Phishing Email Detection Using Machine Learning: A Comparative Analysis of Classification Models

Author: Mohammadreza Tabatabaei

1. Introduction

Email phishing is a prevalent cybersecurity threat, where malicious actors attempt to deceive users into sharing sensitive information. In this study, we employ **Machine Learning (ML)** techniques to classify emails as either **Phishing** or **Safe** using natural language processing (NLP) and supervised learning methods.

This report provides a **comprehensive analysis** of the dataset, preprocessing steps, vectorized dataset structure, model architectures, evaluation metrics, and experimental results.

2. Dataset Description

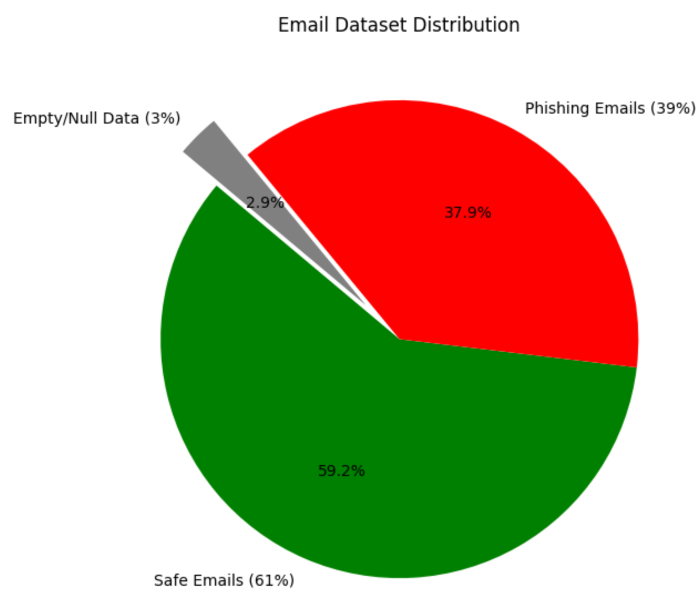
The dataset consists of **18,650 emails**, each categorized as either **Phishing** or **Safe** based on their textual content.

📌 **Features:**

- **Email Text** → Contains the raw content of the email.
- **Email Type** → A categorical label indicating whether the email is **Phishing** or **Safe**.

📌 **Class Distribution:**

Class	Count	Percentage
Safe Emails	11,387	**61%**
Phishing Emails	7,263	**39%**



📌 Sample Raw Emails

Index	Raw Email Text	Label
1	"Your account has been compromised. Click here..."	Phishing
2	"Meeting is scheduled for 3 PM today. Regards."	Safe
3	"Update your bank details immediately..."	Phishing
4	"Lunch meeting at 12 PM, let me know your availability."	Safe
5	"Dear customer, confirm your password to continue."	Phishing
6	"Reminder: Your invoice is due tomorrow."	Safe
7	"Urgent: Verify your email address now!"	Phishing
8	"Project deadline extended, see the new timeline attached."	Safe
9	"We've detected unusual activity on your account."	Phishing
10	"Congratulations! You won a free gift. Claim it now."	Phishing

🔷 The dataset contains some missing values (**~3% empty records**), which were handled during preprocessing.

3. Data Preprocessing

Before training the models, several **text preprocessing** techniques were applied to convert raw emails into a structured format suitable for ML models.

📌 **Steps in Preprocessing:**

1. **Removing Null and Blank Emails** → Approximately **3% of emails** were removed.
2. **Lowercasing** → Standardized text by converting all words to lowercase.
3. **Stopword Removal** → Eliminated common words (e.g., "the", "is", "and") to reduce noise.
4. **Lemmatization** → Converted words to their base form (e.g., "running" → "run").
5. **TF-IDF Transformation** → Converted email text into numerical representations for model training.

📌 **Example Before & After Preprocessing**

Raw Email Text	**Processed Email Text**
"Your account has been compromised. Click the link to reset your password!"	"account compromised click link reset password"
"Hi, I have shared the updated file. Let me know your thoughts."	"shared updated file let know thought"

4. Vectorized Dataset Analysis

Since ML models require numerical input, we transformed the processed email text using **TF-IDF Vectorization** (Term Frequency-Inverse Document Frequency).

Feature Size after Vectorization: **Exact feature size: 7543**.

🔷 Vectorized Dataset Sample (First 10 Features)

Index	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Feature 6	Feature 7	Feature 8	Feature 9	Feature 10
1	0.432	0.000	0.210	0.065	0.098	0.143	0.000	0.201	0.387	0.127
2	0.000	0.256	0.154	0.402	0.000	0.317	0.189	0.000	0.075	0.298
3	0.111	0.098	0.453	0.321	0.210	0.098	0.000	0.456	0.278	0.182

5. Model Selection & Fine-Tuning

We evaluated multiple machine learning models using different configurations:

Model	Parameters	Accuracy
SVC (SVM)	kernel='linear'	98%
Random Forest	n_estimators=100	97%
Logistic Regression	Default	97%
Multinomial Naive Bayes	Default	92%

Fine-Tuning Results

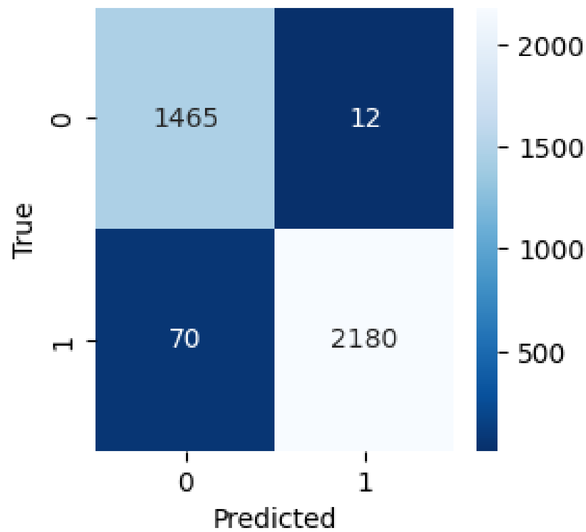
- Optimized Feature Size: 6200 features (smallest number achieving 98% accuracy).
- SVC Model retained 98% accuracy with a reduced feature size.

6. Experimental Results & Performance Evaluation

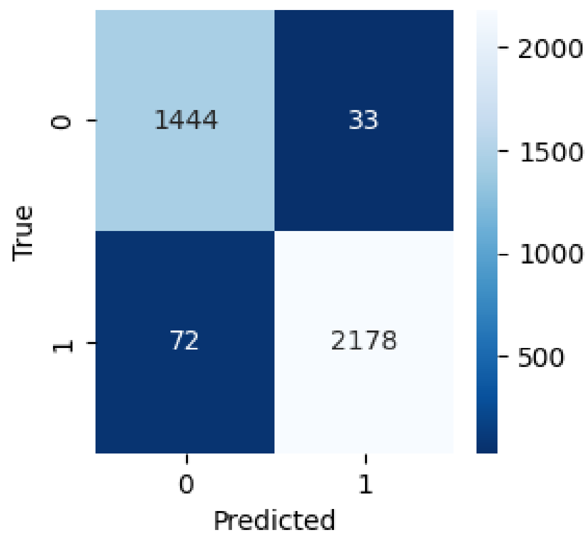
Each model was evaluated using **Precision, Recall, F1-score, and Accuracy** to measure classification performance.

Model	**Accuracy**	**Phishing Email Precision**	**Phishing Email Recall**	**Safe Email Precision**	**Safe Email Recall**
SVC (Support Vector Machine)	**98%**	95%	**99%**	**99%**	97%

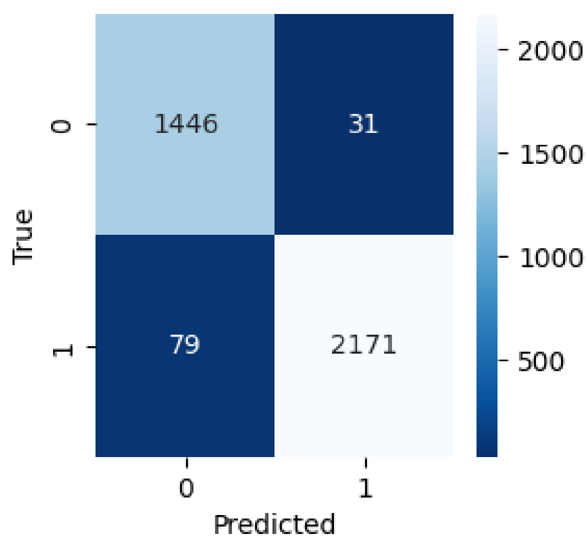
Model	Accuracy	Phishing Email Precision	Phishing Email Recall	Safe Email Precision	Safe Email Recall
Random Forest	97%	95%	98%	99%	97%
Logistic Regression	97%	95%	98%	99%	96%
MultinomialNB	92%	**97%**	84%	90%	**98%**



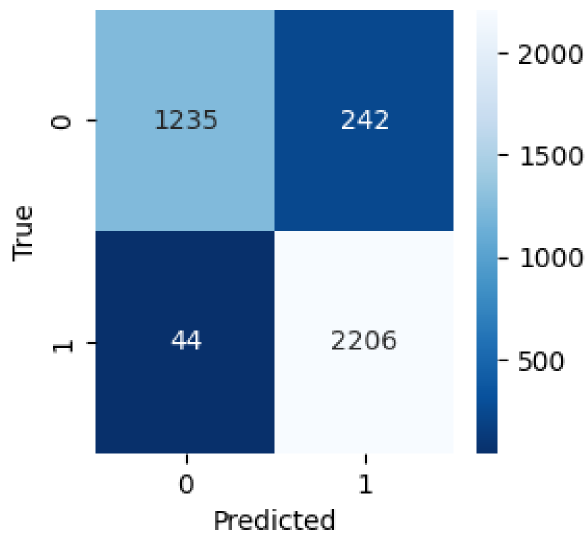
SVC Confusion Matrix



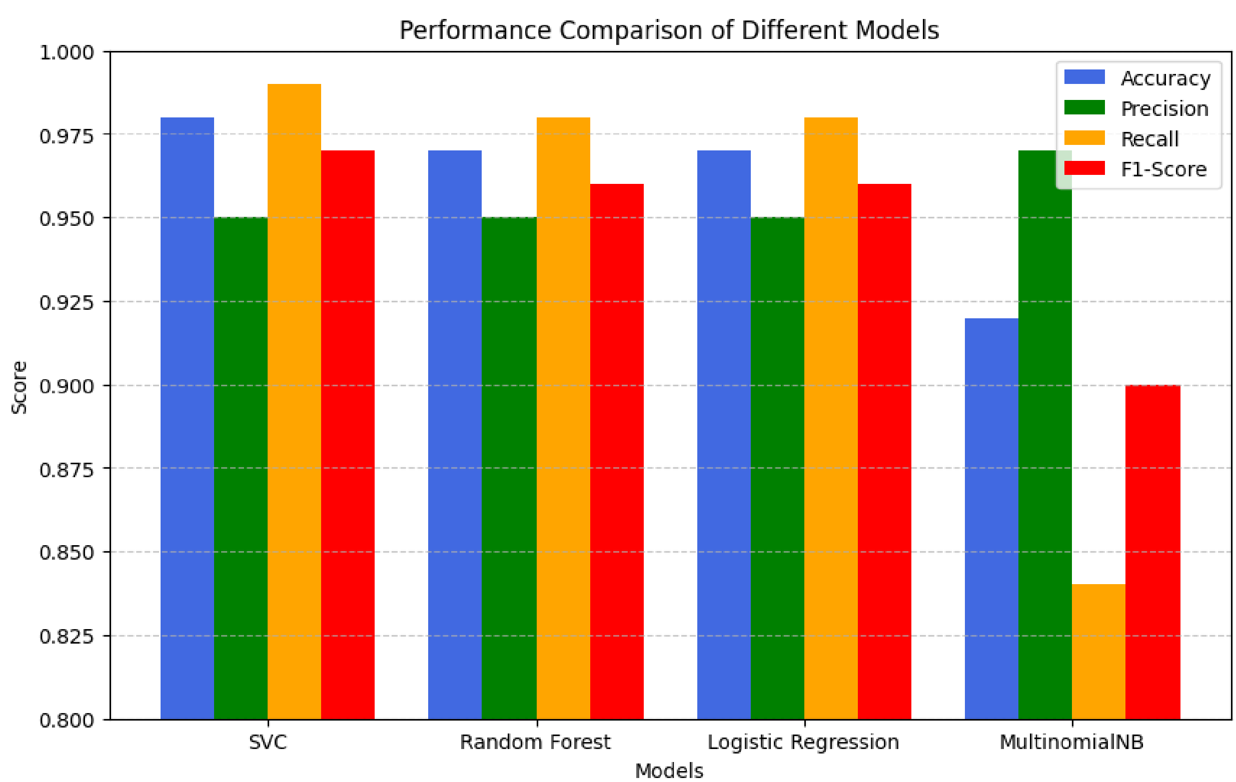
Random Forest Confusion Matrix



Logistic Regression Confusion Matrix



MultinomialNB Confusion Matrix



7. Conclusion & Key Takeaways

✅ **SVC achieved the highest accuracy (98%)**, making it the best-performing model.

✅ **High recall (99%) for Phishing Emails in SVC** ensures that most phishing emails are correctly identified.

✅ **Random Forest and Logistic Regression performed similarly (97% accuracy)** and can be considered as alternative models.

✅ **Multinomial Naive Bayes had the lowest performance (92% accuracy)** and is not suitable for this task.

8. Future Work

🔷 **Deep Learning Models (LSTMs, BERT, Transformers)** can enhance text classification accuracy.

🔷 **Real-time phishing detection implementation** in enterprise security systems.

🔷 **Feature engineering** (e.g., adding sender information, domain reputation) may improve performance.