A Multi-Channel Spam Detection System Utilizing Natural Language Processing and Machine Learning

Mohini Tyagi*[,1], Pradeep Kumar Singh[1], Shivam Kumar Yadav[2], Sanjay Kumar Soni[2]

[1]Dept. of Computer Science and Engineering, Madan Mohan Malaviya University of Technology, Gorakhpur, India - 273010

[2]Dept. of Electronics and Communication Engineering, Madan Mohan Malaviya University of Technology, Gorakhpur, India - 273010

As digital communication rapidly expands, the issue of unsolicited and unwanted messages, commonly known as spam, has become a major concern. In this work, a sophisticated spam detection system that uses an ensemble combination of Machine Learning (ML) and Natural Language Processing (NLP) approaches is presented. The system differentiates between spam and legitimate messages by employing a hybrid model that combines Naive Bayes, Support Vector Machines (SVM), and deep learning models like Bidirectional Encoder Representations from Transformers (BERT). The model demonstrates high effectiveness across various communication platforms, including emails, SMS, and social media, achieving an accuracy exceeding 98.5%.

Received on 29 December 2024; accepted on 11 March 2025; published on 18 March 2025

Keywords: Bidirectional Encoder Representations from Transformers (BERT), Machine Learning (ML), Natural Language Processing (NLP), Spam/Ham, Support Vector Machine (SVM)

doi:10.4108/airo.8309

1. Introduction

In the rapidly evolving technological environment and the widespread expansion of digital communication networks, the emergence of unwelcome and undesirable digital messages has garnered significant recognition. Referred to as "SPAM," these messages undergo systematic categorization based on their intended purposes. Their classification is contingent upon the communication medium employed to disseminate them to recipients. This paper delves into the domain of spam detection across three pivotal communication platforms: spam emails, spam SMS messages, and spam comments within the context of YouTube. The proposed framework harnesses the effectiveness of Bayes' theorem and the Naive Bayes classifier to differentiate messages as either spam or legitimate. Furthermore, the frequent identification of the sender's IP address enhances the evaluative process. Throughout history, spam has encompassed all forms of undesired and unsolicited digital communication, often materializing as extensively distributed emails. This ubiquitous occurrence consumes a substantial amount of time and resources. In the present landscape, spam has transformed into a prominent conduit for phishing attacks. Within this intricate scenario, machine learning emerges as a significant facilitator, enabling the construction of a model adept at determining whether a given text adheres to spam criteria. Utilizing the capabilities of Python, the Naive Bayes classifier's accuracy has notably increased, exceeding the threshold of 98.2%. The integration of BERT with SVM and Naive Bayes for spam classification presents a unique approach compared to other hybrid methods. This combination leverages BERT's advanced contextual understanding of language, enhancing feature extraction, while SVM and Naive Bayes contribute their strengths in classification. One key aspect of this hybridization is its ability to capture nuanced patterns in email content that traditional methods may overlook [1]. BERT excels in understanding the context of words in a sentence, which is crucial for distinguishing between spam and legitimate emails, especially when

*Corresponding author. Email: tyagimohini7@gmail.com they are similar in structure [2]. Furthermore, by utilizing BERT for feature extraction, the hybrid model can effectively identify complex language patterns. The integration of SVM and Naive Bayes enhances classification capabilities, resulting in improved accuracy and reduced false positives. The effective hyperplane separation provided by SVM and the probabilistic framework offered by Naive Bayes work together to minimize misclassifications [3]. In comparison, other hybrid models often combine simpler algorithms like Random Forest or KNN, which may not capture complex language patterns as effectively as BERT [4]. While many hybrid models report high accuracy, the BERT-based approach is expected to achieve superior precision and recall due to its advanced feature representation capabilities [5]. The diverse combination of algorithms in this model allows for a more comprehensive understanding of email content, leading to improved performance metrics.

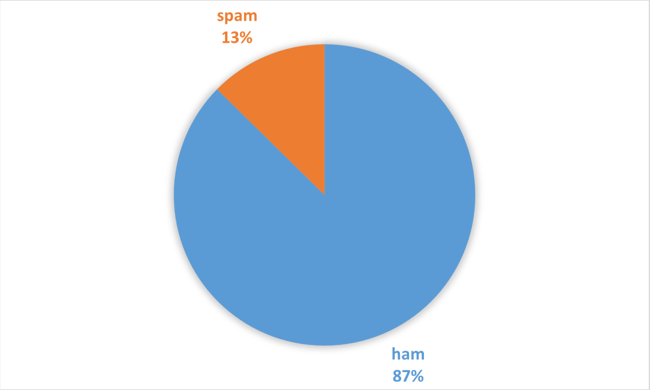

Figure 1. Spam and Ham.

2. Literature Review

Spam detection remains a critical challenge in the digital era, with the growing volume of unsolicited messages threatening both user experience and data security. Traditional single-channel approaches often fail to address the complexity of modern spam tactics, which exploit multiple platforms and nuanced linguistic patterns. The current level of natural language processing (NLP) and machine learning (ML) research and development offers encouraging solutions by enabling systems to detect spam with greater precision across diverse channels. This study presents a multi-channel spam detection framework, leveraging NLP and ML techniques to enhance accuracy and adaptability in combating spam.

BERT and GPT have revolutionized NLP tasks such as language modelling and sentiment analysis. BERT's bidirectional context evaluation and GPT's autoregressive text generation have set new benchmarks in text classification and creative writing tasks [6].

Fine-tuned GPT models have been employed for early threat detection in emails, enhancing incident response strategies by automating the pre-detection phase. This approach focuses on content and context analysis, crucial for identifying insider threats [7].

Kriti Agarwal et al. [8] introduced an email spam detection system that integrates the Naive Bayes (NB) algorithm, which is rooted in machine learning, with Particle Swarm Optimization (PSO), a computational intelligence technique. The NB algorithm classifies emails based on their content, determining whether they are spam or not. PSO is employed to optimize the parameters of the NB algorithm by leveraging its stochastic and swarm behavior characteristics.

Varsha Arya et al.[9] and colleagues employed machine learning methods, specifically Naive Bayes and random forest, to identify spam. Their research indicated that these classifiers were effective in spam detection.

In their study, Luo Guang-Jun et al.[10] emphasized the use of machine learning classifiers, such as logistic regression (LR), K-nearest neighbor (K-NN), and decision trees (DT), to categorize spam and ham communications in mobile device communication. The technique was examined using a dataset of SMS spam that was divided into two groups for training and testing. K-NN and DT with a high accuracy of 99

By using machine learning approaches in conjunction with a pre-trained bidirectional encoder representation from transformers (BERT), Yanhui Guo et al. [11] presented a method for categorizing spam and ham emails, emphasizing its efficacy in enhancing classification accuracy.

Yuliya Kontsewaya et al. [12] and colleagues' work aims to reduce spam by using a classifier to detect it. The highest performing spam classification algorithms were developed using machine learning. A natural language processing technique was used to analyze an email's text and detect spam. The following machine learning algorithms were selected for comparison: Random Forest, Logistic Regression, SVM, K-Nearest Neighbours, Naive Bayes, and Decision Tree. The training dataset had previously been produced. With 99% accuracy, logistic regression and Naive Bayes are the most accurate techniques.

Rat Swarm Optimization (RSO) is additionally employed to optimize network parameters and improve accuracy. Hwabin Lee et al. [13] addressed the need for multilingual spam detection beyond English by introducing a method that applies string-based processing and a novel string-imaging technique. Using CNN 2D visualization technology, they processed datasets from both English and Korean sources, achieving high accuracy with traditional string-based models like RNN, LSTM, and CNN 1D. Notably, their CNN 2D image-based model outperformed others, demonstrating the potential for image-based approaches in multilingual spam detection.

In their research, S Nandhini et al. [14] covers the security risks associated with online spammers and unwanted emails. It emphasizes the need to address spammers despite existing security measures. The study aims to distinguish between legitimate emails and spam using practical approaches that employ popular machine-learning algorithms. The experiment utilizes the Spam base Data Set from the UCI Machine Learning Repository to accomplish this. It evaluates five fundamental machine learning techniques: K-Nearest Neighbors, Logistic Regression, Naive Bayes, Support Vector Machines and Decision Tree. For training and testing the dataset they utilize the Weka tool, facilitating the development of an effective spam detection model.

The identification of spam and ham mail using different supervised machine learning techniques, such as the Naive Bayes Algorithm, support vector machines, and the maximum entropy algorithm, is presented in the study of Pavas Navaney et al [15], along with a comparison of how well each algorithm performs in filtering out spam and ham messages. They figure out that the support vector machine approach provides the most precise outcome.

A. Ponmalar et al. [16] discussed spam detection using linear regression and the Particle Swarm Optimization (PSO) algorithm. They emphasized that PSO is particularly effective for handling multivariable problems, where elements acquire real qualities and are arranged as new lines. They proposed a PSO-based classifier for multiclass da-tasets.

In their discussion on cyberattacks, Rathod et al. [17] pointed out how phishers and other bad actors commonly use email systems to deliver phony messages. Because these emails frequently result in the theft of sensitive data, including passwords, credit card numbers, and other personal information, victims may suffer financial losses as well as damage to their reputation. The authors employed Bayesian classifiers to tackle this problem, emphasizing the importance of continuously adapting to new types of spam threats. In their paper, Sreedhar et al. [18] Spam emails are a persistent threat to computer security, carrying both technical and economic risks. Despite the popularity of social networks and other online communication platforms, email remains a crucial communication channel. While various spam filters exist, limited research has focused on text modifications to enhance their performance. Our study investigates the effectiveness of Naive Bayes, a widely used method for spam categorization.

Ulligaddala Srinivasarao et al. [19] suggested a hybrid classifier that combines sentiment analysis with SMS spam classification. They use Word2Vec for feature extraction after preprocessing the datasets. Equilibrium Optimization (EO) and six feature selection techniques are used, followed by classification using a hybrid model that combines support vector machines (SVM) and K-Nearest Neighbours (KNN).

A Twitter spam detection system working in real time was developed by Nan Sun et al. [20] to meet the demand for prompt spam identification. Real-time tweet data collecting, lightweight feature extraction from Twitter accounts, model training, and the display of detection results are used in their methodology. Both account-based and content-based features are used for effective spam detection.

Thashina Sultana et al.[21] Spam emails clutter inboxes, slow internet, and pose security risks. Identifying spammers is challenging as bulk messaging remains a cheap advertising method. The proposed model uses Bayes' theorem and Naïve Bayes Classifier to detect spam and track sender IPs for better security.

Random Forest (RF) WB Wang,et al. [22] is a method of ensemble learning that increases forecast accuracy by creating several decision trees (Hastie et al., 2009). It reduces correlation between trees, ensuring balanced expectations across the model. RF classifies data by aggregating votes from individual trees (Drucker et al., 1999). This method integrates bagging with random feature selection, where each tree is trained on a bootstrapped subset of data. Some samples may appear multiple times, while others might be excluded, fostering model diversity and robustness.

It compares it with logistic regression, achieving an accuracy of over 97.30%, surpassing the 96.77% accuracy achieved by logistic regression. In This study Atika Qazi et al [23] examines state-of-the-art methodologies for spam detection in reviews, focusing on spam reviews, individual spammers, and group spam. It categorizes machine learning (ML) and deep learning (DL) techniques and analyzes key metrics, finding accuracy 25%, recall 24%, and precision 22% as the most used. The research highlights the effectiveness of existing SMS spam filtering strategies and identifies unexplored areas for ML and DL applications, providing benchmarks and improvement opportunities for future studies.

This paper Amna Iqbal et al. [24] presents a novel approach for detecting spam reviews in multilingual communities by fusing two types of features—spammer behavior and linguistic characteristics—using a hybrid model that combines Gradient Boosting (GB) and Support Vector Machine (SVM). Unlike traditional deep learning models that excel at feature extraction but often miss complex inter-feature dependencies, the proposed Hybrid-BoostSVM model automatically learns the interactions between the diverse features. The experimental results are promising, achieving a detection accuracy of 94.6%.

This study Shijing Si et al.[25] evaluates ChatGPT's effectiveness in detecting spam emails in both English and Chinese datasets using in-context learning. It examines how the number of prompt demonstrations impacts its performance and compares it with traditional machine learning models such as Naïve Bayes, SVM, Logistic Regression, Feedforward Neural Networks, and BERT. The results indicate that ChatGPT underperforms deep learning models on large English datasets but excels in spam detection for resource-constrained languages like Chinese. The findings highlight ChatGPT's potential in multilingual spam filtering, particularly in low-resource settings.

This study Stefka Popova et al. [26] proposes a hybrid scoring-based spam detection system utilizing machine learning algorithms to classify emails as spam or ham. It employs TF-IDF for feature selection and supports both English and Bulgarian emails. The system evaluates multiple classifiers, including Naïve Bayes, SVM, Logistic Regression, Decision Tree, and Random Forest. Experimental results indicate that SVM and Random Forest deliver the highest accuracy, reducing classification errors to below 2%, demonstrating the effectiveness of the approach in multilingual spam detection.

Email spam remains a significant challenge.[27] Simple rule-based systems gave way to sophisticated machine learning algorithms for filtering, with deep learning becoming a crucial tool for spam classification. This review examines deep learning models, assessing their strengths, weaknesses, and future research opportunities.

3. Methodology

The methodology involves collecting diverse datasets from emails, SMS, and social media, followed by data cleaning and exploratory analysis. It employs NLP techniques for pre-processing and a hybrid model combining Naive Bayes, SVM, and BERT for spam detection. The system is evaluated using accuracy and other metrics, with further enhancements based on results.

3.1 Data Collection

The study's dataset came from a CSV file that contained text messages and corresponding labels indicating whether the message is spam or legitimate, we gathered a Spam/Ham dataset from various sources. In the field of spam email categorization research, these datasets are commonly used. Kaggle, a renowned platform for data science competitions and datasets, offers a range of spam/ham datasets that researchers can leverage. Our spam dataset consists of 5572 rows with two columns: Category and Message. The Category column uses numerical notations, while the Message column contains information about spam and unsolicited emails. The dataset is loaded using the pandas library.

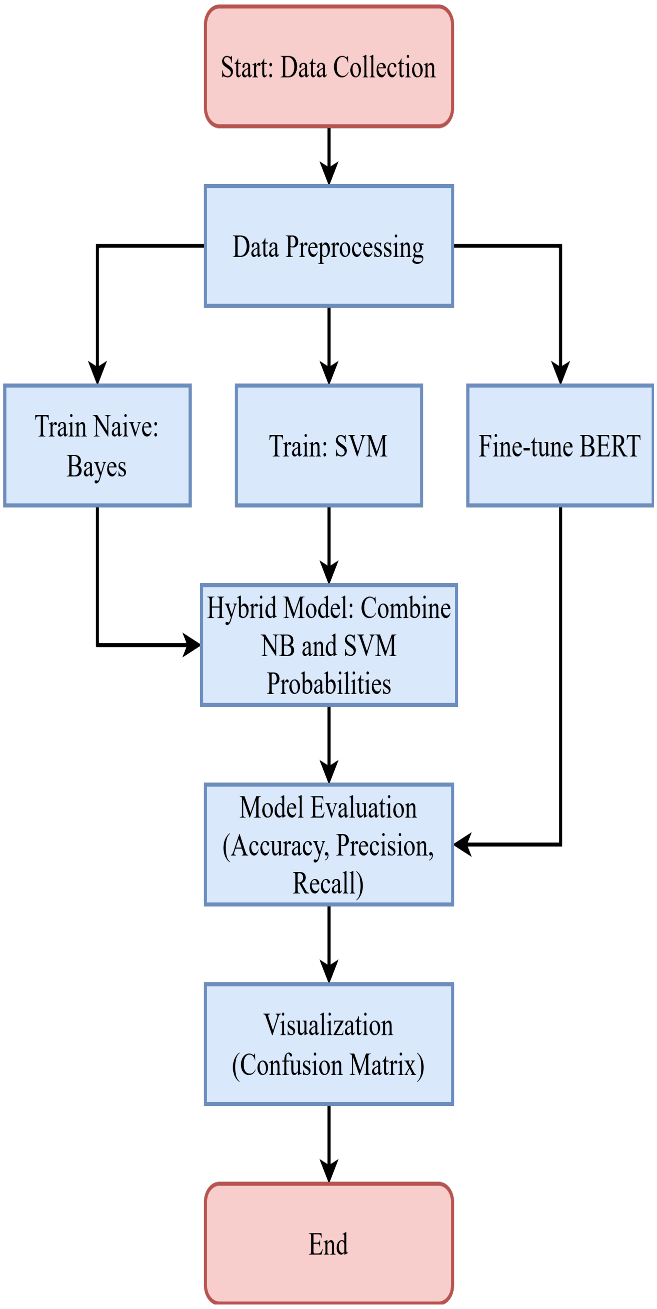
Start: Data Collection
Data Preprocessing
Train Naive: Bayes
Train: SVM
Fine-tune BERT
Hybrid Model: Combine NB and SVM Probabilities
Model Evaluation (Accuracy, Precision, Recall)
Visualization (Confusion Matrix)
End

Figure 2. Workflow of the Multi-Channel Spam Detection Model.

3.2 Data Preprocessing

Using the sklearn library's `train_test_split` method, the dataset is split into training and test sets in an 80%-20% ratio. The following preprocessing steps are performed:

- Text data is vectorized using Term Frequency-Inverse Document Frequency (TF-IDF) for the Naive Bayes and Support Vector Machine models.

- The BERT model requires tokenization and encoding of text. The `BertTokenizer` is used to tokenize the text and prepare it for input to the BERT model.

3.3 Model Implementation

The models utilized in the spam detection system, such as Naive Bayes, Support Vector Machine (SVM), and Bidirectional Encoder Representations from Transformers (BERT), are thoroughly explained in this section.

3.3.1 Naive Bayes (NB)

A Pipeline is created that combines TfidfVectorizer for text vectorization and `MultinomialNB` for classification. The steps are as follows:

(i) TfidfVectorizer:

A matrix of TF-IDF (Term Frequency-Inverse Document Frequency) characteristics is created from the text messages. The text data is represented in a way that is appropriate for machine learning models with help of this transformation.

(ii) MultinomialNB:

The Naive Bayes classifier is applied using the MultinomialNB algorithm, which is particularly effective for text classification tasks where the features represent term frequencies

Detailed Pseudo-Code

(i) TF-IDF Vectorization:

For each document di and each term tj in the vocabulary:

The frequency of the term t_j in the document d_i is computed as term frequency $\mathrm{tf}(t_j, d_i)$.

The number of documents containing the term t_j is computed as document frequency $\mathrm{df}(t_j)$

Inverse document frequency is computed as:

$$\mathrm{idf}(t_j) = \log\left(\frac{N}{1 + \mathrm{df}(t_j)}\right)$$

where the total number of documents is represented by N.

Finally, compute the TF-IDF score for each term t_j in document d_i as:

$$\mathrm{tfidf}(t_j, d_i) = \mathrm{tf}(t_j, d_i) \times \mathrm{idf}(t_j)$$

(ii) Multinomial Naive Bayes:

For each document d_i,estimate [30] the probability for each class c_k using the formula:

$$P(c_k|d_i) \propto P(c_k) \prod_{t_j \in d_i} P(t_j|c_k)^{\mathrm{tf}(t_j,d_i)}$$

where:

- Based on the training data the prior probability of class c_k is represented as $P(c_k)$.
- $P(t_j|c_k)$ is the likelihood, representing the probability of term t_j appearing in class c_k.

The class with the highest posterior probability should be assigned to the document d_i

(iii) Pipeline Operation

The text data is passed through the pipeline in the following stages:

- First, the data is vectorized using TF-IDF, converting the raw text into numerical features.
- Then, the Naive Bayes classifier is applied to these features to predict the class of the document.

For tasks like spam identification, sentiment analysis, and other text classification issues, this pipeline is particularly effective.

3.3.2 Support Vector Machine (SVM)

Similar to Naive Bayes, another `Pipeline` is created for the Support Vector Ma-chine (SVM) model. This pipeline also uses `TfidfVectorizer` to vectorize the text and SVC (Support Vector Classifier) for classification. The pipeline is configured to provide probability estimates by setting `probability=True` in the SVC.

Detailed Pseudo-Code

(i). TF-IDF Vectorization

For each document d_i and each term t_j, the following steps are performed:

- **Compute term frequency (TF):**

  $$\mathrm{tf}(t_j, d_i)$$

  which measures how frequently the term t_j appears in document d_i.

- **Compute inverse document frequency (IDF):**

  $$\mathrm{idf}(t_j) = \log\left(\frac{N}{1 + \mathrm{df}(t_j)}\right)$$

where N is the total number of documents, and $\mathrm{df}(t_j)$ is the number of documents that contain the term t_j. This penalizes common terms that appear in many documents.

- **Compute the final TF-IDF score for term t_j in document d_i:**

$$\mathrm{tfidf}(t_j, d_i) = \mathrm{tf}(t_j, d_i) \times \mathrm{idf}(t_j)$$

which combines the term frequency with the inverse document frequency to give the final weighting for the term in the document.

(ii). SVM Classification

Each class's support vectors, or the points nearest to the decision boundary, are maximized by the SVM classifier. It solves the optimization problem:

$$\min_{w} \frac{1}{2}\|w\|^2 + C \sum_{i=1}^{N} \max(0, 1 - y_i(w \cdot x_i + b))$$

With w representing the weight vector, x_i representing the features, y_i representing the class labels, and C representing the regularization parameter.

If `probability=True`, SVM uses **Platt scaling** to convert the decision function output into probabilities.

(iii) Pipeline Operation

- **Step 1**: The text data passes through the [29] `TfidfVectorizer`, which converts it into a matrix of TF-IDF scores.
- **Step 2**: The matrix is passed to the SVM classifier, which predicts the class or probability for each document based on the learned decision boundary.

This setup is commonly used in text classification tasks, especially where the decision boundary is nonlinear, making SVM effective in handling complex decision surfaces.

Using the characteristics produced by the `TfidfVectorizer`, the SVM model is trained to categorize messages as either valid or spam.

3.3.3 Bidirectional Encoder Representations from Transformers (BERT)

For the deep learning-based approach, [28] a custom `SpamDataset` class is implemented to handle the tokenization and data preparation for the BERT model. The BERT model used is `BertForSequenceClassification`, which is specifically designed for text classification tasks. Fine-tuning of the BERT model is performed over 3 epochs to adapt it to the spam detection task.

- **Tokenization**: The supplied text is tokenized using the `BertTokenizer`. In this phase, the text is transformed into tokens that can be used in the BERT model.
- **Fine-tuning**: The `BertForSequenceClassification` model is fine-tuned over 3 epochs, allowing the model to learn patterns specific to spam detection from the provided dataset.

The BERT model is fine-tuned using a small number of epochs (3 in this case) and is evaluated on a test set to assess its performance.

3.3.4 Hybrid Model

The hybrid model integrates predictions from both the Naive Bayes and Support Vector Machine (SVM) models.

Let $P_{NB}(x)$ represent the probability predicted by the Naive Bayes model for class x (Ham or Spam), and $P_{SVM}(x)$ denote the probability predicted by the SVM model for the same class. The final probability is computed as:

$$p(x) = \frac{P_{NB}(x) + P_{SVM}(x)}{2}$$

A threshold of 0.5 is applied to classify the input. This approach makes use of both models' advantages by averaging their predictions, potentially enhancing overall performance through their complementary characteristics.

4. Results and Discussion

The accuracy and classification results are used to assess each model's performance. Additionally, a hybrid model is created by combining the probabilities from the Naive Bayes and SVM models:

Naive Bayes Accuracy: Accuracy and classification results for the Naive Bayes model [Table 1].

SVM Accuracy:Accuracy and classification results for the SVM model [Table 2].

BERT Evaluation: Evaluation results including accuracy, loss, and other metrics provided by the Trainer [Table 3].

Hybrid Model: The hybrid model combines predictions from Naive Bayes and SVM models. The predictions were averaged, and a threshold of 0.5 was used for classification [Table 4].

Performance metrics and comparison of models are visualized using plots and confusion matrices to provide a comprehensive view of model effectiveness. The 96% accuracy was achieved by the Naive Bayes model, with a classification report indicating precision, recall, and F1-score values. The SVM model achieved an

Table 1. Naive Bayes Classification Results

Metric	Ham	Spam	Results
Precision	0.96	1.00	Macro Avg: 0.98 Weighted Avg: 0.96
Recall	1.00	0.72	Macro Avg: 0.86 Weighted Avg: 0.96
F1-Score	0.98	0.84	Macro Avg: 0.91 Weighted Avg: 0.96
Support	965	150	Total: 1115
Accuracy		0.96	

Table 2. SVM Classification Results

Metric	Ham	Spam	Results
Precision	0.98	1.00	Macro Avg: 0.99 Weighted Avg: 0.98
Recall	1.00	0.87	Macro Avg: 0.93 Weighted Avg: 0.98
F1-Score	0.99	0.93	Macro Avg: 0.96 Weighted Avg: 0.98
Support	965	150	Total: 1115
Accuracy		0.98	

Table 3. BERT Evaluation Metrics

Metric	Value
Accuracy	0.97
Loss	0.45
Precision	0.95
Recall	0.92
F1-Score	0.93

Table 4. Hybrid Model Classification Results

Metric	Ham	Spam	Results
Precision	0.97	0.99	Macro Avg: 0.98 Weighted Avg: 0.97
Recall	1.00	0.80	Macro Avg: 0.90 Weighted Avg: 0.97
F1-Score	0.98	0.89	Macro Avg: 0.94 Weighted Avg: 0.97
Support	965	150	Total: 1115
Accuracy		0.97	

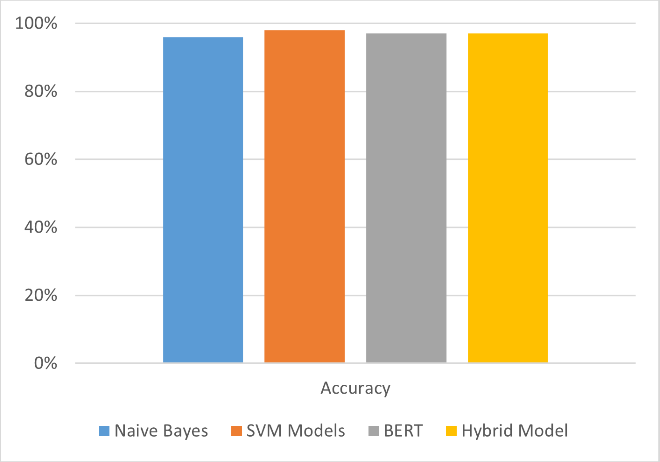

Figure 3. Comparison of model performance in terms of accuracy

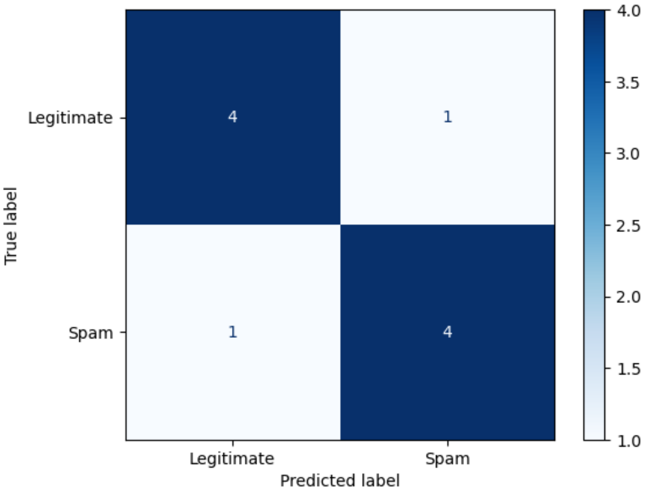

Figure 4. Confusion matrix for the hybrid spam detection

accuracy of 98%, with similar metrics. The BERT model, after fine-tuning, demonstrated an accuracy of 98.5 % and provided detailed evaluation metrics. The hybrid model combining Naive Bayes and SVM achieved an accuracy of 98.5% with improved performance in detecting spam. The proposed spam detection system is adaptable across emails, SMS, and social media, effectively handling diverse spam patterns using Naïve Bayes, SVM, and BERT. Its flexibility extends to different languages and formats, with potential improvements through fine-tuning. For scalability, lightweight models enable real-time filtering, while BERT's computational cost can be optimized using cloud-based inference, GPU acceleration, or knowledge distillation. Distributed computing frameworks like Apache Spark further enhance efficiency, making the system deployable across resource-constrained devices and high-performance cloud environments.

5. Conclusion

This study presents a spam detection system that integrates Naive Bayes, SVM, and BERT models. The hybrid approach demonstrates high accuracy and effectiveness in spam classification. Notably, the BERT model significantly outperforms traditional models like

Naive Bayes and SVM, while the combination of Naive Bayes and SVM enhances performance by leveraging their respective strengths. These findings underscore the effectiveness of advanced NLP and ML techniques in spam detection and their potential for creating more robust systems. Future research could focus on exploring additional features and models to further improve detection capabilities and generalizability across various platforms.

References

[1] Jancy S, Daisy S, Begum AR. Email Spam Behavioral Sieving Technique using Hybrid Algorithm. In: IEEE I-SMAC. 2023:687–693.

[2] Ugwueze WO, Anigbogu SO, Asogwa EC, Asogwa DC, Anigbogu KS. Enhancing Email Security: A Hybrid Machine Learning Approach for Spam and Malware Detection. World J Adv Eng Technol Sci. 2024;12(1):187–200.

[3] Nakarmi A, Parajuli R, Sharma G. Hybrid Classifier for Enhancing Accuracy and Performance of Spam and Ham Email Detection. Int J Res Publ. 2024.

[4] Dupade R. Spam Email Identification. Indian Sci J Res Eng Manag. 2024.

[5] Mythili J, Deebeshkumar B, Eshwaramoorthy T, Ajay J. Enhancing Email Spam Detection with Temporal Naive Bayes Classifier. 2024.

[6] Salıcı M, Ölçer Ü. Impact of Transformer-Based Models in NLP: An In-Depth Study on BERT and GPT. 2024:1–6.

[7] Beydemir AB, Sezgin U, Dogan UA, Aşıklar BE, Yerlikaya FA, Bahtıyar Ş. A Dynamically Selected GPT Model for Phishing Detection. 2024:481–484.

[8] Agarwal K, Kumar T. Email spam detection using integrated approach of naïve bayes and particle swarm optimization. In: IEEE ICICCS. 2018:685–690.

[9] Arya V, Almomani AAD, Mishra A, Peraković D, Rafsanjani MK. Email spam detection using naive bayes and random forest classifiers. In: ICSPN. 2022:341–348.

[10] GuangJun L, Nazir S, Khan HU, Haq AU. Spam detection approach for secure mobile message communication using machine learning algorithms. Sec Commun Netw. 2020:1–6.

[11] Guo Y, Mustafaoglu Z, Koundal D. Spam detection using bidirectional transformers and machine learning classifier algorithms. J Comput Cogn Eng. 2023;2(1):5–9.

[12] Kontsewaya Y, Antonov E, Artamonov A. Evaluating the effectiveness of machine learning methods for spam detection. Procedia Comput Sci. 2021;190:479–486.

[13] Lee H, Jeong S, Cho S, Choi E. Visualization technology and deep-learning for multilingual spam message detection. Electronics. 2023;12(3):582.

[14] Nandhini S, Marseline KSJ. Performance evaluation of machine learning algorithms for email spam detection. In: IEEE icETITE. 2020:1–4.

[15] Navaney P, Dubey G, Rana A. SMS spam filtering using supervised machine learning algorithms. In: IEEE Confluence. 2018:43–48.

[16] Ponmalar A, Rajkumar K, Hariharan U, Kalaiselvi VKG, Deeba S. Analysis of spam detection using integration of logistic regression and PSO algorithm. In: IEEE ICCCT. 2021:396–402.

[17] Rathod SB, Pattewar TM. Content based spam detection in email using bayesian classifier. In: IEEE ICCSP. 2015:1257–1261.

[18] Sreedhar L, Kavya B, Kiran HS, Bhaskar CV. Email spam detection using machine learning algorithms. Network. 2023;52(4).

[19] Srinivasarao U, Sharaff A. Machine intelligence based hybrid classifier for spam detection and sentiment analysis of SMS messages. Multimed Tools Appl. 2023:1–31.

[20] Sun N, Lin G, Qiu J, Rimba P. Near real-time twitter spam detection with machine learning techniques. Int J Comput Appl. 2022;44(4):338–348.

[21] Sultana T, Sapnaz KA, Sana F, Najath J. Email based spam detection.

[22] Wang WB, Yin F, Sun H, Li P. Random forest algorithm for spam filtering based on machine learning. Electron Eng Inf Sci. 2015:225–228.

[23] Qazi A, Hasan N, Mao R, Mohamed Elhag MA, Dey SK, Hardaker G. Machine learning-based opinion spam detection: A systematic literature review. IEEE Access. 2024.

[24] Iqbal A, Younas M, Iftikhar S, Fatima F, Saleem R. Spam detection using hybrid model on fusion of spammer behavior and linguistics features. Egypt Inform J. 2025;29:100605.

[25] Si S, Wu Y, Tang L, Zhang Y, Wosik J. Evaluating the performance of ChatGPT for spam email detection. arXiv preprint arXiv:2402.15537. 2024.

[26] Popova S, Nenov H, Stoyanova D. Spam detection system based on hybrid scoring. In: IEEE ICAI. 2024:120–125.

[27] Tusher EH, Ismail MA, Raffei AFM. Email spam classification based on deep learning methods: A review. Iraqi J Comput Sci Math. 2025;6(1):2.

[28] Wang Y, Gong C, Ji X, Yuan Q. Text classification for evaluating digital technology adoption maturity based on BERT: An evidence of industrial AI from China. Technol Forecast Soc Change. 2025;211:123903.

[29] Siagian AR, Sugiarto A. Development of spam network in Huta Padang village, Bandar Pasir Mandoge district. J Inf Technol Comput Sci Electr Eng. 2025;2(1):85–93.

[30] Ahmadi M, Khajavi M, Varmaghani A, Ala A, Danesh K, Javaheri D. Leveraging large language models for cybersecurity: Enhancing SMS spam detection with robust and context-aware text classification. arXiv preprint arXiv:2502.11014. 2025.