

SPAM MAIL DETECTOR USING RANDOM FOREST

DHARUN J

220901020

PPML MINI PROJECT

Abstract

Spam detection is a critical task in modern communication systems to ensure security and improve user experience. This project implements a spam detection system using machine learning techniques. The dataset, comprising labeled messages as "spam" or "ham," is preprocessed and analyzed. The system utilizes the Random Forest classifier to predict whether a given message is spam. Text data is converted into numerical format using CountVectorizer, enabling the classifier to process it efficiently. The model achieves high accuracy and provides real-time predictions for user-input messages, demonstrating its utility in identifying spam effectively.

Report

Introduction

Spam messages pose a significant challenge by cluttering inboxes and potentially delivering malicious content. Automated systems for spam detection reduce this burden by leveraging machine learning. This report outlines the development of a spam detection system that uses the Random Forest algorithm for classification, a robust ensemble technique known for high accuracy and scalability.

Objective

The goal of this project is to develop a machine learning model that identifies spam messages, providing users with a reliable way to filter unwanted content.

Methodology

1. Data Collection:

The dataset used for training and testing is sourced from publicly available spam message datasets (spam.csv).

- Columns include v1 (label) and v2 (message content).

2. Data Preprocessing:

- Labels (v1) were mapped to numerical values (ham = 0, spam = 1).
- Features (message content) were split into training and test sets (80-20 split).

3. Text Vectorization:

- CountVectorizer converted text data into numerical vectors by representing word frequency.

4. Model Training:

- A Random Forest Classifier with 100 estimators was used to train the model on vectorized data.

5. Evaluation Metrics:

- Accuracy and classification reports (precision, recall, F1-score) were used to evaluate the model.

Results

- **Accuracy:** The model achieved high accuracy on the test data.
- **Classification Report:** Provided detailed insights into the model's performance for "spam" and "ham" classes, highlighting balanced precision and recall values.

Key Features

1. Real-Time Predictions:

Users can input messages and receive instant spam predictions.

2. **Robust Model:**

The Random Forest classifier ensures stability and high prediction accuracy.

3. **User-Friendly Interface:**

Iterative input system with an exit option for ease of use.

Future Scope

- **Improved Text Processing:** Integrate lemmatization, stemming, or advanced vectorization techniques.
- **Deep Learning Models:** Explore transformer-based architectures like BERT for enhanced contextual understanding.
- **Scalability:** Deploy the system for real-world datasets with larger volumes and multilingual support.

Conclusion

The spam detection system successfully identifies spam messages with high accuracy. Its real-time prediction capability makes it practical for deployment in various applications. Future improvements could involve using advanced NLP techniques like TF-IDF or neural networks to enhance performance further.

Program :

```
import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.feature_extraction.text import CountVectorizer

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy_score, classification_report

data = pd.read_csv('spam.csv.csv', encoding='latin-1')
```

```

data = data[['v1', 'v2']]

data.columns = ['label', 'message']

data['label'] = data['label'].map({'ham': 0, 'spam': 1})

X = data['message']

y = data['label']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

vectorizer = CountVectorizer()

X_train_vectorized = vectorizer.fit_transform(X_train)

X_test_vectorized = vectorizer.transform(X_test)

model = RandomForestClassifier(n_estimators=100, random_state=42)

model.fit(X_train_vectorized, y_train)

y_pred = model.predict(X_test_vectorized)

print(f'Accuracy: {accuracy_score(y_test, y_pred)}')

print(classification_report(y_test, y_pred))

def predict_message(message):

    message_vectorized = vectorizer.transform([message])

    prediction = model.predict(message_vectorized)

    return 'Spam' if prediction[0] == 1 else 'Not Spam'

for i in range(15):

    user_input = input(f"Enter message {i+1} to check if it's spam (or type 'exit' to quit): ")

    if user_input.lower() == 'exit':

        print("Exiting the spam detection program.")

        break

    result = predict_message(user_input)

    print(f'The message is: {result}')

```

output :

Accuracy: 0.9757847533632287

	precision	recall	f1-score	support
0	0.97	1.00	0.99	965
1	1.00	0.82	0.90	150
accuracy		0.98		1115
macro avg	0.99	0.91	0.94	1115
weighted avg	0.98	0.98	0.97	1115

Enter message 1 to check if it's spam (or type 'exit' to quit): You have been selected as the lucky winner of our Global Lottery Draw. To claim your prize of \$1,000,000, reply with your name, address, phone number, and bank details. Hurry! Your prize expires in 48 hours! Warm regards, Global Lottery Committee

The message is: Spam

Enter message 2 to check if it's spam (or type 'exit' to quit): Subject: Urgent: Your Account Will Be Deactivated Dear Customer, We noticed unusual activity in your bank account. To ensure your security, we need you to verify your information. Click the link below to update your details: Click here to secure your account Failure to do so within 24 hours will result in account suspension. Thank you, Bank Security Team

The message is: Not Spam

Enter message 3 to check if it's spam (or type 'exit' to quit): Subject: Work From Home & Earn \$5,000 Weekly! Hi, Are you tired of your 9-to-5 job? Join our team and start earning \$5,000+ per week from the comfort of your home! No experience needed. Sign up now at JoinOurTeam.com. Limited spots available! Cheers, John - Recruitment Manager

The message is: Spam

Enter message 4 to check if it's spam (or type 'exit' to quit): exit

Exiting the spam detection program.