AI Powered Phishing Email Detection System

Mr. Rueben van der Westhuizen
u21434809

May 16, 2025

Project Repository:

https://github.com/21434809/AI-Powered-Phishing-Detection

Contents

1 Research Overview

1.1 How does AI detect Phishing attacks:

AI is increasingly used to detect phishing attacks by analyzing various email characteristics such as sender behavior, message content, and embedded URLs to identify anomalies or patterns indicative of phishing attempts [1]. Machine learning algorithms, including natural language processing (NLP) and deep learning, are employed to recognize subtle cues that traditional rule based systems might miss [2] [3]. ML algorithms, especially classifiers like Logistic Regression, Random Forests, and Neural Networks, are trained to identify patterns in email content, metadata, and even the tone of the language used can continuously learn from new data, improving their detection accuracy over time without manual updates [4] [5].

1.2 Traditional methods

Compared to traditional methods, AI can handle large volumes of email traffic and detect novel, previously unseen phishing tactics. Its ability to adapt to new attack strategies makes it more effective in protection that is real time [6] [7] [8].

1.3 Advantages

AI powered phishing detection is also less prone to false positives, ensuring legitimate emails are not mistakenly flagged. Additionally, AI systems can automate responses, reducing the need for manual intervention [8]. This leads to faster threat mitigation and enhanced user security [9]. By integrating AI, organizations can build a more robust defense against the evolving landscape of phishing threats [10] [11].

2 Model selection process

Selection of Kaggle's Phishing Email Dataset: After close inspection, a combination off the "CEAS 08.csv" dataset, "NigerianFraud.csv", "SpamAssasin.csv", and "Nazario.csv" from has approximately 48,762 emails sum total after cleaning.

2.1 Data Preprocessing and Feature Extraction

- Subject: Has sufficient data without any missing values
- Body: Has sufficient data without any missing values
- label (Spam or legitimate)

2.2 Machine Learning Models Considered

- Logistic Regression
- Random Forest
- Neural Networks

2.3 Model Selection Justification

The model selection process for phishing email detection begins with evaluating the type of data available (email content, sender information, subject line, etc.) and the complexity of the task. Given that phishing emails often contain subtle cues, we need models that can identify these implied meaning in unstructured (text) data.

Initially, simpler models like Logistic Regression (LR) and Random Forest (RF) are considered. LR serves as a baseline because of its simplicity and interpretability, making it easier to understand which features are influencing the predictions. However, it may struggle with capturing nonlinear relationships and complex interactions between features.

On the other hand, Random Forest is an ensemble method that performs better on diverse datasets because it combines multiple decision trees, which helps in reducing overfitting. It is also able to capture nonlinear relationships in the data, making it more robust than LR.

For more complex patterns in the data, we consider Neural Networks (NN). Specifically, Multilayer Perceptrons (MLPs) or Recurrent Neural Networks (RNNs) are employed for text data, as they excel at recognizing sequential patterns in the email content, such as unusual phrasing or tone that might signal phishing. Additionally, long-short-term memory (LSTM) networks are considered for their ability to learn from sequences, making them highly effective for analyzing email texts with contextual meaning.

Model performance using metrics such as Accuracy, Precision, Recall, and F1-Score is crucial in evaluating phishing email detection systems. While Random Forest and Neural Networks typically offer the best results in terms of recall (catching phishing emails), Neural Networks—especially those with deep learning layers—tend to perform exceptionally well in detecting subtle, unseen phishing tactics. However, these models generally require longer training times, especially deep learning models, due to the complexity and the large amount of data they process during training.

Ultimately, Random Forest was selected for its balance between performance and interoperability and also provide feature importance scores.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**Notes**
Logistic Regression	98.08	97.88	98.50	98.19	Simple, interpretable, fast
Random Forest	98.02	97.52	97.87	97.69	Handles non-linear data well
Neural Networks	99.40	98.70	99.00	98.80	Captures complex patterns

3 Requirements Specification

The AI Powered Phishing Email Detection System is designed as a full stack web application that enables users to submit email content and receive real time predictions on whether the email is phishing or legitimate. The system leverages a machine learning backend and a user friendly frontend to provide accurate, fast, and interpretable results.

3.1 Functional Requirements

- Users can submit email text (subject and body) via a web interface.
- The system processes the input and returns a prediction (phishing/legitimate) with a confidence score.
- The backend exposes a REST API endpoint for predictions.
- The frontend displays results, highlights suspicious words, and visualizes confidence.
- The system supports retraining and updating the ML model with new data.

3.2 Non-Functional Requirements

- The system must respond to prediction requests within 2 seconds.
- The model must achieve at least 97% accuracy on the test set.
- The web interface must be accessible and responsive on desktop and mobile devices.
- The backend and frontend must be easily deployable on standard Windows or Linux environments.
- Security: Only sanitized input is processed; no user data is stored.

3.3 System Components

- **Frontend:** Angular SPA for user interaction.
- **Backend:** Flask REST API for ML inference.
- **ML Model:** Trained Random Forest, serialized as `model.pkl`.
- **Data:** Preprocessed datasets from multiple sources (CEAS, NigerianFraud, SpamAssassin, Nazario).

3.4 UML Diagrams

- **System Architecture:** Shows the interaction between frontend, backend, and model.

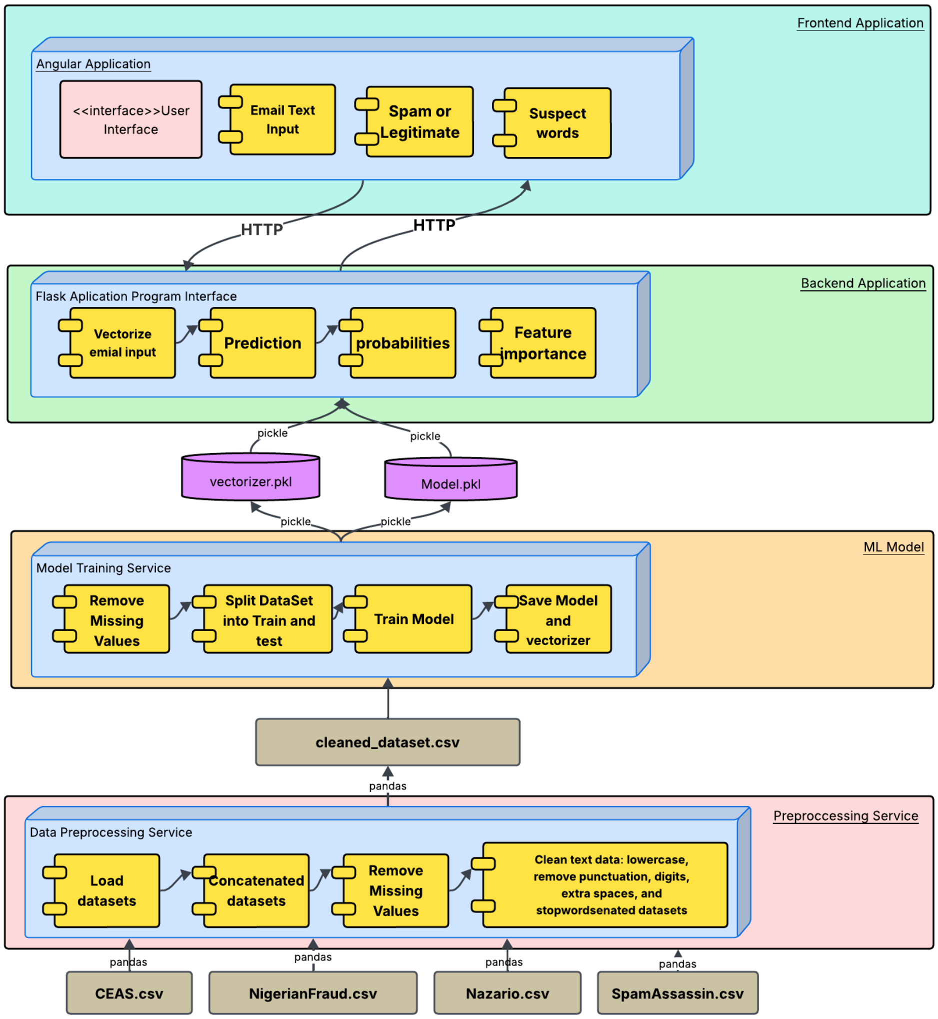


Figure 1: System Architecture of the AI Powered Phishing Email Detection System

- **Workflow:** Sequence of user input, API call, prediction, and result display.

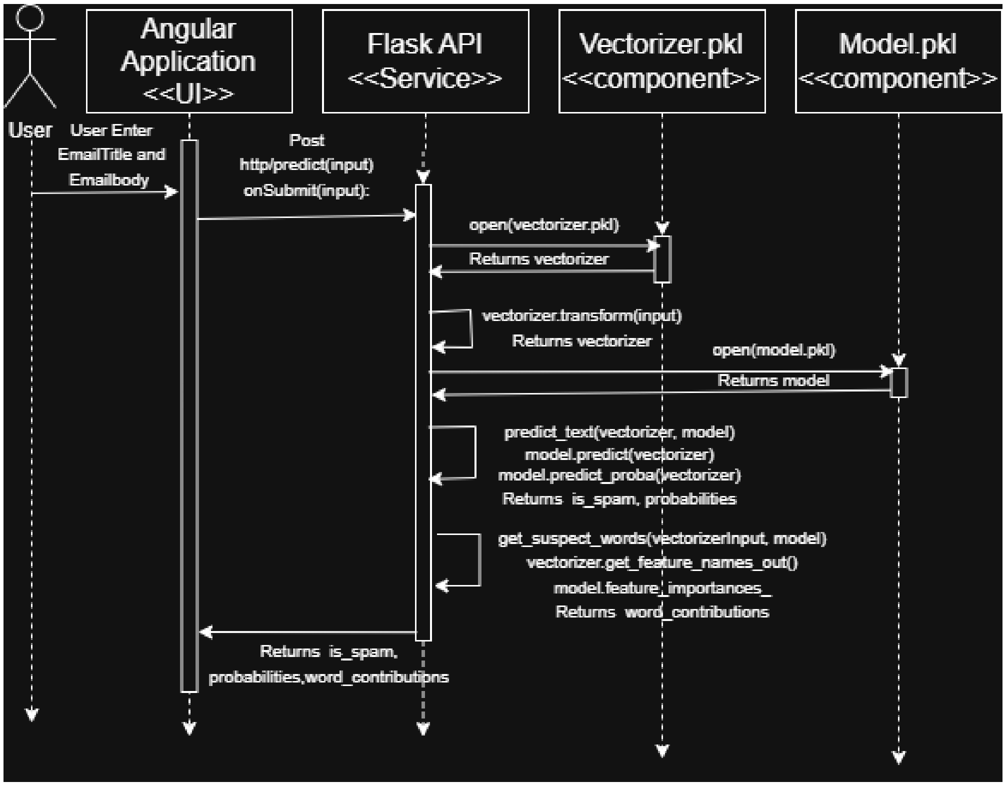

User
Angular Application <<UI>>
Flask API <<Service>>
Vectorizer.pkl <<component>>
Model.pkl <<component>>
User Enter EmailTitle and Emailbody
Post http/predict(input) onSubmit(input):
open(vectorizer.pkl)
Returns vectorizer
vectorizer.transform(input)
Returns vectorizer
open(model.pkl)
Returns model
predict_text(vectorizer, model)
model.predict(vectorizer)
model.predict_proba(vectorizer)
Returns is_spam, probabilities
get_suspect_words(vectorizerInput, model)
vectorizer.get_feature_names_out()
model.feature_importances_
Returns word_contributions
Returns is_spam, probabilities,word_contributions

Figure 2: Workflow of the AI Powered Phishing Email Detection System

4 Testing and Evaluation

4.1 Testing Methodology

We conducted several tests to evaluate the prototype's performance in realistic scenarios. Each test involved using either real or simulated emails to observe the system's behavior:

1. **Known Phishing Example:** We used a known phishing email from the Gmail spam folder. The test performed well, with the model correctly identifying the email as phishing.

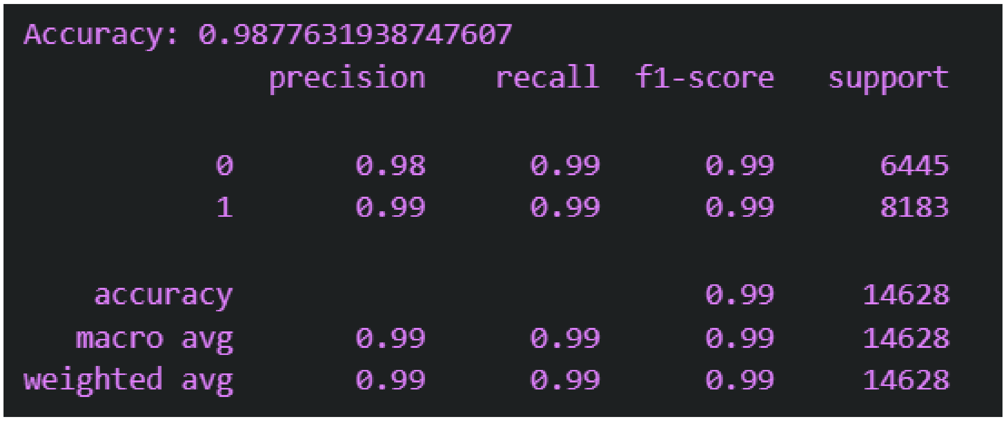
2. **Legitimate Email Example:** We used a known legitimate email from the Gmail inbox. The test performed fairly, with the model correctly identifying the emails as legitimate. However, emails that follow a marketing template can be flagged as phishing, but with a low confidence score.

3. **Curated Spear Phishing Email:** We used a curated spear phishing email based on a real legitimate email. The test performed fairly, with the model leaning more toward legitimate, but with a low confidence score and clear explanation of the words that were flagged as legitimate.

4. **Foreign Language Email:** We used a foreign language email based on a real legitimate email. The test performed poorly; it identified both legitimate and phishing emails as phishing, with a high confidence score and no clear explanation of the flagged words.

4.2 Evaluation Metrics

We evaluated based on accuracy and and a classification report on 48,762 emails. We also used confusion matrix to visualize the performance of the false positives and false negatives. In total there were only 111 false negatives and 68 false positives. The model is more likely to classify a legitimate email as phishing.



```
Accuracy: 0.9877631938747607
              precision    recall  f1-score   support

           0       0.98      0.99      0.99      6445
           1       0.99      0.99      0.99      8183

    accuracy                           0.99     14628
   macro avg       0.99      0.99      0.99     14628
weighted avg       0.99      0.99      0.99     14628
```

Figure 3: Classification Report of the AI Powered Phishing Email Detection System

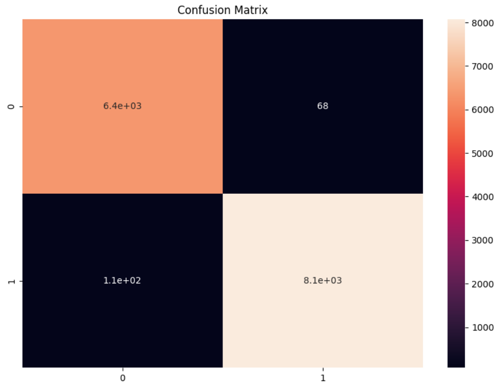


Figure 4: Confusion Matrix of the AI Powered Phishing Email Detection System

5 Conclusion

5.1 Summary of AI Principles

The AI Powered Phishing Email Detection System leverages machine learning, specifically a Random Forest classifier, to distinguish between phishing and legitimate emails. The system uses natural language processing (NLP) techniques to extract features from the subject and body of emails. By training on a large, labeled dataset, the model learns to identify patterns and cues indicative of phishing attempts. The backend exposes a REST API for real-time inference, while the frontend provides an interactive interface for users to submit emails and receive predictions.

5.2 Limitations

Despite its strong performance, the system has several limitations:

- **Dataset Bias:** The model is trained on public datasets that may not fully represent the diversity of real world emails, potentially reducing its effectiveness on novel or sophisticated phishing attacks.
- **False Positives:** Legitimate marketing emails and foreign language emails are sometimes misclassified as phishing, especially with high confidence, indicating limited generalization to non English or template based emails.
- **Limited Feature Set:** The model primarily uses the subject and body text, ignoring other useful features such as sender reputation, email headers, embedded URLs, and attachments.
- **Static Model:** The model does not learn continuously and requires manual retraining to adapt to new phishing tactics.
- **Language and Context:** The model struggles with non English emails and may not capture cultural or contextual nuances, leading to misclassification.

5.3 Suggested Improvements

To address these limitations, the following improvements are recommended:

- **Expand and Diversify the Dataset:** Continuously collect and incorporate new, real-world phishing and legitimate emails, including multilingual and region-specific samples.
- **Feature Engineering:** Incorporate additional features such as sender domain analysis, URL reputation, attachment scanning, and email header inspection.
- **Continuous Learning:** Integrate user feedback with permission to retrain the model to adapt to new threats with minimal manual intervention.

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, pp. 139–154, 2021.

[2] D. Wang, J. Su, and H. Yu, “Feature extraction and analysis of natural language processing for deep learning english language,” *IEEE Access*, vol. 8, pp. 46 335–46 345, 2020.

[3] I. Lauriola, A. Lavelli, and F. Aiolli, “An introduction to deep learning in natural language processing: Models, techniques, and tools,” *Neurocomputing*, vol. 470, pp. 443–456, 2021.

[4] A. Karim, S. Azam, B. Shanmugam, K. Kannoorpatti, and M. Alazab, “A comprehensive survey for intelligent spam email detection,” *IEEE Access*, vol. 7, pp. 168 261–168 295, 2019.

[5] Y. S. Murti and P. Naveen, “Machine learning algorithms for phishing email detection,” *Journal of Logistics, Informatics and Service Science*, 2023.

[6] S. Jalil, M. Usman, and A. Fong, “Highly accurate phishing url detection based on machine learning,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, pp. 9233–9251, 2022.

[7] Y. Alsariera, V. E. Adeyemo, A. Balogun, and A. K. Alazzawi, “Ai meta-learners and extra-trees algorithm for the detection of phishing websites,” *IEEE Access*, vol. 8, pp. 142 532–142 542, 2020.

[8] 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, pp. 139 – 154, 2020.

[9] M. Sameen, K. Han, and S. Hwang, “Phishhaven—an efficient real-time ai phishing urls detection system,” *IEEE Access*, vol. 8, pp. 83 425–83 443, 2020.

[10] S. Kumar, A. Menezes, S. Giri, and S. D. Kotikela, “What the phish! effects of ai on phishing attacks and defense,” *International Conference on AI Research*, 2024.

[11] P. Chinnasamy, P.Krishnamoorthy, K. Alankruthi, T. Mohanraj, B. Kumar, and L. Chandran, “Ai enhanced phishing detection system,” *2024 Third International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, pp. 1–5, 2024.