SPAM MESSAGE DETECTION SYSTEM: MSG HARM OR SPAM DETECT USING MACHINE LEARNING

Sagar fulzele [1], Nayan Janbandhu [2], Prof. Iqra Sabri[3]
[1,2]PG Scholar, [3]Assistant Professor, Department of Computer Application
K.D.K.College of Engineering, Nagpur,Maharashtra, India
fulzelessiddharth.mca24f@kdkce.edu.in, nayanjanbandhu.mca24f@kdkce.edu.in Iqra.sabri@kdkce.edu.in

Abstract
Spam messages have become a major problem in digital communication, causing inconvenience, security risks, and loss of important information. A spam message detection system aims to automatically identify and filter unwanted or malicious messages from legitimate (ham) messages. This project presents a machine learning–based spam message detection system that classifies messages using text processing and supervised learning techniques. The system first preprocesses the message text by removing noise such as stop words, punctuation, and special characters, and then converts the text into numerical features using techniques like Bag of Words or TF-IDF. Various machine learning algorithms such as Naïve Bayes, Support Vector Machine (SVM), and Logistic Regression are trained on labeled datasets to distinguish between spam and non-spam messages. The performance of the system is evaluated using metrics such as accuracy, precision, recall, and F1-score. This system can be used in email services, messaging applications, and social media platforms to improve user experience and enhance communication security.

I. Introduction

With the rapid growth of mobile phones, email services, and online messaging platforms, spam messages have become a serious and widespread problem. Spam messages are unwanted, irrelevant, or fraudulent messages sent in bulk to users. These messages not only waste time but may also contain harmful links, advertisements, or phishing attempts that threaten user privacy and security. Therefore, an efficient spam message detection system is essential to ensure safe and reliable communication.

A spam message detection system is designed to automatically classify incoming messages as spam or legitimate (ham). Traditional rule-based filtering methods are no longer sufficient due to the increasing variety and complexity of spam content. As a result, machine learning techniques are widely used to improve detection accuracy by learning patterns from historical message data.

In this system, text data from messages is first preprocessed to remove unnecessary elements such as stop words, symbols, and punctuation. Feature extraction techniques like Bag of Words or TF-IDF are then applied to convert text into numerical form. Machine learning algorithms such as Naïve Bayes, Support Vector Machine (SVM), and Logistic Regression are trained on labeled datasets to classify messages effectively.

The spam message detection system helps in reducing unwanted messages, protecting users from potential scams, and improving overall communication quality. It is widely applicable in SMS services, email systems, and social media platforms, making it an important application of machine learning in real-world scenarios.

II. Literature Review and Motivation

A. Spam message detection

Spam message detection has attracted significant research attention due to the rapid increase in unwanted messages across SMS, email, and online communication platforms. Early spam detection systems were mainly **rule-based**, relying on predefined keywords and manually created filtering rules. Although easy to implement, these systems lacked adaptability and failed to handle new or evolving spam patterns.

B. Feature extraction techniques

Feature extraction techniques like **Bag of Words (BoW)** and **Term Frequency–Inverse Document Frequency (TF-IDF)** became standard for converting text messages into numerical representations. Research indicates that TF-IDF combined with machine learning classifiers significantly improves spam detection accuracy.

C. The increasing use of mobile

The increasing use of mobile phones, email services, and social media platforms has resulted in a rapid rise in spam messages. These messages cause inconvenience, reduce productivity, and may include malicious content such as phishing links and fraudulent offers. Manual filtering and traditional rule-based systems are no longer sufficient to handle the dynamic and evolving nature of spam messages.

III. Proposed System Architecture and Design

A. System Overview

The proposed Spam Message Detection System is a machine learning–based text classification system designed to automatically identify and filter spam messages from legitimate (ham) messages. The system follows a modular architecture to ensure scalability, accuracy, and ease of integration with existing communication platforms such as SMS services, email systems, and messaging applications.

The system operates by analyzing message content locally using text preprocessing, feature extraction, and supervised machine learning algorithms. It does not require external cloud processing, ensuring data privacy and faster response time.

B. System Modules and Functional Components

1. Message Input Module

This module is responsible for receiving messages from users or datasets. Messages can be entered manually or loaded from a stored dataset for training and testing purposes.

2. Text Preprocessing Module

This module cleans and prepares raw message text for analysis. The preprocessing operations include:

- Conversion of text to lowercase
- Removal of punctuation and special characters
- Removal of stop words
- Tokenization

This step reduces noise and improves classification accuracy.

3. Feature Extraction Module

The feature extraction module converts preprocessed text into numerical form using techniques such as:

- Bag of Words (BoW)
- Term Frequency–Inverse Document Frequency (TF-IDF)

These features represent message content in a format suitable for machine learning models.

4. Machine Learning Classification Module

This module implements the core detection logic using supervised learning algorithms such as:

- Naïve Bayes
- Support Vector Machine (SVM)
- Logistic Regression

The classifier is trained on labeled datasets to distinguish between spam and non-spam messages.

5. Result Output Module

This module displays the final classification result as **Spam** or **Legitimate (Ham)**. It can be integrated with messaging platforms to automatically block or flag spam messages.

C. System Architecture Layers

The proposed system follows a **three-layer architecture**:

- **Data Layer**
  Handles message data collection and storage, including training datasets and input messages.
- **Processing Layer**
  Performs text preprocessing, feature extraction, and machine learning–based classification.
- **Application Layer**
  Displays results to the user and supports integration with SMS, email, or messaging systems for real-time spam filtering.

IV. Methodology and System Development

A. Development Methodology

The Spam Message Detection System was developed using an **iterative and experimental methodology**. Initially, emphasis was placed on understanding message data and applying effective text preprocessing techniques. Subsequent iterations focused on feature extraction, model training, testing, and performance evaluation.

The system follows a **machine learning pipeline approach**, where each stage processes data sequentially to ensure accurate and reliable spam classification.

B. Algorithm Implementation (Hybrid Logic)

To achieve effective spam detection, the system implements the following steps:

- **Data Collection**
  A labeled dataset containing spam and legitimate (ham) messages is used. This dataset serves as input for training and testing the machine learning models.
- **Text Preprocessing**
  Raw message text is cleaned by:
- Removing punctuation and special characters
- Converting text to lowercase
- Removing stop words
- Tokenizing text into meaningful words

C. Data Persistence Strategy

The dataset is divided into:

- **Training set** – Used to train the model
- **Testing set** – Used to evaluate performance

This ensures unbiased performance measurement and prevents overfitting.

V. Experimental Evaluation and Results

A. Evaluation Methodology

The proposed Spam Message Detection System was evaluated using a labeled dataset containing both spam and legitimate (ham) messages. The dataset was divided into training and testing sets to ensure fair evaluation of the machine learning models..

B. Results and Analysis

The experimental results demonstrated that the proposed system effectively classified spam and non-spam messages with high accuracy.

- **Naïve Bayes** showed fast training time and good accuracy, making it suitable for real-time applications.
- **SVM** achieved higher accuracy but required more computational resources.
- **Logistic Regression** provided balanced performance with stable results.

The evaluation metrics indicated:

- High **accuracy** in detecting spam messages
- Improved **precision**, reducing false spam alerts
- Strong **recall**, ensuring most spam messages were detected

VI. Comparative Analysis with Existing Solutions

Table I: Comparative Analysis

Dimension	**Proposed System**	**Standard Antivirus**	**Task Manager**
Detection Approach	Machine Learning	Fixed Rules	User Dependend
Adaptability	High	Low	Very Low
Accuracy	High	Moderate	Low
False Positives	Low	High	Very High
Real-Time Detection	Yes	Limited	No
Scalability	High	Low	Very Low

Positioning:

The proposed system effectively fills the gap between traditional rule-based spam filters and advanced deep learning systems. It offers:

- High accuracy with low computational cost
- Automatic learning of spam patterns
- Easy deployment and integration

This makes it suitable for academic projects as well as real-world applications such as SMS filtering, email spam detection, and messaging platforms.

VII. Technical Stack and Implementation Details

The Spam Message Detection System is implemented using a robust and widely used machine learning and software development stack. The selected technologies ensure efficiency, scalability, and ease of development.

- **Data Loading**
  The dataset is loaded using Pandas and analyzed to understand message distribution.
- **Text Preprocessing**
  Raw message text is cleaned using NLTK to remove noise and irrelevant tokens.
- **Feature Extraction**
  TF-IDF or Bag of Words vectorizers convert text into numerical features.
- **Model Training**
  Machine learning models are trained using Scikit-learn classifiers.
- **Prediction and Output**
  The trained model predicts whether an input message is spam or legitimate

VIII. Limitations and Considerations

A. System Limitations

Despite the effectiveness of the proposed spam message detection system, certain limitations exist:

- **Dependence on Dataset Quality**
  The accuracy of the system highly depends on the quality and size of the training dataset. Poor or imbalanced data may affect performance.
- **Language Dependency**
  The system performs best on messages written in the language used during training. Messages in other languages or mixed-language text may reduce accuracy.

B. Considerations

- **Regular Model Updates**
  The system should be retrained periodically with new data to maintain high accuracy.
- **Computational Resources**
  Although lightweight, real-time detection still requires minimal processing power.
- **False Positives** Some legitimate messages may be incorrectly classified as spam, especially if they contain promotional keywords

IX. Future Enhancements and Extensions

A. Advanced Machine Learning Models Future versions of the system can integrate **deep learning techniques** such as:

- Recurrent Neural Networks (RNN)
- Long Short-Term Memory (LSTM)
- Transformer-based models

These models can better understand context and sequential patterns in messages, improving detection accuracy.

B. Multilingual Spam Detection The current system can be enhanced to support **multiple languages** by training models on multilingual datasets. This will make the system suitable for global communication platforms.

C. Real-Time Deployment The system can be deployed as:

- A mobile application
- A browser extension
- An email server plugin

This will enable **real-time spam filtering** in live communication environments.

D. User Feedback Mechanism A feedback system can be added where users can mark messages as spam or non-spam. This feedback can be used to **retrain and improve** the model continuously.

.E. Cloud and API Integration Future enhancements may include exposing the spam detection model through **REST APIs** for easy integration with third-party applications and cloud-based services.

X. Conclusion

This project presented a **Spam Message Detection System** based on machine learning techniques to effectively classify messages as spam or legitimate (ham). With the rapid growth of digital communication platforms, spam messages have become a major concern due to their negative impact on user experience and potential security risks.

The proposed system utilizes text preprocessing, feature extraction techniques such as Bag of Words and TF-IDF, and supervised machine learning algorithms including Naïve Bayes, Support Vector Machine, and

Logistic Regression. Experimental evaluation demonstrated that the system achieves high accuracy, reduces false positives, and adapts well to different spam patterns.

References

1. T. Almeida, J. Hidalgo, and A. Yamakami, "Contributions to the Study of SMS Spam Filtering: New Collection and Results," *Proceedings of the 11th ACM Symposium on Document Engineering*, 2011.

2. G. V. Cormack, "Email Spam Filtering: A Systematic Review," *Foundations and Trends in Information Retrieval*, vol. 1, no. 4, pp. 335–455, 2007.
3. T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," *European Conference on Machine Learning (ECML)*, 1998.
4. A. McCallum and K. Nigam, "A Comparison of Event Models for Naïve Bayes Text Classification," *AAAI Workshop on Learning for Text Categorization*, 1998.
5. F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
6. S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, O'Reilly Media, 2009.
7. J. Ramos, "Using TF-IDF to Determine Word Relevance in Document Queries," *Proceedings of the First Instructional Conference on Machine Learning*, 2003.
8. C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
9. UCI Machine Learning Repository, "SMS Spam Collection Dataset," [Online]. Available: https://archive.ics.uci.edu/ml