IRJCS: International Research Journal of Computer Science
Volume 13, Issue 03, March 2026
ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.27

Spam Detection System Using Machine Learning

S.Malathi
Assistant Professor, Department of CSE (Cyber Security)
Sengunthar Engineering College (Autonomous), Tiruchengode,India
malathicse@gmail.com
https://orcid.org/0009-0007-6183-2264
Chandan Kumar,Deepak kumar,Gaurav singh,Harsh Raj
Department of CSE (Cyber security)
Sengunthar Engineering College (Autonomous), Tiruchengode,India
ck009184@gmail.com, deepak6355kr@gmail.com, gk0669720@gmail.com
harshraj7361078011@gmail.com

Publication History
Manuscript Reference: IRJCS/RS/Vol.13/Issue03/CSMR26.MRCS10106
Research Article | Open Access | Double-Blind Peer Reviewed Article ID: IRJCS/RS/Vol.13/Issue03/CSMR26.MRCS10106
Received: 30, January 2026, Revised: 13, February 2026, Accepted: 28 February 2026 Published Online: 25 March 2026
https://www.irjcs.com/volumes/Vol13/iss-03/27.CSMR26.MRCS10106.pdf
Article Citation:Malathi,Chandan,Deepak,Gaurav,Harsh(2026),Spam Detection System Using Machine Learning, IRJCS: International Research Journal of Computer Science, Volume 13,Issue 03 of 2026 pages 254-258
Doi:> https://doi.org/10.26562/irjcs.2026.v1303.27 **BibTeX Key Malathi@2026Spam**
Orcid: https://orcid.org/0009-0004-9398-7488
IRJCS papers should be cited as IRJCS (International Research Journal of Computer Science, AM Publications, India 2026, ISSN 2393-9842, https://doi.org/10.26562/ijrcs.2025.v1303.27 The journal's official abbreviation is IRJCS.

Abstract: Spam detection is an important application of machine learning that helps identify unwanted messages such as advertisements, phishing attempts, and malicious links. With the rapid growth of digital communication through email, SMS, and social media platforms, the amount of spam messages has increased significantly. These messages not only cause inconvenience to users but also pose serious security threats such as identity theft, financial fraud, and malware attacks. This project presents a spam detection system that automatically classifies messages as spam or non- spam (ham) using machine learning techniques. The system performs several steps including data preprocessing, feature extraction, model training, and classification. Text preprocessing removes unnecessary characters, stop words, and punctuation to improve the quality of the dataset. Feature extraction techniques such as TF-IDF convert textual data into numerical vectors suitable for machine learning algorithms. Machine learning models such as Naive Bayes and Support Vector Machine are used to train the classification system. These algorithms analyze patterns in the dataset and learn to distinguish between spam and legitimate messages. The trained model is then tested using performance metrics such as accuracy, precision, recall, and F1-score. Experimental results show that the proposed system achieves high accuracy in detecting spam messages. The developed spam detection system can be integrated into email services and messaging applications to automatically filter unwanted messages and improve communication security.
Keywords: Spam Detection, Machine Learning, Text Classification, NLP, Email Filtering

I. INTRODUCTION

With the rapid development of the internet and digital communication systems, email and messaging services have become an essential part of daily life. Millions of messages are exchanged every day across various platforms such as email, SMS, and social networking applications. However, along with the growth of digital communication, the problem of spam messages has also increased. Spam messages are unsolicited messages sentin bulk to a large number of users. These messages often contain advertisements, fraudulent offers, phishing links, or malicious software. Spam messages not only waste users' time but also create serious security risks such as identity theft and financial fraud. Traditional spam filtering methods relied on rule-based systems that used predefined keywords to detect spam messages. However, these methods are not effective in detecting modern spam techniques because spammers continuously change their message patterns. Machine learning techniques provide an efficient solution to this problem. By analyzing patterns in large datasets, machine learning algorithms can learn the characteristics of spam messages and automatically classify new messages. This project focuses on developing a spam detection system using machine learning to improve the accuracy and efficiency of spam filtering.

LITERATURE REVIEW

Spam detection has become an important area of research due to the rapid increase of unwanted and malicious messages across communication platforms such as email, SMS, and social media. Many researchers have proposed different techniques using machine learning, natural language processing, and statistical methods to automatically classify messages as spam or non-spam. One of the earliest approaches to spam detection was based on the Naïve Bayes classifier, which is a probabilistic machine learning algorithm widely used for text classification. Researchers found that Naïve Bayes performs effectively for filtering spam emails by analyzing the probability of words appearing in spam messages compared to legitimate messages.

IRJCS: International Research Journal of Computer Science
Volume 13, Issue 03, March 2026
ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.27

This method is simple, efficient, and provides good accuracy for large datasets. Another commonly used technique is the Support Vector Machine (SVM) algorithm. Studies have shown that SVM provides high accuracy in spam classification by identifying optimal boundaries between spam and non- spam messages. SVM is particularly effective when dealing with high-dimensional data such as text features extracted from messages. In recent years, researchers have also explored Natural Language Processing (NLP) techniques to improve spam detection systems. NLP methods help in analyzing the structure and meaning of text messages. Techniques such as tokenization, stop-word removal, and term frequency–inverse document frequency (TF-IDF) are widely used to extract meaningful features from text before applying machine learning algorithms.

PROPOSED METHODOLOGY ARCHITECTURE

The proposed spam detection system is designed to automatically analyze incoming text messages and classify them as spam or not spam (ham). The system uses machine learning techniques and text preprocessing methods to identify patterns commonly found in spam messages. The architecture of the proposed system consists of several stages including data collection, data preprocessing, feature extraction, model training, and message classification.

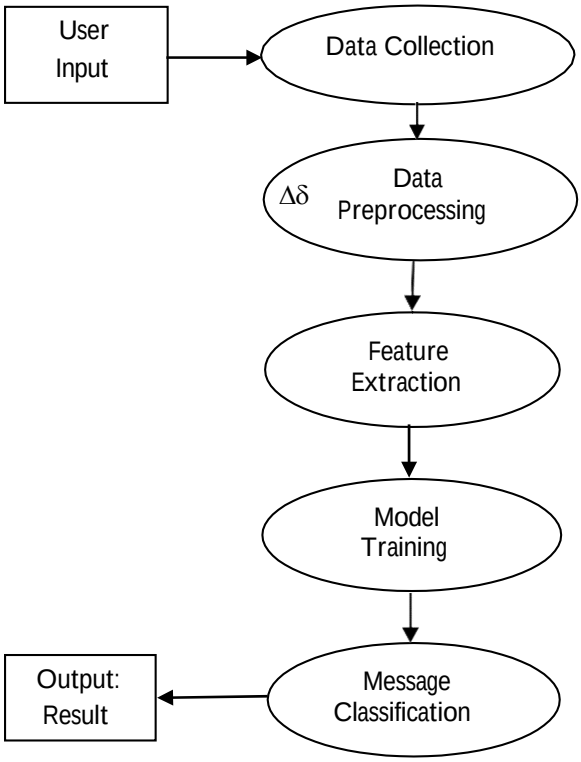

Fig.1.Architecture Diagram

A. Data Collection

Data collection is the first step in building the spam detection system. In this stage, a dataset containing a large number of messages is gathered from publicly available sources. The data set typically consists of two categories of messages: spam messages and legitimate (ham) messages. These messages are labeled so that the machine learning algorithm can learn the difference between spam and non- spam content. The dataset must contain a sufficient number of examples from both categories to train the model effectively. A balanced dataset helps improve the accuracy of the classification model and prevents bias toward one class. Popular datasets used for spam detection include SMS spam datasets and email spam datasets collected from real communication systems.

B. Data Preprocessing

Raw text data often contains noise, unnecessary characters, and inconsistent formatting, which can reduce the performance of the machine learning model. Therefore, data preprocessing is an important step in preparing the data for analysis. During preprocessing, several techniques are applied to clean the text data. First, all characters are converted to lowercase to ensure uniformity. Special symbols, punctuation marks, and unnecessary characters are removed. Stop words such as "the,""is,""and," and "to" are eliminated because they do not contribute significant meaning to the message classification process. Tokenization is also performed, which involves breaking down the message into smaller units such as individual words or tokens. In some cases, stemming or lemmatization is applied to reduce words to their root forms. These preprocessing steps help simplify the text and improve the efficiency of feature extraction and model training.

C. Feature Extraction

Feature extraction is the process of converting textual data into numerical values that can be processed by machine learning algorithms. Since machine learning models cannot directly interpret raw text, it is necessary to represent text data in a mathematical format. One commonly used technique for feature extraction is Term Frequency–Inverse Document Frequency (TF- IDF).

ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.27

This method measures the importance of each word in a message relative to the entire dataset. Words that appear frequently in a particular message but rarely in other messages are considered more important features. Another simple technique used in spam detection is Bag- of-Words (BoW), which represents each message based on the frequency of words appearing in it. These feature extraction techniques transform each message into a vector representation that can be used by machine learning algorithms for classification.

D. Model Training

After the features are extracted from the dataset, the next step is to train a machine learning model. The model learns patterns and relationships between the features and their corresponding labels (spam or non-spam). During this stage, the dataset is usually divided into training and testing sets. Several machine learning algorithms can be used for spam detection, including Naïve Bayes, Support Vector Machine (SVM), Logistic Regression, and Decision Trees. Among these, Naïve Bayes is widely used due to its simplicity and effectiveness in text classification tasks. The training process involves feeding the feature vectors into the algorithm along with their labels so that the model can learn how to distinguish spam messages from legitimate ones. The trained model is then evaluated using the testing dataset to measure its performance.

E. Message Classification

Once the model has been trained successfully, it can be used to classify new incoming messages. When a user inputs a message into the system, the message first undergoes the same preprocessing and feature extraction steps used during training. The processed message is then passed to the trained machine learning model. Based on the learned patterns and probability calculations, the model predicts whether the message belongs to the spam category or the non- spam category. This automatic classification process helps users quickly identify suspicious messages without manually analyzing each message.

TECHNOLOGIES USED

The spam detection system is developed using various technologies and tools that support data processing, machine learning, and system implementation. These technologies help in analyzing text messages, training classification models, and predicting whether a message is spam or not. The major technologies used in the development of the proposed system are described below.

A. Python

Python is the primary programming language used for developing the spam detection system. It is widely used in machine learning and data science applications due to its simplicity and readability. Python provides a large number of libraries that support data analysis, natural language processing, and machine learning model development. The flexibility of Python allows developers to easily implement algorithms, process large datasets, and perform model training efficiently. Its strong community support and open-source ecosystem make it a popular choice for developing intelligent systems like spam detection.

B. Machine Learning Algorithms

Machine learning algorithms play a significant role in identifying spam messages. These algorithms learn patterns from labeled datasets and use those patterns to classify new messages. In the spam detection system, classification algorithms such as Naïve Bayes, Logistic Regression, and Support Vector Machine (SVM) are commonly used. These algorithms analyze word frequency and message patterns to determine whether a message is spam or legitimate. The trained model can then automatically classify new incoming messages with high accuracy.

C. Natural Language Processing (NLP)

Natural Language Processing (NLP) is used to process and analyze textual data in the spam detection system. NLP techniques help the system understand the structure of human language and extract meaningful information from messages. Common NLP techniques include tokenization, stop-word removal, text normalization, and stemming. These methods help remove unnecessary words and reduce noise in the dataset, allowing the machine learning model to focus on the most important features of the text.

D. Scikit-learn Library

Scikit-learn is one of the most widely used machine learning libraries in Python. It provides a variety of tools for data preprocessing, feature extraction, model training, and performance evaluation. In the spam detection system, Scikit-learn is used to implement classification algorithms such as Naïve Bayes and Logistic Regression. The library also provides functions for splitting datasets into training and testing sets, performing cross-validation, and calculating performance metrics like accuracy, precision, and recall.

E. Data Preprocessing Tools

Data preprocessing tools and libraries are used to clean and prepare the text data before training the model. Libraries such as Pandas and NumPy help handle and manipulate large datasets efficiently. Pandas is particularly useful for organizing data into structured formats such as tables or data frames, which makes data analysis and preprocessing easier.

IMPLEMENTATIONS AND RESULTS

A. Implementation

The spam detection system was implemented using a machine learning approach to classify messages as spam or non-spam. The implementation process involved several stages, including data collection, preprocessing, feature extraction, model training, and message classification. The system was developed using the Python programming language along with various machine learning libraries. Initially, a dataset containing labeled messages was collected. The dataset included both spam and legitimate messages, which were used to train the classification model. Before training the model, the dataset was preprocessed to remove unnecessary characters, punctuation marks, and stop words.

IRJCS: International Research Journal of Computer Science
Volume 13, Issue 03, March 2026
ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.27

All text messages were converted into lowercase format to maintain uniformity in the dataset. After preprocessing, feature extraction techniques were applied to convert text data into numerical representations. The TF-IDF (Term Frequency–Inverse Document Frequency) method was used to identify the importance of words in each message. This process transformed the text messages into feature vectors that could be processed by machine learning algorithms. Once the features were extracted, a machine learning algorithm was trained using the prepared dataset. The dataset was divided into training and testing sets to evaluate the performance of the model. The classification algorithm learned patterns from the training data and used those patterns to classify new messages.

B. Results

After implementing the spam detection system, the model was tested using the testing dataset to evaluate its performance. The trained model was able to classify messages as spam or non-spam with a high level of accuracy. Several evaluation metrics were used to measure the performance of the model, including accuracy, precision, recall, and F1-score. Accuracy represents the percentage of messages correctly classified by the model. Precision measures how many messages identified as spam are actually spam, while recall indicates how many actual spam messages were successfully detected by the system. The output of the system displays whether the given message is Spam or Not Spam, allowing users to quickly identify suspicious messages. The system can be further improved by training the model with larger datasets and using advanced machine learning techniques.

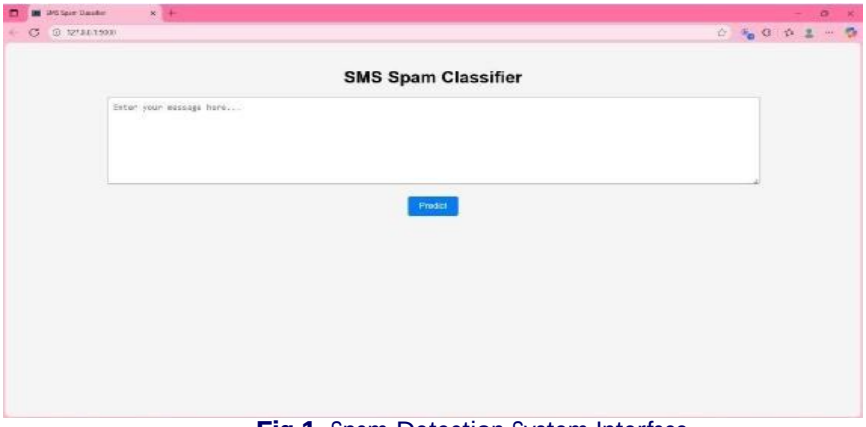

Fig 1. Spam Detection System Interface

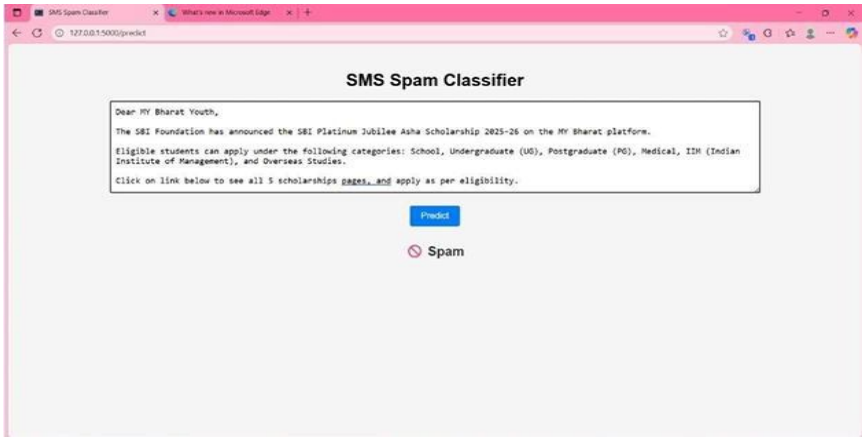

Fig.2. Spam Message Interface

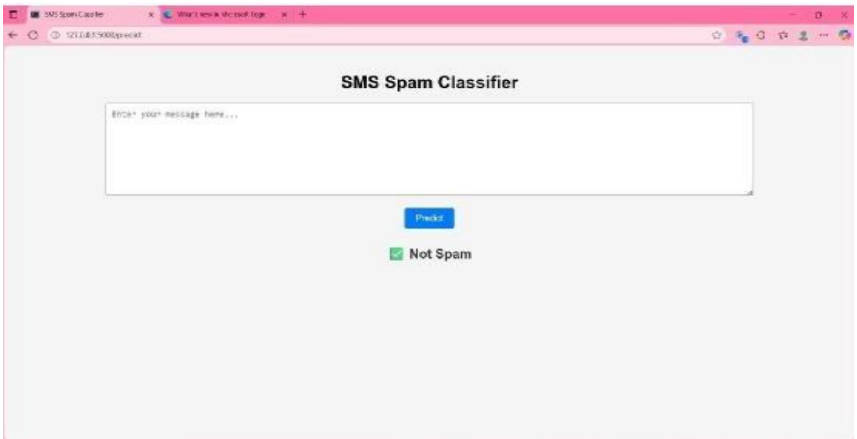

Fig.3. Not Spam Message Interface

ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.27

CONCLUSION

In this project, a spam detection system was developed to automatically identify whether a message is spam or a legitimate message. The system uses machine learning techniques along with text preprocessing and feature extraction methods to analyze the content of messages. By training the model with labeled datasets, the system learns patterns that commonly appear in spam messages and uses this knowledge to classify new incoming messages. The implementation of the system involved several stages, including data collection, data preprocessing, feature extraction using TF-IDF, model training, and message classification. Machine learning algorithms were applied to analyze the message data and predict whether a message belongs to the spam category or the non-spam category. The system was tested using different messages, and the results showed that the model was able to detect spam messages with good accuracy. The spam detection system helps users identify unwanted or suspicious messages quickly, reducing the risk of fraud and improving communication security. By automatically filtering spam messages, the system saves time and prevents users from interacting with harmful or misleading content. Although the proposed system provides effective spam detection, there is still scope for improvement. The system can be enhanced by using larger datasets and advanced machine learning or deep learning techniques to improve classification accuracy. Future improvements may also include integrating the system with email platforms or messaging applications for real-time spam filtering.

REFERENCES

1. S.J.Delany, M.Buckley, and D.Greene, “SMS spam filtering: Methods and data,” Expert Systems with Applications, vol. 39, no. 10, pp. 9899–9908, Aug. 2012.
2. G.V.Cormack, “Email spam filtering: A systematic review,” Foundations and Trends in Information Retrieval, vol. 1, no. 4, pp. 335–455, 2008.
3. T.Joachims,“Text categorization with support vector machines: Learning with many relevant features,” in Proceedings of the European Conference on Machine Learning, Berlin, Germany, 1998, pp. 137–142.
4. J.Rennie, L.Shih, J.Teevan, and D.Karger, “Tackling the poor assumptions of Naïve Bayes text classifiers,” in Proceedings of the 20th International Conference on Machine Learning, Washingt on,DC,USA, 2003, pp. 616–623.
5. F.Sebastiani, “Machine learning in automated text categorization,”ACM Computing Surveys,vol.34,no.1,pp.1–47,Mar.2002.
6. K.Sakkis, I.Androutsopoulos, G.Paliouras, V. Karkaletsis, C. D. Spyropoulos, and P. Stamatopoulos, “Stacking classifiers for anti-spam filtering of email,” Data Mining and Knowledge Discovery, vol. 9, no. 1, pp. 51–90, 2004.
7. A.Almeida, J.Hidalgo, and T.Yamakami, “Contributions to the study of SMS spam filtering: New collection and results,” in Proceedings of the ACM Symposium on Document Engineering, Paris, France, 2011, pp. 259–262.
8. C.D.Manning, P.Raghavan, and H.Schütze, Introduction to Information Retrieval. Cambridge, U.K.: Cambridge University Press, 2008.
9. S.Raschka and V.Mirjalili, Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow. Birmingham,U.K.: Packt Publishing, 2017.
10. J.Han, M.Kamber, and J.Pei, Data Mining: Concepts and Techniques, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann, 2012.