



SMS Spam Detection

MACHINE LEARNING PROJECT OVERVIEW



Project Overview

- ▶ **Problem:** SMS spam affects user safety and experience
- ▶ **Goal:** Classify SMS messages as spam or ham using Machine learning models.
- ▶ **Objectives:**
 - ▶ Develop an automated machine learning system to accurately classify SMS as spam or legitimate (ham) to improve security and user experience.
- ▶ **Success Criteria:**
 - ▶ - Accuracy > 95%
 - ▶ - Low false positives
 - ▶ - Model interpretability



Business Understanding

- ▶ Spam messages pose security and operational risks.
- ▶ Automating detection reduces cost and increases trust.

Key Impacts:

- ▶ - User fraud and phishing threats
- ▶ - Inbox clutter and poor UX
- ▶ - Increased filtering costs for telecoms

Data Understanding

- ▶ The data has 2 columns with categorical data.
- ▶ **Label:** spam or ham
- ▶ **Message:** SMS text content
- ▶ - Dataset loaded using pandas for analysis.

Data Preparation.

- ▶ - Remove stop words and punctuation
- ▶ - Lowercase normalization
- ▶ - Label encoding ('spam'=1, 'ham'=0)
- ▶ - Text vectorization (TF-IDF)

Modelling.

Two modeling techniques were used in analysis of the data. This includes:

- ▶ Logistic Regression
- ▶ Naive Bayes

Model Training

- ▶ Split: 80% train, 20% test
- ▶ Performance evaluated using accuracy, precision.

Evaluation.

Evaluation Metrics

- ▶ Accuracy Score
- ▶ Confusion Matrix

Sample Output

- ▶ High accuracy (~98%)
- ▶ Low false positives and negatives

Summary.

- ▶ Applied text preprocessing and multiple models.
- ▶ Evaluation showed reliable performance.
- ▶ Achieved high accuracy on SMS spam classification.

Conclusion and Recommendations.

- ▶ Using ML, we can automate spam detection to:
- ▶ - Protect users from threats
- ▶ - Improve SMS communication
- ▶ - Cut costs and boost efficiency for providers

Recommendations.

- ▶ Use Naive Bayes for deployment due to speed and simplicity
- ▶ - Integrate with SMS platform to filter messages
- ▶ - Monitor model performance regularly

Next steps.

Future Improvements.

- ▶ Explore deep learning (e.g., LSTM).
- ▶ Add language detection and multi-language support.
- ▶ Improve interpretability using SHAP or LIME.



Thank You

Any questions.