SMS/EMAIL SPAM CLASSIFIER

PROJECT OVERVIEW

Project: SMS Spam Detection Model Using Data Science Techniques in Python

Overview

I developed a machine learning model to classify SMS messages as either spam or non-spam. This project involved various stages of data preprocessing, feature engineering, model selection, evaluation, and deployment using Python and several key libraries. The goal was to create an accurate and efficient classifier to improve user experience and security by filtering out spam messages.

Problem Statement

With the increasing volume of SMS traffic, distinguishing spam from legitimate messages has become crucial. The objective was to build a model that accurately identifies spam messages to enhance user communication experiences and reduce security risks.

Tools & Technologies

- **Python:** Core programming language for the project.
- **Pandas & NumPy**: For data manipulation and analysis.
- **Scikit-Learn:** For implementing and evaluating machine learning models.
- **Matplotlib & Seaborn:** For data visualization.
- **Flask:** For deploying the model as a web application (or mention any other deployment tool used).

Data Preparation

- **Data Collection:** Utilized a dataset of SMS messages labeled as spam or non-spam.
- **Preprocessing:** Cleaned and standardized text data through tokenization, removal of stop words, and lemmatization.

Model Development

- **Feature Engineering:** Transformed text data into numerical features using TF-IDF (Term Frequency-Inverse Document Frequency).

-**Algorithm Selection:** Evaluated multiple algorithms including Logistic Regression, Naive Bayes, Support Vector Machines, and Random Forest.

- **Cross-Validation:** Used cross-validation to tune hyperparameters and select the best model.

- **Model Evaluation:** Assessed models based on accuracy, precision to ensure balanced performance.

Results

- **Performance Metrics:** The optimized model achieved an accuracy of 98.45% , with a precision of 98.41%

- **Visualizations:** Created confusion matrices and ROC curves to visualize and interpret the model's performance.

Deployment

- **Implementation:** Deployed the model using Flask (or any other specified tool), allowing for real-time SMS spam detection.

- **Repository:** https://github.com/sravanthi224/SMS-EMAIL-SPAM-CLASSIFIER.git

Key Learnings

- The significance of thorough data preprocessing in NLP tasks.

- The process of evaluating and tuning different machine learning models for optimal performance.

- Practical experience in deploying a machine learning model for real-world applications.