Spam Email Detection

A Comparative Analysis of Machine Learning Algorithms

Student Name: HARKA ADHIKARI
Roll No: 2315355
Date: 31 January 2026
GitHub Repository: `https://github.com/adhikariharka/spam-mail-detection`

1 Abstract

Spam email remains one of the most common cybersecurity issues, often used for phishing, malware delivery, and online fraud. In this project, a machine learning based spam detection system was developed to automatically classify emails into **Spam** or **Ham**. Four classification models were trained and compared: Support Vector Machine (SVM), Naive Bayes (NB), Random Forest (RF), and Logistic Regression (LR).

The experimental results show that the **Support Vector Machine (SVM)** model achieved the best overall performance with an accuracy of **98.21%**. Due to its strong balance between accuracy, precision, and recall, SVM was selected as the final model for deployment.

2 Introduction

Email spam remains a major threat in modern communication systems. Spam messages are not only unwanted advertisements, but also act as delivery methods for phishing attacks, scams, and malware.

Rule-based filters can detect common patterns, but spammers constantly change wording and formatting to bypass detection. Machine learning offers a stronger solution because it learns patterns from real data and can generalize to new messages.

The aim of this project is to build a spam detection system that performs accurately and can be deployed in a web-based application.

3 Methodology

3.1 Dataset

The dataset used in this project is stored in `dataset.csv`. It contains labeled email messages:

- **0 = Spam**
- **1 = Ham**

3.2 Preprocessing

Text preprocessing is applied to reduce noise and improve model accuracy:

- Converting text to lowercase
- Removing special characters and punctuation
- Cleaning unnecessary spaces and symbols

3.3 Feature Extraction (TF-IDF)

The email messages are transformed into numerical vectors using **TF-IDF**. The system extracts the top **5000 features** to represent important words while keeping the feature space manageable.

4 Algorithms Evaluated

Four machine learning algorithms were implemented and compared:

- **Support Vector Machine (SVM)** with sigmoid kernel
- **Naive Bayes (MultinomialNB)**
- **Random Forest (RF)**
- **Logistic Regression (LR)**

5 Experimental Results

Table 1: Performance Comparison of ML Models

Model	Accuracy	Precision (Spam)	Recall (Spam)	F1-Score
SVM	**98.21%**	98.5%	**88.0%**	**93.0%**
Random Forest	97.94%	**100.0%**	84.7%	91.7%
Naive Bayes	97.04%	100.0%	78.0%	87.6%
Logistic Regression	96.23%	100.0%	72.0%	83.7%

5.1 Discussion

Although Random Forest and Naive Bayes achieved perfect precision, their recall was lower. This means they missed more spam messages. The SVM model achieved the best overall balance, especially with higher recall, which is critical for spam detection.

6 System Implementation

6.1 Backend

The backend is built using **FastAPI**, which loads the trained SVM model and TF-IDF vectorizer. It provides prediction results through an API endpoint.

6.2 Frontend

The frontend is developed using **Next.js (React)**. Users can paste an email message and receive an instant classification result.

7 Conclusion

This project demonstrates that machine learning can effectively classify spam emails with high accuracy. Among the evaluated algorithms, SVM performed best with **98.21% accuracy** and strong recall. The system can be extended in the future using deep learning models and continuous retraining.

8 References

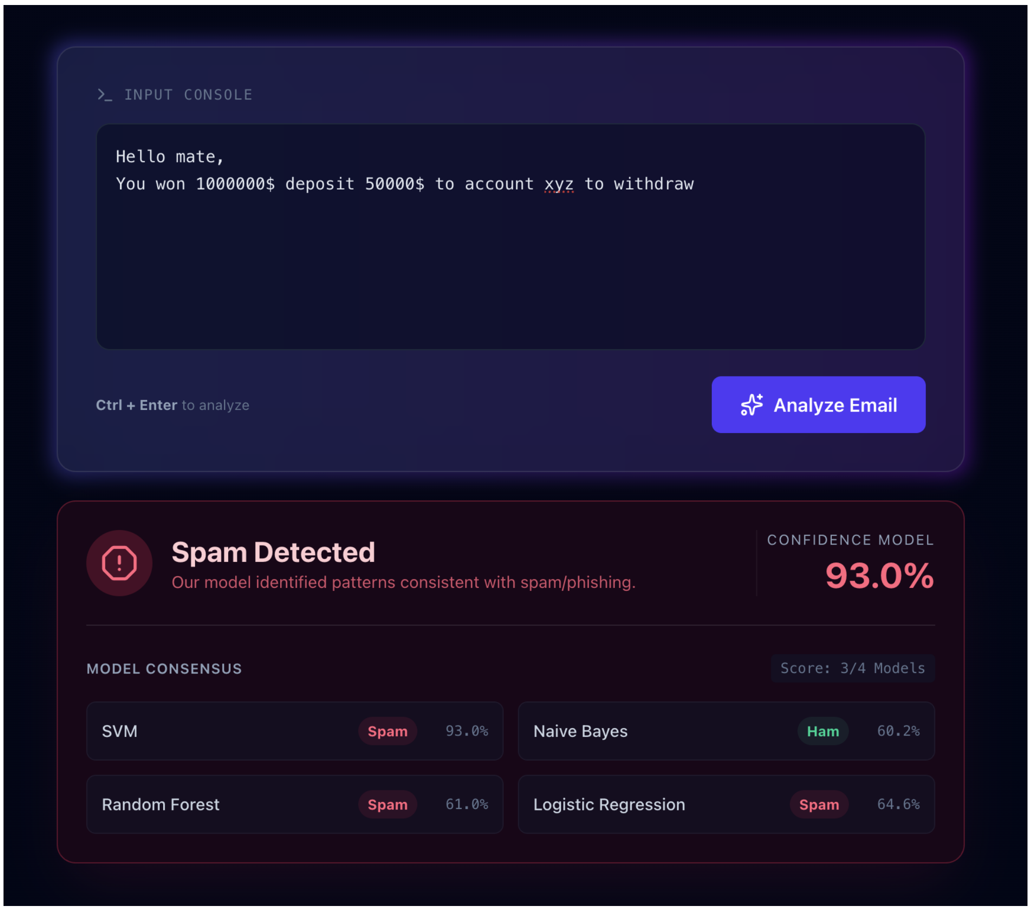
1. Scikit-learn Documentation: https://scikit-learn.org/

2. FastAPI Documentation: https://fastapi.tiangolo.com/

A Appendix

A.1 Email Examples

The following figures demonstrate sample email messages used to test the spam detection system.



```
Hello mate,
You won 1000000$ deposit 50000$ to account xyz to withdraw
```

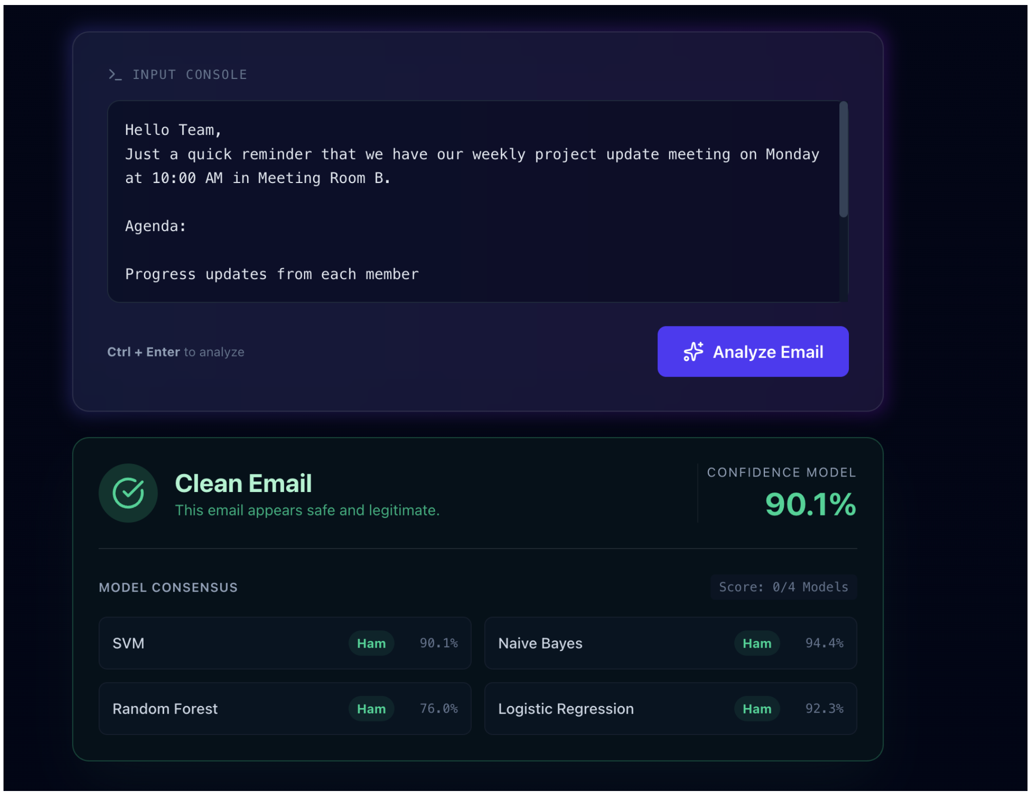
Spam Detected
Our model identified patterns consistent with spam/phishing.

CONFIDENCE MODEL 93.0%

MODEL CONSENSUS — Score: 3/4 Models

SVM	Spam	93.0%
Naive Bayes	Ham	60.2%
Random Forest	Spam	61.0%
Logistic Regression	Spam	64.6%

Figure 1: Example of a Potential Spam Email (Suspicious Content)



```
>_ INPUT CONSOLE

Hello Team,
Just a quick reminder that we have our weekly project update meeting on Monday
at 10:00 AM in Meeting Room B.

Agenda:

Progress updates from each member
```

Ctrl + Enter to analyze

Clean Email
This email appears safe and legitimate.

CONFIDENCE MODEL
90.1%

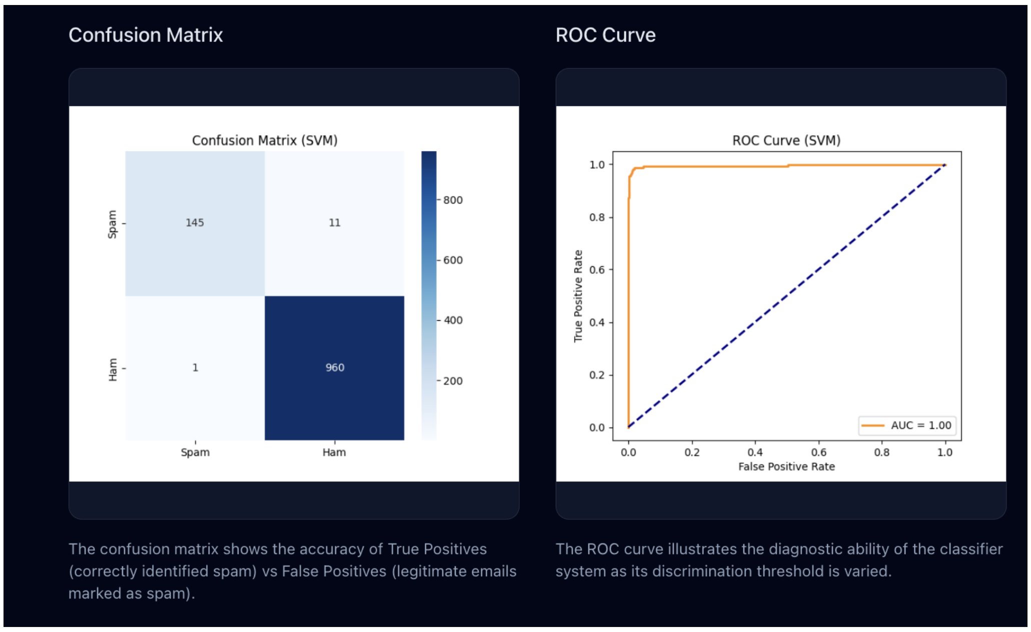
MODEL CONSENSUS — Score: 0/4 Models

SVM	Ham	90.1%
Naive Bayes	Ham	94.4%
Random Forest	Ham	76.0%
Logistic Regression	Ham	92.3%

Figure 2: Example of a Legitimate Email (Ham Message)

A.2 Model Performance

The following figure shows the performance of the selected model.



The confusion matrix shows the accuracy of True Positives (correctly identified spam) vs False Positives (legitimate emails marked as spam).

The ROC curve illustrates the diagnostic ability of the classifier system as its discrimination threshold is varied.

Figure 3: Model Performance Visualization (Confusion Matrix / Accuracy Comparison)