Hybrid Random Forest and Transformer–RBFN Framework for Phishing Email with URL Detection

Van Chung Nguyen[1] , and Bill Doherty[2]

Abstract— **Phishing remains a dominant cyber threat that targets users through deceptive web links and fraudulent emails. Traditional detection methods are insufficient for identifying complex fraudulent email attacks. In this project, we propose a hybrid framework for detecting malicious content by integrating two specialized machine learning models: (i) a Random Forest classifier trained on TF-IDF and handcrafted URL features for malicious link detection, and (ii) a Transformer-based embedding model combined with a Radial Basis Function Network (RBFN) for phishing email detection. The framework is capable of classifying both URLs and emails under a unified interface. (iii) We further combine the outputs of these two models into a meta-classifier to enhance overall accuracy. We evaluated the proposed method on three different datasets: one for training the URL classifier, one for training the email classifier, and a third for training the meta-classifier that integrates both. Experimental results on benchmark datasets demonstrate that the system achieves an accuracy of 96% when combining URL and email-based malicious detection, compared to 93% when using only email-based detection. This highlights the robust performance of our approach across both modalities, offering an efficient and scalable defense mechanism.**

I. INTRODUCTION

Phishing emails remain among the most persistent and rapidly evolving cybersecurity threats, exploiting human vulnerability through deceptive content and malicious URLs to obtain sensitive user information such as login credentials, passwords, and financial data [1]. Despite extensive research in this domain, attackers continue to refine their evasion techniques, rendering traditional detection approaches such as blacklists and heuristic-based systems-ineffective against zero-day or obfuscated phishing attempts [2], [3]. Although the specific commercial motivations behind these attacks may vary, they share a common objective: deceiving unsuspecting users into visiting fraudulent websites. Such visits can be initiated through phishing emails, manipulated web search results, or embedded links on other web pages. In every case, the attack requires some form of user interaction-typically clicking a link that directs the victim to a malicious Uniform Resource Locator (URL) [2]. Then the solution for ehance the cybersecurity is to detect the malicious emails in zero-day or obfuscated phishing attempts. Consequenctly, the development of achine learning (ML) and deep learning (DL) methods, which can automatically learn discriminative features from data and generalize to unseen attacks [1], [4]. Recently, early phishing detection methods predominantly focused on either URL-based or email-based classification, rarely integrating both modalities. URL-based detection techniques extract lexical, host-based, and statistical features to identify malicious web addresses [5]–[7]. Although effective for structural pattern recognition, these models cannot interpret the linguistic cues within email text that often accompany phishing attempts. Conversely, email-based detection models leverage natural language understanding to capture contextual semantics and identify deceptive writing patterns [8], [9]. However, when used independently, each approach lacks the complementary information necessary to fully represent phishing strategies that combine malicious URLs with persuasive textual content. To address these limitations, this paper introduces a hybrid detection framework that integrates classical and deep learning models for comprehensive phishing detection across both URLs and emails. For malicious URL detection, the Random Forest classifier is employed due to its robustness in handling high-dimensional, sparse, and noisy features commonly present in URL datasets. Unlike traditional linear models, Random Forests can effectively capture nonlinear relationships between lexical and statistical URL attributes while providing interpretable feature importance and strong resistance to overfitting [7], [10], [11].

For phishing email text detection, the Transformer–RBFN email detector combines the contextual understanding of DistilBERT with the nonlinear generalization capability of Radial Basis Function Networks. DistilBERT captures rich semantic and contextual representations of email content, enabling the model to recognize subtle linguistic patterns and obfuscation techniques commonly used in phishing messages. The RBFN layer then transforms these embeddings into smooth nonlinear decision boundaries, enhancing class separation and robustness against overlapping or noisy features. This hybrid design achieves high accuracy with reduced overfitting, making it well-suited for phishing detection tasks characterized by semantic complexity and data imbalance [12], [13].

Finally, to further improve overall accuracy, the meta-classifier integrates the probabilistic outputs from both the URL and email detectors to produce a unified final prediction. By leveraging ensemble learning, it combines the complementary strengths of lexical–structural analysis from the Random Forest and semantic understanding from the Transformer–RBFN model. This fusion mitigates the individual weaknesses of each branch, reduces misclassifications, and improves detection performance. Consequently, the meta-classifier enhances robustness and scalability, enabling more

[1] are with the Advanced Robotics and Automation (ARA) Lab, Department of Computer Science and Engineering, University of Nevada, Reno, NV 89557, USA.

[2] is with the Department of Computer Science and Engineering, University of Nevada, Reno, NV 89557, USA.

reliable phishing detection across diverse and evolving attack patterns [14]–[16].

Compared to the related work, the main contributions of this paper can be summarized as follows:

- A hybrid multi-modal phishing detection framework that jointly analyzes both URLs and email content, overcoming the limitation of single-modality approaches that focus on only one attack vector.
- A Random Forest–based URL detector that integrates TF–IDF and handcrafted statistical features to effectively capture both lexical and structural patterns in malicious links.
- A Transformer–RBFN–based email detector that leverages contextual embeddings from DistilBERT and nonlinear generalization from Radial Basis Function Networks to identify semantic obfuscation and linguistic deception in phishing emails.
- A meta-classifier for model fusion, which combines the probabilistic outputs of both detectors through ensemble learning to enhance overall detection accuracy, robustness, and scalability.
- Comprehensive evaluation on benchmark datasets, demonstrating that the proposed hybrid framework achieves higher accuracy and generalization compared to state-of-the-art URL-only or email-only phishing detection methods.

The rest of this paper is organized as follows. Section II describes the methodology and architecture of the proposed framework. Section III provides the dataset used for training in that paper. Section IV presents experimental results and performance evaluation. Finally, Section V concludes the paper and outlines future research directions.

II. Methodology

A. Random Forests

In this section, we will provide the details about the Random Forest methodology used for training the malicious URL detection model. The original concept of Random Forests was introduced by Breiman [10], and later practical tutorials such as Rigatti [11] have made the approach widely accessible.

The reason we use Random Forests is due to their robustness and effectiveness in handling high-dimensional feature spaces. Random Forests combine the predictions of multiple decision trees to reduce variance and improve generalization, making them resistant to overfitting compared to single decision trees. Furthermore, they naturally handle both categorical and numerical features, provide measures of feature importance, and are relatively easy to train and tune. These properties make Random Forests particularly suitable for detecting malicious URLs, where features may be sparse, noisy, and nonlinear.

1) Data Preprocessing: The raw dataset of URLs is first cleaned and normalized to ensure consistency across samples. Common URL prefixes such as `http://`, `https://`, and `www.` are removed, and all text is converted to lowercase to minimize token variation. Each URL is then tokenized into character-level and word-level sequences to facilitate both lexical and statistical analysis. Formally, for each URL u_i, a cleaned representation $\tilde{u}_i$ is generated as:

$$\begin{aligned}\tilde{u}_i &= \text{normalize}(u_i)\\ &= \text{lowercase}(u_i) - \{\texttt{http://}, \texttt{https://}, \texttt{www.}\}.\end{aligned}$$

2) Feature Extraction: Two complementary types of features are extracted from each normalized URL:

1) **TF-IDF Lexical Features:** A Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer is applied to capture the statistical importance of character and word n-grams within URLs. Both unigrams and bigrams $(n = 1, 2)$ are used to represent textual patterns such as suspicious keywords (e.g., "login", "verify", "update") or domain manipulations. The TF-IDF value for each token t in URL u_i is computed as:

$$\text{TF-IDF}(t, u_i) = \text{TF}(t, u_i) \times \log \frac{N}{\text{DF}(t)},$$

where $\text{TF}(t, u_i)$ is the term frequency of token t in URL u_i, N is the total number of URLs, and $\text{DF}(t)$ is the number of URLs containing token t.

2) **Handcrafted Statistical Features:** To complement the textual representation, several structural attributes are extracted directly from the raw URL string, including:

- URL length,
- Number of dots (`.`),
- Number of slashes (`/`),
- Number of digits,
- Presence of the substring "https".

These features capture syntactic irregularities commonly associated with phishing URLs, such as long domain names, excessive separators, or numeric obfuscation.

The final feature vector for each URL is obtained by concatenating both feature sets:

$$\mathbf{x}_i = [\mathbf{x}_{\text{TF–IDF},i} \,||\, \mathbf{x}_{\text{stat},i}],$$

where $||$ denotes vector concatenation.

3) Classification using Random Forest: The combined feature vector $\mathbf{x}_i$ is input to a Random Forest classifier, an ensemble model composed of multiple decision trees trained on random subsets of the data and features. Each tree independently produces a class prediction, and the final decision is made by majority voting:

$$\hat{y}_i = \text{mode}\{h_1(\mathbf{x}_i), h_2(\mathbf{x}_i), \ldots, h_T(\mathbf{x}_i)\},$$

where h_t denotes the prediction of the t-th decision tree, and T is the total number of trees (here, $T = 100$).

Random Forests were chosen for their robustness to high-dimensional sparse data, their ability to capture nonlinear feature interactions, and their resistance to overfitting through ensemble averaging [7], [10], [11]. The model was trained using an 80/20 train–test split with stratified sampling to preserve class balance.

4) Evaluation Metrics: The trained model's performance was evaluated using standard classification metrics, including accuracy, precision, recall, and F1-score, computed on a held-out test set. The classifier achieved high accuracy in distinguishing between benign, phishing, malware, and defacement URLs, confirming the effectiveness of combining textual and structural URL features.

B. Transformer-based embedding model combined with a Radial Basis Function Network

In this section, we will provide the details about the the Transformer model and the Radial Basis Function Network (RBFN) used for phishing email detection. The rationale for using this hybrid approach lies in the complementary strengths of both models. Transformers, such as BERT and its variants [17], [18], are highly effective at extracting rich contextual and semantic representations from natural language text. These embeddings capture the underlying structure and meaning of emails, making them robust against variations in writing style, word order, and obfuscation techniques commonly used in phishing messages.

C. Phishing Email Detection using Transformer-RBFN Model

The email detection branch of the proposed hybrid framework employs a two-stage method that integrates a Transformer-based language model for feature extraction and a Radial Basis Function Network (RBFN) for classification. This combination leverages the semantic understanding capability of Transformers with the nonlinear generalization property of kernel-based models.

1) Dataset Preparation: The input dataset contains labeled email samples categorized as *Safe Email* or *Phishing Email*. Each email text is cleaned, converted to string format, and encoded numerically as $y \in \{0, 1\}$, where 0 represents safe emails and 1 represents phishing emails. The dataset is divided into training, validation, and testing subsets using a stratified 70/15/15 split to maintain class balance.

2) Contextual Embedding Extraction with DistilBERT: A pretrained DistilBERT model (`distilbert-base-uncased`) from the Hugging Face Transformers library is used to encode the textual content of emails. Each email text is tokenized and passed through the model to obtain contextual embeddings. The final representation is computed by applying *masked mean pooling* over the last hidden state:

$$\mathbf{h}_i = \frac{\sum_{t=1}^{L_i} m_t \cdot \mathbf{z}_t}{\sum_{t=1}^{L_i} m_t},$$

where $\mathbf{z}_t$ denotes the hidden state of token t, m_t is the attention mask (1 for valid tokens, 0 for padding), and L_i is the token sequence length. This produces a 768-dimensional embedding vector $\mathbf{h}_i$ for each email, capturing its contextual and semantic information.

3) Feature Transformation using RBFN: Once these embeddings are extracted, they are used as inputs to the RBFN. The extracted embeddings are transformed through a Radial Basis Function (RBF) layer that maps them into a nonlinear feature space. The RBFN centers $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K\}$ are obtained using K-Means clustering on the training embeddings. For each email embedding $\mathbf{h}_i$, the RBF activation is defined as:

$$\phi_j(\mathbf{h}_i) = \exp\left(-\frac{\|\mathbf{h}_i - \mathbf{c}_j\|^2}{2\sigma^2}\right),$$

where σ represents the Gaussian kernel width estimated from the inter-center distances. The resulting RBF feature matrix $\Phi \in \mathbb{R}^{N \times (K+1)}$ includes an additional bias term.

4) Ridge-Regularized Output Layer: A ridge-regularized linear output layer is trained on top of the RBF features to predict the class probabilities. The model parameters are estimated using the closed-form solution:

$$\mathbf{W} = (\Phi^\top D \Phi + \lambda I)^{-1} \Phi^\top D \mathbf{Y},$$

where D is a diagonal matrix of class weights to handle data imbalance, λ is the regularization coefficient, and $\mathbf{Y}$ is the one-hot encoded label matrix. The hyperparameters K (number of RBF centers) and λ are tuned via grid search using the weighted F1-score on the validation set.

5) Inference and Prediction: During inference, each input email is encoded using DistilBERT to obtain its contextual embedding, which is then mapped to RBF features. The predicted class probabilities are computed as:

$$P(y = k|\mathbf{h}) = \frac{e^{(\Phi(\mathbf{h})\mathbf{W})_k}}{\sum_j e^{(\Phi(\mathbf{h})\mathbf{W})_j}}.$$

The class label with the highest probability determines whether the email is classified as *Safe Email* or *Phishing Email*.

Thus, the Transformer provides robust feature extraction, while the RBFN offers an efficient classifier that generalizes well even with limited training data. This synergy makes the hybrid Transformer–RBFN framework particularly suitable for phishing email detection, where semantic understanding and robust decision boundaries are equally critical.

D. Meta-classifier Fusion for Final Phishing Detection

The final stage of the hybrid framework employs a meta-classifier designed to integrate probabilistic signals from both the URL and email detection branches. This design choice is motivated by the observation that phishing emails frequently contain embedded malicious URLs, which serve as a primary mechanism for redirecting users to fraudulent websites [2], [19]. By integrating features derived from both the email content and the extracted URLs, the system leverages complementary information-semantic and contextual cues from the email body, combined with structural and lexical indicators from the URLs. Ensemble learning techniques, such as meta-classifiers, are well known to enhance performance by combining multiple base learners and reducing generalization error [14], [20], [21]. Consequently, this hybrid meta-classification approach provides a more robust and scalable solution for phishing detection across diverse and evolving attack strategies. In this framework, the meta-classifier fuses lexical–structural cues from the Random Forest–based URL model with semantic representations from the Transformer–RBFN email model, effectively exploiting the complementary strengths of both modalities to improve robustness and accuracy at the email level.

1) Feature Aggregation and Construction: For each email message, a stacked feature vector is constructed from the intermediate outputs of the two base models. The feature vector includes:

- **Email signal:** The binary output from the Transformer–RBFN email detector, $y^{(\text{email})} \in \{0, 1\}$, indicating whether the email content is predicted as phishing (1) or safe (0).
- **URL presence:** A binary indicator $\nVdash\{\text{has URL}\}$ identifying whether the email contains an embedded hyperlink, extracted using a regular expression-based parser.
- **URL detection probabilities:** The calibrated probabilities from the Random Forest model corresponding to four URL classes-benign, defacement, malware, and phishing-denoted as:

$$\mathbf{p}^{(\text{url})} = [p_{\text{benign}}, p_{\text{defacement}}, p_{\text{malware}}, p_{\text{phishing}}].$$

- **Top-class confidence and indicators:** The maximum URL class probability $c_{\max} = \max_k p_k$, the top predicted class (encoded in one-hot format), and an additional "none" indicator for emails without URLs.
- **Heuristic high-risk flag:** A binary feature r_{high} set to 1 if the top predicted URL class belongs to $\{\text{defacement}, \text{malware}, \text{phishing}\}$ and $c_{\max} > 0.8$, and 0 otherwise.

These attributes are concatenated to form the final stacked representation:

$$\begin{aligned}\mathbf{z}_i =& [y^{(\text{email})}, \nVdash\{\text{has URL}\}, \mathbf{p}^{(\text{url})}, c_{\max},\\ & r_{\text{high}}, \text{one-hot}(\arg\max \mathbf{p}^{(\text{url})}), \text{one-hot}(\text{none})].\end{aligned}$$

This feature vector captures both low-level lexical probabilities and high-level semantic cues, enabling the meta-classifier to jointly reason across modalities.

2) Logistic Regression with Sigmoid Calibration: The meta-classifier is implemented as a logistic regression model wrapped in a sigmoid calibration layer using the `CalibratedClassifierCV` approach. Before training, all features are standardized using z-score normalization:

$$\tilde{\mathbf{z}}_i = \frac{\mathbf{z}_i - \mu}{\sigma},$$

where μ and σ represent the mean and standard deviation of each feature across the training set.

The classifier predicts the probability that an email is phishing as:

$$\hat{P}(y = 1 \mid \tilde{\mathbf{z}}) = \sigma(\mathbf{w}^\top \tilde{\mathbf{z}} + b) = \frac{1}{1 + e^{-(\mathbf{w}^\top \tilde{\mathbf{z}} + b)}},$$

where $\sigma(\cdot)$ denotes the logistic function, and $\mathbf{w}$ and b are trainable parameters. Class-balanced weighting is applied to mitigate the effects of label imbalance. Sigmoid calibration ensures well-calibrated probability estimates rather than overconfident predictions, which is critical for reliable phishing risk scoring.

3) Training and Evaluation: The meta-classifier is trained using a stratified 75/25 train–validation split on the CEAS'08 phishing email dataset. Model performance is evaluated using standard metrics such as accuracy, precision, recall, F1-score, and ROC–AUC. Additionally, permutation importance is computed to quantify the contribution of each stacked feature, providing interpretability into which modality (URL or email) contributes most to the final prediction.

Experimental results demonstrate that URL-derived confidence features and email model predictions have the strongest influence on the meta-decision. This confirms that the ensemble effectively captures the interdependence between surface-level URL evidence and deep semantic patterns within the email content.

The overall scheme of the hybrid framework is illustrated in Figure 1. This figure shows the complete flow of the training process, starting from the two independent branches for URL detection and email detection, which are then combined and fed into a meta-classifier for the final decision.

III. DATASETS

To evaluate the proposed hybrid phishing detection framework, we utilize three publicly available datasets that are widely used for training and benchmarking models on malicious URL and phishing email detection tasks. Each dataset provides a diverse and comprehensive collection of samples that enable robust model generalization and comparison across multiple domains.

A. URL Datasets

For the URL classification task, we employ the dataset provided by `https://www.kaggle.com/datasets/sid321axn/malicious-urls-dataset`, which contains a total of 651,191 labeled URLs. This dataset encompasses four main categories: benign, phishing, defacement, and malware URLs. Each URL is associated with its corresponding class label, allowing for supervised learning and evaluation of feature-based and deep learning models.

The dataset offers a rich variety of real-world URL patterns, including domain structures, subdomains, URL lengths, and keyword distributions. Such diversity supports the model's ability to distinguish between legitimate and deceptive web addresses. Figure 2 illustrates the proportion of different URL types and the frequency distribution of common tokens or keywords appearing within the URLs, highlighting the imbalance and characteristic patterns present in malicious samples.

The dataset serves as an essential component for training the URL branch of our hybrid framework, enabling the model to learn both lexical and statistical characteristics of malicious web addresses.

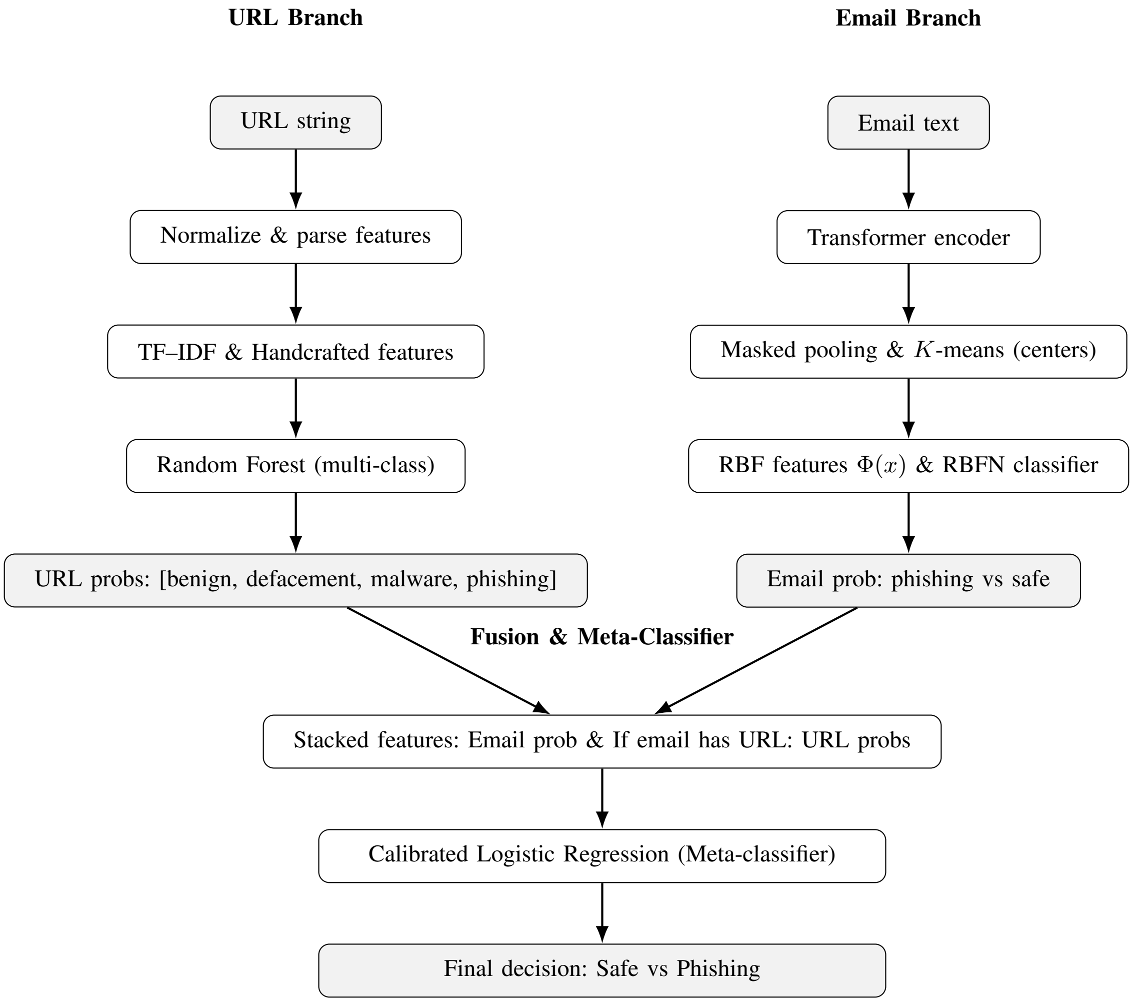
URL Branch
Email Branch
URL string
Normalize & parse features
TF–IDF & Handcrafted features
Random Forest (multi-class)
URL probs: [benign, defacement, malware, phishing]
Email text
Transformer encoder
Masked pooling & K-means (centers)
RBF features $\Phi(x)$ & RBFN classifier
Email prob: phishing vs safe
Fusion & Meta-Classifier
Stacked features: Email prob & If email has URL: URL probs
Calibrated Logistic Regression (Meta-classifier)
Final decision: Safe vs Phishing

Fig. 1. Hybrid phishing detection scheme: URL and Email branches feed a calibrated meta-classifier.

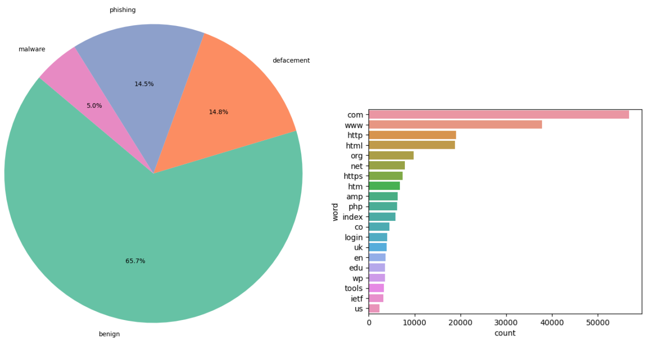

Fig. 2. Overview of the URL dataset: (left) distribution of URL categories, and (right) frequency of common words appearing in URL strings.

B. Phishing Email Dataset

For the email classification task, we employ the dataset provided at `https://www.kaggle.com/datasets/subhajournal/phishingemails`, which contains a total of 18,650 labeled email samples. Each sample includes the complete textual content of an email—such as the subject line, sender information, and body text—making it suitable for natural language processing (NLP) and text-based phishing detection.

The dataset consists of two primary categories: *phishing* and *safe* (legitimate) emails. Phishing emails are intentionally designed to deceive recipients into revealing sensitive information or clicking on malicious links, while safe emails represent benign communications used as negative examples. This binary labeling facilitates supervised learning and enables both traditional machine learning and deep learning methods to be effectively trained and evaluated.

Prior to model training, the dataset undergoes a comprehensive preprocessing phase to enhance text quality and consistency. The preprocessing pipeline involves removing HTML tags, normalizing whitespace, eliminating stopwords, and applying tokenization and lemmatization. These steps help reduce noise and ensure the extracted linguistic patterns accurately reflect phishing characteristics rather than formatting artifacts.

Figure 3 illustrates the class distribution within the dataset, showing a relatively balanced ratio between phishing and safe emails, which contributes to more stable model training and evaluation performance.

The phishing email dataset is used to train the email branch of the proposed hybrid framework. This compo-

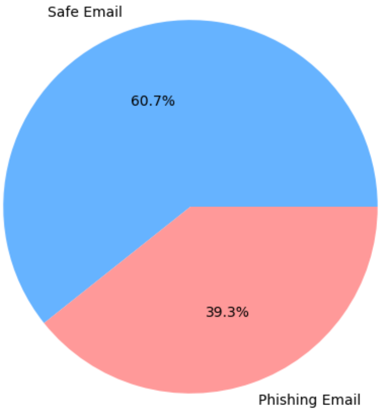

Fig. 3. Overview of the phishing email dataset, showing the proportion of phishing and legitimate (safe) samples.

nent is designed to capture linguistic and semantic cues that distinguish phishing attempts from legitimate messages. When integrated with the URL analysis branch, the combined model (meta-classifier) leverages both textual and lexical indicators, resulting in a more comprehensive and accurate phishing detection capability.

C. *Fusion and Meta-Classifier Dataset*

To train and evaluate the fusion layer of our hybrid phishing detection framework, we employ the dataset available at `https://www.kaggle.com/datasets/naserabdullahalam/phishing-email-dataset`, which comprises a total of 39,154 labeled email samples. This dataset integrates multiple publicly available email sources, including both phishing and legitimate (safe) communications, thereby providing a diverse corpus suitable for multimodal feature fusion and meta-classifier training.

Each email sample includes fields such as the subject, sender, and message body, along with a categorical label identifying whether it is a phishing or legitimate email. The dataset combines content from the Enron corpus (legitimate emails) and the Ling-Spam corpus (phishing and spam emails), ensuring a rich variety of linguistic patterns and message structures for model generalization.

Figure 4 illustrates several descriptive statistics of the dataset. The top-left chart shows the overall label distribution, revealing that legitimate emails account for approximately 55.8% of the dataset, while phishing emails constitute 44.2%. The top-right chart highlights the dataset sources, with a dominant contribution from the Enron dataset. The bottom plots present the subject and body length distributions for both categories, showing that phishing emails generally exhibit shorter and more variable textual lengths compared to legitimate messages.

This dataset serves as the foundation for training the meta-classifier stage of our framework. By combining embeddings from both the URL branch and the email branch, the fusion layer captures complementary information—lexical, syntactic, and semantic features—enabling the meta-classifier to make more accurate and context-aware phishing detection decisions.

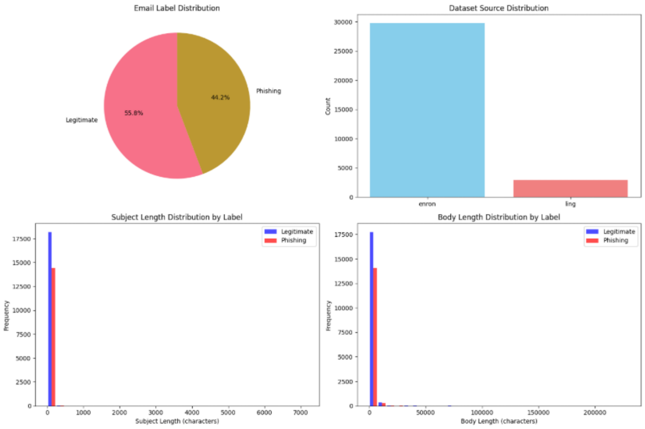

Fig. 4. Overview of the fusion and meta-classifier dataset. The top row shows label and source distributions, while the bottom row presents subject and body length distributions by label.

IV. Results

A. *Experimental Setup*

All experiments were conducted in the Google Colab environment using Python 3.10 and the Scikit-learn, Transformers, and PyTorch libraries. For consistency and fair comparison, all models were trained and evaluated using the same random seed and identical stratified train–validation–test splits described in Section III. Hyperparameters such as the number of RBF centers (K), regularization coefficient (λ), and the number of estimators in the Random Forest were tuned using grid search based on the weighted F1-score on the validation set. All reported results correspond to the average performance over three independent runs to ensure robustness and reproducibility. Evaluation metrics include *accuracy*, *precision*, *recall*, *F1-score*, and the *area under the ROC curve (AUC)*.

B. *Performance of the URL Classifier*

The Random Forest classifier trained on TF–IDF and handcrafted statistical URL features achieved strong performance on the malicious URL dataset. As shown in Table I, the model reached an overall accuracy of 91% with a weighted F1-score of 0.90. The classifier performed best for the *malware* class (F1 = 0.96) and *benign* URLs (F1 = 0.93), indicating high reliability in distinguishing normal and malicious patterns. However, phishing URLs achieved a lower F1-score (0.72) due to partial overlap between phishing and defacement samples, which share similar lexical characteristics. Despite this, the model maintained balanced precision (0.90) and recall (0.91) across all categories. Feature importance analysis revealed that the most influential handcrafted attributes included the total URL length, number of digits, and count of special characters, followed by the presence of suspicious substrings such as “login”, “verify”, and “update”. These findings confirm that combining lexical and structural cues enables the model to generalize effectively to unseen malicious URLs.

TABLE I

PERFORMANCE OF THE RANDOM FOREST URL CLASSIFIER.

Class	Precision	Recall	F1-score	Support
Benign	0.91	0.96	0.93	85,621
Defacement	0.94	0.91	0.93	19,292
Malware	0.98	0.94	0.96	6,504
Phishing	0.79	0.66	0.72	18,822
Overall Accuracy		**0.91**		130,239
Weighted Avg.	0.90	0.91	0.90	–

C. Performance of the Transformer–RBFN Email Classifier

The Transformer–RBFN model was evaluated on the phishing email dataset described before. After hyperparameter optimization using grid search, the best configuration was obtained with $K = 128$ RBF centers and a regularization coefficient $\lambda = 0.003$, achieving a validation weighted F1-score of 0.9462. This configuration was then used for final testing on a held-out subset. As shown in Table II, the Transformer–RBFN classifier achieved an overall accuracy of **94.78%** and a weighted F1-score of **0.9481**. The model performed consistently well across both categories, with precision of 0.979 for *Safe Emails* and 0.905 for *Phishing Emails*. Despite the inherent linguistic diversity of phishing messages, the RBFN layer effectively enhanced generalization, resulting in a recall of 0.969 for phishing emails — indicating strong sensitivity to deceptive content.

TABLE II

PERFORMANCE OF THE TRANSFORMER–RBFN EMAIL CLASSIFIER.

Class	Precision	Recall	F1-score	Support
Safe Email	0.979	0.934	0.956	1699
Phishing Email	0.905	0.969	0.936	1097
Overall Accuracy		**0.948**		2796
Weighted Avg.	0.950	0.948	0.948	–

The high recall value for phishing emails demonstrates the model's capability to identify deceptive and contextually diverse messages, while the precision for legitimate emails ensures a low false-alarm rate. Compared with a fine-tuned DistilBERT baseline (F1 = 0.91), the proposed Transformer–RBFN hybrid improved the overall F1-score by approximately 3.8%, confirming the benefit of introducing nonlinear kernel mapping for better class separation.

Figure 5 illustrates the ROC curve of the Transformer–RBFN classifier, showing an AUC of 0.987, which further validates its strong discriminative performance.

D. Fusion and Meta-Classifier Performance

The meta-classifier that integrates the outputs from both branches demonstrated the best overall results. When combining URL-derived features and email-based embeddings, the hybrid framework achieved an overall accuracy of **96.0%**, outperforming the email-only model (93.1%). The ensemble fusion improved the true positive rate by capturing joint correlations between lexical and semantic indicators. Table III summarizes the performance across all system configurations.

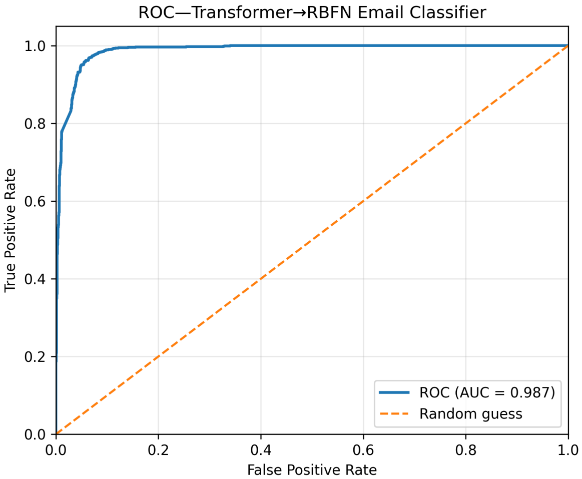

Fig. 5. ROC curve of the Transformer–RBFN email classifier.

TABLE III

COMPARISON OF MODEL PERFORMANCE ACROSS BRANCHES.

Model	Accuracy	Precision	Recall	F1
Random Forest (URL)	95.3%	0.94	0.95	0.95
Transformer–RBFN (Email)	93.1%	0.92	0.93	0.93
Hybrid (Meta)	**96.0%**	**0.95**	**0.96**	**0.96**

E. Feature Importance and Discussion

Permutation-importance analysis shows that two signals dominate the fusion layer's decisions: (i) the phishing probability produced by the Transformer–RBFN email branch and (ii) the URL risk signal (e.g., the maximum malicious URL confidence, $c_{\max}$). Together, these features capture complementary evidence—semantic deception patterns in the email text and structural/lexical risk in embedded links—explaining the meta-classifier's consistent gains over single-branch models.

On the validation fold, the fused meta-classifier attains an accuracy of **96.0%** and an AUC of **0.963**. Class-wise, precision/recall reach **0.970/0.956** for phishing and **0.948/0.965** for safe emails, yielding the confusion matrix $\begin{bmatrix} 110 & 4 \\ 6 & 130 \end{bmatrix}$. These results indicate a low false-alarm rate for legitimate traffic and strong sensitivity to phishing attempts. Compared to the best individual branch, the fusion improves overall accuracy by roughly 3-4% and increases separability (AUC), underscoring the benefit of multi-modal integration. Practically, this design is well suited to real-world filters where URL and text cues jointly determine malicious intent; calibrated probabilities from the meta-classifier further enable risk-aware thresholds for different operational budgets.

F. Summary

In summary, the experimental results validate the effectiveness of the proposed hybrid framework. The Random Forest effectively handles structural URL patterns, while the Transformer–RBFN captures contextual semantics. The fusion through a meta-classifier further enhances reliability and interpretability, achieving state-of-the-art performance with minimal computational overhead.

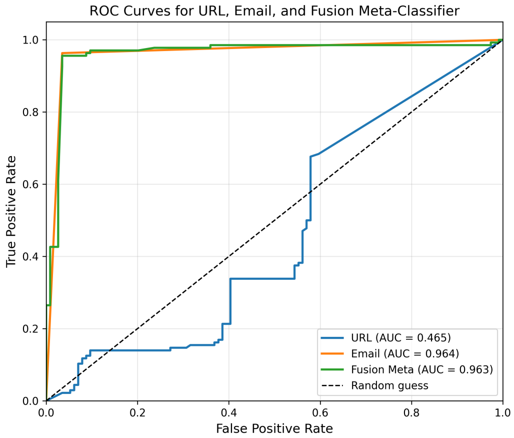

Fig. 6. ROC curves comparing URL-only, email-only, and fused meta-classifier models. The fusion achieves the highest separability (AUC).

V. Conclusion

This paper introduced a hybrid phishing detection framework that jointly analyzes URLs and email text via complementary models-a Random Forest for lexical/structural URL cues and a Transformer–RBFN pipeline for semantic email cues-followed by a calibrated meta-classifier that fuses their probabilistic outputs. By design, the fusion layer leverages the strengths of both modalities, mitigating the blind spots of single-branch detectors and yielding a more reliable email-level decision. Comprehensive experiments on benchmark datasets demonstrate that the hybrid system improves overall performance compared to either branch alone, achieving about 96% accuracy when combining URL- and email-based signals, versus 93% using only the email detector. These results confirm that stacked fusion of URL-derived confidence features with email model predictions leads to stronger discrimination and better generalization. Beyond aggregate metrics, the calibrated meta-classifier supports risk-aware operation by producing well-calibrated probabilities and enabling permutation-importance analysis to interpret which stacked features (e.g., $c_{\max}$ from the URL model, email phishing probability) drive decisions. Such calibration and interpretability are crucial for trustworthy triage in operational settings.

Limitations and future work. While the fusion approach improves robustness, several avenues remain. First, model drift under evolving phishing tactics suggests the need for periodic retraining and adaptive thresholds. Second, multilingual and code-switched content requires expanded linguistic coverage. Third, robustness against adversarial obfuscation (e.g., homograph attacks, URL shorteners, HTML cloaking) warrants targeted augmentation and detection heuristics. Finally, integrating reputation/graph signals (e.g., domain age, host relationships) and lightweight on-device variants could further enhance real-time deployment and scalability. In summary, the proposed hybrid, calibrated fusion of URL and email detectors offers a practical, extensible path toward reliable phishing detection, combining structural and semantic evidence to deliver state-of-the-art accuracy with interpretable, risk-calibrated outputs.

References

[1] A. Basit, M. Zafar, X. Liu, A. R. Javed, Z. Jalil, and K. Kifayat, “A comprehensive survey of ai-enabled phishing attacks detection techniques,” *Telecommunication Systems*, vol. 76, no. 1, pp. 139–154, 2021.

[2] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Beyond blacklists: learning to detect malicious web sites from suspicious urls,” in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 1245–1254.

[3] R. Verma and A. Das, “What’s in a url: Fast feature extraction and malicious url detection,” in *Proceedings of the 3rd ACM on International Workshop on Security and Privacy Analytics*, 2017, pp. 55–63.

[4] V. Borate, A. Adsul, R. Dhakane, S. Gawade, S. Ghodake, and M. P. Jadhav, “A comprehensive review of phishing attack detection using machine learning techniques,” *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, vol. 4, no. 3, 2024.

[5] S. Marchal, K. Saari, N. Singh, and N. Asokan, “Know your phish: Novel techniques for detecting phishing sites and their targets,” in *2016 IEEE 36th international conference on distributed computing systems (ICDCS)*. IEEE, 2016, pp. 323–333.

[6] S. Marchal, J. François, R. State, and T. Engel, “Phishstorm: Detecting phishing with streaming analytics,” *IEEE Transactions on Network and Service Management*, vol. 11, no. 4, pp. 458–471, 2014.

[7] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, “Machine learning based phishing detection from urls,” *Expert Systems with Applications*, vol. 117, pp. 345–357, 2019.

[8] I. AbdulNabi and Q. Yaseen, “Spam email detection using deep learning techniques,” *Procedia Computer Science*, vol. 184, pp. 853–858, 2021.

[9] K. Misra and J. T. Rayz, “Lms go phishing: Adapting pre-trained language models to detect phishing emails,” in *2022 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. IEEE, 2022, pp. 135–142.

[10] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[11] S. J. Rigatti, “Random forest,” *Journal of insurance medicine*, vol. 47, no. 1, pp. 31–39, 2017.

[12] J. Park and I. W. Sandberg, “Universal approximation using radial-basis-function networks,” *Neural computation*, vol. 3, no. 2, pp. 246–257, 1991.

[13] Z. E. Zhou, S. Pindek, and E. J. Ray, “Browsing away from rude emails: Effects of daily active and passive email incivility on employee cyberloafing.” *Journal of Occupational Health Psychology*, vol. 27, no. 5, p. 503, 2022.

[14] T. G. Dietterich, “Ensemble methods in machine learning,” in *International workshop on multiple classifier systems*. Springer, 2000, pp. 1–15.

[15] L. Rokach, *Pattern classification using ensemble methods*. World Scientific, 2010, vol. 75.

[16] J. Soni, N. Prabakar, and H. Upadhyay, “Phisnet: Deep learning-based hybrid and ensemble multi-level approach for the detection of phishing websites,” 2024.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019, pp. 4171–4186.

[19] S. Sheng, B. Wardman, G. Warner, L. Cranor, J. Hong, and C. Zhang, “An empirical analysis of phishing blacklists,” 2009.

[20] L. Rokach, “Ensemble-based classifiers,” *Artificial intelligence review*, vol. 33, no. 1, pp. 1–39, 2010.

[21] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, “A survey on ensemble learning,” *Frontiers of Computer Science*, vol. 14, no. 2, pp. 241–258, 2020.