King Saud university

College of Computer and information Sciences

Department of Computer Science

CSC 361: Artificial Intelligence

Course Project

Spring 2025

Team Members:

- **Mohannad alshahrani 442100744**
- **Saud Hamad Alrayes 443170180**

Project Summary

This project uses Machine Learning algorithms to determine whether an email is Spam or Ham (Not Spam).
We want to build an email classifier that, given a new email, can classify it into one of the two categories accurately.
To achieve this, we used Supervised Learning with a Multinomial Naïve Bayes algorithm.
We trained our model with a collection of over 5,500 emails and applied data balancing and text feature extraction methods to optimize the performance of the model.
The end model can distinguish spam emails correctly and demonstrates the effectiveness of Machine Learning in text classification.

Project description

It is almost universal now for every individual to own at least one email account and the majority of people have more than one account.
Most sites request individuals to provide an email address so that they can send them offers, newsletters, or updates.
But at times, these sites sell or make public user emails to third-party businesses, and a much greater problem is formed.
The threat begins when attackers and firms start showering users with spam emails and phishing

attacks.
Even if an individual manages to block unwanted emails quickly,
spammers continually devise new ways of contacting them once again.
Our project seeks to serve as a buffer between users and spam mail by
automatically identifying and deleting unwanted messages before they even reach the user's inbox.
This helps to protect users' time, privacy, and security, and demonstrates how Machine Learning can be used to solve real-world communication problems.

Background:

With the technology expanding and it has become effortless to spam emails with about 160 billion emails sent every day which account for 46% of email traffic and increasing every day comes a grave problem.

Common approaches include:

- Naïve Bayes Classifiers, which are easy and quick.
- Support Vector Machines (SVMs), which yield good decision boundaries.
- Deep learning models (more recently) for large datasets.

One of the simplest yet still very effective spam filtering algorithms is the Multinomial Naïve Bayes algorithm

TF-IDF (Term Frequency–Inverse Document Frequency) helps to represent text emails numerically by highlighting important and rare words in documents.

TF-IDF vectorization along with Multinomial Naïve Bayes is used in this project to build a spam classifier.

Proposed Approach

We began by gathering a labeled data set of email messages that were spam or ham.

We used Python and libraries such as Pandas to handle data and Matplotlib for basic visualizations.

We cleaned the data by:

- Removing unwanted columns and null values

- Deleting special characters and punctuation marks
- Converting text to lower case
- Tokenizing the text into words

Feature Extraction

We applied **TF-IDF (Term Frequency–Inverse Document Frequency)** to convert the processed text into numerical feature vectors suitable for machine learning.
This technique emphasizes rare and meaningful terms, which helps distinguish spam patterns from normal communication.

Model Training

We trained a **Multinomial Naïve Bayes** classifier on the extracted features.
This algorithm is efficient and particularly effective for **text classification** tasks.
We also fine-tuned the model by adjusting the **alpha** hyperparameter and modifying TF-IDF parameters such as min_df, max_df, and ngram_range.

Optimization

To achieve the highest accuracy, we experimented with different preprocessing techniques, combinations of features, and model parameters.
The last model was selected for its accuracy, precision, recall, and F1-score performance.

Results and discussion:

Our model works with good accuracy, but our problem was with the data we gathered it was 13.4% ham and 86.6% spam which was unbalanced so it results bad model that is kind of scared to choose ham with ham recall 0.66:

Class	Precision	Recall	F1-Score	Support
Ham	0.95	1.00	0.97	1208
Spam	0.99	0.66	0.80	185
Accuracy			**0.95**	1393

confusion Matrix: [1211 0] [57 125]]

But with balanced data I got better:

Class	Precision	Recall	F1-Score	Support
Ham	0.98	0.98	0.98	1228
Spam	0.98	0.98	0.98	1185
Accuracy			**0.98**	2413

confusion Matrix: [1209 19] [20 1165]]

after optimization:

TfidfVectorizer(max_df=0.9, min_df=5,ngram_range=(1,2))

MultinomialNB(alpha=0.1)

Class	Precision	Recall	F1-Score	Support
Ham	0.99	0.99	0.99	1235
Spam	0.99	0.99	0.99	1178
Accuracy			**0.99**	2413

confusion Matrix: [1218 17] [13 1165]].

At the end, our model is very effective and gives high accuracy in detecting spam emails. However, it needs more data to improve and handle different types of emails, especially real-world and mixed-language emails. With more diverse data, the model can be further refined and reliable.

Reference:

- https://www.emailtooltester.com/en/blog/spam-statistics/