SPAM /HAM Internship Project

📄 Project Documentation: SMS Spam-Ham Classification using Logistic Regression

📝 Introduction

This project focuses on classifying SMS messages as **Spam** or **Ham (Not Spam)** using traditional machine learning techniques. It uses the `spam.csv` dataset and applies logistic regression after transforming the text data using **TF-IDF vectorization**. The main objective is to develop an efficient model that can distinguish between spam and legitimate messages with high accuracy.

📚 Dataset Overview

- Source: `spam.csv`
- Columns:
    - `v1`: Label (`ham` or `spam`)
    - `v2`: The actual SMS message text

Data cleaning involves:

- Removing null values
- Renaming labels: `spam → 0`, `ham → 1`

🧹 Data Preprocessing

1. **Text and label extraction**: `X = v2`, `Y = v1`
2. **Label Encoding**: Spam = 0, Ham = 1

SPAM /HAM Internship Project

3. **Splitting the dataset**: 80% training, 20% testing
4. **Vectorization**: TF-IDF (`TfidfVectorizer`) is used with:
    - `min_df=1`
    - `stop_words='english'`
    - `lowercase=True`

This converts text into numerical vectors suitable for modeling.

🧠 Model Training

- **Algorithm used**: Logistic Regression
- Training is done using TF-IDF transformed features.

python
CopyEdit
```
model = LogisticRegression()
model.fit(X_train_features, Y_train)
```

📈 Evaluation

- **Accuracy on Training Set**: Computed using `accuracy_score`
- **Accuracy on Test Set**: Also computed similarly
- **Confusion Matrix**: Visualized using Seaborn heatmap
- **Classification Report**: Includes `precision`, `recall`, and `f1-score`

python
CopyEdit

SPAM /HAM Internship Project

```
print(classification_report(Y_test, prediction_on_test_data,
target_names=['Ham', 'Spam']))
```

🔍 Results

- The model performs well on both training and test data, demonstrating generalization capability.
- Confusion matrix and metrics indicate that the classifier is able to accurately detect spam messages.

✉️ Predictive System

A simple predictive pipeline is included to check custom SMS messages:

python
CopyEdit

```
input_your_mail = ["Your message here"]
features = feature_extraction.transform(input_your_mail)
prediction = model.predict(features)
```

Depending on the output (`1` = Ham, `0` = Spam), the message type is printed.

✅ Conclusion

This machine learning pipeline effectively classifies SMS messages with high accuracy using Logistic Regression and TF-IDF. It can be extended with more advanced models or incorporated into real-world spam detection systems.
