Spam Classification Report

**Author:** Muhammad Ammar Shahid

**Date:** October 2025

# 1. Introduction

Spam messages are one of the most common cybersecurity nuisances, ranging from harmless advertisements to dangerous phishing attempts. This project focuses on building a machine learning-based Spam Detection System to classify SMS messages as 'Spam' or 'Ham' (legitimate).

# 2. Dataset

The dataset used in this project is the MSSpamCollection from the UCI Machine Learning Repository. It contains a collection of labeled SMS messages, tagged as either 'ham' or 'spam'. This dataset serves as the foundation for training and testing the model.

# 3. Data Preprocessing

Preprocessing steps applied on the raw dataset include:  
- Converting text to lowercase  
- Removing punctuation and special characters  
- Tokenization (splitting text into words)  
- Stopword removal (removing common non-informative words)  
- Vectorization using TF-IDF (Term Frequency – Inverse Document Frequency)

# 4. Feature Extraction

Feature extraction was performed using TF-IDF vectorization, which converts text messages into numerical feature vectors based on the frequency of terms relative to their importance across the dataset.

# 5. Model Training and Evaluation

Several machine learning models were tested, including Naive Bayes and Logistic Regression. The models were evaluated using metrics such as accuracy, precision, recall, and F1-score. The chosen classifier achieved \*\*~98-99% accuracy\*\*, demonstrating strong performance in detecting spam messages.

# 6. Model Saving

The trained spam classifier model was saved using the `joblib` library for reusability. This allows future predictions to be made without retraining the model.

# 7. Results

The model performed exceptionally well, with:  
- Accuracy: ~91%  
- High precision: correctly identifying spam with minimal false positives  
- Strong recall: capturing almost all spam messages  
  
These results indicate the classifier is reliable for practical spam detection tasks.

# 8. Future Work

- Deploy model as a Flask/Django web application or REST API  
- Extend dataset with multilingual spam messages  
- Experiment with deep learning models (e.g., LSTMs, BERT) for improved performance

# 9. Conclusion

This project successfully demonstrates the development of a spam classification system using traditional machine learning techniques. With preprocessing, feature extraction, model training, and evaluation, a robust pipeline was created that achieves high accuracy. Future extensions could involve real-time deployment and integration into cybersecurity applications.