Email Spam Detection Project

This project builds a machine learning model to classify emails as **spam** or **ham (not spam)** using the **SMS Spam Collection Dataset**. The pipeline includes text preprocessing, TF-IDF feature extraction, and classification using **Naive Bayes** and **Support Vector Machine (SVM)** algorithms.

Project Overview

Problem Statement: Spam emails reduce productivity and pose security risks. Automating spam detection improves email filtering.

Goal: Build and evaluate models that can accurately classify email messages as spam or ham.

Dataset

Source: [Kaggle - SMS Spam Collection Dataset](https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset)

Structure:

- `v1`: Label (ham or spam)

- `v2`: Message content

Steps Performed

1. **Data Loading**

- Read the `spam.csv` file

- Renamed columns to `label` and `message`

2. **Preprocessing**

- Convert to lowercase

- Remove numbers, punctuation, extra whitespace using regex

- Map labels: `ham` 0, `spam` 1

3. **Feature Extraction (TF-IDF)**

- Used `TfidfVectorizer` from `scikit-learn`

- Removed English stopwords

- Limited vocabulary to top 3000 words

4. **Model Training and Evaluation**

- Train/Test split (80/20)

- Trained and evaluated:

  - **Multinomial Naive Bayes**

- **Support Vector Machine (SVM)**

- Metrics: Accuracy, Precision, Recall, F1-score

- Visualized confusion matrices using Seaborn heatmaps

5. **Model Saving**

- Saved models and TF-IDF vectorizer using `joblib`:
  - `naive_bayes_model.pkl`
  - `svm_model.pkl`
  - `tfidf_vectorizer.pkl`

Results Summary

Both models performed well, with SVM often achieving slightly better accuracy. Confusion matrices were used to identify misclassifications.