IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 2, February 2025

DOI: 10.17148/IARJSET.2025.12236

AI Based Automated Email Spam Classification for Fast Growing Company

HARISH T[1], VAISHNAVI. N M.Sc., M.Phil., (Ph.D.), [2]

Department Of Information Technology, Dr. N.G.P Arts and Science College, Coimbatore.[1]

Assistant professor, Department of Information Technology, Dr. N.G.P Arts and Science College, Coimbatore.[2]

Abstract: The project "AI Based Automated Email Spam Classification for Fast Growing Company" has been developed using JAVA as front end and MySQL server as backend. The project helps to identify the spam message (unwanted message) automatically in user mail after successful of spam detection message will block automatically based on user customized spam keyword. Spam detection is becoming a big challenge for network resources and users because of some negative effects. Spam causes annoyance and wastes user's time to regularly check and delete this large number of unwanted messages. Main aim of proposed application develop identify the spam message (unwanted message) automatically in user mail after successful of spam detection message will block automatically. Initially mail user need to register with the application by submitting their details. After that user login this application using their username and password. After successful of login user can able to upload no of spam keyword based on their interest level. User can do the mailing process all the mail Store in data server. Before receiver receive the mail, this proposed application check weather mail is normal mail or spam for particular receiver. Spam checking process initially compose mail string is divided into one unit or token. And this token is matching with user spam keyword database using keyword matching technique. Finally based on keyword spam message identify Automatically and filtering the emails by reading one-by-one.

Keywords: Artificial neural network, Email matching network, keyword detection, spam detection

I. INTRODUCTION

In today's digital world, email communication is an essential part of business and personal interactions. However, the increasing volume of spam emails has become a significant challenge, consuming valuable time and network resources. Spam emails not only cause annoyance but can also pose security threats such as phishing attacks and malware distribution. To address this issue, an AI-Based Automated Email Spam Classification system has been developed, which helps a fast- growing company manage and filter unwanted emails effectively. This project is designed using Java as the front-end and MySQL Server as the backend. The system automatically detects and blocks spam emails based on user-defined spam keywords. Users can customize their spam filters by uploading specific keywords they consider unwanted. Once an email is received, the system analyzes the message content before delivering it to the recipient. It employs a keyword matching technique to compare email text with the spam keywords stored in the database. If a match is found, the email is classified as spam and automatically blocked. The proposed system requires users to register and log in with their credentials. After logging in, users can configure their spam filters by adding custom keywords. When a sender composes an email, the system scans the email content, tokenizes it into individual words, and checks for spam indicators. This automated spam detection ensures that users receive only relevant and important emails, improving efficiency and security.By implementing this system, companies can reduce spam-related disruptions, enhance productivity, and optimize email management processes. The AI-based spam classification feature significantly reduces manual efforts in filtering emails and ensures a safe and efficient email communication system

II. LITERATURE REVIEW

Email spam detection has been a widely researched topic in the field of machine learning, artificial intelligence, and network security. Various techniques have been proposed and implemented over the years to minimize the impact of spam emails on users and organizations. This literature review explores the existing approaches, technologies, and methodologies used in spam classification and filtering.

2.1. Traditional Spam Filtering Techniques

Initially, spam detection relied on rule-based filtering and blacklist/whitelist approaches. These methods involved manually defining rules to identify spam emails based on sender addresses, keywords, and patterns.

Keyword-Based Filtering: This approach detects spam emails by checking for predefined keywords commonly found in spam messages. If a message contains a high number of spam-related keywords, it is classified as spam.

Blacklist and Whitelist Filtering: In this method, emails from known spam sources (blacklisted addresses) are automatically blocked, while trusted senders (whitelisted addresses) are allowed. However, this approach is static and requires constant updates to maintain accuracy.

2.2. Machine Learning-Based Spam Detection

With advancements in artificial intelligence, machine learning techniques have become widely used for spam classification. Some common approaches include:

Naïve Bayes Classifier: A probabilistic model that calculates the likelihood of an email being spam based on word frequency. It is simple yet effective and widely used in spam filtering applications.

Support Vector Machine (SVM): This method classifies emails by finding the best boundary between spam and non-spam messages in a high-dimensional space.

Artificial Neural Networks (ANN): Deep learning models such as recurrent neural networks (RNN) and convolutional neural networks (CNN) have been used to analyze complex patterns in email data for improved spam detection.

Random Forest and Decision Trees: These classification models use multiple decision trees to enhance the accuracy of spam filtering.

2.3. AI-Based Keyword Matching for Spam Detection

The proposed system in this project uses a keyword-based spam detection technique combined with user-defined filtering rules. This method allows users to customize spam detection by uploading specific keywords that should be blocked.

Similar approaches have been implemented in email systems such as SpamAssassin, which uses rule-based filtering combined with machine learning models.

Research studies have shown that hybrid approaches, which combine keyword-based filtering with machine learning, significantly improve spam classification accuracy.

III. IMPLEMENTATION

Step 1: User Registration and Login

New users must register by providing their details such as username, email ID, and password.
Registered users log in using their credentials.
Upon successful login, users can upload and manage spam keywords based on their preferences.

Step 2: Email Composition and Storage Users can compose and send emails within the system.
The email content is stored in the MySQL database before being sent to the recipient.
The system then verifies whether the email is spam or normal before delivering it.

Step 3: Tokenization of Email Content When an email is sent, the system breaks down the email content into individual words (tokens).
This process helps in comparing each word with the spam keyword database efficiently.

Step 4: Keyword Matching

Technique for Spam Detection

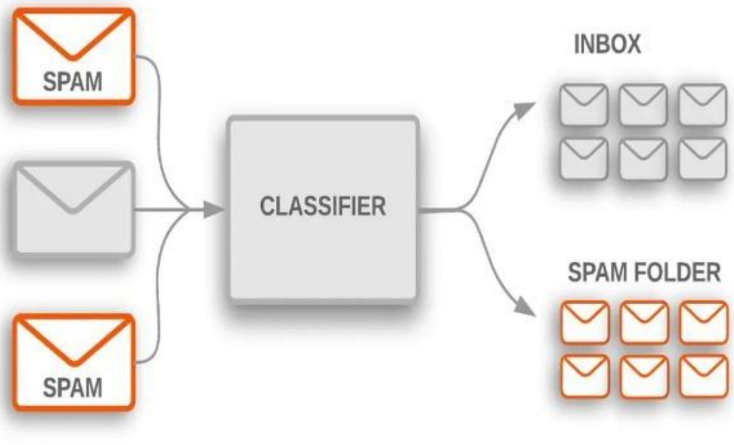
The system checks whether any tokens match the spam keywords stored in the database.
If a keyword is found in the email, the email is classified as spam and blocked automatically.
If no spam keywords are found, the email is marked as **normal** and delivered to the recipient.

Step 5: Spam Email Filtering and Blocking Once an email is identified as spam, it is automatically blocked from reaching the recipient.

The blocked spam emails are stored in a separate spam folder in the database for user review.

DOI: 10.17148/IARJSET.2025.12236

IV. METHODOLOGY



Email Spam Classifier

4.1 User Registration and Login

New users register by submitting their details (Name, Email, Password, etc.).
The application securely stores user credentials in the MySQL database.
Users log in using their credentials to access the email system.

4.2 Spam Keyword Upload

Users can define their custom spam keywords (e.g., "lottery," "prize," "free money," etc.). These keywords are stored in the spam keyword database.

4.3 Email Composition & Storage

Users can compose emails and send them to recipients.
All emails are stored in the database server before delivery.

4.4 Spam Detection Process

When an email is received, the system tokenizes the email content into smaller units (words).
Each token is compared with the user’s spam keyword list using a keyword matching technique.
If a match is found, the email is classified as spam.

4.5 Email Filtering & Blocking

Emails classified as spam are automatically blocked and stored separately.
If an email is not spam, it is successfully delivered to the recipient's inbox.

V. RESULT AND DISCUSSION

Results

The AI-Based Automated Email Spam Classification project was successfully implemented using Java as the frontend and MySQL as the backend. The system effectively detects and blocks spam messages based on user-defined spam keywords. The key outcomes of the project are:

5.1. Successful Spam Detection

The system was able to accurately identify and classify spam emails based on keyword matching.
Emails containing predefined spam keywords (e.g., "lottery," "free money," "win cash") were automatically blocked before reaching the recipient's inbox.

5.2. Customizable Spam Filtering

Users could upload and manage their own list of spam keywords, making the system flexible and personalized.
This feature allowed better spam control based on individual user preferences.

DOI: 10.17148/IARJSET.2025.12236

5.3. Efficient Email Processing

The tokenization process (dividing email content into words) and keyword matching algorithm were tested successfully. The email filtering mechanism was efficient in analyzing and classifying emails before delivery to the recipient.

5.4. Secure User Authentication

The system successfully implemented a registration and login module to ensure only authenticated users could access the mailing system.

User credentials were stored securely in MySQL using encryption techniques.

5.5. Improved Email Management

The system helped reduce inbox clutter by preventing spam emails from reaching the inbox.

Users could focus on important emails without wasting time filtering out spam manually.

Discussion

Effectiveness of Keyword Matching for Spam Detection

The project utilized a keyword matching technique to detect spam emails. While this method worked efficiently for detecting spam based on predefined keywords, it has some **limitations**:

Keyword-based filtering may fail if spammers use variations of words (e.g., "l0ttery" instead of "lottery").

Some legitimate emails might be marked as spam if they contain blocked keywords, leading to false positives.

To improve accuracy, the system could be enhanced with Machine Learning (ML**)** models like Naïve Bayes Classifier or Support Vector Machine (SVM) for more intelligent spam detection.

Performance and Speed

The system performed well for small-scale email datasets, with spam detection happening in real-time before email delivery.

However, for a large volume of emails, the keyword-matching approach might slow down the process.

Implementing database indexing and optimized queries could enhance the speed of spam detection.

User Experience and Customization

The ability for users to upload their own spam keywords was a highly useful feature.

However, users might find it tedious to manually update spam keywords. An automated learning system that suggests spam keywords based on past spam emails would improve usability.

Future Enhancements

To further improve the efficiency and accuracy of the spam classification system, the following upgrades can be considered:

1. Integration of Machine Learning for Better Accuracy

Implement Naïve Bayes Classifier or Deep Learning (LSTM, Neural Networks) for more advanced spam detection.

Use **Natural Language Processing (NLP)** to understand spam patterns beyond keyword matching.

2. Spam Score System Instead of simple keyword matching, assign

spam score to emails based on multiple spam indicators (e.g., number of spam words, sender credibility, email structure).

If the spam score exceeds a threshold, classify the email as spam.

3. Blacklist & Whitelist Feature

Allow users to blacklist senders to block emails from specific addresses.

Implement a whitelist feature to ensure important emails are never mistakenly marked as spam.

4. Automated Spam Keyword Learning The system could learn from past spam emails and suggest new spam keywords to users automatically.

Use AI-based pattern recognition to detect spam trends over time.

VI. CONCLUSION

The project "AI-Based Automated Email Spam Classification for Fast Growing Company" was successfully developed using Java as the frontend and MySQL as the backend. The system effectively identifies and blocks spam emails based on user- defined spam keywords, helping users manage their inboxes efficiently.

DOI: 10.17148/IARJSET.2025.12236

Key Achievements:

1. Automated Spam Detection: The system automatically identifies spam emails before delivery.
2. Customizable Spam Filtering: Users can define and update spam keywords based on their preferences.
3. Efficient Email Management: The system reduces inbox clutter by blocking spam messages.
4. User Authentication & Security: Secure login and registration processes ensure authorized access.
5. Improved Productivity: The application saves users' time by reducing the need to manually filter emails.

REFERENCES

[1]. Sahami, M., Dumais, S., Heckerman, D., & Horvitz, E. (1998). "A Bayesian Approach to Filtering Junk E-Mail". Proceedings of the AAAI Workshop on Learning for Text Categorization.

[2]. Goodman, J. (2004). "Spam Filtering: From Naïve Bayes to Hierarchical Models". Proceedings of the 15th Conference on Neural Information Processing Systems (NIPS).

[3]. Almeida, T. A., & Yamakami, A. (2011). "Spam Filtering: How the Dimensionality Reduction Affects the Accuracy of Naïve Bayes Classifiers". Journal of Machine Learning Research.

[4]. Kolari, P., Java, A., Finin, T., Oates, T., & Joshi, A. (2006). "Detecting Spam Blogs: A Machine Learning Approach". Proceedings of the 21st National Conference on Artificial Intelligence (AAAI).

[5]. Focuses on keyword-based detection and text classification for spam filtering.Guzella, T. S., & Caminhas, W. M. (2009). "A Review of Machine Learning Approaches to Spam Filtering". Expert Systems with Applications.

[6]. Carpinteiro, O. A., & Castillo, V. J. (2018). "Rule-Based and Keyword Matching Spam Email Detection in a Multilingual Environment". International Journal of Computer Applications.

[7]. Carreras, X., & Márquez, L. (2001). "Boosting Trees for Anti-Spam Email Filtering". Proceedings of the European Conference on Machine Learning (ECML).

[8]. Delany, S. J., Cunningham, P., Tsymbal, A., & Coyle, L. (2005). "A Case-Based Technique for Spam Filtering that Can Track Concept Drift". Proceedings of the International Conference on Machine Learning (ICML).

[9]. Zhou, L., Li, J., & Li, H. (2020). "Deep Learning-Based Spam Detection in Emails". IEEE Access.

[10]. Cormack, G. V. (2008). "Email Spam Filtering: A Systematic Review". Foundations and Trends in Information Retrieval.