MINISTRY OF EDUCATION AND TRAINING
FPT UNIVERSITY

ARTIFICIAL INTELLIGENCE APPLICATIONS IN PHISHING EMAIL CLASSIFICATION

by

Pham Trong Kha

A thesis submitted in conformity with the requirements
for the degree of Master of Software Engineering

MINISTRY OF EDUCATION AND TRAINING
FPT UNIVERSITY

ARTIFICIAL INTELLIGENCE APPLICATIONS IN PHISHING EMAIL CLASSIFICATION

by

Pham Trong Kha

A thesis submitted in conformity with the requirements
for the degree of Master of Software Engineering

Supervisor:

Dr. Le Thanh Hai

Artificial Intelligence Applications in Phishing Email Classification

Pham Trong Kha

Degree Master of Software Engineering

FPT University

2024

Abstract

With the increasing sophistication of phishing attacks, the need for advanced security measures in email communication has become critical. Traditional rule-based systems are often inadequate in detecting complex phishing patterns, prompting the integration of Artificial Intelligence (AI) for more accurate detection. This thesis presents the development of a web-based application designed for system administrators, featuring AI integration for phishing email classification.

Leveraging a dataset of phishing and legitimate emails from Kaggle, the system applies various machine learning models, including Naive Bayes, Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU), to enhance email security. The architecture incorporates a frontend built with HTML, CSS, and JavaScript, and a backend using Python's Flask framework, interfacing with a MySQL database for email management.

The system is containerized using Docker to ensure scalability and portability. Key AI models are trained using TensorFlow and deployed to classify incoming emails. The application integrates Microsoft Azure Active Directory for secure user authentication, while real-time phishing alerts are provided through email analysis.

By evaluating model performance through accuracy, precision, recall, and F1-score, the system provides a robust solution for detecting phishing emails in real-time. This research demonstrates the practicality of AI-powered solutions in enhancing email security and offers a scalable tool for system administrators to combat phishing threats.

Acknowledgments

I would like to express my deepest gratitude to my supervisor, Dr. Le Thanh Hai, for his invaluable guidance throughout this project. I am also thankful to my classmates for their helpful advice and support. A special thanks to my wife and my company for their unwavering support and for providing the necessary resources and encouragement to complete this work.

I am also grateful for the knowledge gained from other courses, which contributed to the success of this thesis. Lastly, I would like to thank FPT University for providing an excellent learning environment and offering courses that are closely aligned with the subject matter of my research.

Table of Contents

Contents

1 Reasons for Choosing the Topic

Email is one of the most widely used communication tools for both personal and professional exchanges. However, it has also become a major vector for cyberattacks, particularly phishing attacks. Phishing is a type of cybercrime in which attackers deceive individuals into revealing sensitive information, often leading to data breaches and financial losses. In 2023, it was reported that phishing emails accounted for over 90% of global cyberattacks, making it one of the most pressing cybersecurity issues today [1].

Traditional methods for detecting phishing emails, such as rule-based filtering, are often insufficient in dealing with the sophisticated nature of modern phishing techniques. These systems rely on static rules or keyword matching, which attackers can easily bypass by altering their email patterns. This leads to high false-positive rates, where legitimate emails are incorrectly flagged, and, worse, some phishing emails may go undetected, putting organizations at significant risk [2]. As phishing techniques evolve, there is an increasing need for more advanced detection mechanisms that can adapt to new types of attacks.

Recent advancements in Artificial Intelligence (AI) and Machine Learning (ML) have opened up new possibilities for addressing these challenges. AI, particularly deep learning models like Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), has shown considerable promise in improving phishing detection by identifying complex patterns and behaviors within email content that traditional methods miss [3]. These models can learn from large datasets, continuously improving their ability to distinguish phishing emails from legitimate ones.

Moreover, the COVID-19 pandemic has accelerated the shift towards digital communication, with remote work becoming the norm. This has resulted in a significant increase in phishing attacks, with reports indicating a 220% rise in phishing activity during the pandemic [4]. Cybercriminals have exploited this shift by targeting individuals and organizations with more personalized and convincing phishing emails, making the need for robust detection systems even more urgent.

Given these developments, this research focuses on building a web-based application that integrates AI models for classifying phishing emails in real-time. By utilizing advanced machine learning models like Naive Bayes, LSTM, and GRU, the proposed system aims to provide an effective solution for system administrators, improving their ability to detect and mitigate phishing attacks. This approach not only addresses the limitations of traditional systems but also offers scalability and adaptability, making it suitable for modern cybersecurity environments.

2 Research Objectives

The primary goal of this research is to design and implement a scalable web-based application that provides system administrators with real-time phishing email detection using AI models. The technical architecture of the system can be summarized in the following objectives:

- **Develop a web-based system** for phishing email detection that includes a frontend, backend, and a database for managing email data. The system is built using Flask for the backend, with a frontend developed in HTML, CSS, and JavaScript, and MySQL for data management.

- **Implement AI models** such as Naive Bayes, LSTM, and GRU for phishing classification. These models are trained and deployed using TensorFlow, and Docker is used for containerization, ensuring scalability and deployment across different environments.

- **Integrate Microsoft Azure for authentication and email handling:**

  Authentication: Azure Active Directory is used to provide secure authentication for system administrators, ensuring that only authorized personnel can access the system.

  Email Handling: Azure is also responsible for connecting to the Microsoft Exchange server to retrieve incoming emails. These emails are then processed through the AI models for phishing detection, and the classification results (phishing or legitimate) are updated back into the user's mailbox for immediate action.

- **Provide real-time email classification:** The system processes incoming emails in real-time, classifying them as phishing or legitimate and immediately alerting system administrators to any phishing threats.

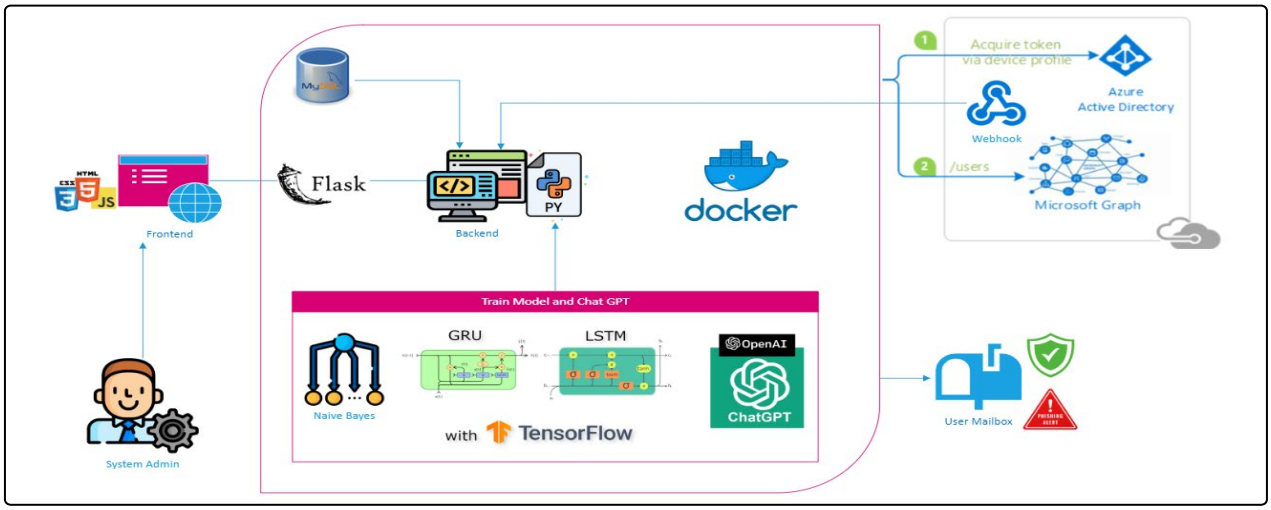

Figure 2.1: Technical Architecture of the Web-Based Phishing Detection System

The **Technical Architecture** of the system, as illustrated in Figure 2.1, presents an overview of how the frontend, backend, AI models, and real-time detection processes are integrated to form a cohesive phishing detection solution.

3 Research Scope and Limitations

3.1 Scope of Research

The primary subject of this research is the development of a web-based application for phishing email classification, aimed at enhancing email security for system administrators. The research focuses on the following key areas:

1. **AI-Powered Phishing Detection**:
    - The core of the research is to integrate advanced machine learning and deep learning models for detecting phishing emails. Models such as Naive Bayes, Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) are used to classify emails as either phishing or legitimate.
2. **Real-Time Email Classification**:
    - The system is designed to operate in real-time, retrieving emails from Microsoft Exchange servers via Microsoft Azure, running them through AI models, and

immediately updating the classification back into the user's mailbox. This real-time aspect is crucial for system administrators to mitigate threats as they arise.

3. **Web-Based Application for System Administrators**:
    - The application provides a user-friendly web interface for system administrators, enabling them to monitor email traffic, view phishing detection results, and receive real-time alerts. The frontend is built with HTML, CSS, and JavaScript, and the backend is powered by Python's Flask framework.
4. **Integration of Microsoft Azure**:
    - Microsoft Azure plays a pivotal role by offering secure authentication through Azure Active Directory and managing email retrieval from Microsoft Exchange servers. This ensures that only authorized users can access the system and that emails are processed securely.

3.2 Limitations of Research

Despite the extensive scope of the research, certain limitations are acknowledged:

1. **Dataset Dependency**:
    - The research relies on publicly available datasets, such as the phishing email dataset from Kaggle [10]. While this dataset is diverse, it may not capture all types of phishing techniques encountered in real-world environments. Additionally, the accuracy of the models may be influenced by the quality and diversity of the training data.
2. **Model Generalization**:
    - The AI models implemented (Naive Bayes, LSTM, GRU) are trained on the selected dataset and may not generalize well to emails containing novel phishing techniques that are not represented in the training data. Continuous updates and retraining of models will be required as phishing tactics evolve [11].

3. **Real-Time Performance**:
    - While the system is designed to operate in real-time, performance limitations may arise depending on the volume of incoming emails and the computational resources available. Scaling the system to handle large volumes of emails efficiently will require the use of cloud-based resources, such as those provided by Azure or similar platforms [12].
4. **Security Considerations**:
    - Although the system leverages Azure Active Directory for secure authentication, potential vulnerabilities may arise if the system is not regularly updated or if other security layers are compromised. Ensuring that the system remains secure against evolving cyber threats will require ongoing maintenance and security updates.
5. **Phishing Detection Focus**:
    - The scope of the research is limited to phishing email detection. Other forms of cyberattacks, such as malware or ransomware delivered through email, are outside the scope of this study. However, the system could be extended in future research to detect a broader range of email-based threats.

4 Research Methodology

The research methodology outlines the specific processes and techniques used to achieve the objectives of the study. This section includes the methods for data collection, preprocessing, model selection, and system implementation.

4.1 Data Collection and Dataset

For this research, the primary data source is the **Phishing Email Dataset** obtained from Kaggle [13]. This dataset contains both phishing and legitimate emails, providing a comprehensive foundation for training and testing machine learning models. The dataset includes various features, such as the email subject, body content, and metadata (e.g., sender address), which are critical for identifying patterns related to phishing emails.

The dataset consists of:

- **Phishing emails**: Emails designed to deceive users into disclosing personal information or installing malicious software.
- **Legitimate emails**: Regular emails that do not contain harmful content.

4.2 Data Preprocessing

Before feeding the data into the AI models, a series of preprocessing steps are required to clean and prepare the email data for analysis:

1. **Text Cleaning**: Remove unnecessary characters, stop words, and HTML tags from the email content using libraries like BeautifulSoup and NLTK [14].
2. **Tokenization**: Split the email content into individual tokens (words) for analysis.
3. **Feature Extraction**: Transform the text data into numerical features that can be used by machine learning models. This includes techniques like Term Frequency-Inverse Document Frequency (TF-IDF) [15].
4. **Label Encoding**: Convert the labels (phishing or legitimate) into binary values (e.g., 0 for legitimate, 1 for phishing).

4.3 Model Selection

Several machine learning and deep learning models are implemented in this research to classify phishing emails. The models chosen for this study include:

1. **Naive Bayes**: A probabilistic classifier that is particularly well-suited for text classification tasks like email filtering. It is simple to implement and often provides strong baseline results [5].
2. **Long Short-Term Memory (LSTM)**: A type of recurrent neural network (RNN) that is effective for processing and classifying sequences of data, such as email content. LSTM networks are capable of learning long-term dependencies, making them ideal for phishing detection [6].

3. **Gated Recurrent Units (GRU)**: Another RNN variant that performs similarly to LSTM but with fewer parameters, making it faster to train while maintaining similar performance for phishing detection tasks [16].

4.4 Model Training and Evaluation

The models are trained on the preprocessed phishing email dataset using **TensorFlow** and **Keras**. The training process involves splitting the dataset into training and test sets, ensuring that the models are evaluated on unseen data to prevent overfitting. The evaluation metrics used to assess the model performance are:

1. **Accuracy**: The ratio of correctly classified emails (both phishing and legitimate) to the total number of emails.

2. **Precision**: The ratio of correctly identified phishing emails to the total emails predicted as phishing.

3. **Recall**: The ratio of correctly identified phishing emails to the total actual phishing emails.

4. **F1-Score**: The harmonic mean of precision and recall, providing a balance between the two.

4.5 System Implementation

The phishing detection system is implemented as a **web-based application** using the following technologies:

1. **Frontend**: Developed using HTML, CSS, and JavaScript to provide a user-friendly interface for system administrators.

2. **Backend**: Built using Python's Flask framework, which handles the interaction between the user interface, the AI models, and the database.

3. **Database**: **MySQL** is used to store and manage email data and classification results.

4. **Microsoft Azure**: Azure Active Directory is integrated to provide secure authentication and access to the system. Azure also facilitates the retrieval of emails from Microsoft Exchange servers, enabling real-time email analysis and classification.

4.6 Real-Time Detection and Updates

The system is designed to retrieve emails in real-time using Microsoft Azure's connection to the Exchange server. Once emails are retrieved, they are processed through the trained AI models, and the classification results are updated in the user's mailbox. This real-time detection feature is essential for enabling system administrators to act on phishing threats immediately [17].

5 Expected Contributions

The expected contributions of this research will span both practical applications and academic insights. The proposed system will significantly enhance phishing email detection capabilities while contributing to the field of cybersecurity. The key contributions include:

1. **AI-Based Phishing Detection System**:
    - The development of a web-based phishing email detection system that integrates AI and machine learning models such as Naive Bayes, LSTM, and GRU to improve phishing detection accuracy. This system will address the limitations of traditional rule-based email filters by recognizing complex phishing patterns [5], [6], [18].
2. **Real-Time Classification**:
    - A real-time email classification system that processes emails using Microsoft Azure, running them through AI models and updating the classification results directly in user mailboxes. This will empower system administrators to take immediate action against phishing threats [19].
3. **Scalability and Adaptability**:
    - The system will be scalable, allowing it to handle various organizational sizes and email traffic volumes. Additionally, its adaptability through continuous retraining

of AI models will ensure effectiveness against evolving phishing tactics [16], [20].

4. **Enhanced Security for Organizations**:
    - The proposed system will offer a secure authentication framework using Azure Active Directory and a user-friendly interface that allows administrators to monitor email security. This will be a vital tool for organizations seeking to mitigate phishing threats [19].
5. **Contributions to AI Research in Cybersecurity**:
    - This research will contribute to the field of AI-driven phishing detection, particularly in demonstrating how LSTM and GRU models outperform traditional email filters. This contribution will add valuable knowledge to the academic research community and practical cybersecurity measures [6], [21].

6 Proposed Structure of the Thesis (Chapter Outlines)

The thesis will be organized into five primary chapters, each addressing a specific aspect of the research:

Chapter 1: Introduction

This chapter introduces the background of phishing detection, the rationale for the research, research objectives, and an overview of the methodology.

Chapter 2: Literature Review

This chapter reviews existing studies on phishing detection systems, traditional email filters, and the integration of AI models (Naive Bayes, LSTM, GRU) into phishing detection. It also explores real-time email classification and the role of Microsoft Azure in secure email handling.

Chapter 3: Methodology

This chapter outlines the research methodology, including data collection, preprocessing steps, model selection (Naive Bayes, LSTM, GRU), system implementation, and real-time detection through Microsoft Azure.

Chapter 4: Results and Discussion

This chapter presents the findings, including model performance metrics (accuracy, precision, recall, F1-score) and compares the proposed system's effectiveness against traditional methods. It also discusses the challenges encountered during the research.

Chapter 5: Conclusion and Future Work

The final chapter summarizes the research outcomes and the effectiveness of the AI-based phishing detection system. It also discusses potential areas for future work, such as extending the system to detect other email-based threats (e.g., malware).

7 Preliminary Timeline

The preliminary timeline for the project is as follows:

Task	Duration	Estimated Completion
Phase 1: Literature Review	15 days	4-Nov-24
Phase 2: Data Collection and Preprocessing	20 days	24-Nov-24
Phase 3: Model Development and Training	25 days	19-Dec-24
Phase 4: System Implementation	10 days	29-Dec-24
Phase 5: Model Testing and Evaluation	7 days	1-Jan-25
Final Submission	N/A	3-Jan-25

References

[1] Verizon, "2023 Data Breach Investigations Report," 2023. [Online]. Available: https://www.verizon.com/business/resources/reports/dbir/.

[2] K. Scarfone, M. Souppaya, and A. Cody, "Guidelines on Electronic Mail Security," *NIST Special Publication 800-45 Version 2*, National Institute of Standards and Technology, 2018.

[3] A. Sahu and A. Kumar, "AI-Based Email Phishing Detection System: A Review," *Journal of Cybersecurity*, vol. 5, pp. 45-55, 2022.

[4] Deloitte, "Phishing: How to Defend Against Growing Cyberattacks," 2021. [Online]. Available: https://www.deloitte.com.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[6] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[7] R. Patel, "Phishing Email Detection AI/ML," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/code/riyapatel1697/phishing-email-detection-ai-ml.

[8] K. Scarfone, M. Souppaya, and A. Cody, "Guidelines on Electronic Mail Security," *NIST Special Publication 800-45 Version 2*, National Institute of Standards and Technology, 2018.

[9] Microsoft Azure, "Azure Active Directory Documentation," [Online]. Available: https://docs.microsoft.com/en-us/azure/active-directory/.

[10] N. Abdullah, "Phishing Email Dataset," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/naserabdullahalam/phishing-email-dataset.

[11] A. Sahu and A. Kumar, "AI-Based Email Phishing Detection System: A Review," *Journal of Cybersecurity*, vol. 5, pp. 45-55, 2022.

[12] R. Lewis, "Real-Time Data Processing in the Cloud: A Comprehensive Guide," *Cloud Computing Journal*, vol. 12, no. 3, pp. 66-75, 2020.

[13] N. Abdullah, "Phishing Email Dataset," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/naserabdullahalam/phishing-email-dataset.

[14] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly Media, 2009.

[15] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513-523, 1988.

[16] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," *arXiv preprint*, 2014.

[17] Microsoft Azure, "Azure Active Directory Documentation," [Online]. Available: https://docs.microsoft.com/en-us/azure/active-directory/.

[18] G. Hinton, L. Deng, and D. Yu, "Deep Neural Networks for Acoustic Modeling in Speech Recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82-97, 2012.

[19] Microsoft Azure, "Azure Active Directory Documentation," [Online]. Available: https://docs.microsoft.com/en-us/azure/active-directory/.

[20] A. Sahu and A. Kumar, "AI-Based Email Phishing Detection System: A Review," *Journal of Cybersecurity*, vol. 5, pp. 45-55, 2022.

[21] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," *arXiv preprint*, 2014.