What's the Deal with Email Spam?

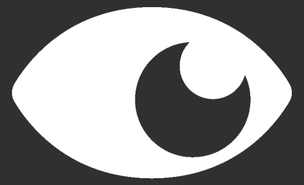
An in depth analysis of spam detection

Presented By:
Bryan Mejia, Roger Leung, Ali Ahmed, Abul Hasan

Research Process

Email Spam in Modern World

How many different types of spam email are there and are there any special ways of dealing with them?



Exploration and Visualization

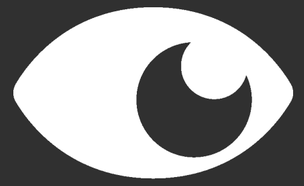
Elaborate on your topic here.

Working with Our Models!

Elaborate on your topic here.

Deliverable and Evaluation!

Elaborate on your topic here.



Identifying the Problem

What exactly is Spam Email?

How often are we checking our spam email to see if something an important email was wrongly classified?

Spam detection is a vital part of securing personal and enterprise communications. As spammers evolve their techniques, we can't rely on previous spam detection models to be as efficient they used to be. Cyber security professionals work with engineers closely to develop methods of identifying these emails, with the help of machine learning models.

However, these machine learning-based systems must adapt and stay ahead. With increasing access to labeled datasets and evolving algorithms, now is an ideal time to explore which models are most effective in spam filtering.

This study focuses on evaluating individual models and ensemble methods to identify a reliable, accurate, and scalable solution.

How Will We Do It?

Working Dataset

We will replicate and extend findings from selected research papers using two public datasets. We will evaluate, test and model the data through machine learning models.

Pre Processing (Preparing the Data!)

To ensure our data is prepared for our models, we will enforce techniques such as lowercasing, stop-word removal, punctuation cleaning, and TF-IDF vectorization for converting email text to numerical features.

Which Models?

We will implement Naive Bayes, Support Vector Machines, Logistic Regression, Random Forest, Artificial Neural Networks and ensemble methods like Adaboost and Stacking.

Evaluating Spam Email Research

- Zhang (2024): Compared Naive Bayes, Decision Tree, and SVM. Found that SVM achieved the highest accuracy. A hybrid model combining SVM and NB showed even better results.
- AIP Publishing (2025): Analyzed RF, NB, MLP, and SVM. Random Forest (RF) achieved the highest accuracy (~98.8%) and demonstrated consistent, stable performance.
- Sevli & Keskin (2024): Evaluated RF, LR, NB, SVM, and ANN. RF performed best. Simpler ensemble models like RF outperformed even neural networks for medium-sized datasets.
- Li (2024): Compared NB and RF. NB had better recall; RF had better precision. The paper emphasized choosing a model based on spam filter goals (recall vs. precision).
- Adnan et al. (2023): Used stacking ensembles (LR, DT, KNN, GNB, AdaBoost). The stacked model outperformed all individual models, with 98.8% accuracy and 98.9% F1 score

Enhancing Spam Filtering: A Comparative Study of Modern ML Techniques (Chenwei Zhang, 2024):
https://www.sciencedirect.com/science/article/pii/S2772662223002308
Measuring the Efficiency of RF, NB, MLP & SVM in Email Spam Detection (AIP Publishing, 2025):
https://pubs.aip.org/aip/acp/article-abstract/3270/1/020052/3343767/Measuring-the-efficiency-of-random-forest-naive
Machine Learning Based Classification for Spam Detection (Sevli & Keskin, 2024):
https://www.researchgate.net/publication/380000625_Machine_Learning_Based_Classification_for_Spam_Detection
Analysis of Spam Classification Based on NB and RF (Li, 2024):
https://www.ewadirect.com/proceedings/aemps/article/view/12660
Improving Spam Email Classification Using Stacking Ensembles (Adnan et al., 2023):
https://link.springer.com/article/10.1007/s10207-023-00756-1

Exploratory Data Analysis (EDA)

Our Working Data!

Our email spam classification data set was produced by Balaka Biswas. This data set contains spam/not spam information from about 5172 emails.

	Email No.	the	to	ect	and	for	of	a	you	hou	...	connevey	jay	valued	lay	infrastructure	military	allowing	ff	dry	Prediction
0	Email 1	0	0	1	0	0	0	2	0	0	...	0	0	0	0	0	0	0	0	0	0
1	Email 2	8	13	24	6	6	2	102	1	27	...	0	0	0	0	0	0	0	1	0	0
2	Email 3	0	0	1	0	0	0	8	0	0	...	0	0	0	0	0	0	0	0	0	0
3	Email 4	0	5	22	0	5	1	51	2	10	...	0	0	0	0	0	0	0	0	0	0
4	Email 5	7	6	17	1	5	2	57	0	9	...	0	0	0	0	0	0	0	1	0	0

5 rows × 3002 columns

```
Column names: ['Email No.', 'the', 'to', 'ect', 'and', 'for', 'of', 'a', 'you', 'hou', 'in', 'on', 'is', 'this', 'enron', 'i'
, 'be', 'that', 'will', 'have', 'with', 'your', 'at', 'we', 's', 'are', 'it', 'by', 'com', 'as', 'from', 'gas', 'or', 'not',
'me', 'deal', 'if', 'meter', 'hpl', 'please', 're', 'e', 'any', 'our', 'corp', 'can', 'd', 'all', 'has', 'was', 'know', 'need
', 'an', 'forwarded', 'new', 't', 'may', 'up', 'j', 'mmbtu', 'should', 'do', 'am', 'get', 'out', 'see', 'no', 'there', 'price
', 'daren', 'but', 'been', 'company', 'l', 'these', 'let', 'so', 'would', 'm', 'into', 'xls', 'farmer', 'attached', 'us', 'in
formation', 'they', 'message', 'day', 'time', 'my', 'one', 'what', 'only', 'http', 'th', 'volume', 'mail', 'contract', 'which
', 'month', 'more', 'robert', 'sitara', 'about', 'texas', 'nom', 'energy', 'pec', 'questions', 'www', 'deals', 'volumes', 'pm
', 'ena', 'now', 'their', 'file', 'some', 'email', 'just', 'also', 'call', 'change', 'other', 'here', 'like', 'b', 'flow', 'n
et', 'following', 'p', 'production', 'when', 'over', 'back', 'want', 'original', 'them', 'below', 'o', 'ticket', 'c', 'he', '
could', 'make', 'inc', 'report', 'march', 'contact', 'were', 'days', 'list', 'nomination', 'system', 'who', 'april', 'number'
, 'sale', 'don', 'its', 'first', 'thanks', 'business', 'help', 'per', 'through', 'july', 'forward', 'font', 'free', 'daily',
'use', 'order', 'today', 'r', 'had', 'fw', 'set', 'plant', 'statements', 'go', 'gary', 'oil', 'line', 'sales', 'w', 'effectiv
```

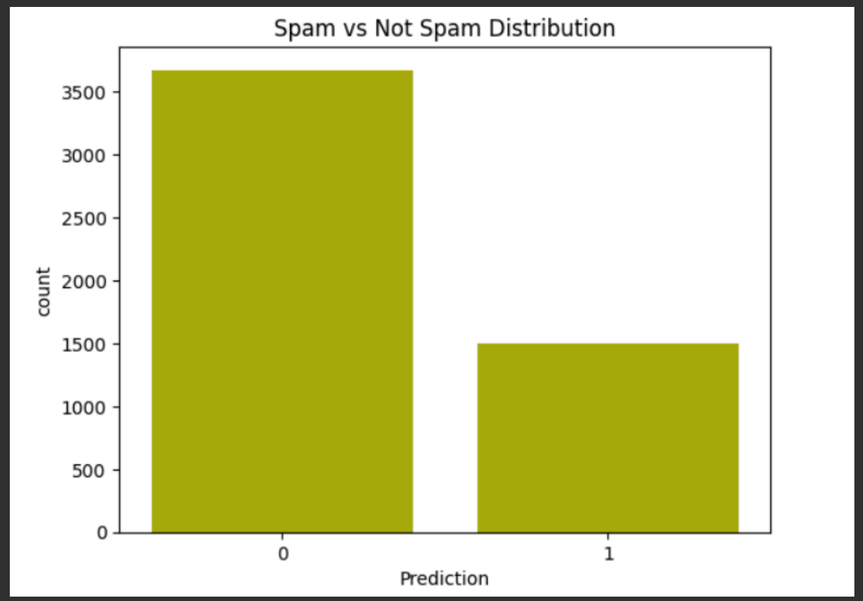
Dealing with Imbalances

Class Imbalance Check

The Prediction column indicates whether an email is spam (1) or not spam (0). Based on the distribution:

- Label 0 (Not Spam) accounts for approximately 71% of the data.
- Label 1 (Spam) accounts for around 29%.

This reveals a noticeable **class imbalance**, with non-spam emails being more frequent than spam.



But **why?**

Machine learning models trained on imbalanced data may become biased toward the majority class (not spam), resulting in poor performance in detecting the minority class (spam).

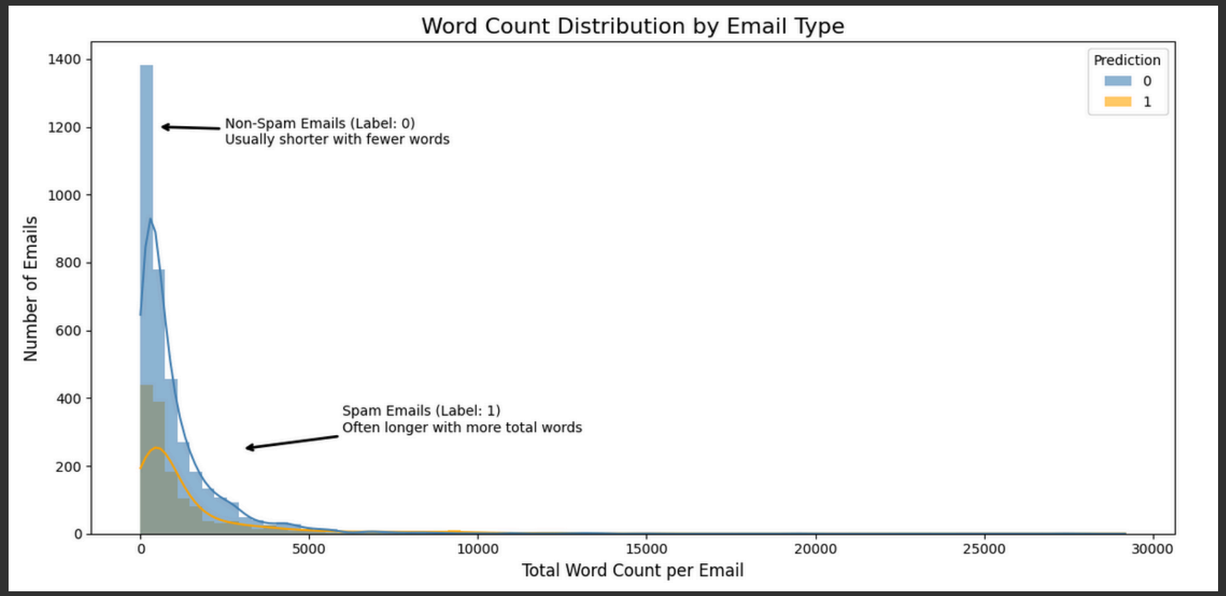
Analyzing Word Count

Word Count

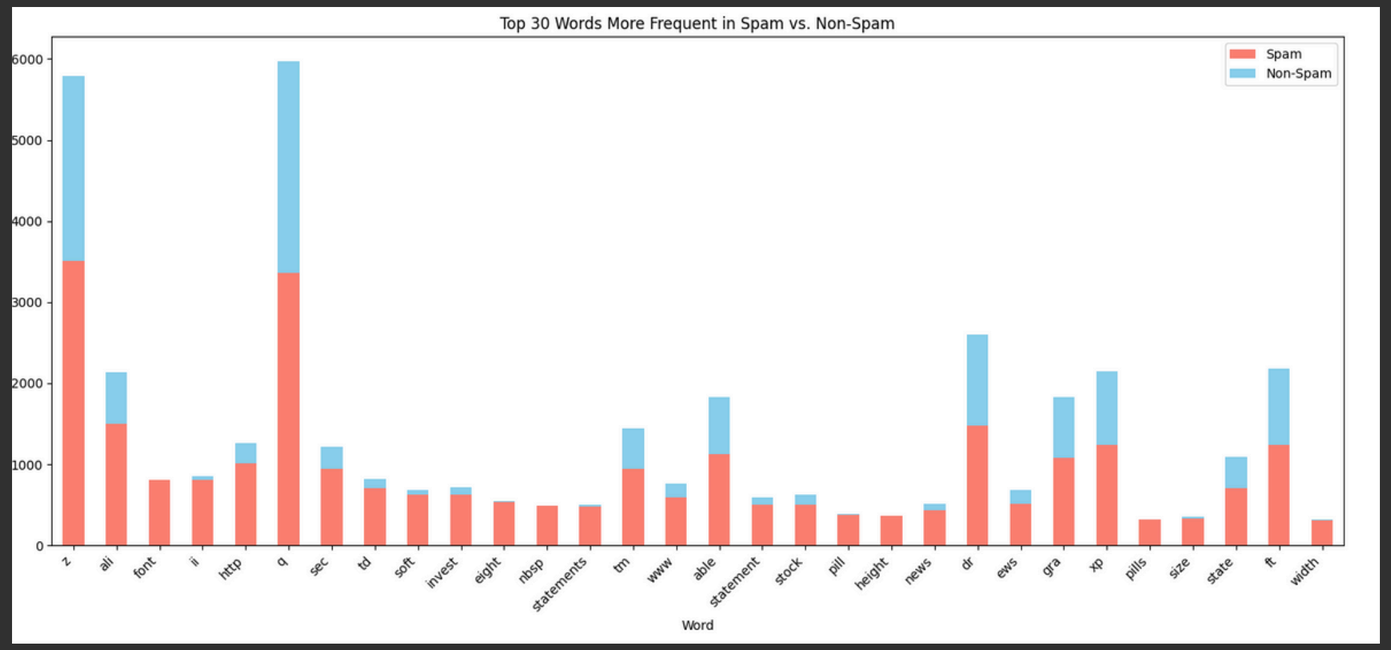
Word Count Distribution by Email Type : This histogram illustrates the total word count per email, separated by class labels:

- **Label 0 (Non-Spam)**: These emails tend to have fewer words, forming a strong peak at lower word counts.
- **Label 1 (Spam)**: These emails typically contain more words, resulting in a longer distribution tail toward higher word counts.

The plot shows that spam emails are often longer and more verbose compared to non-spam emails, which are shorter and more concise.
This difference in word count can be a useful feature for classification models to distinguish between spam and legitimate emails.
The visible class separation in this distribution confirms that word count is a meaningful predictor in this dataset.



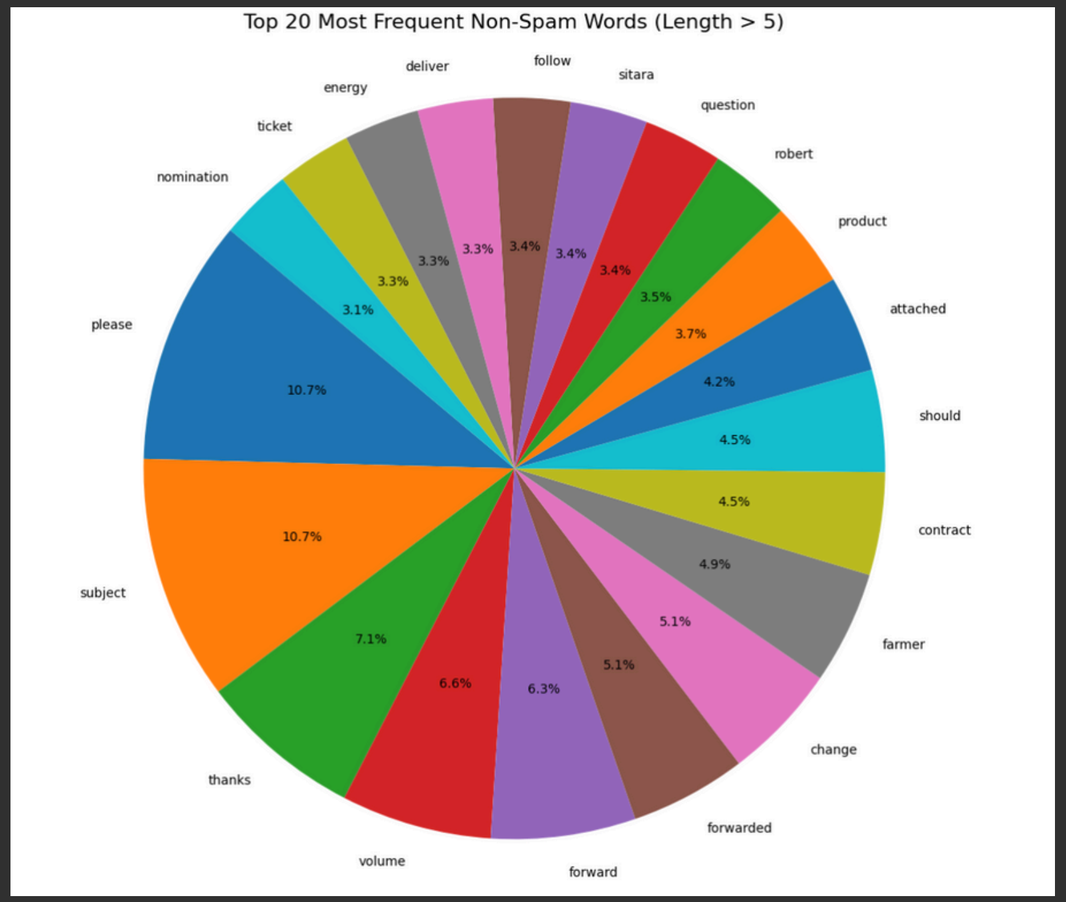
Which Words Are Considered Spam?

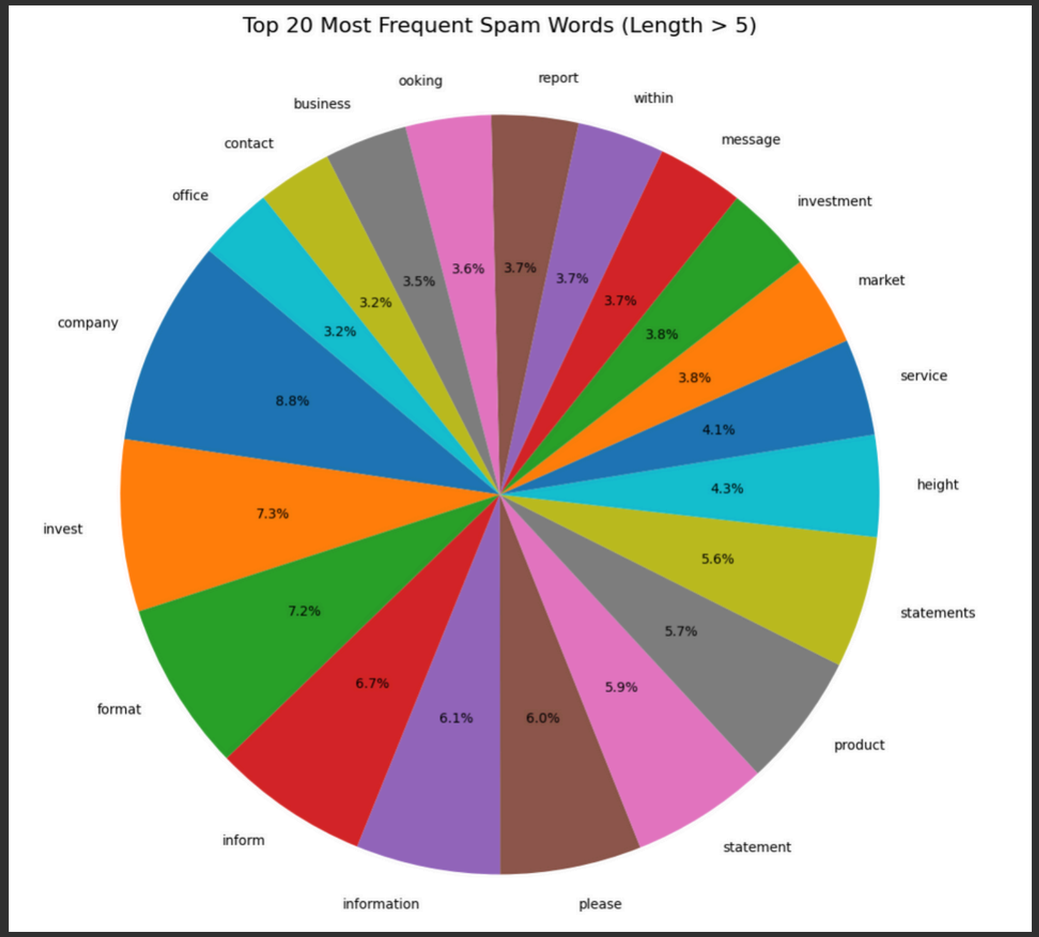


Non-Spam Emails: The top words in non-spam emails often relate to business communications, internal company matters, and technical terms, reflecting the dataset's origin (likely from the Enron corpus). Words such as 'enron', 'please', 'subject' and 'thank'.

Spam Emails: Conversely, the most frequent words in spam emails are typically associated with promotional content, offers, and less formal language. Words like 'price', 'company', 'invest', 'format', 'inform', and terms related to finance or products are common.

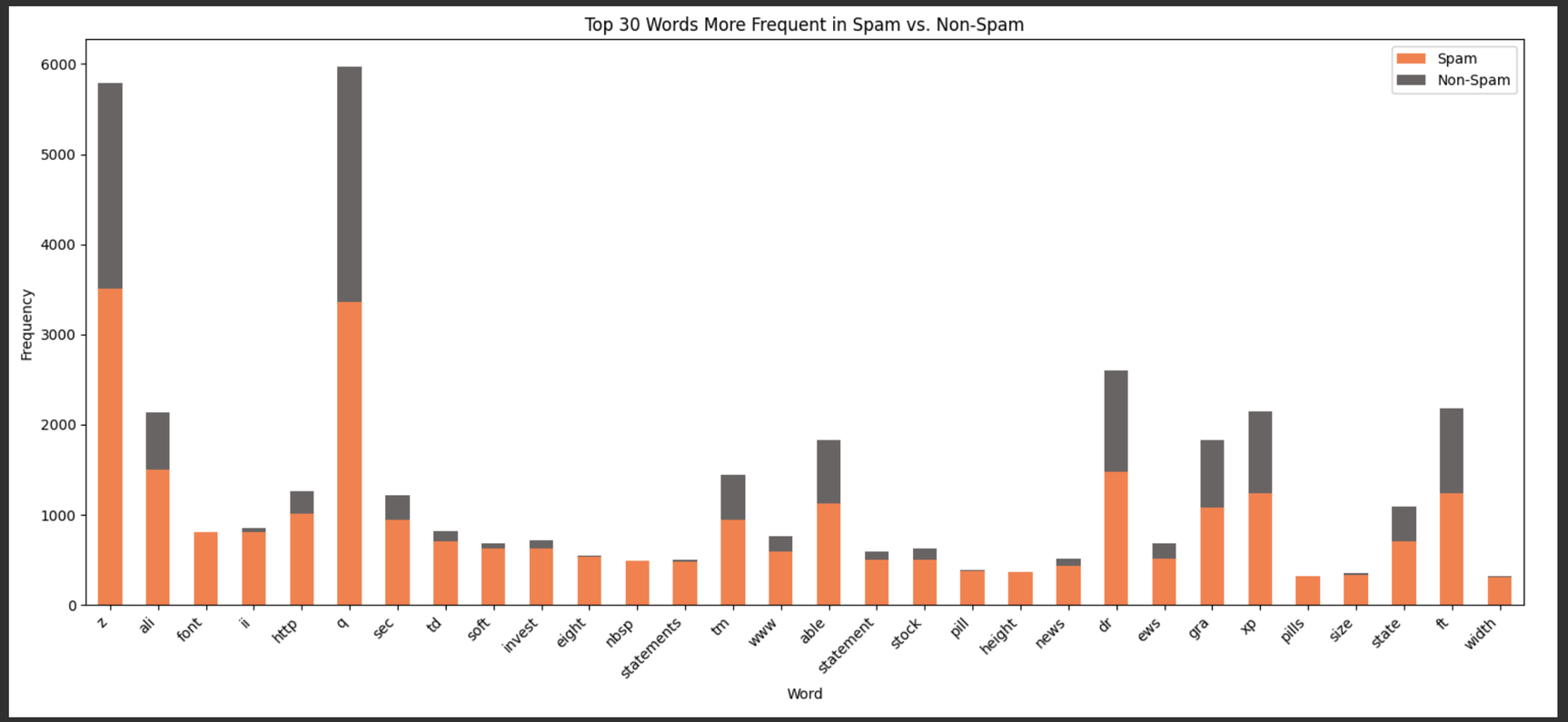
Non-Spam vs Spam





What Shows Up The Most?

Words like "http" and "www" often indicate the links in spam emails, which are likely to be phishing attempts. Terms such as "invest" or "statement" suggest they may be trying to sell something or get you to invest. Additionally, the word "statement" could be associated with a PDF attachment, which might contain a virus.



Working with our Models!

What Models Did We Use?

Now what we've processed our data, what will we work with? We will implement:

- Logistic Regression
- Random Forest
- Naive Bayes
- Support Vector Machines
- Artificial Neural Networks
- Stacking

With many visuals too!

Modifying Our Data

Why We Used PCA (Principal Component Analysis)

Our email data has *thousands* of features (word frequencies per email).

High Dimensionality Problems:
Training models on so many features can be *slow* and computationally expensive.
Risk of overfitting:
Models might learn noise instead of true patterns.

Benefits for Models:
Allows models like SVM and Neural Networks to train more efficiently.
In Short:
PCA helped us handle the large number of word features, making our models more efficient and potentially more accurate.

Logistic Regression

Our objective was to use Binary Classification using Logistic Regression for our dataset.
Addressing Imbalance: Compared models without and with class_weight = balance to handle dataset imbalance

What We Observed :
Strong bias towards class 0 (Majority)
High accuracy, but poor recall for class 1 (Minority)
Our balanced model had significantly improved recall for class 1
Much better responses for precision/recall, leading to better performance

ROC Curve: High AUC indicates strong overall discriminative power, further optimized by balancing.

Insights :
Before Balance : Accuracy at 97%, but class 1 recall was only 0.58.

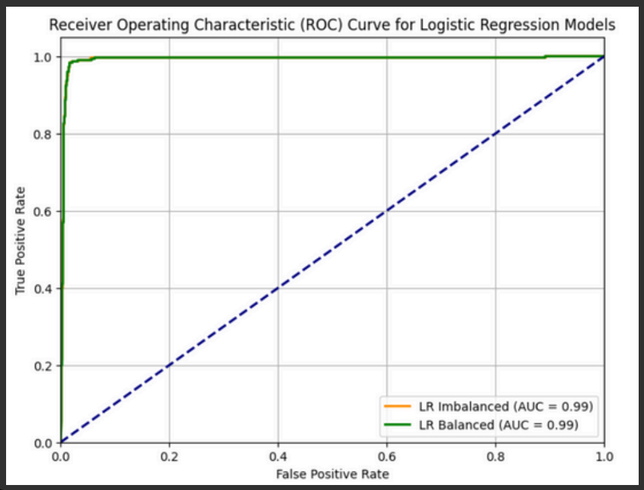
After Balance:Class 1 recall improved to 0.80; Overall AUC remains strong at 0.931.

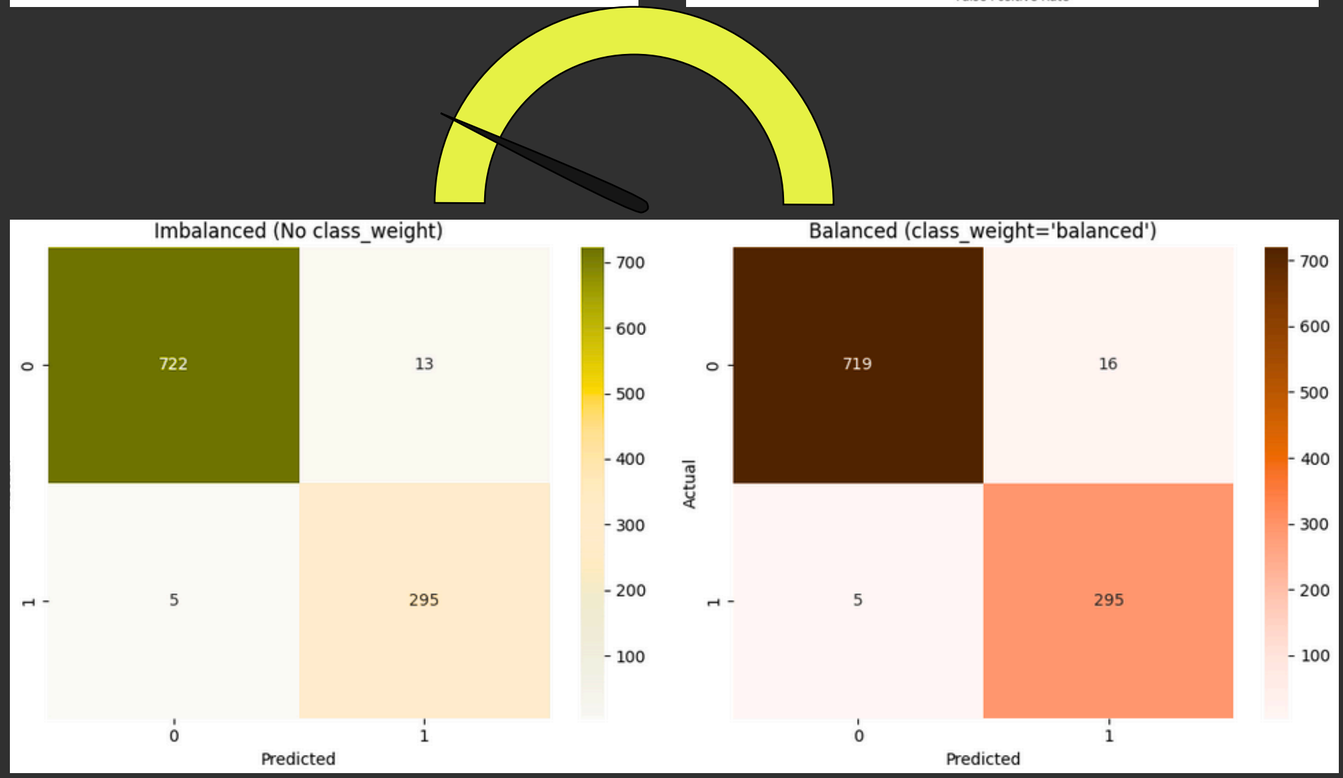
Logistic Regression WITHOUT class_weight (Imbalanced):

	precision	recall	f1-score	support
0	0.99	0.98	0.99	735
1	0.96	0.98	0.97	300
accuracy			0.98	1035
macro avg	0.98	0.98	0.98	1035
weighted avg	0.98	0.98	0.98	1035

Logistic Regression WITH class_weight='balanced':

	precision	recall	f1-score	support
0	0.99	0.98	0.99	735
1	0.95	0.98	0.97	300
accuracy			0.98	1035
macro avg	0.97	0.98	0.98	1035
weighted avg	0.98	0.98	0.98	1035



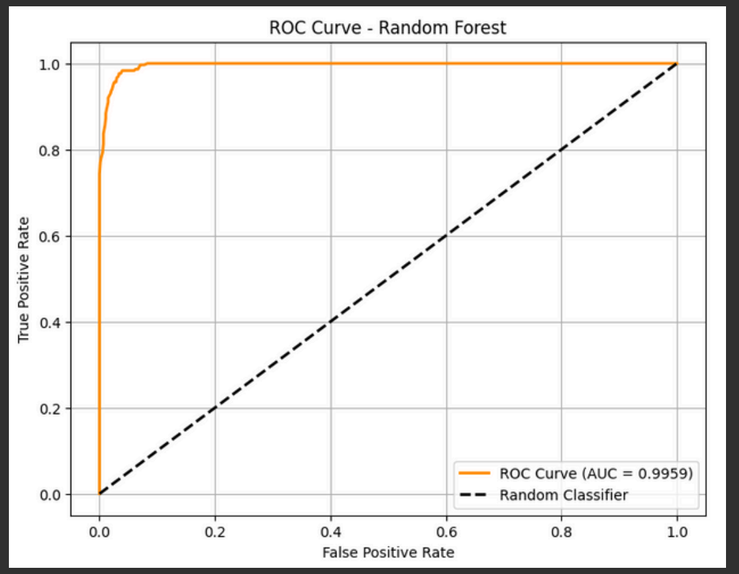


Random Forest

What Was Observed?
Exceptional Performance : The ROC AUC is 0.99, indicating almost perfect classification.
We can see that it's further supported by a high precision, recall and F-1 Score for both columns after balancing

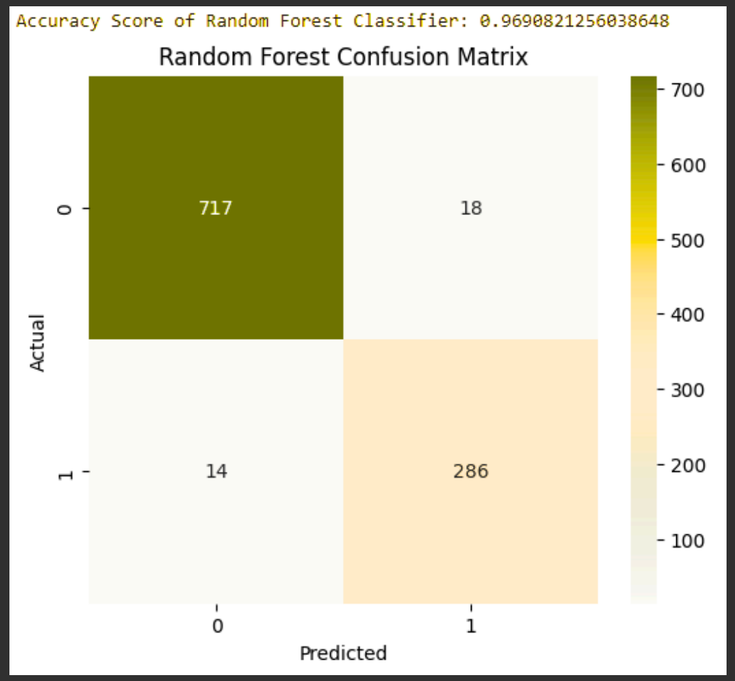
Confusion Matrix Breakdown :
We can see with Class 0, there are 717 true negatives, only 14 false positives!
With Class 1, there were 286 true positives, only 18 false negatives!

This means that very few class 0s were misclassified and very few class 1s were missed.



Classification Report:

	precision	recall	f1-score	support
0	0.98	0.98	0.98	735
1	0.94	0.95	0.95	300
accuracy			0.97	1035
macro avg	0.96	0.96	0.96	1035
weighted avg	0.97	0.97	0.97	1035



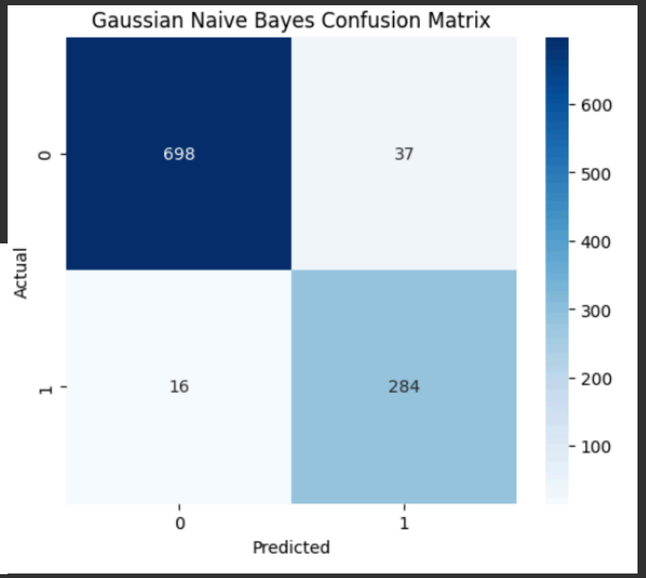
Gaussian Naive Bayes

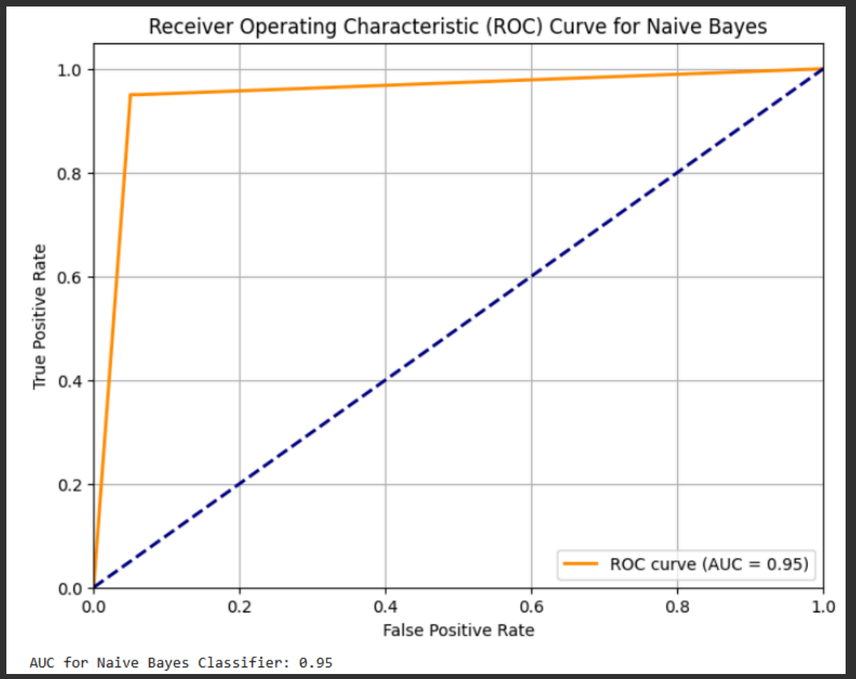
What was observed?
The confusion matrix visually shows that while the model has a higher recall for spam, it also has a higher number of false positives compared to the Logistic Regression and Random Forest models.

And the Models?
Gaussian Naive Bayes Classifier is a reasonably effective model for this email spam classification task, achieving good accuracy and high recall for spam. However, its precision for the spam class is lower than that of Logistic Regression and Random Forest, meaning it is more prone to false positives.

Gaussian Naive Bayes Classifier:

	precision	recall	f1-score	support
0	0.98	0.95	0.96	735
1	0.88	0.95	0.91	300
accuracy			0.95	1035
macro avg	0.93	0.95	0.94	1035
weighted avg	0.95	0.95	0.95	1035





Support Vector Machines (SVMs)

What was observed?

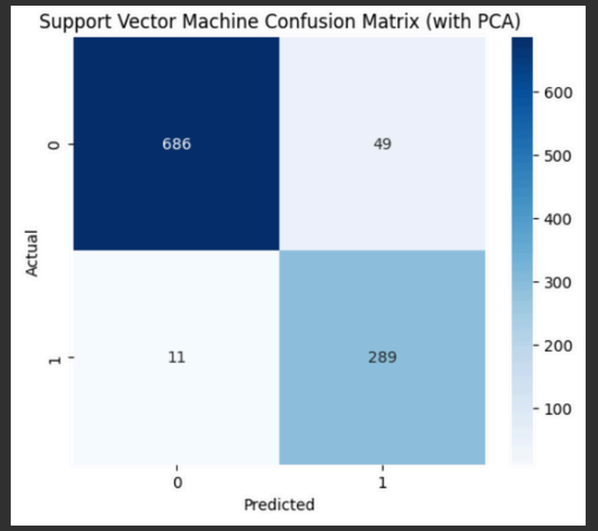
The precision of 86% suggests a moderate number of false positives, where non-spam emails were incorrectly flagged as spam. This is a trade-off observed with high recall – the model is more likely to flag emails as spam to avoid missing true spam, potentially leading to more false alarms

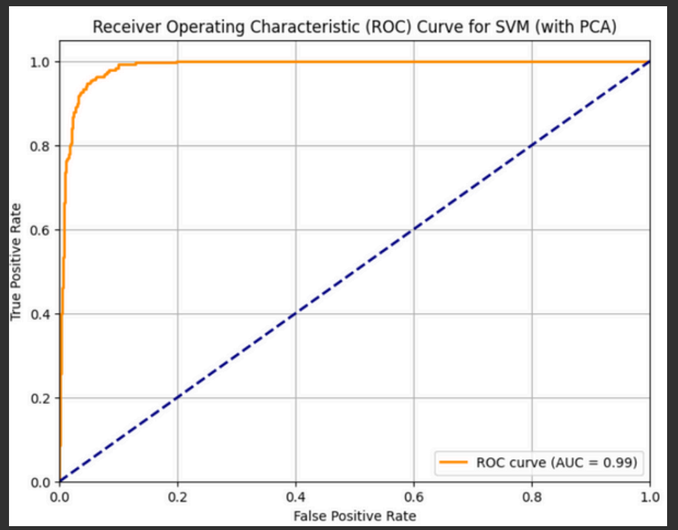
In summary, the Support Vector Machine model, trained on PCA-reduced data with balanced class weights, is a very effective classifier for this task, particularly in its ability to identify the majority of spam emails (high recall). The use of PCA helped manage the high dimensionality of the dataset for this model.

Support Vector Machine (SVM) Classifier (with PCA):

	precision	recall	f1-score	support
0	0.98	0.93	0.96	735
1	0.86	0.96	0.91	300
accuracy			0.94	1035
macro avg	0.92	0.95	0.93	1035
weighted avg	0.95	0.94	0.94	1035

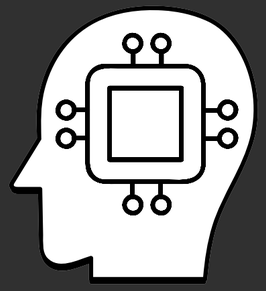
AUC for SVM Classifier (with PCA): 0.99





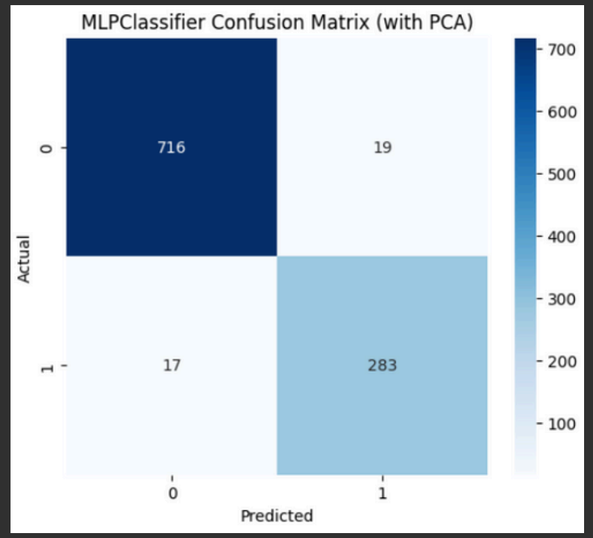
Artificial Neural Networks

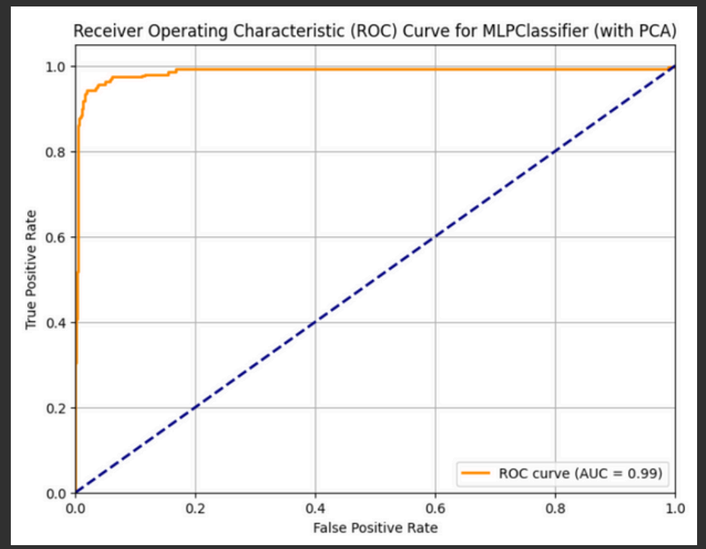
- We utilize the MLPClassifier to explore its effectiveness in capturing intricate patterns in the word frequency features for spam email classification. This model was trained using the dimensionality-reduced data obtained through PCA.
- The confusion matrix visually confirms that the MLPClassifier is effective in distinguishing between the two classes, with a good balance between correctly identifying spam and avoiding false alarms.
- The ROC curve for the MLPClassifier model (with PCA) is shown in the plot. The Area Under the Curve (AUC) is 0.99. An AUC of 0.99 indicates excellent discriminatory power, meaning the model is very good at distinguishing between spam and non-spam emails across various probability thresholds
- This model is a strong contender for spam detection, showcasing the power of neural networks even on a reduced feature set. The use of PCA likely contributed to the model's efficient training and good performance.



MLPClassifier (Neural Network) (with PCA):

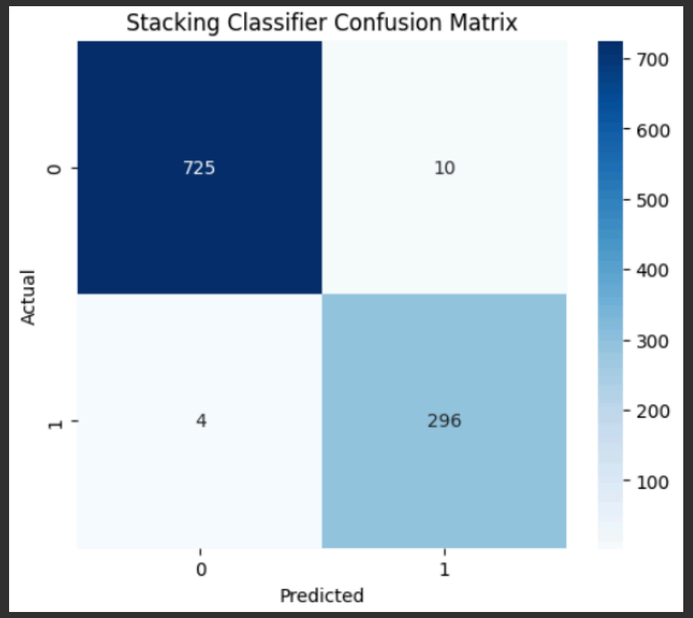
	precision	recall	f1-score	support
0	0.98	0.97	0.98	735
1	0.94	0.94	0.94	300
accuracy			0.97	1035
macro avg	0.96	0.96	0.96	1035
weighted avg	0.97	0.97	0.97	1035

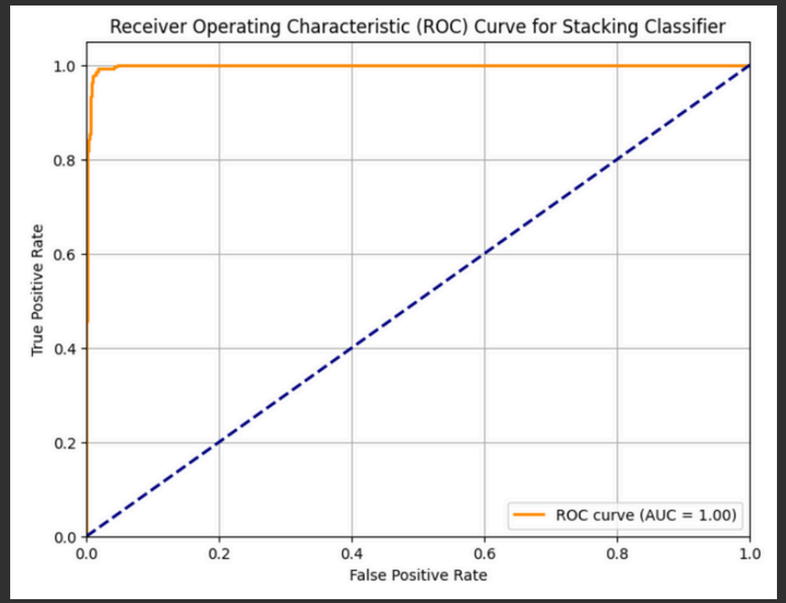




Stacking

- Stacking is an advanced ensemble learning technique that combines the predictions of multiple diverse base models to produce a final prediction. The core idea is to train a meta-model (or final estimator) on the predictions generated by the base models. This often leads to improved predictive performance and robustness compared to using individual models alone.
- our base estimators include the Logistic Regression, Random Forest, Gaussian Naive Bayes, SVM, and MLPClassifier
- The Stacking Classifier is the top-performing model evaluated in this project. By combining the predictions of multiple diverse base models, it achieves exceptional performance metrics across the board, including near-perfect accuracy, precision, recall, F1-score, and AUC. Its ability to minimize both false positives and false negatives makes it an ideal model for a robust spam email classification system. This ensemble approach successfully leveraged the strengths of the individual models to achieve superior overall performance.





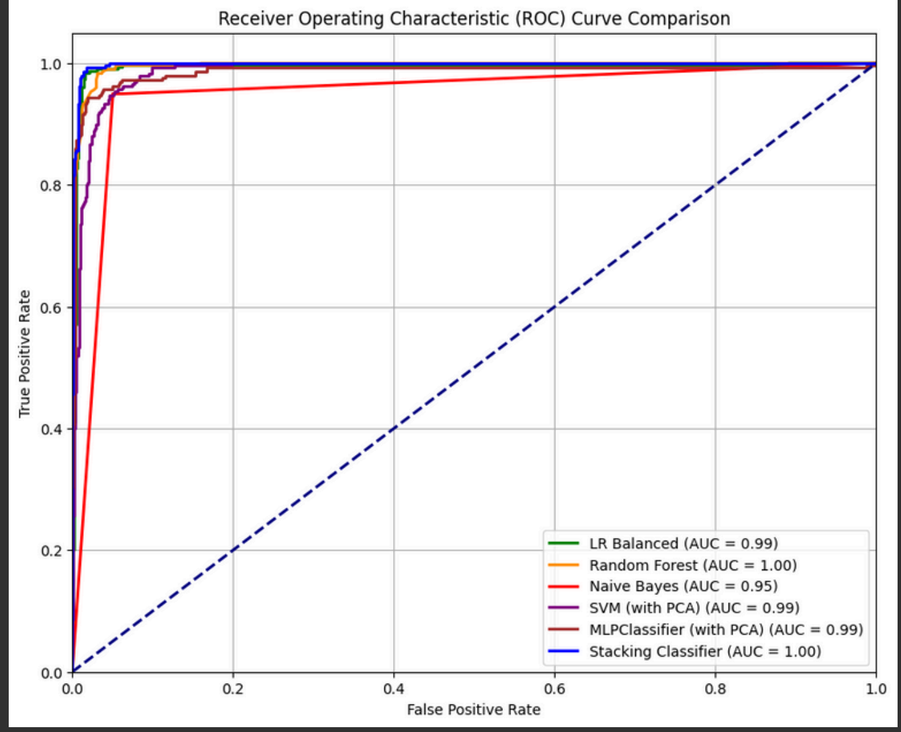
Stacking Classifier:

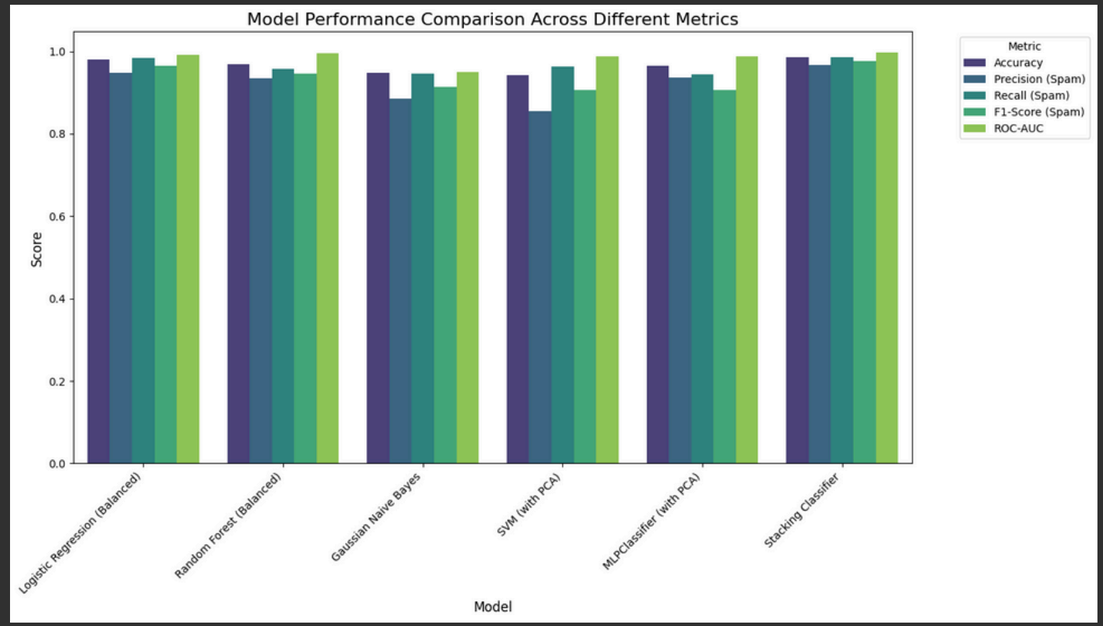
	precision	recall	f1-score	support
0	0.99	0.99	0.99	735
1	0.97	0.99	0.98	300
accuracy			0.99	1035
macro avg	0.98	0.99	0.98	1035
weighted avg	0.99	0.99	0.99	1035

Analysis and Evaluation!

Model Performance Analysis

- **Stacking Classifier:** This ensemble model emerged as a top performer, consistently achieving high scores in Accuracy, Precision (Spam), Recall (Spam), F1-Score (Spam), and ROC-AUC. Its ability to combine the predictions of multiple base models appears to contribute to its robust performance, particularly in balancing precision and recall for the spam class.
- **Logistic Regression (Balanced):** The Logistic Regression model with class weighting also performed exceptionally well, with metrics very close to the Stacking Classifier. Its high recall for spam is a significant strength, indicating its effectiveness in identifying the majority of spam emails.
- **Random Forest (Balanced):** The Random Forest classifier, also using class weighting, delivered strong and competitive results across the evaluation metrics.
- **MLPClassifier (with PCA):** The Neural Network model, trained on PCA-reduced data, showed good performance comparable to Random Forest, demonstrating the benefit of dimensionality reduction for this model.
- **SVM (with PCA):** The Support Vector Machine with PCA achieved good results, particularly high recall for spam. However, its precision for spam was slightly lower compared to some other models, suggesting a potential for more false positives.
- **Gaussian Naive Bayes:** This model had the lowest overall performance among the evaluated models, with lower precision and F1-score for the spam class compared to the others.





Findings and Conclusions

	Accuracy	Precision (Spam)	Recall (Spam)	F1-Score (Spam)	ROC-AUC
Logistic Regression (Balanced)	0.980	0.949	0.983	0.966	0.992
Random Forest (Balanced)	0.968	0.935	0.957	0.946	0.995
Gaussian Naive Bayes	0.949	0.885	0.947	0.915	0.949
SVM (with PCA)	0.942	0.855	0.963	0.906	0.987
MLPClassifier (with PCA)	0.965	0.937	0.943	0.906	0.987
Stacking Classifier	0.986	0.967	0.987	0.977	0.998

Considering the importance of both precision and recall in spam classification (minimizing both false positives and false negatives), the Stacking Classifier and the Logistic Regression (Balanced) models are the most effective based on these results.
The Stacking Classifier holds a slight edge in most performance metrics, suggesting that the ensemble approach provides a marginal improvement in overall classification ability.
However, the Logistic Regression (Balanced) and Random Forest model both offers a strong and competitive alternative with excellent performance characteristics and may be preferred in scenarios where a simpler, more interpretable model is desired.
Ultimately, the choice of the "best" model may also depend on the specific requirements and tolerance for false positives versus false negatives in the deployment scenario. Based on the evaluated metrics, the Stacking Classifier shows the most promising overall performance.