AI-Based Spam Detection: ShieldSecure Protecting Clients

DONNA NAADU BOTCHWAY

2415778

NOVEMBER 30, 2024

Executive Summary

ShieldSecure faces a challenge of client's inboxes flooded with spam. These spams bypassing traditional filters increases phishing and malware risks. To tackle this, we developed an AI-based spam detector using the "cmm541data" dataset. We tested several algorithms, including Naive Bayes and Random Forest with a focus on less false negatives(missing spam).

After testing, the top-performing model was Random Forest. It produced high accuracy and fewer false negatives. It also handles the data imbalance effectively. For now, we can conclude that Random Forest is reliable with a test accuracy of 99.83% and an AUC-ROC close to 1.

Considering vulnerabilities, attackers may attempt to bypass detection using by modifying text. Hence, we must regularly update the model to stay ahead of evolving spam tactics. Also, we addressed privacy concerns by anonymizing email data.

Random Forest is ShieldSecure's best model for spam detection. It delivers high accuracy, reduces false positives, and handles data imbalance well. It's also adaptable to emerging spam techniques.

Methodology

According to SentinelOne(2024), spam is the top vector for cyberattacks. It enables phishing and malware infection which causes operational, financial, and reputational harm. Addressing spam requires both advanced technology, like machine learning-based filters, and user awareness. This presentation focuses on the technology aspect to combat these evolving threats to enhance ShieldSecure’s security.

Problem Definition : Understand the spam problem and possible challenges.

Exploratory Data Analysis : Analyze dataset to identify patterns in classifying spam and ham.

Data Preparation : Clean and preprocess data to extract relevant features.

Modelling: Train and test machine learning model to identify the most effective fit.

Evaluation : Measure model performance to minimize false positives using metrics like precision, confusion matrix and F1-score.

Exploratory Data Analysis (EDA)

Findings

❑ **Class Distribution**

- Spam: 50,197 emails (66.6%)
- Ham: 25,218 emails (33.4%)

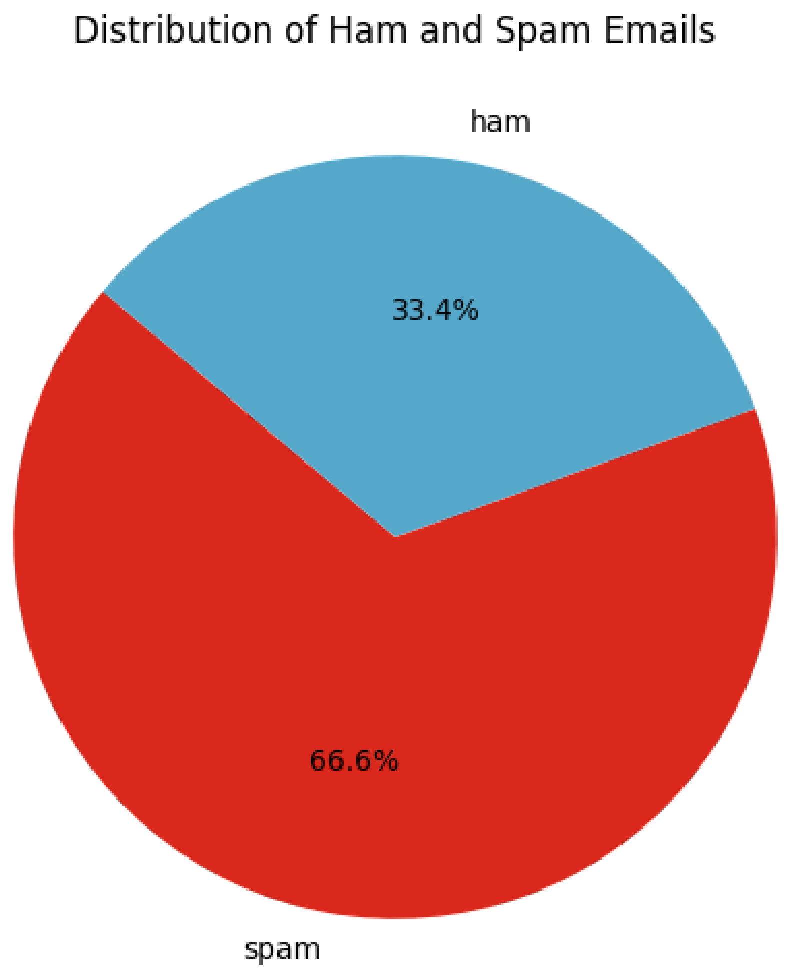
❑ **Text Length Analysis**

- Spam: 7,363.99 characters (Avg.)
- Ham: 8,049.82 characters (Avg.

❑ **Data Quality**

- No duplicates or missing values
- Data types: Text (object), Length (int)

❑ **Reviewed email samples for cleaning and feature extraction.**



Data Preprocessing

Remove HTML Tags & Links

Clean HTML tags like <p>, <a>, <img> and links which add noise.

Remove Headers & Metadata

Headers and metadata like Subject, To, From, Content-Type, and Date which are common in emails are removed.

Normalization

Replace dates, currency, emails, and symbols with placeholders to reduce variation and improve pattern recognition.

Stopword Removal & Tokenization

Remove common words like “the” and split the text into individual words for model to learn in context.

Lemmatization

Reduce words to their root form, minimizing the vocabulary. Words like ‘buying’ become ‘buy’, reducing vocabulary size.

Feature Engineering

To improve model classification, both content patterns and email metadata was used by performing;

- **Text Vectorization:**

  Used Count and TF-IDF to capture both word frequency and importance.

- **Feature Combination**:

  Then the extracted email metadata and vectorized content was combined to train the model.

Feature	**Description**
Subject Length	Short or overly long subject lines often indicate spam.
Uppercase Ratio	High uppercase ratio suggests spam.
Spam Keywords	Presence of terms like "free", "urgent", "claim" are red flags.
Sender Information	Suspicious or unfamiliar senders may signal spam.
Priority Flag	High priority often seen in spam emails.
Word Count	Spam emails may have unusually short or long text.
Special Characters	Frequent use of !, $, %, # can indicate spam.
Links	Multiple links, especially embedded in html , are common in spam.
Attachments	Spam emails may include multiple attachments.
Hidden Text	Spam may contain hidden text to evade detection.
Time of Day	Emails outside business hours are likely to be spam.
Reply/Forward	Spam may use **Re:** or **Fwd:** in subject to appears as replies or forwards

Machine Learning Algorithms

After research, the following algorithms were selected to identify the most effective ones for spam detection.

Naive Bayes is a classifier good for text data. It is known for its simplicity. It's easy to implement, fast, and accurate (Raschka, 2014).

Logistic Regression model is simple yet powerful. It works well for linear data predicting 0 or 1. It is fast and scales well with large datasets

(Kontsewaya, Antonov, and Artamonov, 2020).

Support Vector Classification produces good accuracy and good with clear boundaries although its slow and not good with large data.

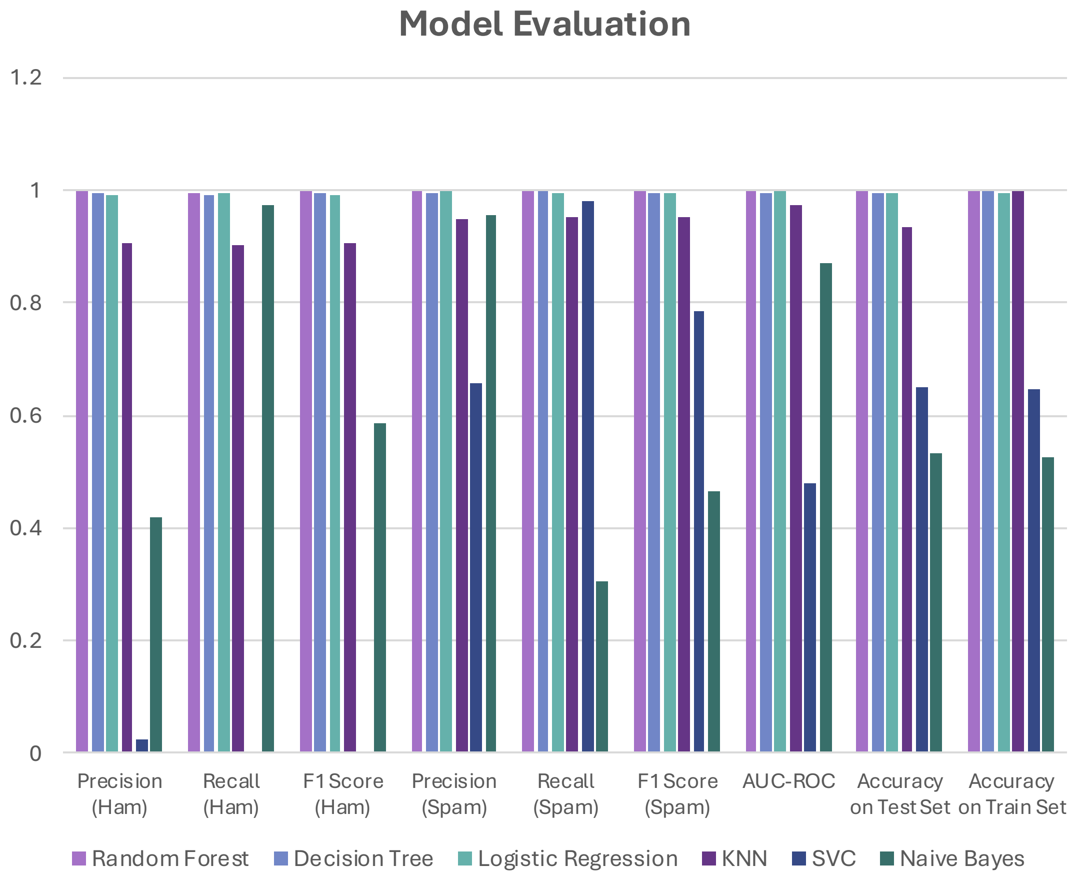
Decision Tree is simple and interpretable but prone to overfitting without tuning.

Random Forest has high accuracy , handling complex data and imbalances well. It also scales well with large dataset.

K-Nearest Neighbors (KNN) is a simple classification method that compares data to the closest labeled examples. It works well with clean data but can be slow with large datasets.

Isolation Forest is an anomaly detection algorithm efficient for detecting rare spam. However, it is not ideal for standard classification tasks.

Model Performance and Vulnerabilities



Vulnerabilities include;

1. Spamming ML System with Chaff Data (AML.T0046)

Attackers flood the system with irrelevant data to overwhelm the model and waste analysts' time.

Mitigation: Limit the number of queries users can perform at a time.

2. Trained on Sensitive Data (AML.T0020)

Exposure of sensitive user data via model leakage.

Mitigation: Sanitize data by removing sensitive data from training sets.

3.Adversarial Inputs

Attackers craft inputs to evade spam detection (e.g., emails that try to bypass filters).

Mitigation: Implement adversarial training and regularly update spam filter rules to detect new evasion techniques.

Final Model Selection: Random Forest

Random Forest produced fewer false negatives (misclassified spam) and false positives (misclassified ham).

Random Forest outdid other models with high precision, recall and AUC-ROC showing its effectiveness in spam detection.

F1 score balances precision and recall , valued in imbalanced dataset. Random Forest had a high F1 score and overall high accuracy compared to the other models.

Random Forest is well suited for handling imbalanced dataset, as shown in its high AUC-ROC and F1 score.

Random Forest's ability to model complex patterns allows it to handle different types of spam emails effectively. This makes it scalable and able to adapt to evolving spam tactics.

Random Forest	Predicted Ham	Predicted Spam
Actual Ham	4,603	15
Actual Spam	8	8,949

Random Forest	Precision	Recall	F1-Score	Support
Ham	0.998265	0.996752	0.997508	4618
Spam	0.998327	0.999107	0.998717	8957
Accuracy	0.998306	0.998306	0.998306	0.998306
Macro Avg	0.998296	0.997929	0.998112	13575
Weighted Avg	0.998306	0.998306	0.998305	13575

Challenges and Ethical Considerations

Challenges encountered during the development of the spam detector include:

- **Data Imbalance**: Spam was more than legit emails, so I used class balancing during training to address this issue.
- **Evolving Spam Tactics**: Some tactics are not always identifiable, so I applied Isolation Forest to detect and remove outliers before training.
- **Feature Selection**: To identify relevant features, I reviewed sample emails, which raised privacy concerns, but it was necessary for effective feature extraction.
- **False Positives/Negatives**: I chose the Random Forest algorithm as it produced fewer false positives and negatives compared to other models.

Ethical concerns as to using AI based spam detectors include:

- **Privacy and Data Security**: Data must be anonymized as spam filters analyze email data which may contain sensitive information.
- **False Positives and False Negatives**: Misclassifying legitimate emails as spam (false positives) or missing spam (false negatives) could lead to missing vital messages or exposure to threats.
- **Freedom of Expression**: Over filtering can limit legitimate communication.
- **Social Impact and Trust**: Spam reduces trust in email, hence the need for accuracy.

Reflection

- Data Preparation is Key: Most of the work lies in understanding and preparing the data.
- Preprocessing: Effective data preprocessing is essential for better performance.
- Address Data Imbalance: Solving label imbalance improves model results.
- Modeling & Evaluation: A smaller portion of the work, but selecting the right model and evaluation metrics is crucial.
- Evaluation Metrics: Focus on accuracy across different labels, not just total accuracy.

References

- SentinelOne (2024) 'What is Spam? Types, Risks, and How to Protect Your Business', SentinelOne Cybersecurity 101, 27 August. Available at: : https://www.sentinelone.com/cybersecurity-101/cybersecurity/what-is-spam/ (Accessed: 11 November 2024).
- Kontsewaya, Y., Antonov, E. and Artamonov, A. (2020) 'Evaluating the Effectiveness of Machine Learning Methods for Spam Detection', Procedia Computer Science. Available at: https://doi.org/10.1016/j.procs.2021.06.056 (Accessed: 20 November 2024).
- Raschka, S. (2014) 'Naive Bayes and Text Classification', Sebastian Raschka Blog, 4 October. Available at: https://sebastianraschka.com/Articles/2014_naive_bayes_1.html (Accessed: 25 November 2024).
- MITRE (2023) 'AML.M0004: Mitigation for Adversarial Machine Learning', MITRE ATLAS. Available at: https://atlas.mitre.org/mitigations/AML.M0004 (Accessed: 30 November 2024)
- MITRE (2023) 'AML.M0007: Mitigation for Adversarial Machine Learning', MITRE ATLAS. Available at: https://atlas.mitre.org/mitigations/AML.M0007 (Accessed: 30 November 2024).
- Jang, D. (2024) 'Spam Detection: The Classification Techniques of Supervised Learning in Artificial Intelligence', Medium, 16 October. Available at: https://medium.com/@jangdaehan1/spam-detection-the-classification-techniques-of-supervised-learning-in-artificial-intelligence-23e64343983d (Accessed: 30 November 2024).
- OpenAI. (2024). ChatGPT Conversation. Available at: https://chatgpt.com/share/674d0419-da94-8000-923e-7f52f69b277a (Accessed 1 Dec. 2024).
- Wickramasinghe, I. and Kalutarage, H., 2021. Naive Bayes: Applications, Variations and Vulnerabilities: A Review of Literature with Code Snippets for Implementation. Soft Computing, 25(3), pp.2277-2293.

Thank You