Email Spam Classifier using Machine Learning

1. Abstract / Introduction

Spam emails are unwanted messages that clutter inboxes and waste user time. An effective spam classifier helps filter out these messages and ensures only relevant emails reach the user. This project aims to build a machine learning model that classifies emails into 'Spam' or 'Not Spam (Ham)' with high accuracy.

2. Problem Statement

To design a machine learning model that classifies incoming emails into spam or ham based on their content, with high accuracy and minimal false positives.

3. Dataset Used

Common datasets include SpamAssassin, UCI SMS Spam Dataset, and the Enron Email Dataset. These datasets typically include email text and labels (spam/ham). Text data is preprocessed by removing punctuation, stopwords, and applying stemming or lemmatization.

4. Text Preprocessing Techniques

- Lowercasing
- Removing punctuation and special characters
- Tokenization
- Removing stopwords
- Stemming or Lemmatization
- Vectorization using Bag of Words (BoW), TF-IDF, or word embeddings

5. Machine Learning Algorithms Used

- Naive Bayes (highly effective for text)

- Logistic Regression

- Support Vector Machine (SVM)

- Decision Tree / Random Forest

- Optional: Deep Learning models like LSTM or BERT

Models are evaluated and compared based on their performance.

6. Evaluation Metrics

To assess the model performance, the following metrics are used:

- Accuracy

- Precision

- Recall

- F1 Score

- Confusion Matrix

High precision reduces false positives, and high recall ensures spam is detected.

7. Result Analysis

Models show high accuracy, with typical performance like:

- Accuracy: 97.5%

- Precision: 95.2%

- Recall: 96.8%

Performance can be visualized using confusion matrices and comparison tables.

8. Deployment (Optional)

If deployed, a web app (e.g., using Flask or Streamlit) allows users to input email content. The model classifies it as Spam or Not Spam and displays the result instantly.

9. Conclusion

Machine learning provides an efficient solution for spam detection. Naive Bayes is especially suitable for text classification. Such models can be adapted and improved for real-time spam filtering.

10. Future Enhancements

- Incorporate deep learning models (e.g., BERT)
- Develop real-time spam detection pipelines
- Support for multi-language email filtering