# Detecting Spam Emails

## Student Details

| | |
|---|---|
| Name: | SHERIN BANU Z.G |
| NM Id: | au61772111092 |
| College Name: | GOVERNMENT COLLEGE OF ENGINEERING, SALEM |

## Disclaimer

The content is curated from online/offline resources and used for educational purpose only

## Course Outline

- Abstract
- Problem Statement
- Aims, Objective & Proposed System/Solution
- System Deployment Approach
- Model Development & Algorithm
- Future Scope
- Video of the Project
- Conclusion
- Reference

# Detecting Spam Emails

## Abstract

Email spam continues to be a pervasive issue, causing inconvenience and potential harm to users. In this project, we propose a machine learning approach to automatically detect spam emails. The project involves collecting a dataset of labeled emails, where each email is classified as spam or non-spam (ham). The project aims to develop a robust and efficient spam detection system that can be deployed to protect users from unwanted and potentially harmful emails.

# Detecting Spam Emails

## Problem Statement

- Detecting Spam Emails Using TensorFlow. Implement and build a deep-learning model for Spam Detection. The model we will try to implement will be a Classifier, which would give binary outputs- either spam or ham.

## Aim and Objective

**Aim:** This project aims to develop an effective spam email detection system using machine learning techniques. The project will involve collecting a dataset of labeled emails, preprocessing the email text data, extracting relevant features, and training a machine learning model to classify emails as spam or non-spam (ham).

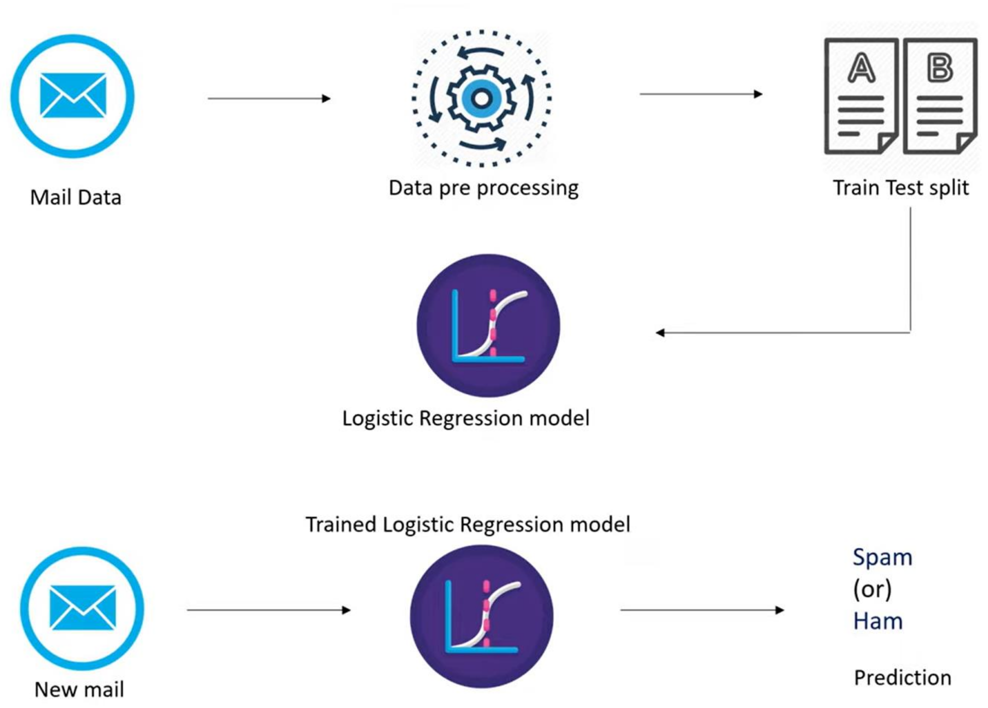
## Objectives

- Collect a dataset of labeled emails, including spam and non-spam (ham) emails.
- Preprocess the email text data to prepare it for machine learning model training.
- Extract relevant features from the email text data using techniques such as TF-IDF and N-grams.
- Train a machine learning model, such as a Naive Bayes classifier or a Support Vector Machine (SVM), using the extracted features.
- Evaluate the performance of the trained model using metrics such as accuracy, precision, recall, and F1 score..
- Develop a user-friendly interface for the spam detection system, allowing users to easily classify emails as spam or non-spam.

## Proposed Solution

- **Solution:** Gather a dataset of labeled emails, where each email is labeled as spam or non-spam (ham). Preprocess the text data to convert it into a format suitable for machine learning models. This may include removing stop words, tokenization, and converting text to numerical representations. Extract features from the preprocessed text data. Common features include word frequency, TF-IDF (Term Frequency-Inverse Document Frequency), and N-grams. Choose a machine-learning model for spam detection. Common models include logistic regression, support vector machines (SVM), and naive Bayes classifiers.

## System Deployment Approach



Mail Data

Data pre processing

Train Test split

Logistic Regression model

Trained Logistic Regression model

New mail

Spam
(or)
Ham

Prediction

# Detecting Spam Emails

## Model Development & Algorithm

**Dataset Description:**

- The dataset contains set of Emails.
- Size of dataset is 5572 rows
- Categorized into two classes
- Spam, Ham
- Each class has around 2780 sentences

## Model Development & Algorithm

Algorithm:

**1.Input:** Email text data

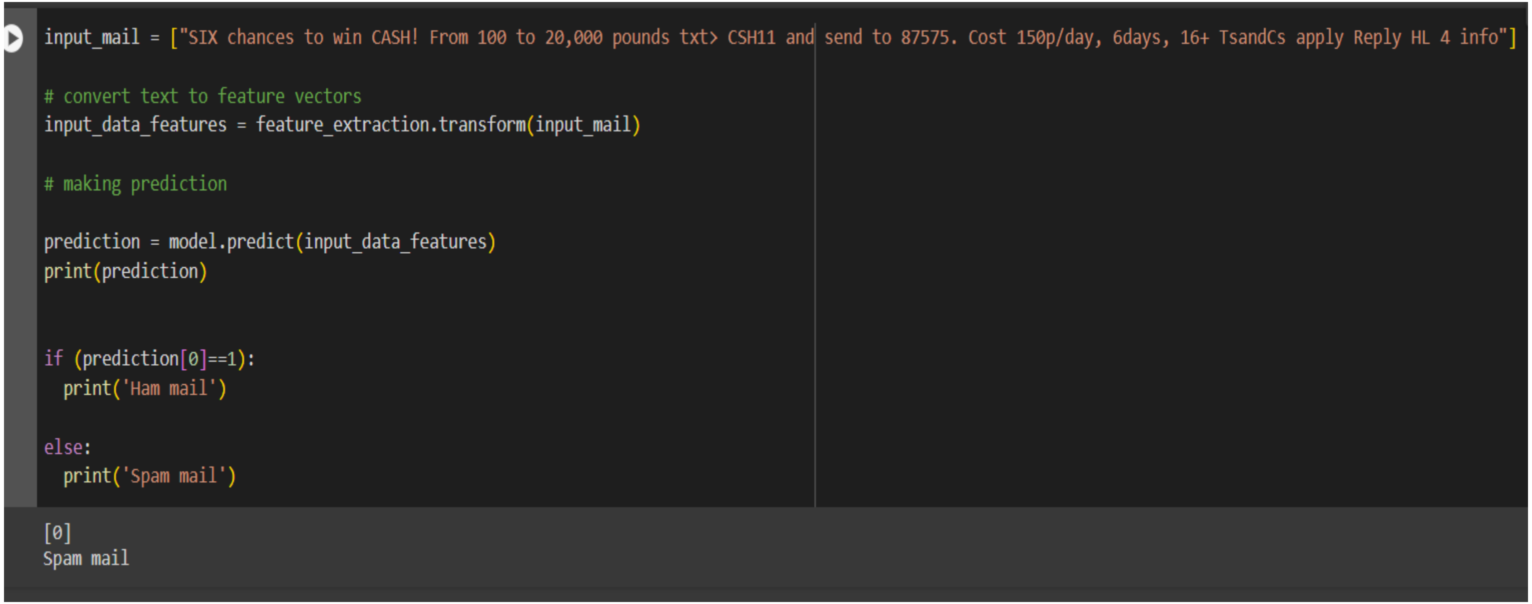
**2.Output:** Spam or non-spam (ham) classification

Algorithm Steps:

- Preprocess the email text data (e.g., remove stop words, tokenize).
- Extract features from the pre-processed text data (e.g., TF-IDF, N-grams).
- Train a machine learning model on the extracted features (e.g., logistic regression, SVM, naive Bayes).
- Evaluate the trained model on a test dataset using metrics such as accuracy, precision, recall, and F1 score.
- Fine-tune the model by experimenting with different preprocessing techniques, feature extraction methods, and model parameters to improve performance.
- Deploy the trained model as a spam detection system for real-world use.

# Detecting Spam Emails

## Result



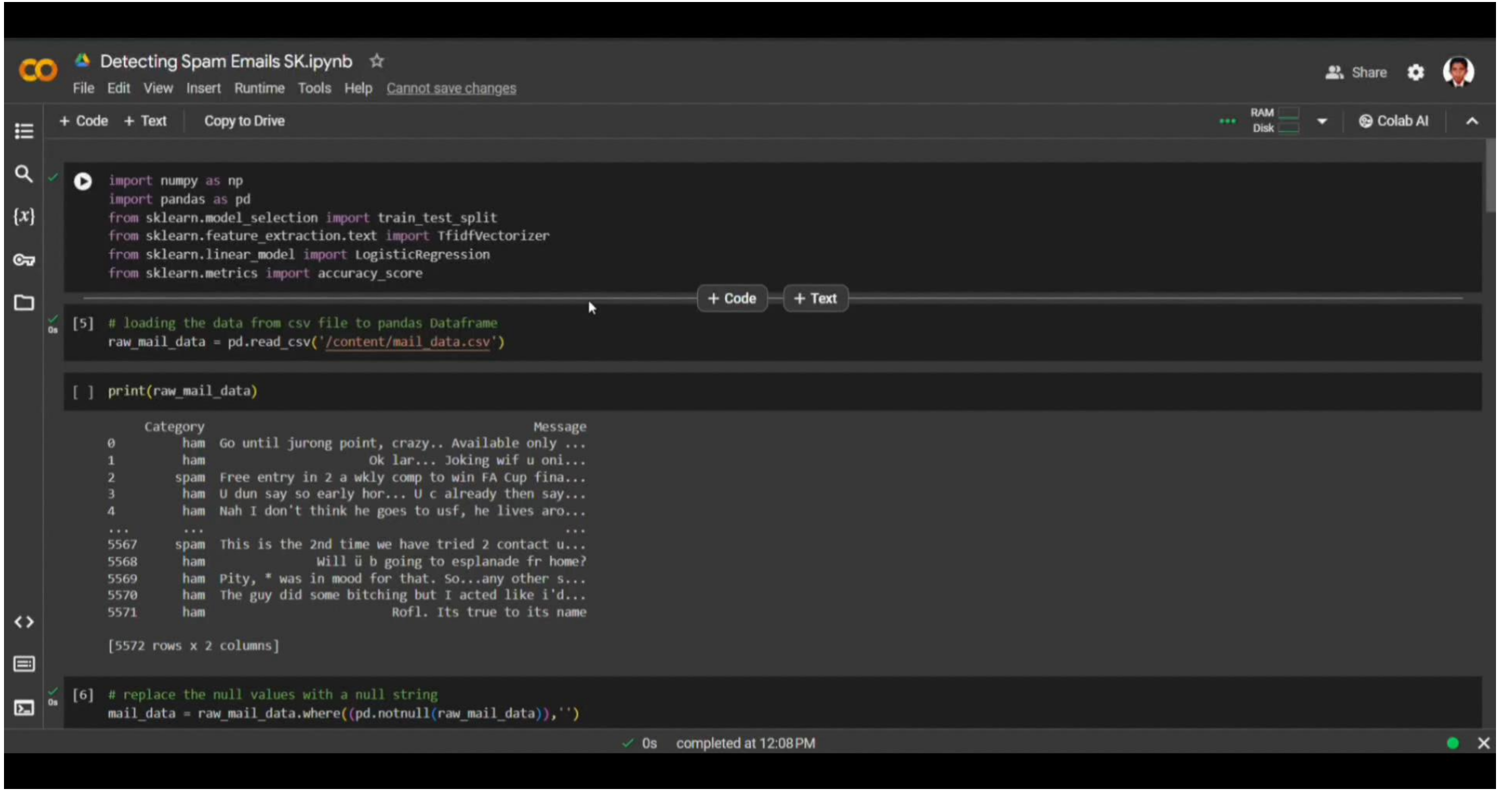
```
input_mail = ["SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info"]

# convert text to feature vectors
input_data_features = feature_extraction.transform(input_mail)

# making prediction

prediction = model.predict(input_data_features)
print(prediction)


if (prediction[0]==1):
  print('Ham mail')

else:
  print('Spam mail')
```

```
[0]
Spam mail
```

## Future Scope

1. **Advanced Machine Learning Techniques:** Explore advanced machine learning techniques such as deep learning, ensemble methods, and natural language processing (NLP) to improve the accuracy and efficiency of spam email detection.
2. **Real-Time Detection:** Develop real-time spam detection systems that can quickly identify and filter out spam emails as they are received, providing users with immediate protection.
3. **User Feedback Integration:** Incorporate user feedback into the spam detection system to continuously improve its performance and adapt to new spamming techniques.
4. **Multimodal Detection:** Combine text-based features with other modalities such as images and metadata to improve the detection of sophisticated spam emails.

## Video of the Project


Detecting Spam Emails SK.ipynb
File Edit View Insert Runtime Tools Help Cannot save changes
Share
+ Code + Text Copy to Drive
RAM
Disk
Colab AI
+ Code
+ Text
0s completed at 12:08 PM

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

```
[5] # loading the data from csv file to pandas Dataframe
    raw_mail_data = pd.read_csv('/content/mail_data.csv')
```

```
[ ] print(raw_mail_data)
```

```
         Category                                            Message
    0         ham  Go until jurong point, crazy.. Available only ...
    1         ham                      Ok lar... Joking wif u oni...
    2        spam  Free entry in 2 a wkly comp to win FA Cup fina...
    3         ham  U dun say so early hor... U c already then say...
    4         ham  Nah I don't think he goes to usf, he lives aro...
    ...       ...                                                ...
    5567     spam  This is the 2nd time we have tried 2 contact u...
    5568      ham               Will ü b going to esplanade fr home?
    5569      ham  Pity, * was in mood for that. So...any other s...
    5570      ham  The guy did some bitching but I acted like i'd...
    5571      ham                         Rofl. Its true to its name

    [5572 rows x 2 columns]
```

```
[6] # replace the null values with a null string
    mail_data = raw_mail_data.where((pd.notnull(raw_mail_data)),'')
```

## Conclusion

- In conclusion, our project on spam email detection using machine learning techniques has shown promising results in automatically identifying and filtering out unwanted emails. By leveraging machine learning models such as logistic regression, support vector machines (SVM), and naive Bayes classifiers, we could effectively classify emails as spam or non-spam (ham).
- Moving forward, there is potential to enhance this system by exploring advanced machine learning techniques, implementing real-time detection, and incorporating user feedback. These improvements could further enhance the accuracy and efficiency of spam email detection systems, ultimately improving user experience and security.

## Reference

- https://www.coursera.org/
- https://www.udacity.com/
- https://www.kaggle.com/learn
- https://codelabs.developers.google.com/

# Thank you!