Spam Mail Prediction Using Machine Learning

**Summitted in partial fulfillment of the requirements for the awards of**

**Degree of**

**BACHELOR OF COMPUTER APPLICATION**

**By**

**Vinit kumar (22BCA012)**

**Sanjay Dwivedi (22BCA046)**

**Abhishek Yadav (22BCA001)**

**Under the Supervision of**

**Er. Vineesh Cutting**

**(Assistant Professor)**

****

**FACULTY OF ENGINEERING AND TECHNOLOGY**

**SAM HIGGINBOTTOM UNIVERSITY OF AGRICULTURE, TECHNOLOGY & SCIENCES**

**(FORMERLY ALLAHABAD AGRICULTURAL INSTITUTE)**

**NAINI, PRAYAGRAJ-211007, INDIA (2024)**

**1. Introduction**

This program utilizes Logistic Regression to classify emails as "spam" or "ham" (non-spam), enhancing inbox management and security. It preprocesses email data, extracts features using TF-IDF vectorization, trains a model, evaluates its accuracy, and provides a predictive system for real-time classification.

**2**. **Objectives**

* Utilize Logistic Regression for accurate classification of emails into "spam" or "ham" categories.
* Improve inbox management by effectively filtering out unwanted spam messages.
* Filtering of large volumes of email data for both personal and professional use.

**3. Existing Systems**

**3.1. Traditional Approach:**

* Relies on rule-based methods or basic keyword matching.
* Often results in inaccuracies and high false-positive rates.

**3.2. Limitations:**

* Unable to effectively adapt to evolving spam email tactics.
* May overlook subtle variations in spam email content.

**3.3. User Experience:**

* Users may encounter difficulties in managing inbox clutter.
* Security risks remain due to inadequate spam filtering.

**3.4. Need for Improvement:**

* Demand for more sophisticated techniques to combat spam.
* Requirement for solutions that balance accuracy and user convenience.

**4. Proposed System**

* Logistic Regression for accurate email classification, enhancing inbox management. Seamless integration into existing email platforms ensures user convenience. Continuous improvement mechanisms adapt to evolving spam patterns.
* This system aims to address the objectives of accurately classifying emails, seamlessly integrating the classification feature into existing email platforms, and incorporating mechanisms for continuous improvement.
* It involves the development, testing, and deployment of the classification model, along with user interface enhancements to ensure a smooth user experience.

**5. Hardware and Software Use**

**5.1. Hardware:**

|  |  |
| --- | --- |
| Processor | Intel® Core™ i5-1135G7 Processor 2.4 GHz (8M Cache, up to 4.2 GHz, 4 cores) |
| Resolution | 1920 x 1080 x 60 hertz |
| Memory (RAM) | 4GB DDR4 |
| Input Devices | Keyboard and Mouse |
| Storage (SSD) | 512GB M.2 NVMe™ PCIe® 3.0 SSD |

|  |  |
| --- | --- |
| Technology | Python 3.11.3 |
| Python Libraries | Numpy, Pandas, train\_test\_split, TfidfVectorizer, LogisticRegression, accuracy\_score |
| Operating System | Windows 11 Home (11th Gen) |
| Browers | Microsoft Edge (Version 124.0.2478.105) |
| Code Editor | Google Colab Notebook |

**5.2. Software:**

**6. System Design**

**6.1. Modular Approach:**

* Program is divided into separate parts like data handling, model training, and evaluation.
* Allows for easier management and scalability.

**6.2. Workflow:**

* Begins with data collection and preprocessing.
* Splits data for training and testing.
* Converts text to numerical features for analysis.
* Trains a Logistic Regression model and evaluates its performance.

**6.3. Adaptability:**

* Designed to accommodate future upgrades and different datasets.
* Offers flexibility for integration with advanced techniques.

**7. Work Flow**

**8. Flow Diagram**

Data Preparation

Start Data Preparation

Load the necessary libraries and modules for the spam detection task, including NumPy, Pandas, and various scikit-learn components.

Load Data  
Read the 'mail\_data.csv' file and store the raw mail data in the 'raw\_mail\_data' variable.

Clean Data  
Handle missing values in the raw mail data by replacing them with empty strings. Also, convert the 'spam' and 'ham' categories to numerical values of 0 and 1, respectively.

Split Data into  
Training and Test Sets  
Use the train\_test\_split function from scikit-learn to divide the mail data into training and test sets, with a test size of 20% and a random state of 3.

Extract Feature

Convert the text-based mail messages into numerical feature vectors using the TfidfVectorizer from scikit-learn. Fit the vectorizer on the training data and transform both the training and test data.

Prepare Targets  
Convert the 'Category' column in the training and test sets to integer data types.

End Data Preparation

Provides Prepared Data

Model Training

Start Model Training

Train Logistic Regression Model  
Create a Logistic Regression model instance and fit it to the training data features and targets.

Evaluate Model on  
Training Data  
Use the trained model to make predictions on the training data and calculate the accuracy score.

Evaluate Model on  
Test Data  
Use the trained model to make predictions on the test data and calculate the accuracy score.

End Model Training

Provides Trained Model

Predict New Mail

Start Prediction

Prepare New Mail  
Create a list containing a new mail message to be classified as spam or ham.

Extract Features from New Mail  
Use the previously trained TfidfVectorizer to transform the new mail message into a numerical feature vector.

Predict Mail Type  
Use the trained Logistic Regression model to predict whether the new mail message is spam 0 or ham 1 .

Display Result  
Print the predicted mail type spam or ham based on the model's output.

End Prediction

**9. Future Scope**

* This program aims to integrate advanced ML algorithms, implement real-time email processing, incorporate user feedback mechanisms, and enhance user experience for efficient email management.
* This program may expand to handle multi-class classification, integrate with email clients, collaborate with service providers, and analyze metadata or attachments for comprehensive spam detection.

**10. Conclusion**

This program demonstrates the effectiveness of Logistic Regression in email spam classification, improving inbox management and security. With potential for further enhancements such as advanced algorithms and real-time processing, it represents a promising solution for combating email spam and enhancing user experience.