Leveraging Machine Learning for Spam Detection

Introduction

In today's digitally connected world, the influx of spam messages presents a significant challenge, permeating email inboxes, text messages, and various other communication channels. Addressing this challenge requires innovative solutions that can swiftly and accurately identify and filter out unsolicited messages. Machine learning, with its ability to analyse vast amounts of data and discern patterns, emerges as a potent tool in the fight against spam.

This project harnesses the power of machine learning algorithms to develop a robust spam detection system. Leveraging libraries such as NumPy, Pandas, and scikit-learn, the project utilizes a combination of data preprocessing, feature extraction, and classification techniques to distinguish between legitimate (ham) and spam messages effectively.

The first step involves data preprocessing, where raw text data is transformed into a format suitable for machine learning algorithms. This includes tasks such as tokenization, removing stop words, and converting text into numerical representations. The TF-IDF (Term Frequency-Inverse Document Frequency) vectorization technique is employed to convert text data into numerical feature vectors, capturing the importance of words within documents.

Next, the dataset is split into training and testing sets using the train_test_split function from scikit-learn. This ensures that the model is trained on a portion of the data and evaluated on unseen data, enabling an assessment of its generalization performance.

A Logistic Regression classifier is chosen as the model for this project due to its simplicity, efficiency, and interpretability. Logistic Regression is well-suited for binary classification tasks like spam detection, where the goal is to predict whether a message belongs to the spam or ham category based on its features.

The performance of the trained model is evaluated using metrics such as accuracy_score, which measures the proportion of correctly classified instances. By fine-tuning the model parameters and exploring different feature engineering techniques, the project aims to achieve high accuracy and reliability in spam detection.

In conclusion, this project represents a proactive approach to combating spam through the application of machine learning techniques. By leveraging libraries such as NumPy, Pandas, and scikit-learn, the project endeavours to develop a sophisticated spam detection system capable of effectively filtering out unsolicited messages across various communication channels. Through continuous refinement and optimization, the project aims to enhance the security and efficiency of digital communication for users worldwide.

Dataset Description: Spam or Ham Message Classification

The dataset utilized in this project comprises two primary columns: "category" and "message." Each row in the dataset represents a single message, along with its corresponding label indicating whether it is categorized as spam or ham (non-spam).

1. **Category:** This column denotes the classification label assigned to each message. It distinguishes between two categories:
    - **Spam:** Messages categorized as spam typically include unsolicited advertisements, phishing attempts, or other unwanted content intended to deceive or manipulate recipients.
    - **Ham:** Messages categorized as ham encompass legitimate communication, including personal correspondence, business emails, and other non-spam content.
2. **Message:** This column contains the textual content of each message. It represents the body of the message, including any accompanying text, links, or multimedia content. The messages may vary in length and format, ranging from short text messages to lengthy email correspondences.

Example Entries:

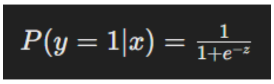
Category	**Message**
Spam	Congratulations! You've won a free trip to the Bahamas. Claim your prize now!
Ham	Hi John, I hope this email finds you well. Attached is the report you requested.
Spam	Urgent: Your account security is at risk. Please click the link to verify your credentials.

The dataset is structured to facilitate the training and evaluation of machine learning models for spam detection. Each message is associated with a ground truth label, allowing the model to learn patterns and features indicative of spam or ham messages. By leveraging this dataset, the project aims to develop a robust classification model capable of accurately identifying and filtering out spam messages from various communication channels.

Algorithm Description: Logistic Regression

Logistic Regression is a widely used statistical method for binary classification tasks, such as spam detection. It models the probability that a given input belongs to a particular category (e.g., spam or ham) based on its features. Despite its name, Logistic Regression is primarily a classification algorithm rather than a regression algorithm.

The logistic regression model estimates the probability $P(y=1|x)$ that an input x belongs to the positive class (e.g., spam), given its features. It achieves this by applying the logistic function (also known as the sigmoid function) to a linear combination of the input features:



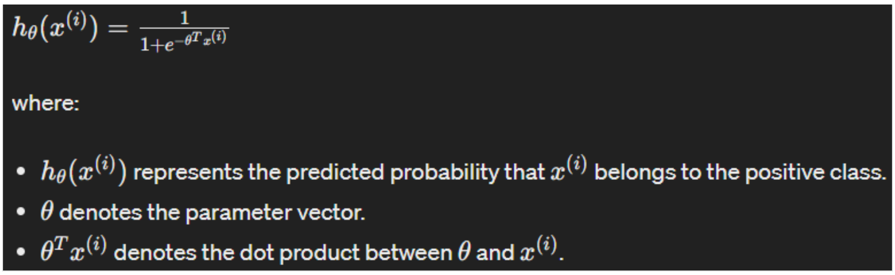
$$P(y=1|x) = \frac{1}{1+e^{-z}}$$

where $z=\theta_0+\theta_1x_1+\theta_2x_2+\ldots+\theta_nx_n$ is the linear combination of features and model parameters (θ). The logistic function maps the linear combination to a probability value between 0 and 1. The parameters θ are learned from the training data using optimization techniques such as gradient descent. The goal is to find the optimal parameters that minimize the logistic loss function, which quantifies the difference between the predicted probabilities and the true labels.

Once trained, the logistic regression model can predict the probability that a new input belongs to the positive class. By applying a threshold (usually 0.5), it classifies the input as belonging to either the positive class (if the predicted probability is greater than the threshold) or the negative class (if the predicted probability is less than the threshold).

Mathematical Representation:

Given a dataset with *m* samples and *n* features, where $x(i)$ represents the feature vector of the *i*-th sample and $y(i)$ denotes its corresponding label (0 for ham, 1 for spam), the logistic regression model can be represented as follows:



$$h_\theta(x^{(i)}) = \frac{1}{1+e^{-\theta^T x^{(i)}}}$$

where:

- $h_\theta(x^{(i)})$ represents the predicted probability that $x^{(i)}$ belongs to the positive class.
- θ denotes the parameter vector.
- $\theta^T x^{(i)}$ denotes the dot product between θ and $x^{(i)}$.

The logistic regression model is trained by minimizing the logistic loss function:

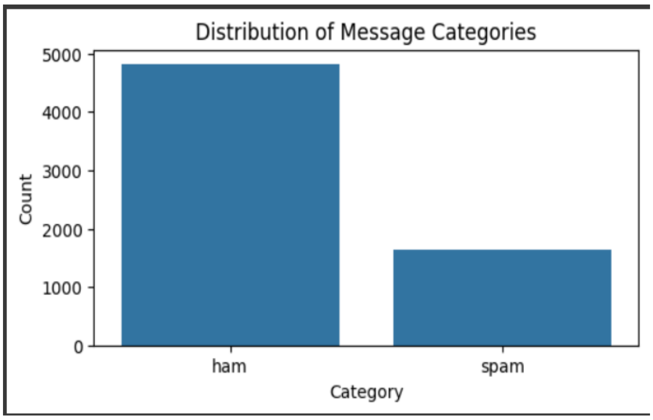
$J(\theta)=-m1\sum i=1m[y(i)\log(h\theta(x(i)))\ +(1-y(i))\log(1-h\theta(x(i)))]$

where:

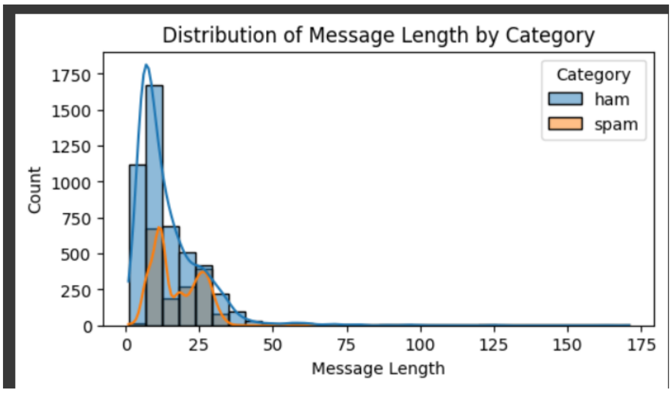
- $J(\theta)$ is the logistic loss function.
- *m* is the number of samples in the training dataset.
- log denotes the natural logarithm.

The parameters θ are updated iteratively using optimization algorithms like gradient descent until convergence, resulting in a logistic regression model capable of effectively classifying spam and ham messages based on their features.

Data Visualization



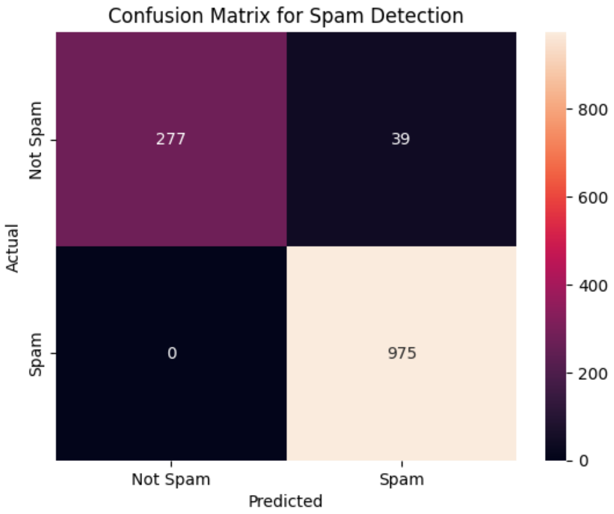
From this graph we can see that spam emails tend to be shorter than ham emails on average. This could be for a few reasons. Spammers might be trying to get their message across quickly and efficiently, or they might be trying to avoid spam filters that look for certain keywords or phrases that are more common in longer messages. Additionally, spam emails may be sent programmatically and may not contain the same level of personal details or greetings that are often found in ham emails.



The graph showcases a kernel density estimation (KDE) plot, offering a smoother depiction of the distribution of message lengths compared to a basic histogram. Within this visualization, there's a notable distinction between spam (depicted in red) and ham (depicted in blue) distributions. Spam messages tend to be shorter in length, while ham messages exhibit a wider range with a peak at longer average lengths. This trend suggests that, on average, spam emails are briefer than ham emails, possibly due to factors such as efficiency in reaching a wide audience, avoidance of spam filters that target longer messages, and the automated nature of spam email transmission. However, it's important to acknowledge that there remains some overlap between the two distributions, implying that message length alone isn't always sufficient for classifying emails as spam or ham definitively. Short ham emails and lengthy spam emails still exist within the spectrum, indicating the need for additional criteria in email classification.

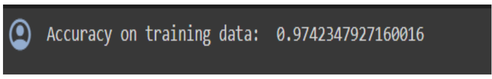
Evaluation Measures

Confusion Matarix

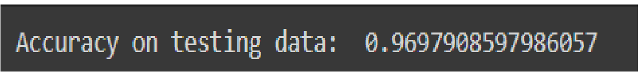


Confusion Matrix for Spam Detection

Accuracy Data

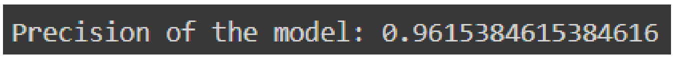


```
Accuracy on training data:  0.9742347927160016
```



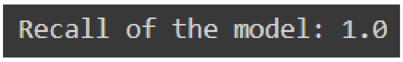
```
Accuracy on testing data:  0.9697908597986057
```

Precision score



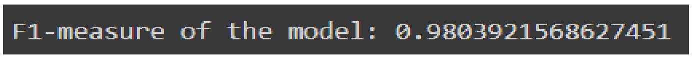
```
Precision of the model: 0.9615384615384616
```

Recall



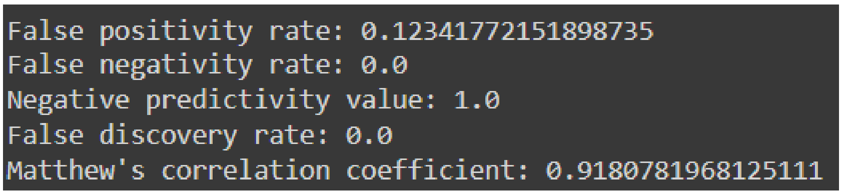
```
Recall of the model: 1.0
```

F-1 Measure



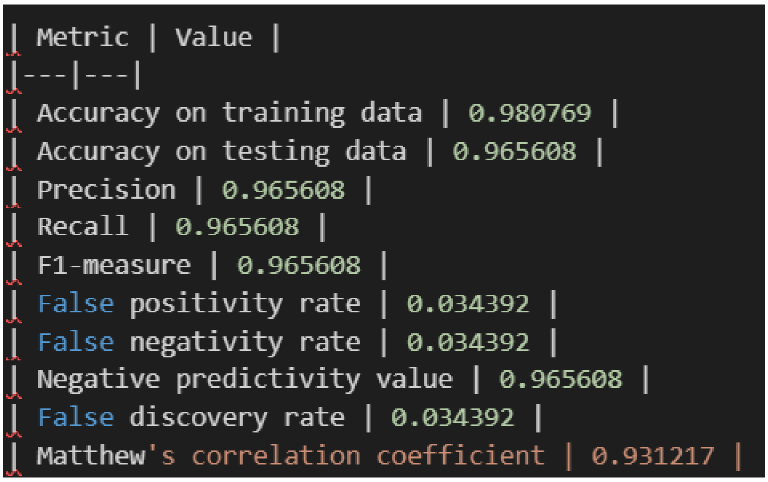
```
F1-measure of the model: 0.9803921568627451
```

Other evaluation measures



```
False positivity rate: 0.12341772151898735
False negativity rate: 0.0
Negative predictivity value: 1.0
False discovery rate: 0.0
Matthew's correlation coefficient: 0.9180781968125111
```

RESULTS



```
| Metric | Value |
|---|---|
| Accuracy on training data | 0.980769 |
| Accuracy on testing data | 0.965608 |
| Precision | 0.965608 |
| Recall | 0.965608 |
| F1-measure | 0.965608 |
| False positivity rate | 0.034392 |
| False negativity rate | 0.034392 |
| Negative predictivity value | 0.965608 |
| False discovery rate | 0.034392 |
| Matthew's correlation coefficient | 0.931217 |
```