

Spam Detection Report

Natural language processing



**Introduction**

The goal of this project is to classify emails as spam or non-spam using natural language processing (NLP) techniques and machine learning models. The process involves data exploration, text preprocessing, feature extraction, and model building to achieve optimal classification performance.

**Data Exploration and Visualization**

**Dataset Information**

The dataset contains email messages labeled as either spam (1) or non-spam (0).

A screenshot of a message

Description automatically generated

**Key Insights:**

1. **Data Summary:**
   * Total rows and columns: 3 columns (CATEGORY, MESSAGE, FILE\_NAME), 5795 rows.
   * Data types and null checks: Verified.

A screenshot of a computer program

Description automatically generated

1. **Category Distribution:**
   * Spam: 32.72%
   * Non-Spam: 67.28%
2. **Visualization:**
   * A bar graph with blue squares

     Description automatically generatedA bar plot showing the distribution of spam and non-spam emails.

* A Word cloud for spam and non-spam emails

A close-up of words

Description automatically generatedA close-up of words

Description automatically generated

* A graph with a line

  Description automatically generatedMost frequent words

**Text Cleaning and Normalization**

**Preprocessing Steps:**

1. **Tokenization:** Split each email into words.
2. **Lowercasing:** Convert all text to lowercase.
3. **Removing Stopwords:** Exclude common words like "the", "and", etc.
4. **Removing Non-Alphabetic Characters:** Eliminate numbers and punctuation.
5. A screen shot of a computer program

   Description automatically generated**Lemmatization:** Reduce words to their base form (e.g., "running" to "run").

**Feature Extraction**

**TF-IDF Vectorization**

The text data was transformed into numerical features using the TF-IDF (Term Frequency-Inverse Document Frequency) vectorizer with a maximum of 1000 features.

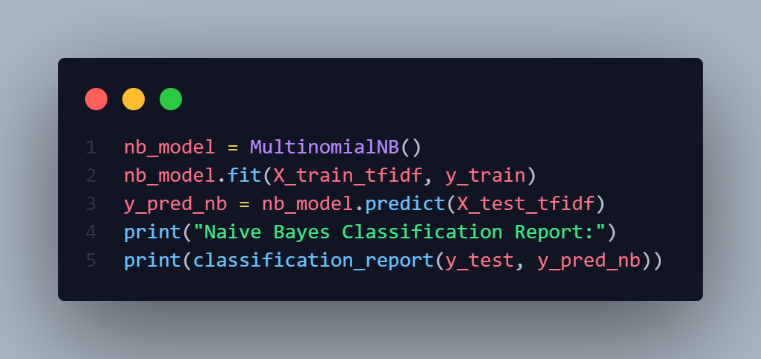
A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated**Model Building**

**Models Tested:**

1. **Logistic Regression:**
   * Accuracy: 99%
2. **Naive Bayes:**
   * Accuracy: 98%
   * Suitable for text data due to simplicity and efficiency.
3. A screen shot of a computer program

   Description automatically generated**Support Vector Machine (SVM):**
   * Accuracy: 99%
   * good for high-dimensional text data.
4. A computer screen shot of a program code

   Description automatically generated**Random Forest:**
   * Accuracy: 98%
   * Ensemble learning.

**Results and Discussion**

**Model Comparison:**

| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
| --- | --- | --- | --- | --- |
| Logistic Regression | 99% | 99% | 98% | 99% |
| Naive Bayes | 98% | 98% | 97% | 97% |
| SVM | 99% | 99% | 99% | 99% |
| Random Forest | 98% | 98% | 98% | 98% |

**Key Observations:**

1. Logistic Regression and SVM performed best in terms of F1-Score.
2. Naive Bayes achieved good accuracy but struggled with imbalanced data.
3. Random Forest provided good results in general but required longer training time.

**Conclusion**

The project successfully classified spam and non-spam emails using multiple NLP and machine learning techniques. Based on the results, SVM emerged as the most suitable model for this task due to its high accuracy and generalization ability.