

# Detection Of AI-Generated Phishing Emails Using Hybrid Attention-Based NLP Models

Mahankali Ravisankar<sup>1</sup>, Dr. B. Anuradha<sup>2</sup>

<sup>1</sup>*M. Tech Student, Department of Electronics and Communication Engineering, S. V. University College of Engineering, Tirupati, A. P. India*

<sup>2</sup>*Professor, Department of Electronics and Communication Engineering, S. V. University College of Engineering, Tirupati, A. P. India*

*doi.org/10.64643/IJIRT208745*

**Abstract**—In recent years, artificial intelligence and natural language processing have transformed the field of cybersecurity, particularly in the detection of phishing emails. This research focuses on detecting AI-generated and traditional phishing emails that closely resemble legitimate messages and are difficult for conventional filtering systems to identify. The proposed system combines advanced text preprocessing with TF-IDF and RoBERTa to extract both statistical and contextual features from email content. These features are fused with URL-based information and balanced using SMOTE-Tomek to improve classification performance. A BiLSTM model is employed for deep feature learning, while XGBoost is used for final email classification and performance evaluation. The model is further optimized using Optuna, validated through Stratified K-Fold cross-validation, and interpreted using SHAP explainability. The proposed approach achieved 99.29% accuracy, 99.23% precision, 99.36% recall, 99.30% F1-score, and 99.96% ROC-AUC, demonstrating its effectiveness in identifying phishing emails. However, the system still faces challenges such as false predictions, computational cost, dataset limitations, and the need for continuous adaptation to evolving AI-generated phishing attacks.

**Index Terms**—AI Generated Phishing Emails, Phishing Detection, Natural Language Processing (NLP), TFIDF, RoBERTa, BiLSTM, XGBoost, SMOTE-Tomek, Optuna, SHAP Explainability.

## I. INTRODUCTION

Phishing is a common cybersecurity threat in which attackers use deceptive emails to obtain sensitive information such as passwords, financial details, and personal data. The increasing sophistication of phishing techniques has made detection more challenging [3]. The emergence of Artificial

Intelligence (AI) and Generative AI has further increased this challenge by enabling the generation of realistic and personalized phishing emails [4] and [14]. Therefore, automated techniques are required to analyze email content and identify phishing characteristics. Natural Language Processing (NLP) and machine-learning methods have been widely applied to analyze linguistic patterns in email messages. Earlier studies explored NLP-based phishing email detection [1], while recent work investigated NLP and machine-learning approaches for suspicious email classification [15]. Malicious URLs can also provide useful information for identifying phishing emails [2]. Deep-learning and transformer-based approaches have further improved phishing detection by learning complex textual and contextual patterns [5] and [6]. Bidirectional Encoder Representations from Transformers (BERT)-based representations have also been combined with Extreme Gradient Boosting [XGBoost] for phishing email classification [8]. Class imbalance can affect phishing classification performance; therefore, SMOTE-Tomek can be used to balance training data and reduce overlapping samples [7]. Bidirectional Long Short-Term Memory [BiLSTM] models can capture sequential patterns in email text [9], while explainable AI techniques such as SHAP and LIME can improve the interpretability of phishing detection systems [10] and [12]. Dataset and feature analysis have also been investigated for machine-learning-based phishing detection [11] and [13]. Based on these studies, this work proposes a framework combining NLP preprocessing, TF-IDF, attention-based contextual representations, URL features, SMOTE-Tomek, BiLSTM, and XGBoost, with hyperparameter

optimisation, and SHapley Additive exPlanations (SHAP)-based explainability. The framework aims to provide accurate and interpretable detection of modern phishing emails, including AI-generated phishing messages.

## II. LITERATURE SURVEY

R. Verma et al, Verma and his co-authors investigated phishing email detection using the natural language contained in email messages. Their work demonstrated that words, expressions, and linguistic patterns can provide useful information for identifying suspicious emails. This study established an early foundation for applying Natural Language Processing (NLP) to phishing detection. It supports the text-analysis component of the present research. However, the study does not employ recent contextual transformer-based representations such as BERT or RoBERTa.[1]

D. Ranganayakulu and C. Chellappan Ranganayakulu and Chellappan investigated the detection of malicious URLs contained in email messages. Their study demonstrated that URL-related characteristics can provide useful information for identifying potentially harmful emails. This work supports the inclusion of URL-related features in the proposed framework. However, their approach primarily focuses on URL detection rather than combining URL information with contextual email-text representations.[2]

Z. Alkhalil et al. Alkhalil and co-authors presented a comprehensive study of phishing attacks and their evolving characteristics. Their work discussed different phishing techniques and the ways attackers make fraudulent communication appear trustworthy. This study provides important background for understanding the phishing problem addressed in the present research. However, it mainly focuses on the characteristics and anatomy of phishing attacks rather than developing a specific deep-learning detection framework.[3]

C. S. Eze and L. Shamir Eze and Shamir examined the role of Artificial Intelligence in phishing email attacks. Their study highlighted how AI-based techniques can assist attackers in producing convincing and realistic phishing messages. This work is highly relevant to the present research because it establishes the importance of addressing AI-assisted phishing threats. However, the study primarily focuses on the analysis and

prevention of AI-based phishing rather than proposing the complete feature-fusion and classification framework used in this research.[4]

N. Altwaijry et al. Altwaijry and co-authors investigated and compared deep-learning approaches for phishing email detection. Their study examined the ability of neural-network-based models to learn complex patterns from email data. The findings support the use of deep learning for automated phishing detection. This research provides a foundation for incorporating deep-learning techniques into the proposed framework. However, it does not combine all the textual, contextual, URL, balancing, and explainability components considered in the present study.[5]

A. Tawil et al. Tawil and co-authors compared different machine-learning approaches for email phishing detection using TF-IDF, Word2Vec, and BERT representations. Their work demonstrated the importance of selecting appropriate text representations for capturing different characteristics of phishing emails. This study is closely related to the present research because TF-IDF and transformer-based representations are used to capture lexical and contextual information. The proposed framework extends this approach by incorporating URL-related information, class balancing, deep learning, and XGBoost classification.[6]

R. Zaimi et al. Zaimi and co-authors investigated a deep-learning approach for phishing detection while addressing class imbalance using SMOTE-Tomek. Their work demonstrated the usefulness of combining synthetic sample generation with Tomek-link removal to improve the representation of minority and overlapping samples. This study directly supports the use of SMOTE-Tomek in the present research. However, their work focuses primarily on phishing detection and does not combine the same contextual text, URL, BiLSTM, XGBoost, and explainability components used in the proposed framework.[7]

M. Antonijevic et al. Antonijevic and co-authors investigated phishing email identification using BERT representations combined with an optimized XGBoost classifier. Their study is particularly relevant because it demonstrates the effectiveness of combining contextual language representations with a tree-based classification model. This work provides methodological support for the use of contextual embeddings and XGBoost in the present research. The

proposed framework extends this direction by incorporating additional feature types, class balancing, deep learning, hyperparameter optimization, and explainability.[8]

Y. Sanjalawe et al. Sanjalawe and co-authors proposed a BiLSTM-based framework for interpretable email threat detection. Their study demonstrated that BiLSTM networks can learn sequential and contextual patterns from email-related text for threat identification. This work supports the use of BiLSTM for learning sequential patterns in the present research. However, the proposed framework uses BiLSTM as part of a broader pipeline that also includes transformer-based representations, URL features, XGBoost, class balancing, and SHAP-based interpretation.[9]

C. Mayo et al. Mayo and co-authors investigated phishing detection using a temporally dynamic feature-extraction approach with LIME and SHAP explanations. Their work emphasized the importance of interpretability in phishing-detection systems by identifying the features contributing to model predictions. This study directly supports the explainability component of the present research, where SHAP is used to analyze model predictions. However, the proposed framework integrates explainability with contextual feature extraction, class balancing, deep learning, and XGBoost classification.[10]

N. A. Alam et al. Alam and co-authors provided the *Phish No More* phishing email dataset, which combines multiple publicly available email collections, including Enron, Ling, CEAS, Nazario, Nigerian, and SpamAssassin datasets. This work is relevant to the present study because diverse email sources can provide a broader basis for training and evaluating phishing-detection models. The dataset supports the data-collection and experimental stages of the proposed framework. However, the dataset itself does not constitute a complete detection methodology.[11]

A. Al-Subaiey et al. Al-Subaiey and co-authors developed an interpretable web-based AI platform for phishing email detection. Their work focused on combining automated phishing detection with interpretability, allowing users to obtain information about the basis of model predictions. This study supports the use of explainable AI in cybersecurity applications and is relevant to the SHAP-based

interpretation component of the present research. However, the proposed framework focuses specifically on integrating contextual representations, URL information, class balancing, BiLSTM, and XGBoost within the detection pipeline.[12]

A. I. Champa et al. Champa and co-authors investigated curated datasets and feature characteristics for machine-learning-based phishing email detection. Their study highlighted the importance of dataset preparation and feature analysis in developing reliable phishing classifiers. This work supports the dataset preparation and feature-engineering stages of the present research. However, it does not provide the complete deep-learning and contextual feature-fusion framework proposed in this study.[13]

R. Gutierrez et al. Gutierrez and co-authors evaluated phishing detectors against emails generated using Large Language Models (LLMs) and provided associated datasets, code, and experimental results. Their work is particularly relevant to the present research because it addresses the emerging challenge of detecting phishing emails generated by modern language models. The study provides motivation for evaluating phishing-detection systems against AI-generated email content. However, it does not implement the complete attention-based NLP, BiLSTM, XGBoost, class-balancing, and SHAP framework proposed in the present study.[14]

G. Alkhodhairi and K. Saleem Alkhodhairi and Saleem investigated machine-learning-based detection of suspicious email messages using Natural Language Processing techniques. Their work examined NLP-based email representation and classification, providing evidence that textual characteristics can be effectively used to distinguish suspicious messages. This study is closely related to the present research because it supports the use of NLP and text-based feature extraction for email classification. However, the proposed framework extends this direction by incorporating contextual transformer-based representations, URL features, SMOTE-Tomek, BiLSTM, XGBoost, hyperparameter optimization, and SHAP-based explainability.[15]

### III. PROPOSED METHODOLOGY

For phishing email detection, the suggested architecture integrates text, URL, data balancing, and machine learning algorithms. TF-IDF, RoBERTa, and URL characteristics are extracted, scaled, and fused

from preprocessed email content. To address class imbalance, SMOTE-Tomek is applied to the training data. BiLSTM and XGBoost are utilized to classify the fused features, with Optuna being employed to optimize XGBoost. Lastly, interpretability and robustness analysis are provided by SHAP and 5-fold stratified cross-validation, and evaluation metrics include Accuracy, Precision, Recall, F1-Score, and ROC-AUC.

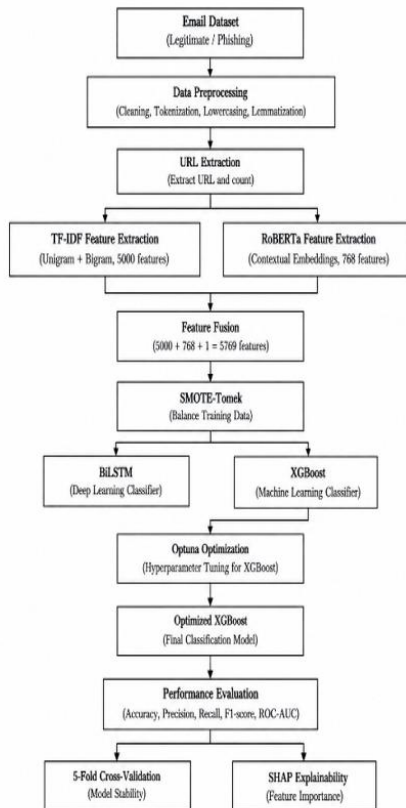


Fig. 1: Block Diagram

#### A. Email dataset

The proposed method uses a multi-source email dataset to identify both conventional and AI-generated phishing emails. The dataset combines publicly available legitimate, spam, and phishing email collections, including Enron, Ling, CEAS, Nazario, Nigerian Fraud, and SpamAssassin from the Phish No More collection [11]. These datasets provide diverse email content and phishing patterns for email classification. Previous studies have also used these collections for phishing detection [12], while curated datasets containing CEAS, Enron, Ling, Nazario, Nigerian Fraud, SpamAssassin, and TREC provide

additional sources for machine-learning-based detection [13]. To represent emerging AI-generated phishing attacks, LLM-generated phishing emails from multiple large language models are also incorporated [14]. After combining the sources, duplicate and empty records are removed, and the emails are standardized into Email\_Text and Label, where 0 represents legitimate emails and 1 represents phishing emails. The final dataset is shuffled and divided into training and testing sets using stratified sampling.

#### B. Data Preprocessing

To enhance the quality of the data and eliminate extraneous information, the gathered email dataset is first preprocessed. Unwanted letters, HTML elements, and extraneous spaces are eliminated from the email text. For uniform analysis, URLs are taken out of the email content and the text is changed to lowercase. After processing, the email text is lemmatized and normalized to reduce words to their most meaningful versions. These pre-processing procedures enhance the effectiveness of the suggested phishing email detection model by preparing the email data for efficient feature extraction using TF-IDF and RoBERTa.

#### C. Splitting Data

An 80:20 ratio is used to split the processed dataset into training and testing subsets. While the test data is kept apart for ultimate assessment, the training data is utilized for model construction and optimization.

#### D. Extraction of URLs

Prior to URL removal during text preparation, URLs are found in the original email content. Because URL data can offer helpful structural information for phishing identification, the number of URLs in each email is kept as an extra numerical attribute.

#### E. TF-IDF Feature Extraction

By giving significant and infrequent words greater weights, TF-IDF transforms preprocessed email text into numerical features. Individual words and significant word combinations are captured using both unigram and bigram characteristics. These characteristics aid in differentiating phishing emails from authentic ones, and they are then integrated with

contextual and URL-based characteristics to enhance detection efficiency.

#### *F. RoBERTa Feature Extraction*

Contextual features are extracted from email content using RoBERTa (Robustly Optimized BERT Pretraining Approach). In contrast to traditional word-based methods, RoBERTa captures the contextual meaning and semantic relationships of words in an email. The RoBERTa tokenizer is used in the suggested system to tokenize the preprocessed email text, which is then fed into the pretrained RoBERTa model. Each email is represented in 768 dimensions by the contextual embeddings produced by RoBERTa. These embeddings aid in the detection of minute linguistic patterns frequently connected to phishing and phishing emails produced by artificial intelligence. A complete feature representation is then created by combining the extracted contextual features with TF-IDF and URL-based features. The model can make use of both statistical and semantic aspects of email messages thanks to this feature fusion. The suggested hybrid machine learning framework is then utilized to classify the generated representation.

#### *G. Feature Fusion*

Complementary information from various feature representations is combined through feature fusion. The suggested system uses the URL count to represent URL-related properties, RoBERTa to give contextual and semantic information, and TF-IDF to capture lexical patterns. A 5,769-dimensional feature vector including 5,000 TF-IDF, 768 RoBERTa, and 1 URL-count feature is created by concatenating these features. Phishing email detection is subsequently enhanced by using the fused characteristics for SMOTE-Tomek balancing and classification.

#### *H. Data Balancing Using SMOTE-Tomek*

Class imbalance in the training data is addressed using SMOTE-Tomek. Tomek Links eliminates samples that overlap or are unclear between classes, while SMOTE creates artificial examples for the minority phishing class. This lessens classification bias and enhances class separation. To avoid data leaking and preserve an impartial test set, SMOTE-Tomek is exclusively used on training data. The XGBoost classifier is then given the balanced data to detect phishing emails.

#### *I. Deep Learning Using BiLSTM*

A Bidirectional Long Short-Term Memory (BiLSTM) network is also employed in the proposed study to assess a deep learning model's capacity to extract intricate patterns from fused email features. BiLSTM allows information from several locations in the input representation to be taken into account during learning by processing the input both forward and backward. The fused feature representation from TF-IDF, RoBERTa, and URL-based features is transformed into an appropriate three-dimensional format and fed into the BiLSTM network in the implemented model. Dropout and thick layers come after a 64-unit bidirectional LSTM layer in the model. The model is trained using the Adam optimizer with a learning rate of 0.001 and a dropout rate of 0.30 to minimize overfitting. When the validation performance stops improving, early halting is also used to end training. The efficacy of the BiLSTM model in identifying patterns associated with phishing from the fused feature representation is assessed independently.

#### *J. XGBoost Categorization*

The balanced feature-fused dataset was utilized to classify phishing and legal samples using the Extreme Gradient Boosting (XGBoost) classifier. Optuna-based hyperparameter optimization was used to find the best model parameters in order to enhance the classification performance. With 220 estimators, a maximum tree depth of 8, a learning rate of 0.1055, a subsample ratio of 0.9848, and a column sampling ratio of 0.9161, the optimized XGBoost model was set up. The balanced training dataset was used to train the final optimal model, and the independent test dataset was used to assess it. 98.35% accuracy, 97.90% precision, 98.82% recall, and 98.36% F1-score were all attained by the model. Additionally, the model's ROC-AUC score of 99.86% shows how well it can differentiate between phishing and authentic data. The high recall value is very important for phishing detection because it shows that a considerable percentage of phishing cases are correctly identified by the suggested model. Overall, the findings show that the modified XGBoost classifier offers the suggested phishing detection system very efficient and dependable classification performance.

#### K. Hyperparameter Optimization Using Optuna

Optuna was used to enhance the XGBoost classifier's classification performance by optimizing its hyperparameters. Important XGBoost parameters, such as the number of estimators, maximum tree depth, learning rate, subsampling ratio, and column sampling ratio, are searched for appropriate values during the optimization process. Stratified sampling was used to separate the balanced training dataset into optimization-training and validation subsets. An XGBoost model was trained with a different set of hyperparameters for every Optuna trial, and the optimization goal was the validation accuracy. The parameter combination that produced the best validation accuracy was chosen after five trials in total. The final XGBoost classifier was then trained on the balanced training dataset using the optimized parameters. This automatic optimization procedure offers a methodical way to choose appropriate model parameters and enhance the suggested phishing email detection system's classification performance.

#### L. 5-Fold Stratified Cross-Validation

The suggested categorization model's consistency and stability are assessed using stratified 5-fold cross-validation. The class distribution is preserved in each of the five folds created from the training data. In each iteration, four folds are utilized for training, while the remaining fold is used for validation. The mean and standard deviation of the model performance are determined using the accuracy values that are acquired after five repetitions of this process.

#### M. Explainability of SHAP-Based Models

The optimized XGBoost classifier's predictions were interpreted using SHAP (SHapley Additive exPlanations) to determine which features have the biggest impact on the classification of phishing emails. Individual input features are given contribution values by SHAP according to their impact on the model's output. A feature's contribution to the phishing class is shown by positive SHAP values, while its contribution to the legitimate class is indicated by negative SHAP values. To lower computing needs, a portion of the test data in this study is subjected to SHAP analysis. Information regarding the relative significance and contribution of the characteristics in the fused representation can be found in the ensuing SHAP summary plot. By offering an

understandable perspective of the variables affecting the final phishing email predictions, this analysis enhances the transparency of the suggested model.

### IV. EVALUATION METRICS

The performance of the proposed phishing email detection model is evaluated using Accuracy, Precision, Recall, F1-Score, and ROC-AUC. These metrics provide a comprehensive assessment of the model's ability to correctly distinguish between legitimate and phishing emails.

#### A. Confusion Matrix

The confusion matrix summarizes the classification results using four values:

- TP (True Positive): Phishing emails correctly classified as phishing.
- TN (True Negative): Legitimate emails correctly classified as legitimate.
- FP (False Positive): Legitimate emails incorrectly classified as phishing.
- FN (False Negative): Phishing emails incorrectly classified as legitimate.

- B. Accuracy measures the overall proportion of correctly classified emails.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- C. Precision measures how many emails predicted as phishing are actually phishing.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- D. Recall measures how many actual phishing emails are correctly detected. This is particularly important because missing a phishing email can pose a security risk.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- E. F1-Score is the harmonic mean of Precision and Recall and provides a balanced measure of classification performance.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Equivalent form:

$$\text{F1-Score} = \frac{2TP}{2TP + FP + FN}$$

- F. ROC-AUC measures the model's ability to distinguish between legitimate and phishing classes across different classification thresholds.

A higher AUC indicates better class discrimination.

$$AUC = \int_0^1 TPR(FPR) d(FPR)$$

G. TPR is another name for Recall or Sensitivity.

$$TPR = \frac{TP}{TP + FN}$$

H. FPR measures the proportion of legitimate emails incorrectly classified as phishing.

$$FPR = \frac{FP}{FP + TN}$$

## V. RESULTS AND DISCUSSIONS

Table 1. Performance of the Proposed Model

| S.no | Evaluation Metric | XGBoost (Initial) | Optimized XGBoost |
|------|-------------------|-------------------|-------------------|
| 1    | Accuracy          | 98.03%            | 98.35%            |
| 2    | Precision         | 97.62%            | 97.90%            |
| 3    | Recall            | 98.46%            | 98.82%            |
| 4    | F1-Score          | 98.04%            | 98.36%            |
| 5    | ROC-AUC           | 99.82%            | 99.86%            |

Accuracy, Precision, Recall, F1-Score, and ROC-AUC were used to assess the suggested model. 98.03% accuracy, 97.62% precision, 98.46% recall, 98.04% F1-Score, and 99.82% ROC-AUC were attained with the original XGBoost model. Improved classification and discriminating performance was demonstrated by the model's 98.35% accuracy, 97.90% precision, 98.82% recall, 98.36% F1-Score, and 99.86% ROC-AUC following Optuna tuning.

### A. Confusion Matrix

Prior to and following Optuna optimization, the XGBoost confusion matrix was compared. The model recorded 4,880 TN, 120 FP, 77 FN, and 4,923 TP with 197 mistakes prior to optimization. It reduced errors to 165 after optimization, achieving 4,894 TN, 106 FP, 59 FN, and 4,941 TP. As a result, Optuna optimization decreased false positives and false negatives.

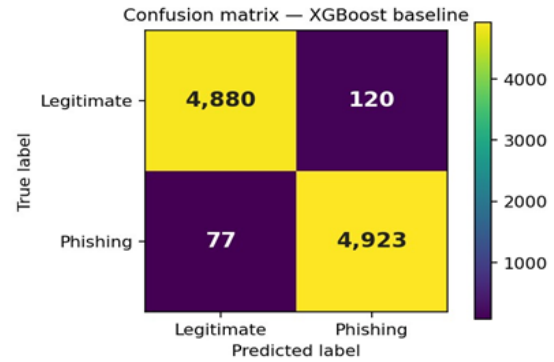


Fig. 2. Confusion Matrix of the XGBoost Classifier

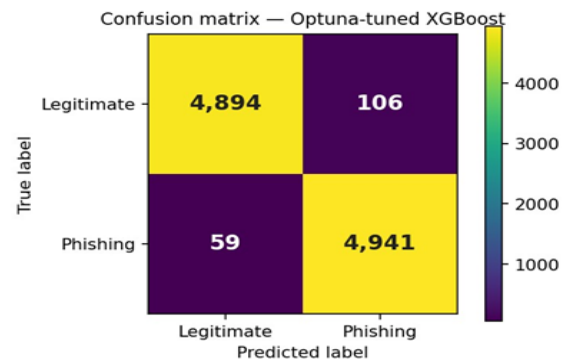


Fig. 3. Confusion Matrix of Optuna-Tuned XGBoost Model After Hyperparameter Optimization

### B. Analysis of ROC Curves

The XGBoost classifiers' ability to discriminate was assessed using the Receiver Operating Characteristic (ROC) curve. The ROC-AUC for the Optuna-tuned configuration was 0.9986, compared to 0.9982 for the default XGBoost. The suggested feature-fusion representation offers a high distinction between authentic and phishing emails, as seen by the near-unity AUC.

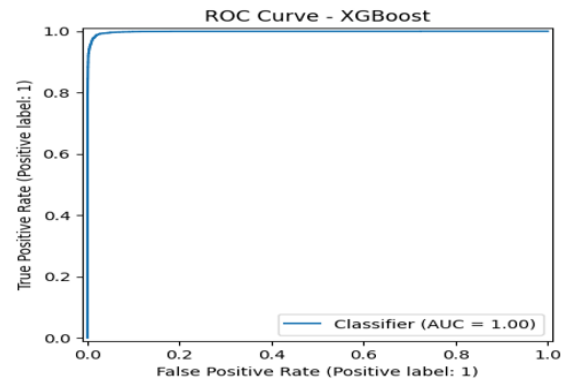


Fig. 4: ROC Curves of Default and Optuna-Tuned XGBoost Classifiers

C. Cross-Validation Result

The balanced training pool was subjected to stratified 5-fold cross-validation in order to assess the classification approach's stability. For each of the five folds, the obtained accuracy was 98.23%, 97.98%, 98.20%, 97.95%, and 98.25%. With a standard deviation of 0.13%, the final mean accuracy was 98.12%. Stable classification performance across several training-validation partitions is indicated by the low fluctuation between folds.

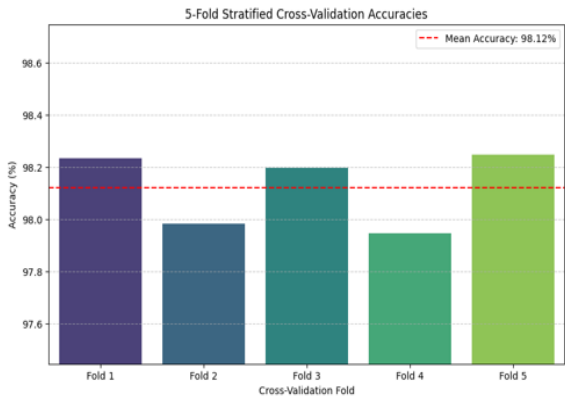


Fig. 5: Per-fold cross-validation accuracies with the mean line

Table 2: Stratified 5-fold cross-validation accuracies

| Fold 1  | Fold 2  | Fold 3  | Fold 4  | Fold 5  | Mean / Std       |
|---------|---------|---------|---------|---------|------------------|
| 98.23 % | 97.98 % | 98.20 % | 97.95 % | 98.25 % | 98.12 % / 0.13 % |

D. SHAP Explainability

A 200-sample subset of the test data was used to apply TreeSHAP to the Optuna-tuned XGBoost classifier in order to enhance the interpretability of the suggested detector. Several RoBERTa embedding dimensions were among the most significant features, according to the SHAP analysis, suggesting that contextual semantic information is crucial for categorization. The URL-count feature

added another metadata signal, and TF-IDF features also helped with the predictions. These results show that the classifier benefits from the contextual representation offered by RoBERTa rather than relying only on keyword frequency.

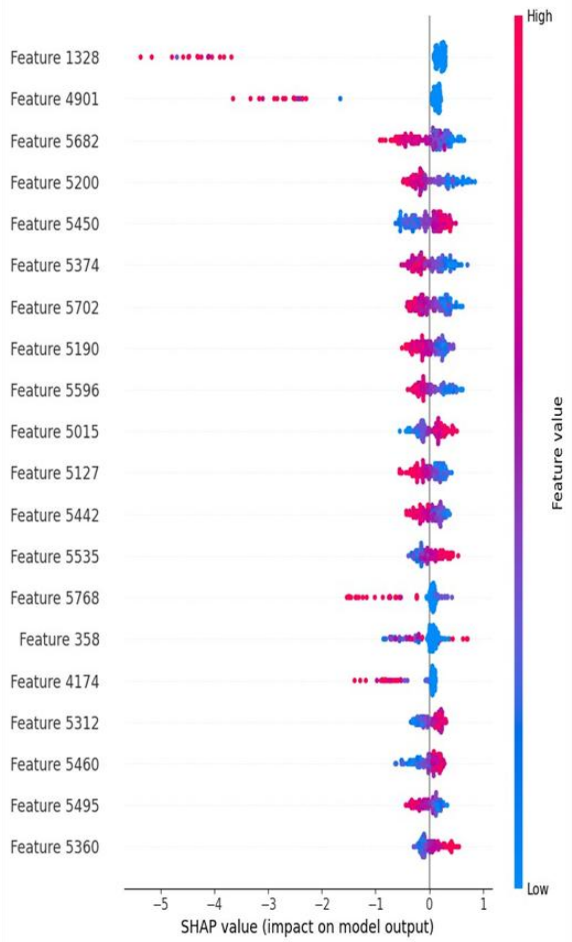


Fig.6: SHAP Feature Importance for the Optuna-Tuned XGBoost Classifier

E. Overall Discussion

Table 3. Comparative Performance of the Evaluated Models

| S.no | Model                | Accuracy | Precision | Recall  | F1-Score | ROC-AUC |
|------|----------------------|----------|-----------|---------|----------|---------|
| 1    | BiLSTM               | 78.04 %  | —         | —       | —        | —       |
| 2    | Default XGBoost      | 98.03 %  | 97.62 %   | 98.46 % | 98.04 %  | 0.9982  |
| 3    | Optuna-Tuned XGBoost | 98.35 %  | 97.90 %   | 98.82 % | 98.36 %  | 0.9986  |

The suggested method uses a combination of URL-count metadata, TF-IDF textual features, and RoBERTa semantic features to identify AI-generated phishing emails. The 50,000 balanced emails in the dataset are split into training and testing sets. For the classification of phishing emails, two classifiers—BiLSTM and XGBoost—were assessed. The default XGBoost obtained 98.03% accuracy, compared to 78.04% for the BiLSTM. The XGBoost model had the highest accuracy of 98.35% after using Optuna hyperparameter tweaking. 97.90% precision, 98.82% recall, 98.36% F1-score, and 0.9986 ROC-AUC were achieved by the optimized model. These findings demonstrate that employing the fused feature representation, the optimized XGBoost model offers very successful phishing email detection.

## VI. CONCLUSION

A transformer-based NLP framework for identifying phishing emails created by AI was provided in this project. To represent email content, the suggested method incorporated URL-based characteristics, frozen RoBERTa embeddings, and TF-IDF features. BiLSTM and XGBoost classifiers were employed to assess the fused features, whereas SMOTE-Tomek was only used on the training set. According to the experimental data, the Optuna-tuned XGBoost performed best with 98.35% accuracy, 97.90% precision, 98.82% recall, 98.36% F1-score, and 0.9986 ROC-AUC, whereas the BiLSTM achieved 78.04% accuracy. Additionally, the model demonstrated stable performance in 5-fold cross-validation with an accuracy of  $98.12\% \pm 0.13\%$ . Overall, the findings show that machine learning and transformer-based semantic features work well together to identify phishing emails created by artificial intelligence.

## REFERENCES

- [1] R. Verma, N. Shashidhar, and N. Hossain, "Detecting phishing emails the natural language way," in *Proc. European Symposium on Research in Computer Security (ESORICS)*, Springer, 2012, pp. 824–841.
- [2] D. Ranganayakulu and C. Chellappan, "Detecting malicious URLs in e-mail—An implementation," *AASRI Procedia*, vol. 4, pp. 125–131, 2013.
- [3] Z. Alkhalil, C. Hewage, L. Nawaf, and I. Khan, "Phishing attacks: A recent comprehensive study and a new anatomy," *Frontiers in Computer Science*, vol. 3, p. 563060, 2021.
- [4] C. S. Eze and L. Shamir, "Analysis and prevention of AI-based phishing email attacks," *Electronics*, vol. 13, no. 10, p. 1839, 2024.
- [5] Altwaijry, I. Al-Turaiki, R. Alotaibi, and F. Alakeel, "Advancing phishing email detection: A comparative study of deep learning models," *Sensors*, vol. 24, no. 7, p. 2077, 2024.
- [6] A. Tawil, L. Almazaydeh, D. S. Qawasmeh, B. Qawasmeh, M. Alshinwan, and K. Elleithy, "Comparative analysis of machine learning algorithms for email phishing detection using TF-IDF, Word2Vec, and BERT," *Computers, Materials & Continua*, vol. 81, no. 2, pp. 3395–3412, 2024.
- [7] R. Zaimi, M. Hafidi, and M. Lamia, "A deep learning mechanism to detect phishing URLs using the permutation importance method and SMOTE-Tomek link," *The Journal of Supercomputing*, vol. 80, no. 12, pp. 17159–17191, 2024.
- [8] M. Antonijevic, L. Jovanovic, N. Bacanin, M. Zivkovic, J. Kaljevic, and T. Zivkovic, "Using BERT with modified metaheuristic optimized XGBoost for phishing email identification," in *Proc. International Conference on Artificial Intelligence and Smart Energy*, Springer, 2024, pp. 358–370.
- [9] Y. Sanjalawe, S. Fraihat, G. Kurdi, and B. Allehyani, "A Cognitive BDI-BiLSTM hybrid framework for interpretable email threat detection," *IEEE Access*, vol. 13, pp. 208192–208211, 2025.
- [10] C. Mayo, M. Tchuindjang, S. Brohi, and N. Ersotelos, "A temporally dynamic feature-extraction framework for phishing detection with LIME and SHAP explanations," *Future Internet*, vol. 18, no. 2, p. 101, 2026.
- [11] A. Alam, A. Al-Subaiey, M. Al-Thani, K. F. Antora, A. Khandakar, and S. A. U. Zaman, "Phishing email dataset: Phish No More—The Enron, Ling, CEAS, Nazario, Nigerian & SpamAssassin datasets," Kaggle, 2024.
- [12] Al-Subaiey, A. Al-Thani, N. A. Alam, K. F. Antora, A. Khandakar, and S. A. U. Zaman,

“Novel interpretable and robust web-based AI platform for phishing email detection,” *Computers & Electrical Engineering*, vol. 119, Art. no. 109625, 2024.

- [13] I. Champa, M. F. Rabbi, and M. F. Zibran, “Curated datasets and feature analysis for phishing email detection with machine learning,” in *Proc. 3rd IEEE International Conference on Computing and Machine Intelligence (ICMI)*, 2024, pp. 1–7.
- [14] R. Gutierrez, W. Villegas-CH, and J. Govea, “Cross-model evaluation of phishing detectors against LLM-generated emails: Dataset, code and results,” Zenodo, 2026, doi: 10.5281/zenodo.20250116.
- [15] Alkhodhairi, G., & Saleem, K. (2025). Machine learning algorithm for detecting suspicious email messages using natural language processing nlp. *Alexandria Engineering Journal*, 128, 153-165.