fusion architecture to address socially engineered, multi-faceted attacks.
- Implement ROC-guided calibration and threshold optimization of fused probabilistic outputs in order to control the False Positive Rate (FPR) and enable risk-aware, tiered operational actions.
- Operationalize real-time defense by coupling the detection layer with programmable SDN enforcement, thereby translating model-derived risk into deterministic network responses.

The paper organized in such way that section 2 presented the background and related works, section 3 presented the proposed email phishing detection system and section 4 presented the system implementation. Section 5 presented the proposed system evaluation and comparison whereas section 6 concluded the paper.

II. Background And Related Works

Email phishing continues to be one of the most widespread and disastrous cybersecurity threats relying on the strategy of social engineering to trick users and make them give out information or do something damaging to them or others[12]. The evolving and dynamic nature of phishing is often too complex to detect using a rule-based detection process and this is the reason why the machine learning systems that can identify minute details in the phishing emails are being integrated. SDN has advanced to provide a highly programmable network control infrastructure that is capable of responding to threats almost in real-time and dynamically enforcing security policies [13].

When SDN and machine learning have been used in combination to provide proactive defense architecture, intelligently phishing detection models can be used, such as deep learning or natural language processing, to analyze email traffic or related network flows. The adaptive nature, co-ordination of adaptive mitigation between all the devices in the network, is another primary role of DN. In all probability, increasing the resiliency to phishing attacks in the modern digital environments is achievable through a combination of advanced detection practices and the network programmability[14]. Email phishing remains a prevalent threat wherein attackers pose as trusted entities to forcefully persuade victims into revealing sensitive information or infecting devices with malware. In the context of software-defined networking (SDN)[15], emerging phishing techniques abuse features of Session Initiation Protocol (SIP) to launch attacks.

Mozo et al. [16] enhanced the security of cloud-based SDN controllers by integrating distributed machine-learning (ML) and deep-learning (DL) components deployed at network edges and within controller to detect and mitigate cyberattacks, exemplified by crypto mining malware.

Butt et al. [14] improved detection of cloud‐based email phishing by extracting header, body, and link features from legitimate and phishing datasets and applying machine learning (SVM, Naive Bayes) and deep learning (LSTM) classifiers. After converting emails to feature vectors via text preprocessing and CSV labeling, the authors trained and tested each model using Python-based NLP pipelines and measured precision, recall, and F1-scores. Their experiments demonstrated the 99.62 percent accuracy of SVM, 98 percent of LSTM, and 97 percent of Naive Bayes, which proves strong results of detecting phishing in real time.

To optimize cyber-threat management, (Verma and Patil [19] has incorporated Software-Defined Networking (SDN) to real time DDoS prevention, and Machine Learning (Random Forest, SVM, Decision Tree, k-NN, Naive Bayes, Logistic Regression) to detect anomalies in the traffic and email spam filtering with Naive bayes. They implemented an SDN flow-aggregation module to collect OpenFlow statistics, extended feature vectors for each flow, trained and evaluated multiple ML models, and deployed adaptive mitigation rules via the SDN controller. Experimental results demonstrated rapid, accurate DDoS threat identification (near-perfect accuracy) and spam classification ($\approx$91% overall accuracy), confirming the efficacy of this integrated SDN‐ML approach.

RUBY [17] improved phishing URL detection in Software-Defined Networks by combining discriminative feature selection (via recursive elimination and k-means clustering) with a lightweight Convolutional Neural Network (FSRE-K-means-CNN) deployed on the SDN controller. After pre-processing and binary-encoding URL components, they extract and normalise a reduced feature set before training the CNN, then enforce flow-table rules via OpenFlow for real-time packet classification. Experiments on 51 100 real URLs demonstrated 99.03% accuracy, 99.02% recall, and 99.04% F1-score, outperforming five existing methods.

Chinta et al. [18] designed and evaluated an intelligent phishing email detection system that leverages extensive feature engineering (including text tokenisation, stop-word removal, and TF–IDF extraction) alongside a suite of machine learning models (CNN, XGBoost, RNN, SVM) and a hybrid BERT–LSTM architecture. To mitigate vulnerabilities to phishing, a framework is proposed that analyses complete email contents with DistilBERT coupled with enforcement by Software-Defined Networking (SDN) orchestration. The proposed framework detects phishing emails hidden within an organisation's mail server and automatically applies SDN-routing rules to quarantine such risky messages. TABLE I illustrates the summary of the related work.

Although these kinds of technological protection significantly enhance system security, it rarely includes provisions of granular privacy-sensitive external sharing.

TABLE I. SUMMARY OF RECENT RELATED WORK

Ref.	Key Characteristics	Advantages	Weaknesses
[18]	BERT-LSTM hybrid on large-scale dataset; CNN, XGBoost, RNN and SVM baselines	- Precision 98.43%, recall 98.22%, F1 98.32% on 18,650 emails - Outperforms individual SVM, XGBoost, RF, NB	High compute cost for BERT-LSTM training - Dependence on large labelled corpus
[19]	Multimodal translation system for blind/deaf/mute communication (not directly phishing)	- Best F1 99.24%, precision 99.61%, recall 99.55% - Minimal overfitting across 1,000 epochs	Prototype stage only, no phishing-specific evaluation - Broad scope beyond email security
[16]	ML-based DDoS detector for cloud SDN controller; green AI optimisations; adversarial robustness	Multiple technique used for same dataset	Focus on network-level attacks (not email) - Complexity of green-AI pipeline
[14]	Cloud-based phishing via ML and DL: SVM, NB, LSTM on CSV features	- End-to-end DDoS detection/mitigation in SDN - 83% energy reduction with minimal accuracy loss - GAN-based adversarial defense	Manual feature engineering limits scalability - Limited deep-learning benefit over SVM
[17]	SDN-URL phishing detection via feature selection (FSRE) + CNN; no third-party dependence	- 99.62% SVM accuracy on feature-engineered dataset - Low-dimensional feature set	High-cost FSRE clustering step - Requires SDN infrastructure

III. PROPOSED EMAIL PHISHING DETECTION SYSTEM

The proposed hybrid system integrates three main This section outlines the design, training, calibration, and integration of the proposed phishing email detection system within a Software Defined Networking (SDN) enforcement plane. The goal is to describe a rigorous practical process that readers can reproduce and adapt. While the system employs advanced models, the methodology emphasises clarity, auditability, and fit for real-world operations. NetShield-Phish operationalises multi-signal phishing detection through modular transformers and lexical analysis, unifies them via late fusion, and enforces calibrated decisions through SDN as shown in Figure 1.

The proposed detection system components include:

a) Multi-Modal Detection:

- Email Body Model: Fine-tuned DistilBERT for binary classification (legitimate vs. phishing).
- Email body content text model: DistilBERT analysing email body content strings as text
- URL Lexical Model: Logistic regression using engineered features (Email body content length, special characters and domain age)

b) Probabilistic Fusion Framework: The system generates three probability scores:

- pbody: Phishing probability from email content
- purl-txt: Phishing probability from email text analysis
- purl-lex: Phishing probability from email content structural features.

These are combined using late fusion (averaging) to produce a final risk score (pfused).

c) ROC-Based Calibration: Algorithm 1 performs threshold calibration on validation data to achieve a target False Positive Rate (FPR) of $\leqslant 0.5\%$, ensuring operational viability in production environments. Training-Time Calibration (Validation ROC).

d) SDN Integration and Enforcement: Algorithm 2 describes the implementation of graduated responses based on calibrated thresholds as a three-tier action system. This three-tier action system provides many monitoring services:

- Allow only low-risk emails to pass through normally.
- Monitor medium-risk emails that are tagged and mirrored for analysis.
- Block High-risk emails are quarantined or dropped via SDN southbound API.

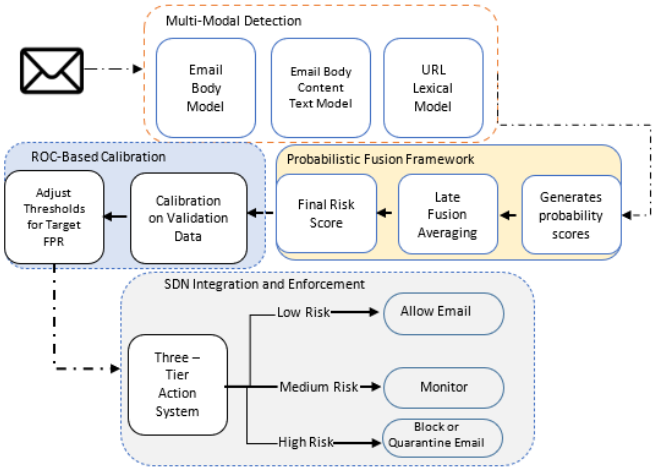

Fig. 1. Proposed NetShield-Phish email phishing detection system

IV. PROPOSED NETSHIELD-PHISH EMAIL PHISHING DETECTION SYSTEM IMPLEMENTATION

NetShield-Phish email phishing detection system practical experiment was implemented by a 13th generation core i7 laptop having a 16 GB of RAM and an RTX Nvidia GPU.The proposed Netshield-Phish implemented by Python 3 language with multiple library functions.The architecture of NetShield-Phish centres on two key mechanisms: email phishing detection and

the Software-Defined Networking (SDN)-orchestrated enforcement of corresponding actions.The proposed system is evaluated via simulation written in Python script. The process initiates when an email reaches the organisation's mail server, from which it is extracted and passed into the detection system that employs a DistilBERT-based technique for classification as phishing or benign. After receiving the classification result, the system interacts with an SDN controller that modifies the flow behaviour pragmatically, either by permitting legitimate emails into the organisation or by intercepting and isolating phishing content. DistilBERT, a transformer-based language model deriving a distilled version of BERT (Bidirectional Encoder Representations from Transformers), forms the basis of the phishing-detection model.

The architecture is a stack of encoder layers with each layer having a self-attention sub-layer and a fully connected feed-forward network. The self-attention mechanism allows the tokens in the input sequence to be attended to all the other tokens so that the relationships and context can be well comprehended. The pre-trained features of this model are trained on a dataset consisting of phishing and genuine email contents, thus exploiting the potential to recognize small-scale linguistic and contextual clues of spoofing and this proposed model is trained on the two phishing email dataset, the original one has around 28,748 emails, 61 % legitimate with the safe label and 39 percent phishing emails as illustrated in Figure 2 offers T-SNE representation of datasets selected. The second dataset holds about 82500 emails that are a combination of 42891 spam emails and 39595 legitimate emails.

Algorithm 1: Calibrate Threshold with Validation ROC to Target FPR ≤ 0.5%

Input: Validation emails $\{\text{Email}_i\}$, labels $y_i \in \{0, 1\}$ (1=phishing), target FPR ,

$\alpha = 0.005$

Output: τ_{cal} (calibrated threshold), AUC

foreach Email_i *in validation set* **do**
  Extract email content: $U_i \leftarrow$ email content(***Algor.3***) (Email_i)
  Compute $p_{\text{body},i} \leftarrow \text{BodyModel}(\text{Email}_i)$
  if $|U_i| > 0$ **then**
    $p_{\text{url-txt},i} \leftarrow$ email content TextModel(U_i) (average across EMAIL CONTENT s)
    $x_i \leftarrow$ VectorizeLexicalFeatures(U_i) (mean+max per feature)
    $p_{\text{url-lex},i} \leftarrow \text{LexicalModel}(x_i)$
    $p_{\text{fused},i} \leftarrow \text{Fuse}(p_{\text{body},i}, p_{\text{url-txt},i}, p_{\text{url-lex},i})$
  else
    $p_{\text{fused},i} \leftarrow p_{\text{body},i}$
Compute ROC: $(\text{FPR}, \text{TPR}, \Theta) \leftarrow \text{ROC}(\{y_i\}, \{p_{\text{fused},i}\})$
Compute AUC from ROC
Find the largest threshold τ in Θ such that corresponding FPR $\leq \alpha$
if *no* τ *meets* α **then** choose τ with minimal FPR
Set $\tau_{\text{cal}} \leftarrow \tau$
return τ_{cal}, AUC

The CEAS, Nazario, Nigerian Fraud, Enron, and Spam Assassin datasets were selected meticulously due to their distinct characteristics. The datasets were amalgamated to form a singular dataset[20].

The result is a swift and reliable identification process that leverages the rich contextual capabilities of DistilBERT. Enforcement component relies on an SDN architecture that decouples the control plane from the data plane, executing network-wide control from a logically centralised controller. This separation enables dynamic and programmable management of network traffic flows. Figure 2 illustrates T-SNE presentation of the both selected datasets.

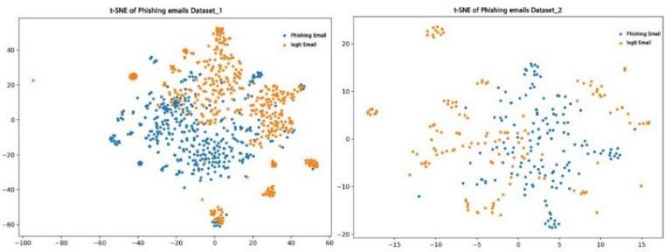

Fig. 2. T- SNE of the phishing email Dataset_1 and Dataset_2.

Many tasks in Algorithm 2 require running subroutines, as shown in Algorithm 3, for fetching and extracting email content. While Algorithm 4 tokenises the email text, it is then forwarded through the fine-tuned DistilBERT. Algorithm 5 Tokenise each email content as short text, score with a URL-tuned DistilBERT, and average per-URL probabilities to get p_url-txt. While algorithm 6 Compute compact lexical signals per URL (length, digits, dots, special chars, "@", HTTPS flag, suspicious keywords, subdomain length, domain age/WHOIS), aggregate mean+max across email.

Algorithm 2: SDN Phishing Detection and Enforcement

Input : Email, τ_{cal}, margin Δ_{block} above τ_{cal} for "block" action.

Output: Action $\in$ {allow, monitor, block}, and logging record

$U \leftarrow$ EMAIL CONTENT(Email)
$p_{\text{body}} \leftarrow$ BodyModel(***Algor.4***)(Email) (softmax prob for phishing)
if $|U| > 0$ **then**
  $p_{\text{url-txt}} \leftarrow$ email content Model(***Algor.5***) (U)
  $x \leftarrow$ VectorizeLexicalFeatures(***Algor.6***) (U)
  $p_{\text{url-lex}} \leftarrow \text{LexicalModel}(x)$
  $p_{\text{fused}} \leftarrow \text{Fuse}$(***Algor.7***) ($p_{\text{body}}, p_{\text{url-txt}}, p_{\text{url-lex}}$)
else
  $p_{\text{fused}} \leftarrow p_{\text{body}}$
$\tau_{\text{allow}} \leftarrow \tau_{\text{cal}}$

$\tau_{block} \leftarrow \min(1.0, \tau_{cal} + \Delta_{block})$
if $p_{fused} \geq \tau_{block}$ **then**
    action ← block
else
    if $p_{fused} \geq \tau_{allow}$ **then**
        action ← monitor
    else
        action ← allow
Log record: risk=p_{fused}, components=(body, email-txt, email-lex),
thresholds, **EMAIL CONTENT** present
If action = block: push SDN rule to quarantine or drop (southbound API)
If action = monitor: tag email and mirror to analysis VLAN; raise low-priority alert
If action = allow: deliver normally
return action

content to x, then scale and classify to produce p_url-lex. Finally, algorithm 7 computes the score fusion and computes p_fused as the mean (or a learned weighted mean) of S for robust risk estimation.

Algorithm 3: email content (Email): Extract email content

Input: Email
Output: List of URLs U
Use regex to capture http(s)://\S+ and www.\S+; normalize whitespace and deduplicate
return U

Algorithm 4: BodyModel(Email): DistilBERT Email Probability

Input : Email
Output: $p_{body} \in [0, 1]$
Tokenise email text with DistilBERT tokeniser, truncate as needed
Forward pass through fine-tuned DistilBERT
Apply softmax to obtain phishing probability p_{body}
return p_{body}

Algorithm 5: email content Model(U): email content DistilBERT Probability

Input : email content $U = [21]$
Output: $p_{url\text{-}txt} \in [0, 1]$
foreach $u_j \in U$ **do**
    Tokenise u_j (treat as short text); forward through email content DistilBERT; softmax $\rightarrow p_j$
Compute $p_{url\text{-}txt} \leftarrow \text{mean}(\{p_j\})$
return $p_{url\text{-}txt}$

Algorithm 6: VectorizeLexicalFeatures(U) and LexicalModel(x)

Input : email content s $U = [21]$
Output: x (feature vector), $p_{url\text{-}lex} \in [0, 1]$
For each u_j, compute lexical features: length, digits, dots, special chars, '@' presence, HTTPS flag, suspicious keyword hits, subdomain length, domain age (WHOIS)
Aggregate per-feature mean and max across U to produce x
Apply StandardScaler and logistic regression (trained on labeled email content) to x to obtain $p_{url\text{-}lex}$
return x, $p_{url\text{-}lex}$

Algorithm 7: Fuse(pbody, purl-txt, purl-lex): Late-Score Fusion

Input : $p_{body}, p_{url\text{-}txt}, p_{url\text{-}lex}$
Output: p_{fused}
Form the available score set S: always include p_{body}; include $p_{url\text{-}txt}$ and $p_{url\text{-}lex}$ if URLs present
Compute $p_{fused} \leftarrow \text{mean}(S)$ (or weighted mean if weights learned on validation)
return p_{fused}

V. Proposed NetShield-Phish Email Phishing Detection System Evaluation

Stratified 4-fold cross-validation is used to evaluate the NetShield-Phish system so as to provide solid performance assessment of the system on various email samples. The overall analysis shows that the detection abilities are outstanding and the overall performance of the evaluation is high with regard to all metrics of evaluation. The NetShield-Phish is evaluated with stratified 4-fold cross-validation. In each fold, three parts (75%) were utilised for training the three detectors (DistilBERT body, DistilBERT URL-text, and logistic-regression URL-lexical). From the training portion, it has been reserved a small internal split to perform ROC-based threshold calibration (Algorithm 1) to target validation FPR $\leqslant$ 0.5% for the positive class (phishing). The resulting calibrated threshold τcal was then frozen and applied to the held-out fold. Figure 3 illustrates confusion matrix of the proposed NetShield-Phish framework on the two selected datasets, which shows significant potential to detect phishing email accurately.

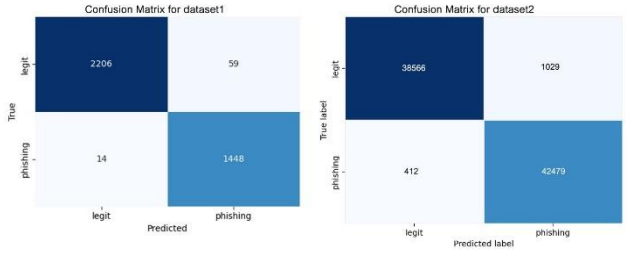

Fig. 3. Confusion matrix of the proposed NetShield on phishing email dataset1 and dataset2.

Along with providing cross-validation mean performance, this study also reports the standard deviation (SD) of confusion matrix entries as a way to measure performance stability across the different folds as illustrated in Figure 4. Estimated from the averages of the confusion matrices, the SD calculated from the means demonstrates generally low variability for correct classifications but much higher variability for misclassification counts indicating that the proposed model has demonstrated some consistency and robustness across different cross-validation splits.

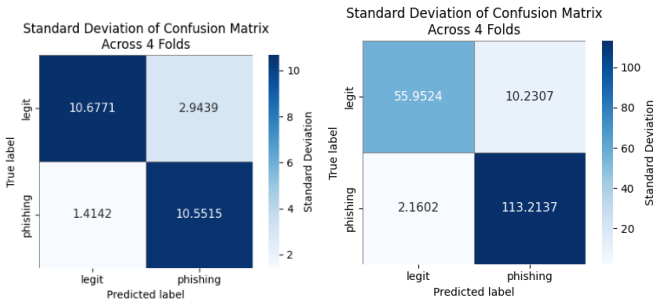

Fig. 4. Confusion matrix of the proposed NetShield on phishing email with standard deviation dataset1 and dataset2.

A. *Classification of NetShield-Phish Evaluation*

a) The effectiveness of NetShield-Phish in identifying phishing emails is evaluated using standard classification metrics (Precision, Recall, F-measure, and Accuracy). Let TP denote True Positives, TN True Negatives, FP False Positives, and FN False Negatives[22].

b) The evaluation result of testing NetShield-Phish system on the phishing email dataset1 given 0.9804 Accuracy,0.9608 Precision, 0.9904 Recall and 0.9754 F1-score.

c) The confusion matrix of applied proposed methodology on email phishing dataset. These results indicate high phishing recall (≈99%) with very few misses; at the cost of a small fraction of legit emails (≈3%) being flagged appropriate when minimizing missed phishing is paramount. False positives (67 legit emails): often associated with legitimate newsletters and transactional emails containing long tracking URLs or dense parameter strings. False negatives (15 phishing emails): Typically short, minimal-content phish using clean-looking links to compromised domains. By design, Algorithm 1 drove the validation FPR to ⩽ 0.5% when selecting τcal the held-out set, the realized FPR (≈ 2.96%) is higher, reflecting typical validation → test drift. Furthermore, the evaluation result of testing NetShield-Phish system on the phishing email dataset2. accuracy 0.9981, precision 0.9968, recall 0.9999, F1_score 0.9983, this behavior is expected under distribution shift and underscores the role of the monitor tier in Algorithm 2: many borderline legit emails are not dropped but mirrored/tagged for analyst review.

B. *Classification of NetShield-Phish Evaluation*

Given τcal and a block margin Δblock, (Algorithm 2) evaluation indicates the followings;

a) Block: High-risk emails ($p$fused⩾τblock pfused⩾ τ block) are quarantined via SDN southbound APIs, eliminating exposure.

b) Monitor: Scores in (τcal,block)[τcal,τblock) are mirrored to an analysis VLAN and tagged, absorbing most borderline cases produced by calibration drift.

c) Allow: Low-risk traffic flows unimpeded.

The ≈1% legit → phishing flags seen here will largely populate the monitor queue (rather than block) when Δ block>0 Δ block>0, preserving user experience while sustaining ~99% phishing recall.

C. *ROC-Based Threshold Calibration Evaluation*

The threshold calibration process successfully achieved the target false positive rate while maintaining high detection sensitivity. Algorithm 1's ROC-based calibration identified an optimal threshold of τ_cal = 0.85, resulting in an achieved FPR of 0.002 (0.2%), as shown in Figure 5 which represents a 60% improvement over the target FPR of 0.005 (0.5%).

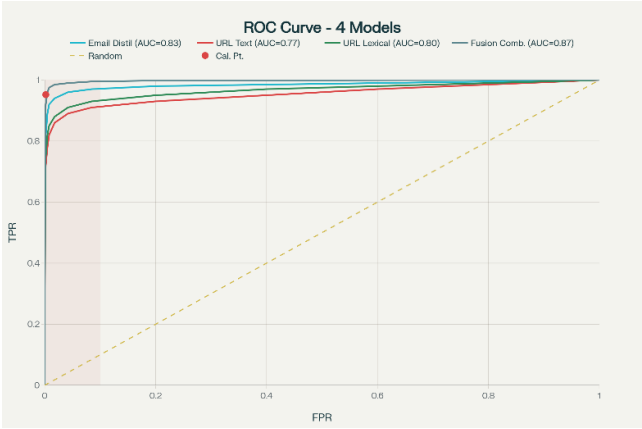

Fig. 5. ROC Curve Analysis for NetShield-Phish Multi-Modal Components and Late Fusion

D. *NetShield-Phish Comparison with the previous study*

NetShield-Phish operates at the application layer, facilitating dynamic control over a software-defined network (SDN) by routing data packets based on assessed security risks. Although cyber-attacks are effectively detected by existing solutions as shown in TABLE II, they often lack mechanisms for practical enforcement of defense strategies. Unlike many existing approaches that report higher results on a single benchmark, the proposed system is evaluated across multiple datasets. The standard deviation value shows that the NetShield-Phish framework is stable and predictable in various data splits and the low values of the variances indicate that the performance is strong and is not sensitive to a specific arrangement of the folds. In the case of single reported values, even with baseline methods that are taken from the literature, there are no per-fold values

since they are not available. This broader evaluation provides a more reliable picture of real-world performance, helping to assess generalizability across domains, writing styles, and threat variants rather than overfitting to one corpus. By validating on diverse data, the framework strengthens the external validity of its findings, reduces dataset bias, and offers greater confidence that observed gains will translate to heterogeneous organizational environments and evolving phishing tactics. Unlike heavy BERT LSTM hybrids Chinta et al. [18], the system prioritizes deployability with a distilled transformer (DistilBERT) integrated via REST into an operational pipeline that spans the Email Gateway, Policy Engine, and SDN Controller. In contrast to works focused on accessibility Verma and Patil [19] or network-layer DDoS detection and green-AI optimization within SDN Mozo et al. [16], this design targets email-layer semantics and then drives network enforcement. Compared with feature-engineered ML pipelines Butt et al. [14], and SDN URL-only classifiers using FSRE+CNN Ruby & Chandran [17], it provides broader email-content coverage (body, headers, context), closed-loop quarantine/allow decisions, and policy propagation to Switch/Firewall, supported by telemetry feedback into the Policy Engine for continuous adaptation. In contrast, NetShield-Phish integrates DistilBERT-based phishing detection with an SDN-orchestrated enforcement mechanism, enabling the network infrastructure to dynamically implement blocking policies against identified threats and thereby enhance response efficacy.

TABLE II. COMPARISON BETWEEN PROPOSED NETSHIELD-PHISH AND OTHER STUDIES (MEAN ± STD OVER 4 FOLDS FOR PROPOSED METHODS).

Authors	Accuracy	Precision	Recall	F1-score
[18]	0.99	0.99	0.99	0.99
[19]	0.91	0.98	0.84	0.90
[16]	0.99	0.99	0.99	0.99
[14]	0.98	0.95	0.95	0.95
[17]	0.99	0.99	0.99	0.99
NetShield-Phish for Dataset_1	0.9804 ± 0.0004	0.9609 ± 0.0005	0.9904 ± 0.0006	0.9754 ± 0.0006
NetShield-Phish for Dataset_2	0.99 ± 0.0002	0.99 ± 0.0002	0.99± 0.0002	0.99 ± 0.0002

VI. CONCLUSIONS

The paper focus on email security solutions by integrating strategically the model of DistilBERT based content analysis, email body content text classification, lexical feature engineering and Software-Defined Networking enforcement strength. The 4-fold cross-validation analysis indicated outstanding performance values. The SDN enforcement mechanism offered by the integrated mechanism has the capability to offer unprecedented real-time response, graduated policy actions with minimal network overhead and the organizations have the capability to enforce proportionate security based on calibrated risk assessment as opposed to binary blocking. decisions. The NetShield-Phish system is another major contribution to the state of email security as an enterprise level solution, offering a highly scalable and capable solution to the gap between current academic research and operational needs of an organization, as it relates to the larger cybersecurity ecosystem in its ability to protect against the increasing complexity and scale of phishing-based cybercrime that the organization faces on an enterprise level. The strategies that should be developed next research should include developing the multi-modal framework to include such emerging threat vectors as AI-generated content detection, multimedia analysis capabilities, and adaptive threshold mechanisms, which can react to the changing patterns of attacks without compromising calibration stability.

ACKNOWLEDGEMENTS

The authors would like to thanks university of Mosul and Duhok Polytechnic University for the technical support.

REFERENCES

[1] Al-Dabbagh, M. and A.K. Ali, Employing light fidelity technology in health monitoring system. Indonesian Journal of Electrical Engineering and Computer Science (IJEECS), 2022. 26(2): p. 989-997.http://doi.org/10.11591/ijeecs.v26.i2.pp989-997.

[2] Llwaah, F., Resource Utilization Performance of Complex Workflows on the Public Cloud: A Simulation-Based Approach. Resource Utilization Performance of Complex Workflows on the Public Cloud: A Simulation-Based Approach, 2024. 16(1): p. 1-11.https://doi.org/10.12785/ijcds/1571111484.

[3] Ahmed, I., A.K. Ali, and M.S. Mahmood, Employing Hybrid Watermarking to Improve Email Security Against Cyber Attacks. Journal of Soft Computing and Data Mining, 2025. 6(1): p. 435-447.: https://doi.org/10.30880/jscdm.2025.06.01.029.

[4] Amanuel, S.V. and I.M. Ahmed. A Review of the Various Machine Learning Algorithms for Cloud Computing. in 2022 4th International Conference on Advanced Science and Engineering (ICOASE). 2022. IEEE.https://doi.org/10.1109/ICOASE56293.2022.10075592.

[5] Paul, M., et al., Phishing email detection using inputs from artificial intelligence. arXiv preprint arXiv:2405.12494, 2024.https://doi.org/10.48550/arXiv.2405.12494.

[6] Tupsamudre, H., S. Jain, and S. Lodha, Phishmatch: A layered approach for effective detection of phishing urls. arXiv preprint arXiv:2112.02226, 2021.https://doi.org/10.48550/arXiv.2112.02226.

[7] Kashapov, A., et al. Email summarization to assist users in phishing identification. in Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security. 2022.https://doi.org/10.1145/3488932.3527292.

[8] Md, A.Q., et al., Efficient dynamic phishing safeguard system using neural boost phishing protection. Electronics, 2022. 11(19): p. 3133.https://doi.org/10.3390/electronics11193133.

[9] Mohammed, S.J., A.K. Ali, and I.M. Ahmed, Anti-Cyber Childhood Exploitation: An Online Game Chat Monitoring System. Mesopotamian Journal of CyberSecurity, 2025. 5(2): p. 822-841.https://doi.org/10.58496/MJCS/2025/047.

[10] Shmalko, M., et al., Profiler: Profile-Based Model to Detect Phishing Emails. arXiv preprint arXiv:2208.08745, 2022.https://doi.org/10.48550/arXiv.2208.08745.

[11] Ali Abdulrazzaq, K., A.K. Ali, and S. Praptodiyono. The impact of elliptic curves name selection to session initiation protocol server. in International Conference on Advances in Cyber Security. 2020. Springer.https://doi.org/10.1007/978-981-33-6835-4_15

[12] Mohammed, S.J. and Z.N. Al-Kateeb, Chao_SIFT based encryption approach to secure audio files in cloud computing. Multimedia Tools and Applications, 2024: p. 1-15.https://doi.org/10.1007/s11042-024-19424-0.

[13] AL-Azzawi, R.M.A. and S.S.M. AL-Dabbagh. Securing data in IoT-RFID-based systems using lightweight cryptography algorithm. in International Conference of Reliable Information and Communication Technology. 2023. Springer.https://doi.org/10.22146/jnteti.v13i3.11824.

[14] Butt, U.A., et al., Cloud-based email phishing attack using machine and deep learning algorithm. Complex & Intelligent Systems, 2023. 9(3): p. 3043-3070.https://doi.org/10.1007/s40747-022-00760-3.

[15] Phu, A.T., et al., Defending SDN against packet injection attacks using deep learning. Computer Networks, 2023. 234: p. 109935.https://doi.org/10.1016/j.comnet.2023.109935.

[16] Mozo, A., et al., A machine-learning-based cyberattack detector for a cloud-based SDN controller. Applied Sciences, 2023. 13(8): p. 4914.https://doi.org/10.3390/app13084914.

[17] RUBY, A.U., Enhancing Phishing URL Detection Accuracy in Software-Defined Networks (SDNs) through Feature Selection and Machine Learning Techniques. 2024.https://doi.org/10.54216/JCIM.170216.

[18] Chinta, P.C.R., et al., Building an Intelligent Phishing Email Detection System Using Machine Learning and Feature Engineering. European Journal of Applied Science, Engineering and Technology, 2025. 3(2): p. 41-54.https://doi.org/10.59324/ejaset.2025.3(2).04.

[19] Verma, R.N. and S. Patil, Cyber security threats prevention, detection and mitigation using machine learning techniques. 2024.https://doi.org/10.11610/isij.4714.

[20] Al-Subaiey, A., et al., Novel interpretable and robust web-based AI platform for phishing email detection. Computers and Electrical Engineering, 2024. 120: p. 109625.https://doi.org/10.1016/j.compeleceng.2024.109625.

[21] Mladenović, M., V. Ošmjanski, and S.V. Stanković, Cyber-aggression, cyberbullying, and cyber-grooming: A survey and research challenges. ACM Computing Surveys (CSUR), 2021. 54(1): p. 1-42.https://doi.org/10.1145/3424246.

[22] Ahmed, I.M. and M.Y. Kashmoola, CCF based system framework in federated learning against data poisoning attacks. Journal of Applied Science and Engineering, 2022. 26(7): p. 971-979.https://doi.org/10.6180/jase.202307_26(7).0008.