Can The Spam: Detecting Junk Emails Using Machine Learning

Renae Alailima, Saharsh Bhargava, Megan Nguyen, Somrat Sen

Joint Science and Technology Institute West 2024

Introduction

Machine learning involves the study of statistical models that learn from data and perform tasks without explicit instructions. Through recent developments in machine learning, it has become increasingly viable to apply it to the field of cybersecurity. **The purpose of this project is to explore the application of various machine learning methods to detect and filter spam emails.** Additionally, this project will evaluate and analyze the effectiveness of different machine learning methods for spam filtering.

Background

This project aims to explore the application of machine learning methods in spam detection. By leveraging advanced algorithms, it is possible to detect and filter spam emails with high accuracy. The algorithms include:

- **K-Nearest Neighbors (KNN)**: Predicts the class of a data point based on the majority class of its k nearest neighbors.
- **Decision Trees**: Splits data into subsets based on the value of features to form a tree-like structure of decisions.
- **Random Forest**: Builds multiple decision trees and merges their outputs for classification.
- **Logistic Regression**: Assigns probabilities to outcomes using the sigmoid function.

The metrics used to evaluate the models are:

- **Accuracy:** The ratio of correct predictions to the total number of instances.
- **Precision**: The ratio of true positive predictions to the total predicted positives.
- **Recall:** The ratio of true positive predictions to the total actual positives.

Materials & Methods

An email dataset consisting of 500,000+ emails including 7,500+ spam emails was curated by mentors from Sandia National Laboratories. Python libraries such as Pandas, NumPy, Scikit-Learn were used to extract and engineer features and train machine learning models. An 80/20 train-test split was used for model training and testing. The dataset was balanced to have an equal split of spam and non-spam emails before training.

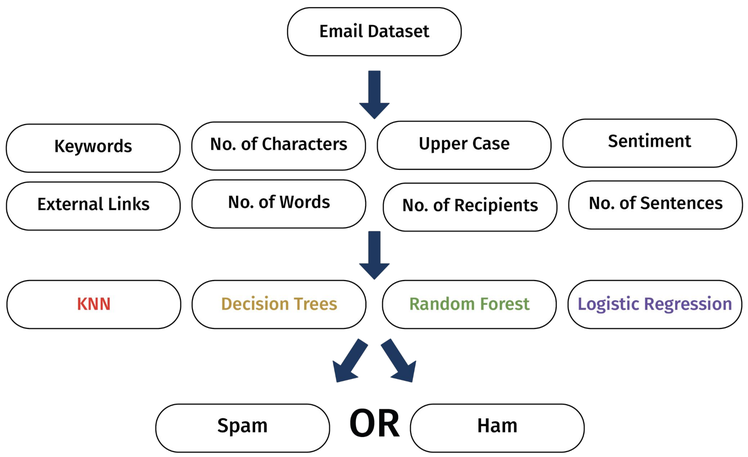

Figure 1. Project Workflow and Methodology

Results

K-Nearest Neighbors (KNN)

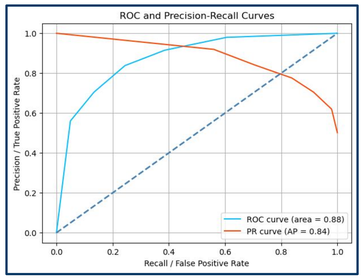

Figure 2. Receiver Operating Characteristic (ROC) Curve and Precision-Recall (PR) Curve for KNN

The KNN algorithm has a simple implementation, however it does not work well in datasets with high dimensionality, such as in this project. The KNN model has a moderately high accuracy of **79.55%**. It has a precision value of **82.58%** and a recall value of **82.57%**. The area under the ROC is **0.88**, which shows the model is a good distinguisher between the positive and negative classifications.

Decision Trees

Decision Trees are very interpretable but are prone to overfitting and can become biased. Similarly to KNN models, decision trees do not operate well in high-dimensional datasets. This model has a very high accuracy of **94.34%**, a precision of **94.33%**, and a recall of **94.43%**. The area under the ROC is **0.94**, which demonstrates that this model is an excellent distinguisher between the positive and negative classifications.

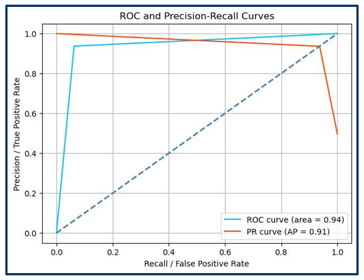

Figure 3. Receiver Operating Characteristic (ROC) Curve and Precision-Recall (PR) Curve for Decision Trees

Random Forest

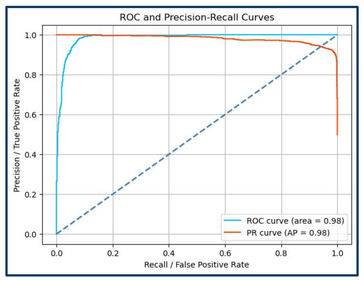

Figure 4. Receiver Operating Characteristic Curve & Precision-Recall (PR) Curve for Random Forest

The Random Forest algorithm reduces the risk of overfitting and provides the ability to analyze feature importances for classification. This model achieves a high accuracy of **95.79%**, a precision of **98.07%**, and a recall of **94.06%**. The area under the ROC is **0.98**, suggesting that this model provides great performance with other desirable properties for positive and negative classifications.

Logistic Regression

Logistic Regression is interpretable, but it may not perform well with complex and non-linear relationships. Logistic regression also assumes a linear relationship between the features. This model has a slightly lower accuracy of **91.9%**, a precision of **96.69%**, and a recall of **88.17%**. The area under the ROC is **0.93** demonstrating potential for inclusion in a ensemble classifier, but not as performant on its own relative to others.

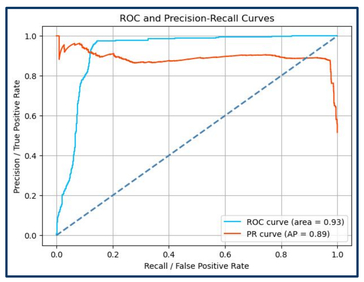

Figure 5. Receiver Operating Characteristic (ROC) Curve and Precision-Recall (PR) Curve for Logistic Regression

Conclusion

Algorithm → Statistics ↓	KNN	Decision Trees	Random Forest	Logistic Regression
Accuracy	79.55%	94.34%	***95.79%***	91.90%
Precision	82.58%	94.33%	***98.07%***	96.69%
Recall	82.57%	***94.43%***	94.06%	88.17%
AUC	0.88	0.94	***0.98***	0.93
Ranking	4th	2nd	1st	3rd

Figure 6. Table Comparing Statistics of All Four Algorithms

- Machine learning provides an incredibly effective method for identifying spam emails.
- **Random Forest** was demonstrated to be the best spam email classifier of the four models tested in this project.
- The most important features in the Random Forest model were the presence of links, the number of characters in the email text, and the email's sentiment.
- The models developed for this project may exhibit bias toward specifically detecting phishing emails due to the composition of the training dataset.
- The Support Vector Machine and XGBoost algorithms were initially tested, but abandoned due to time constraints and limited computational resources.

Acknowledgements

Thanks to Head Mentor Meifan Chen and Mentors Bryan Arguello, Carol Chen, Brian Gaume, Tian Ma, Doug McGeehan, Anton Sumali and Alumnus Undergraduate Assistant Peter Escamilla. We appreciate the opportunity provided to us by Sandia National Laboratories, Oak Ridge Institute for Science and Education, The University of New Mexico, and the Defense Threat Reduction Agency's Joint Science and Technology Office.

Figure 7. Group Photo