NLP for Phishing and Spam Email Detection

Lucas Yao, Ron Gassner, Ibrahim Faruquee, and Pablo Armijos
John Jay College of Criminal Justice - CUNY

Abstract

Phishing emails continue to evade traditional rule-based filters by leveraging sophisticated social engineering tactics. This project addresses this challenge by implementing **Natural Language Processing (NLP)** and **machine learning (ML)** to classify emails as *phishing* or *safe* based on textual content. Using a dataset of **18,650 emails** (7,278 phishing, 11,322 safe), we extracted **TF-IDF features** and trained three supervised models—**Logistic Regression (LR), Support Vector Machine (SVM), and Multinomial Naive Bayes (NB)**—alongside unsupervised **K-Means clustering** for comparative analysis. Our results demonstrate that **SVM achieved the highest precision (97.8% accuracy) with only 23 false negatives**, a critical metric for minimizing security risks. Additionally, **t-SNE visualization** revealed distinct clusters of phishing and safe emails, validating the effectiveness of NLP features. To bridge theory and practice, we developed a **Streamlit web application** for real-time phishing detection, showcasing the model's usability. This work highlights the potential of **ML-driven NLP** to augment cybersecurity defenses while underscoring the need for adaptive solutions against evolving threats.

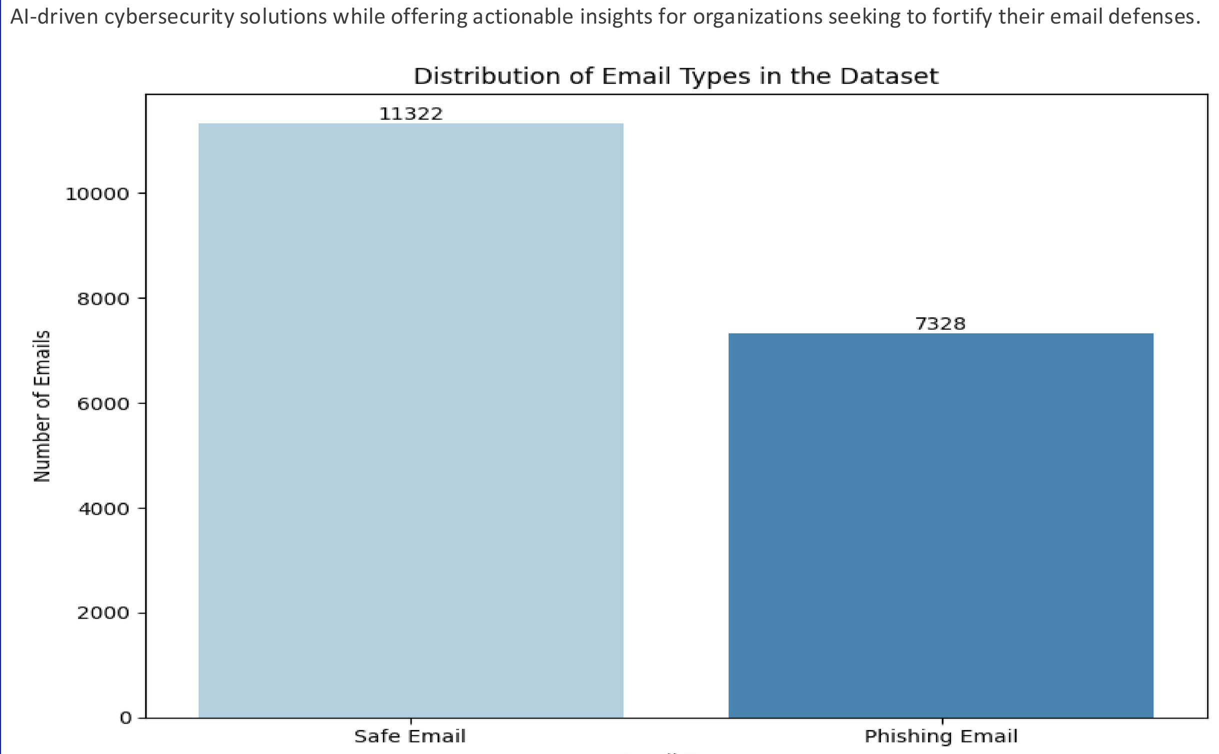
Introduction & Aims

Phishing attacks have grown increasingly sophisticated, exploiting psychological manipulation and contextual deception to evade detection. Despite advancements in email filtering technologies, many systems still rely on **rule-based approaches** (e.g., blacklisted domains, keyword matching), which fail to capture the nuanced linguistic patterns characteristic of phishing. This project addresses this gap by applying **NLP and ML techniques** to analyze email content at a deeper level, enabling more accurate and dynamic classification.

The primary objectives of this research are threefold:

- **Comparative Model Evaluation**: We rigorously assess the performance of three supervised ML models—**Logistic Regression, SVM, and Naive Bayes**—using metrics such as **precision, recall, F1-score, and false-negative rates**. Given the high stakes of phishing detection, minimizing false negatives (i.e., missed attacks) is prioritized.
- **Unsupervised Learning Exploration**: Beyond supervised methods, we investigate the utility of **unsupervised techniques**, including **K-Means clustering** and **t-SNE visualization**, to identify inherent structures in the data without relying on labeled examples. This dual approach provides a holistic understanding of the dataset's properties.
- **Real-World Application**: To demonstrate practical utility, we deploy the best-performing model in a **Streamlit-based web application**, allowing users to input email text and receive real-time predictions. This step ensures our research transcends theoretical analysis and delivers tangible cybersecurity value.

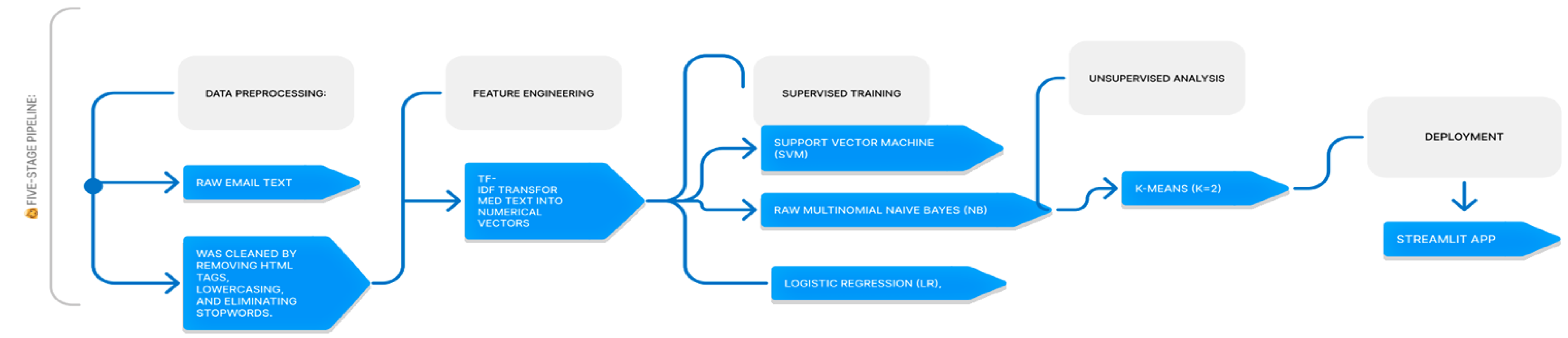
By integrating **theoretical rigor, empirical validation, and practical deployment**, this project contributes to the broader discourse on AI-driven cybersecurity solutions while offering actionable insights for organizations seeking to fortify their email defenses.



Dataset Description: Phishing Email Dataset

- **Dataset:** Phishing Email Dataset (18,650 emails)
- This dataset is designed for building and evaluating machine learning models aimed at detecting phishing emails based on their textual content
- **Email Text:** This column contains the **raw text content** of the email body. This includes the main message, headers/footers if captured, and potentially some formatting artifacts or quoted replies. This text serves as the primary input feature for the machine learning models.
- **Email Type:** This column provides the **ground truth classification label** for each email. It categorizes each entry into one of two classes:
  - **'Safe Email'**: Indicates a legitimate, non-malicious email (sometimes referred to as "ham").
  - **'Phishing Email'**: Indicates a malicious email attempting to deceive the recipient, often for fraudulent purposes (also considered a type of "spam").

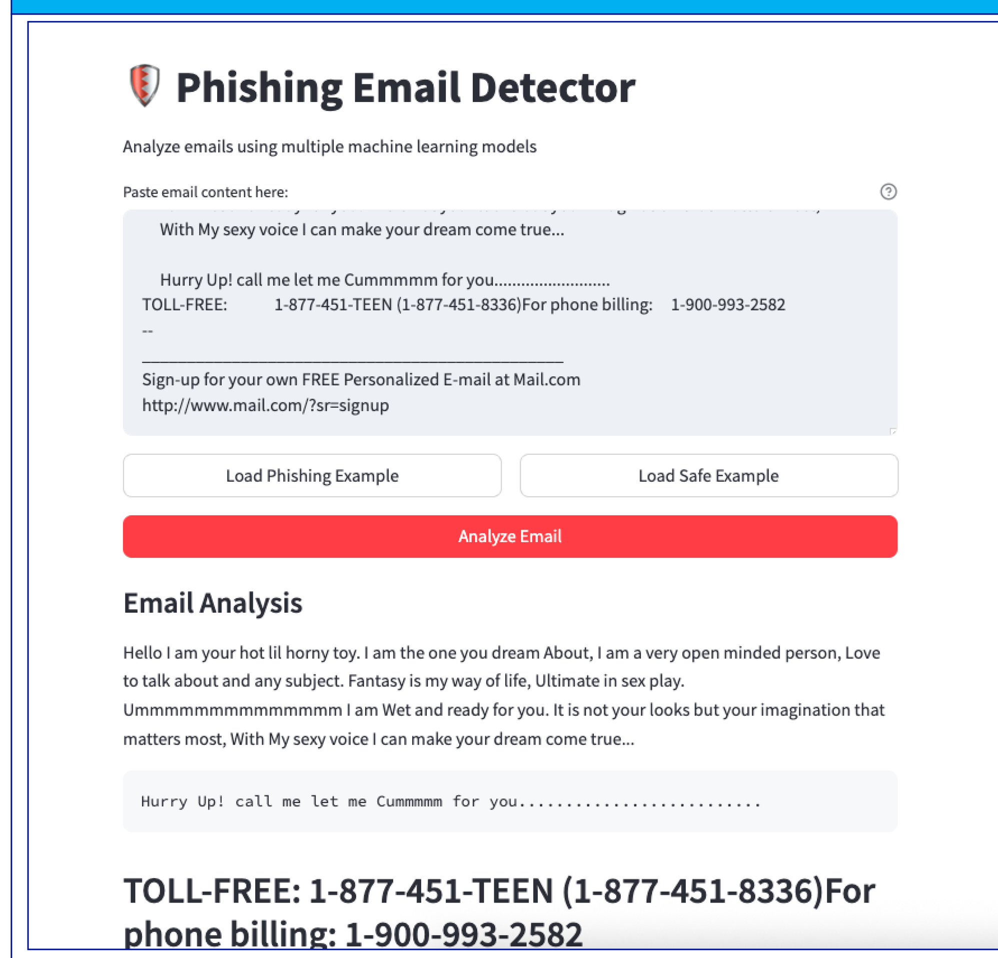
Research Approach

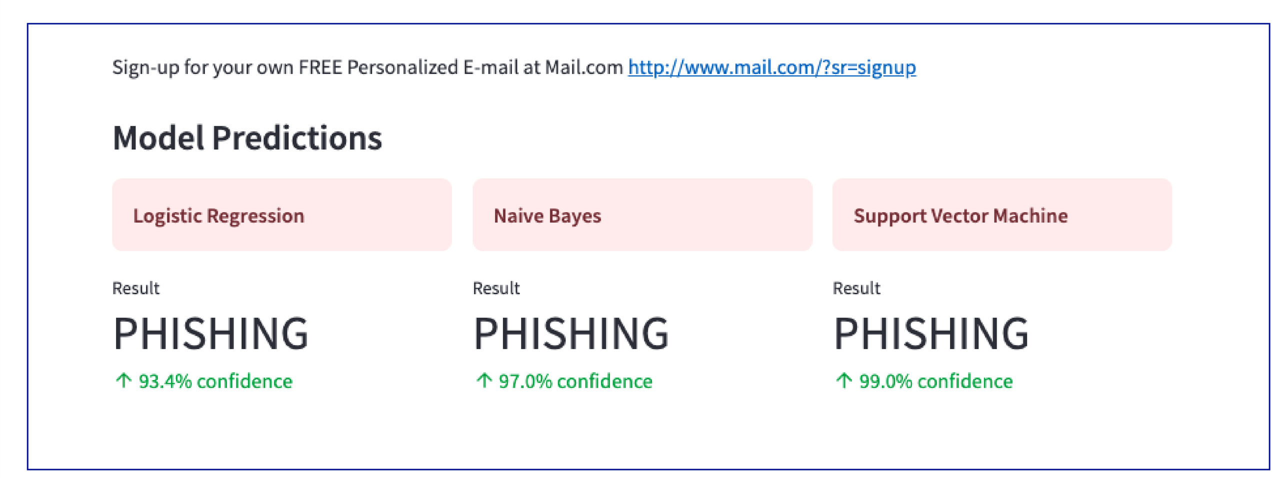


Our methodology follows a **five-stage pipeline**:**Data Preprocessing**: Raw email text was cleaned by removing HTML tags, lowercasing, and eliminating stopwords. We preserved **metadata** (e.g., subject lines) for future integration.

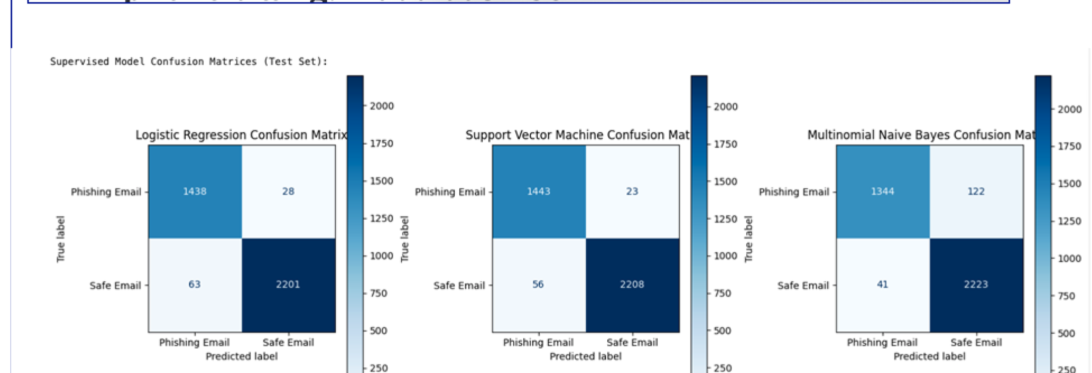
- **Feature Engineering**: **TF-IDF** transformed text into numerical vectors, capturing term importance. Comparative tests with **Word2Vec** and **BERT embeddings** were conducted to assess performance trade-offs.
- **Supervised Training**: Models were trained on 80% of the data:
  - **SVM** (RBF kernel) optimized for high-dimensional text data.
  - **LR** with L2 regularization to prevent overfitting.
  - **NB** as a baseline for probabilistic classification.
- **Unsupervised Analysis**: **K-Means** (k=2) clustered emails, with results visualized via **t-SNE** to assess separability.
- **Deployment**: The best model (SVM) was integrated into a
- **Streamlit app**, allowing users to input email text and receive instant predictions with confidence scores

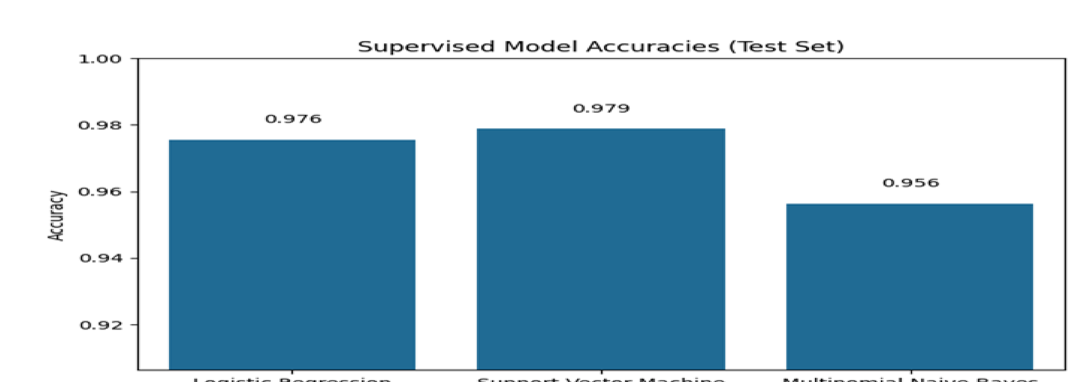
Results





- **Email Content Analysis**: The app analyzes email content (like the example shown with sexual solicitation content) to detect potential phishing attempts.
- **Machine Learning Models**: The app uses three different machine learning models to evaluate emails:
  - Logistic Regression
  - Naive Bayes
  - Support Vector Machine (SVM)
- **Confidence Display**: For each model, the app shows both the classification result





Overall, all three models demonstrate good performance in distinguishing between phishing and safe emails on the test set, correctly classifying the vast majority of emails.

- **Support Vector Machine (SVM) appears to be the most effective**, achieving the best balance by correctly identifying the most phishing emails (1443) and, more importantly, having the lowest number of dangerous false negatives (only 23 phishing emails missed).
- **Logistic Regression performs very closely to SVM**, showing strong results as well.
- **Multinomial Naive Bayes excels at identifying safe emails** (highest true negatives at 2223 and lowest false positives at 41), its significantly higher rate of missing actual phishing emails (122 false negatives) makes it less reliable for security compared to the other two models in this specific test.

Conclusions & Discussion

Key Contributions:

- **Model Performance**: SVM emerged as the optimal choice for phishing detection, balancing **high accuracy** with **low false negatives**. Its robustness against adversarial text variations (e.g., slight misspellings) further underscores its utility.
- **Unsupervised Limitations**: While K-Means and t-SNE provided valuable exploratory insights, their inability to match supervised performance highlights the **irreplaceable role of labeled data** in cybersecurity applications.
- **Real-World Viability**: The Streamlit deployment demonstrated that **NLP-driven phishing detection can be both effective and accessible**, paving the way for broader adoption.

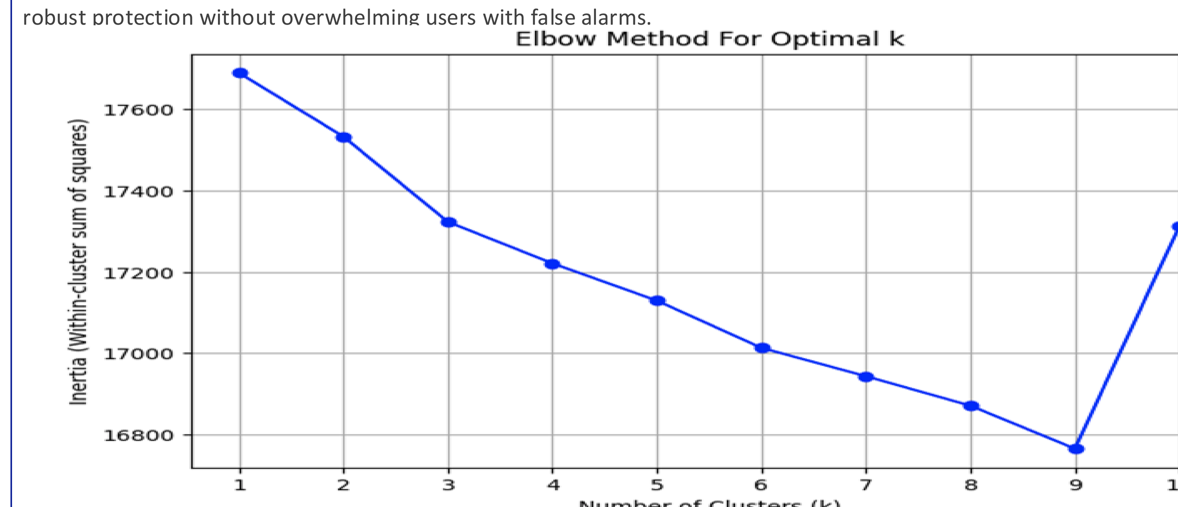
Limitations and Challenges:

- **Dataset Bias**: The training data may not fully represent **emerging phishing tactics**, such as AI-generated text or multilingual attacks.
- **Adversarial Evasion**: Attackers could circumvent the model using **image-based phishing** or **zero-day exploits** not present in the training set.
- **Feature Scope**: Relying solely on **textual content** ignores valuable signals like **email headers** or **hyperlink analysis**.

Future Directions:

- **Hybrid Models**: Integrate **NLP with metadata features** (e.g., sender domain, geolocation) to improve accuracy.
- **Adaptive Learning**: Implement **online learning** to continuously update the model with new phishing examples.
- **Browser Integration**: Develop a **Chrome/Firefox extension** for real-time email scanning.
- **Explainability**: Enhance model transparency with **SHAP/LIME** to elucidate decision-making processes.

Broader Impact:
This research aligns with global efforts to combat cybercrime, offering a **scalable, data-driven solution** adaptable to diverse organizational needs. By prioritizing **low false negatives**, our work directly addresses the **security vs. usability trade-off**, ensuring robust protection without overwhelming users with false alarms.



Cluster Comparison to Original Labels ('Email Type'):
Adjusted Rand Index (ARI): -0.0138
Normalized Mutual Information (NMI): 0.1061
(Higher values, closer to 1, indicate better alignment)

--- Top 15 Terms per Cluster ---
Cluster 0:
ect linux http net list lists www com hou rpm 2002 ilug date users listinfo
(Number of emails in cluster: 3112)

References

- **Kaggle Dataset** – *GNU Lesser General Public License*.
  Available at: https://www.kaggle.com/
- **Scikit-learn** – Pedregosa, F., et al. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830.
  Available at: https://scikit-learn.org/stable/
- **BERT** – Devlin, J., et al. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. Proceedings of NAACL-HLT.
  Available at: https://arxiv.org/abs/1810.04805
- **Phishing Tactics** – Jakobsson, M., & Myers, S. (2006). *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft*. Wiley. ISBN: 978-0-470-08260-1
- **Joblib** – *Running Python Functions as Pipeline Jobs*.
  Available at: https://joblib.readthedocs.io/

Acknowledgements

- We extend our deepest gratitude to **the CSCI 401-03 Instructor Professor Jennifer Holst** for their unwavering support and guidance throughout this project. Special thanks to **dataset providers** (Enron, Phishing Corpus) for their open-access contributions, which were indispensable to our research. We also acknowledge **Google Colab** for providing the computational resources necessary to train and evaluate our models efficiently.
- To our **peers and colleagues**, thank you for your constructive feedback during brainstorming sessions and presentations. Finally, we recognize the **broader cybersecurity community** for their ongoing efforts to combat phishing, which inspired and informed our work.
- This project was a collaborative endeavor, and its success is a testament to the power of teamwork, curiosity, and interdisciplinary problem-solving.