UNIVERSITY OF GUJRAT
A WORLD CLASS UNIVERSITY

Email Spam Identifier

NAME:

Abdullah Khan

ROLL NO:

20014119-037

CLASS:

BSCS-A (EVENING)

COURSE CODE:

CS-447

SUBJECT:

Machine Learning

DATE:

6th June 2024

DEPARTMENT:

COMPUTER SCIENCE

SUBMITTED TO:

Dr. Zahid Iqbal

Problem Statement:

The objective is to ***develop a machine learning model that can accurately classify SMS messages as either spam or not spam***. The model will be evaluated based on its ability to correctly identify spam messages while minimizing false positives for legitimate messages.

Dataset:

SMS Spam Collection Dataset:
This dataset contains 5,572 SMS messages in English, labeled as either "ham" or "spam". I have sourced this dataset from Kaggle - SMS Spam Collection Dataset. The format of the dataset is in CSV file with two columns: "label" (spam or ham) and "message" (the text of the SMS).

The content dataset consists of real SMS messages, providing a realistic set of examples for training and evaluating a spam detection model. Each message is pre-labeled, simplifying the supervised learning process. "Ham" indicates a legitimate message, and "spam" indicates an unsolicited or unwanted message. The dataset will be split into training and testing sets. The messages will undergo preprocessing to transform them into a format suitable for machine learning models.

Classifiers to Apply:

We will use the following classifiers:

1. **Naive Bayes Classifier:**
    - **Advantages**: Simple, fast, and effective for text classification problems.
2. **Support Vector Machine (SVM):**
    - **Advantages**: High accuracy, particularly effective for text data, and good at handling high-dimensional data.
3. **Random Forest Classifier:**
    - **Advantages**: High accuracy, robustness to overfitting, and good performance with imbalanced datasets.

Implementation Plan:

Data Preprocessing:

- Load and clean the datasets.
- Remove duplicates and irrelevant information.
- Normalize the text by converting it to lowercase, removing stop words and stemming.

Feature Extraction:

- Convert text data into numerical features using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings.

Model Training:

- Split the datasets into training and testing sets.
- Train the Naive Bayes, SVM, and Random Forest classifiers on the training data.Tune hyperparameters using cross-validation.

Model Evaluation:

- Evaluate the models on the test data using metrics such as accuracy, precision, recall, and F1-score.
- Compare the performance of the different classifiers.

Model Deployment:

- Develop a simple web interface or API to demonstrate the spam detection capability of the best performing model.