A Comparative Analysis of Selected Machine Learning Technique for Spam Email Detection

[1]Egbezemoghie F, P., [2]Muyideen, & [3]Ayeni, J. K

[1] Ph.D Student, Computer Science Department, University of Ilorin

[2]Dr. Muyideen, Computer Science Department, University of Ilorin

[3]Computer Science Department, Kwara State Polytechnic, Ilorin

DOI: https://doi.org/10.51244/IJRSI.2026.1306000247

Received: 11 June 2026; Accepted: 16 June 2026; Published: 03 July 2026

ABSTRACT

This study explores comparative analysis of selected machine language techniques for spam email detection system. The proposed solution aims to improve user experience by accurately identifying and filtering out unwanted and malicious emails, thereby reducing clutter. Our experiments show high accuracy and efficiency in detecting spam emails, highlighting the system's potential to significantly mitigate email-based cyber threats. This research contributes to ongoing cybersecurity efforts, offering a robust tool for safer and more efficient email communication. The project assessed three machine learning algorithms: Naïve Bayes, SVM, and KNN. The system was designed and implemented using the best-performing algorithm to detect unwanted and malicious emails. Performance metrics such as accuracy, precision, recall, and F1-score were used to evaluate the algorithms. These metrics were derived from a confusion matrix generated by the algorithms, trained using a dataset from Kaggle. The spam email detection system's interface was designed using the Flask framework, a popular Python web framework. The machine learning models powering the system's threat detection capabilities were trained using Jupyter Notebook, an interactive computing environment within the Anaconda Navigator platform. Flask's flexibility and modularity allowed the development of a user-friendly web-based interface for interacting with the spam detection capabilities. Jupyter Notebook enabled the research team to experiment with various natural language processing and machine learning techniques, leading to the development of the comprehensive threat detection algorithms employed by the system. Combining Flask for the web interface and Jupyter Notebook for model training and development allows the spam email detection system to offer real-time email analysis, comprehensive threat detection, and customizable settings. This protects users from the evolving landscape of online scams and malicious activities.

Keywords:: Detection, Email, Ham, Machine Learning, Spam.

INTRODUCTION

The rapid growth of email usage in recent years has been accompanied by a significant increase in the volume of unsolicited and unwanted emails, commonly known as spam. Spam emails can be a nuisance and even pose security risks by spreading malware or phishing scams. As a result, effective spam detection and filtration have become critical challenges for email service providers and users alike. Traditional approaches to spam filtering have relied on techniques such as blacklisting, whitelisting, and simple keyword matching. However, these methods have proven to be ineffective against the increasingly sophisticated tactics employed by spammers. More advanced techniques are needed to accurately classify emails as spam .In the contemporary age of information technology, sharing information has become more accessible than ever before. Users can easily exchange information across various platforms from any part of the world. Presently, email stands out as the most convenient, cost-effective, and swift method for transmitting information globally. However, emails are susceptible to various attacks, with spam being the most prevalent and damaging due to its simplicity (Faris *et al.,* 2019). Apart from consuming recipients' time and resources, irrelevant emails may contain malicious

content such as attachments or URLs, posing a threat to the security of the host system (Olatunji, 2017). Spam refers to undesired messages sent by attackers to numerous recipients via email or other communication means (Olatunji, 2017). Consequently, safeguarding the email system demands significant attention. In spam emails, viruses, rats, and Trojans may be concealed, enticing users towards online services and potentially leading to data and financial fraud as well as identity theft (Alghoul *et al*., 2018). Although some email providers offer keyword-based rules for filtering messages, this approach is impractical as it is cumbersome and users are reluctant to customize their messages, making their accounts susceptible to spammers (Ferrag *et al*., 2020).Machine learning has emerged as a powerful tool for spam detection, with algorithms such as Naive Bayes, Support Vector Machines (SVMs), and Random Forests showing promising results. These algorithms work by learning patterns and features from a labeled dataset of spam and ham emails, and then using that knowledge to classify new incoming messages. One key aspect of effective spam detection using machine learning is the selection of relevant features from the email content. The high dimensionality of email data, with many possible words and phrases, can lead to over fitting and poor generalization if not handled properly. Feature selection techniques aim to identify the most discriminative terms that best distinguish spam from ham, while reducing the overall feature space. Several feature selection strategies have been explored in the literature, including filter methods for example information gain, chi-square), wrapper methods for example genetic algorithms, particle swarm optimization), and embedded methods for example L1 regularization. However, there is still room for improvement in terms of the accuracy, efficiency, and robustness of these techniques. This study aims to perform comparative analysis of select machine learning for spam email detection with state-of-the-art machine learning algorithms. By leveraging the power of machine learning for spam email detection, the proposed system aims to achieve high accuracy in spam detection while maintaining low computational complexity and adaptability to evolving spam tactics.

Related Work

Sharmin, et. al, (2022) addresses the challenge of image-based spam detection by employing Convolutional Neural Networks (CNNs). Spammers often embed text within images to evade traditional text-based filters. The authors developed a CNN model that analyzes both raw images and their Canny edge representations to detect spam content effectively. The model was evaluated on real-world and synthetic image spam datasets, demonstrating superior performance compared to traditional machine learning techniques .The study underscores the efficacy of CNNs in handling image spam and highlights the importance of incorporating image analysis into spam detection systems to counter evolving spamming techniques. Jamal and Wimmer (2024), introduced the Improved Phishing, Spam, and Ham Detection Model (IPSDM), a fine-tuned BERT-based transformer tailored for classifying email content. Their model leverages deep contextual embeddings to distinguish between phishing, spam, and legitimate (ham) messages. The research addresses shortcomings in existing systems that often misclassify complex or hybrid messages. IPSDM showed improved precision and recall across multiple benchmark datasets, making it suitable for real-world deployment. The authors emphasize the importance of interpretability and propose using attention visualization for transparency. Their study illustrates how transformer-based architectures can advance the accuracy and reliability of spam and phishing detection in modern communication platforms. Talaei et. al., (2020), study proposes a spam detection method that integrates feature selection using the Sine–Cosine Algorithm (SCA) with an artificial neural network (ANN).Tusher et al. (2024) carried out a thorough review of methods for spamming detection from email, optimization procedures, issues, and emerging research gaps. They thoroughly reviewed disparate machine learning and deep learning spam detection methods and emphasized optimization techniques for model improvement. While the review highlighted significant achievements, it also raised such existing challenges as spam concept drift, adaptive detection system requirements, and the lack of standardized metrics of evaluation, noting significant areas calling for further research and innovation. Oluchukwu et al. Talaei et. al., (2020), study proposes a spam detection method that integrates feature selection using the Sine–Cosine Algorithm (SCA) with an artificial neural network (ANN). By optimizing the feature set with SCA, the model achieved higher precision, accuracy, and sensitivity compared to traditional classifiers like MLP, Bayesian networks, decision trees, and random forests. Implemented on the Spambase dataset, the approach demonstrated a precision of 98.64%, accuracy of 97.92%, and sensitivity of 98.36%. The research highlights the effectiveness of metaheuristic algorithms in enhancing spam detection performance and reducing errors associated with irrelevant or redundant features.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

METHODOLOGY

This study investigates the performance of Support Vector Machines (SVM), Decision Trees, Random Forests, and Logistic Regression for Aspect-Based Sentiment Analysis (ABSA) of Apple laptop customer reviews. The dataset, obtained from Kaggle.com (Figure 1) contains labeled reviews annotated with aspect terms and sentiment polarity (positive, negative, or neutral). The data preprocessing steps include text normalization, tokenization, stop-word removal, and lemmatization. Aspect terms are extracted using a combination of part-of-speech tagging and dependency parsing. Text data is vectorized using Term Frequency-Inverse Document Frequency (TF-IDF) to convert it into a machine-readable format. The dataset is split into training and testing sets using an 80:20 ratio, each model undergoes training with optimized hyper parameters. Performance evaluation is based on four key metrics: accuracy, precision, recall, and F1-score. All models are implemented using Python libraries including scikit-learn and NLTK. Comparative analysis helps identify the most effective model for ABSA in e-commerce review analysis.

	A	B	C	D	E	F	G	H
1		label	text	label_num				
2	605	ham	Subject: enron methanol ; meter # : 988291	0				
3	2349	ham	Subject: hpl nom for january 9 , 2001	0				
4	3624	ham	Subject: neon retreat	0				
5	4685	spam	Subject: photoshop , windows , office . cheap . main	1				
6	2030	ham	Subject: re : indian springs	0				
7	2949	ham	Subject: ehronline web address change	0				
8	2793	ham	Subject: spring savings certificate - take 30 % off	0				
9	4185	spam	Subject: looking for medication ? we ` re the best	1				
10	2641	ham	Subject: noms / actual flow for 2 / 26	0				
11	1870	ham	Subject: nominations for oct . 21 - 23 , 2000	0				
12	4922	spam	Subject: vocable % rnd - word asceticism	1				
13	3799	spam	Subject: report 01405 !	1				
14	1488	ham	Subject: enron / hpl actuals for august 28 , 2000	0				
15	3948	spam	Subject: vic . odin n ^ ow	1				
16	3418	ham	Subject: tenaska iv july	0				
17	4791	spam	Subject: underpriced issue with high return on equity	1				
18	2643	ham	Subject: re : first delivery - wheeler operating	0				
19	3137	ham	Subject: swift - may 2001 vols	0				
20	1629	ham	Subject: meter variances - ua 4 clean - up	0				
21	1858	ham	Subject: additional recruiting	0				
22	3261	ham	Subject: fw : ercot load comparison	0				
23	3447	ham	Subject: meter 6461 , concorde churchill	0				
24	2459	ham	Subject: hpl nom for january 25 , 2001	0				

spam_ham_dataset

Figure 1: Screenshot of dataset (Kaggle.com)

Support Vector Classifier (SVC) Performance Report

The Support Vector Classifier (SVC) achieved an overall accuracy of 96%, demonstrating strong performance in classifying spam and non-spam emails based on the provided dataset. The classifier shows high precision and recall for both classes, indicating its effectiveness in identifying spam emails while minimizing false positives and false negatives. These results suggest that SVC is a robust choice for spam detection tasks, offering high accuracy and reliability in distinguishing between spam and non-spam emails. The provided classification report evaluates the performance of a Support Vector Classifier (SVC) on a dataset, specifically for spam detection.

	precision	recall	f1-score	support
0	0.98	0.97	0.97	1071
1	0.93	0.96	0.94	481
accuracy			0.96	1552
macro avg	0.95	0.96	0.96	1552
weighted avg	0.97	0.96	0.96	1552

Figure 2: SVC

The K-Nearest Neighbors (KNN)

The K-Nearest Neighbors (KNN) classifier achieved an overall accuracy of 87%, demonstrating moderate performance in classifying spam and non-spam emails based on the provided dataset. While the classifier shows higher precision and recall for class 0 (non-spam), indicating better performance in identifying non-spam emails, it exhibits lower precision and recall for class 1 (spam). These results suggest that while KNN is effective in certain contexts, further optimization or feature engineering may be necessary to improve its performance in accurately identifying spam emails.

	precision	recall	f1-score	support
0	0.94	0.87	0.91	1108
1	0.73	0.87	0.80	444
accuracy			0.87	1552
macro avg	0.84	0.87	0.85	1552
weighted avg	0.88	0.87	0.88	1552

Figure 3: KNN

Naive Bayes Classification for Spam Detection

After pre-processing, the dataset was split into training and testing sets using a 70-30 split ratio. The Naive Bayes classifier was trained on the training set and evaluated using the testing set.

	precision	recall	f1-score	support
0	0.97	0.99	0.98	1101
1	0.98	0.94	0.96	451
accuracy			0.98	1552
macro avg	0.98	0.96	0.97	1552
weighted avg	0.98	0.98	0.98	1552

Figure 4: Naive Bayes

Model Evaluation Parameters

In this study, metrics such as accuracy, precision, recall, and F1-measure are used to assess how well text classification systems perform.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2)$$

$$\text{Precision} = \frac{TN}{TP+FP} \quad (3)$$

$$\text{F1-Score} = 2x \frac{\text{Precision x Recall}}{\text{Precision + Recall}} \quad (4)$$

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

The terms TP, FP, TN, and FN denote the number of true positives, false positives, true negatives, and false negatives, respectively, as given in equations (1) through (4).

Model Evaluation Results

The model evaluation is accomplished through the use of Anaconda Python programming language using the metrics; precision, r ecall, F1-score, and Accuracy are all metrics to be used to evaluate the performance of text categorization systems. After implementing these algorithms, the models are evaluated using a testing dataset. Key evaluation metrics include accuracy, precision, recall, and F1-score. These metrics help determine the performance of each algorithm and provide insights into which approach is most effective for spam detection. In spam detection, evaluating the performance of machine learning models is crucial to determine their effectiveness and reliability. The key evaluation parameters commonly used are precision, recall, F-measure (F1-score), and accuracy. These metrics provide insights into the classifier's performance, helping to assess the quality of spam detection. To understand these parameters, it's important to first explain the concepts of True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN).

Table 1: Accuracy Comparison

Model/Metric	**Accuracy**
Naïve Bayes	98%
KNN	87%
SVM	96%

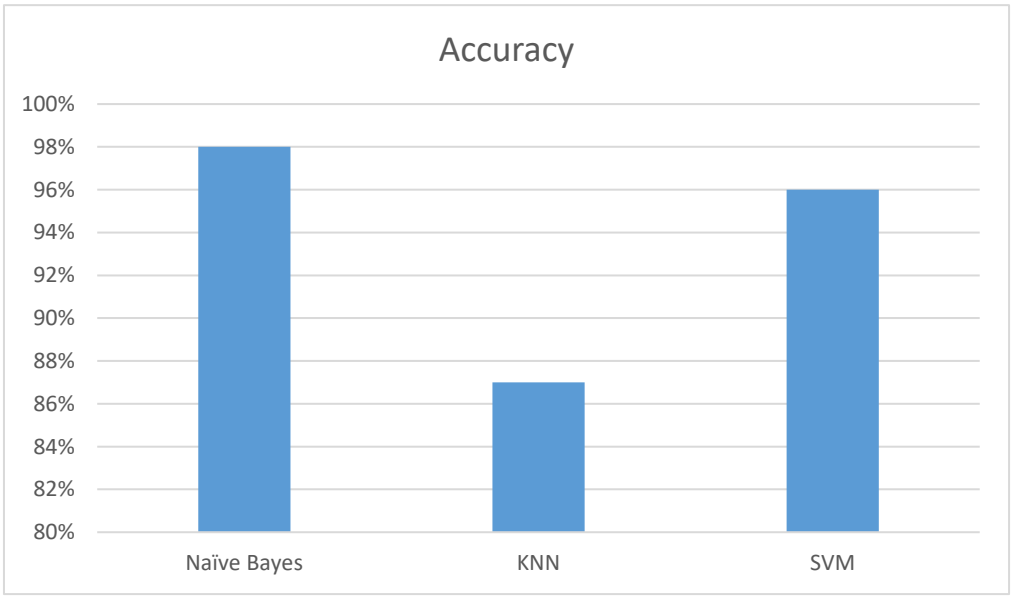


Figure 6: Chart showing Accuracy

Precision

The table 1 & Figures 6 highlight the **precision scores** of three machine learning models Precision is a key performance metric in machine learning that measures the quality of a models positive predictions, it answers the question: in the table ,the SVN model is the most precise among the three, making it the best choice if the cost of a false positive error is very high. Precision .only 2% of its positive predications are fasle alarms, KNN 94% ,this models has the lowest precision meaning it generates more fasle postives than the other two.6%of its positive precisions are incorrect.Naive Bayes 97% , the model predicts an outcome as positive it is correct 97% of the time .

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

Table 2: Precision

Model/Metric	**Precision**
Naïve Bayes	97%
KNN	94%
SVM	98%

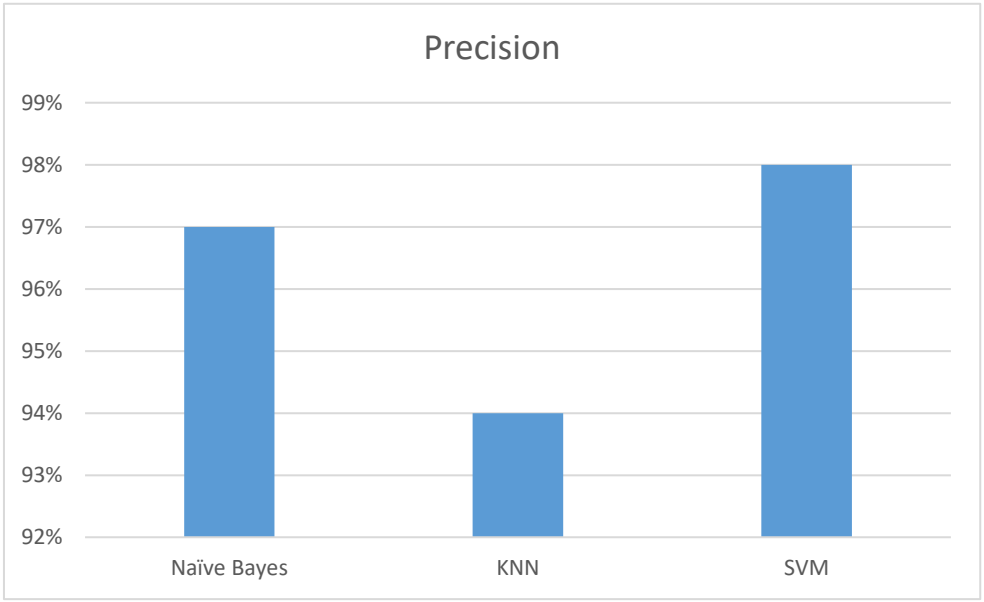


Figure 7: chart showing Precision

Recall

The table 2 & Figures 7 highlight the **recall scores** of three machine learning models, the Naïve Bayes models is the most highest 99% recall among the three. The model is often preferred in scenarios where false negatives. Knn 87%,this model has the lowest recalls its missed 13% of the actual cases., Svn 97% a very strong recall score , meaning this models also did an excellence job of finding nearly all the relevant cases, missing only 3% of them.

Table 3: Recall

Model/Metric	**Recall**
Naïve Bayes	99%
KNN	87%
SVM	*97%*

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

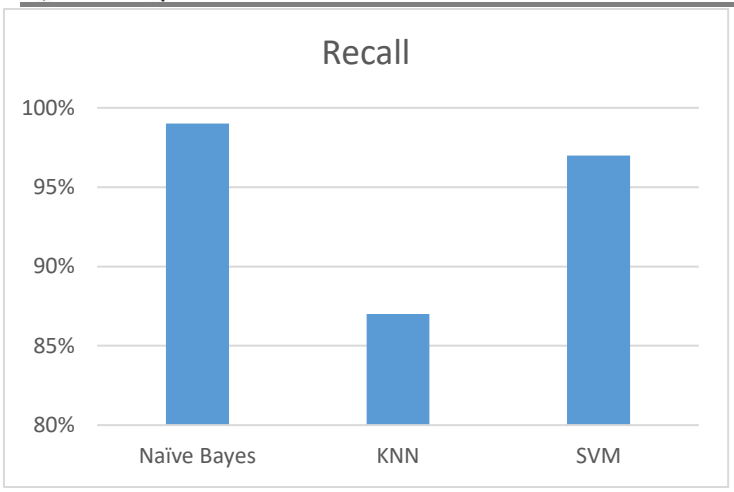


Figure 8: Chart Showing Recall

F1-Score

The table 3 & Figures 8 highlight the **F1 scores** of three machine learning models,F1 Score is a single metric that represents a machine learning models accuracy on the dataset. the results shows that naïve bayes models model is the best performer with the highest F1- score of 98%.it appears to be most reliable model for this classification problem ,successfully achieving a strong balance between identifying. KNN 91% this models had the lowest F1 Score among the three suggesting its performance wasn't as balanced or accurate as the others. SVN 97% very close to the best showing excellent predictive power with a strong balance of precision and recall.

Table 4: F1-Score

Model/Metric	**F1-Score**
Naïve Bayes	98%
KNN	91%
SVM	97%

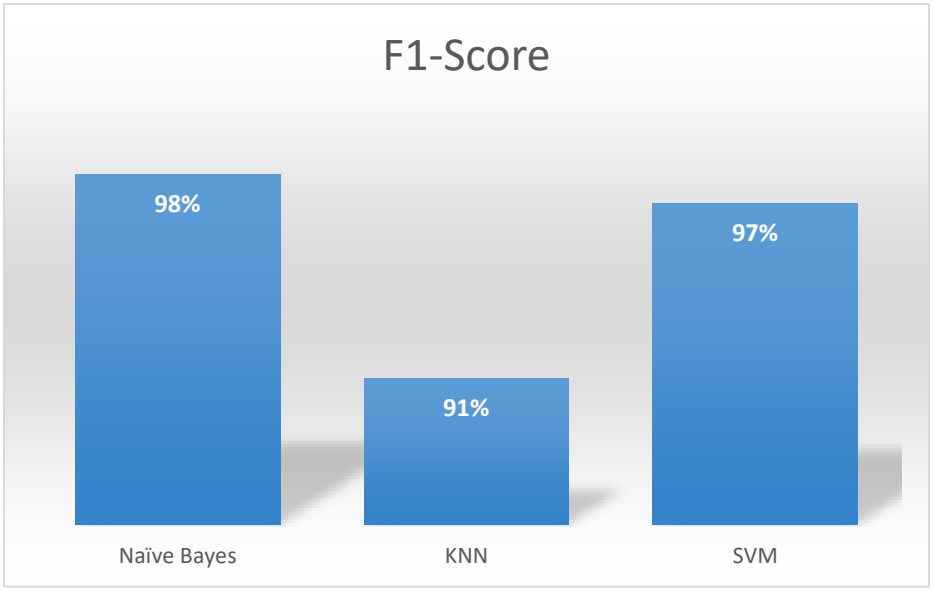


Figure 9: chart showing F1-Score

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue VI June 2026

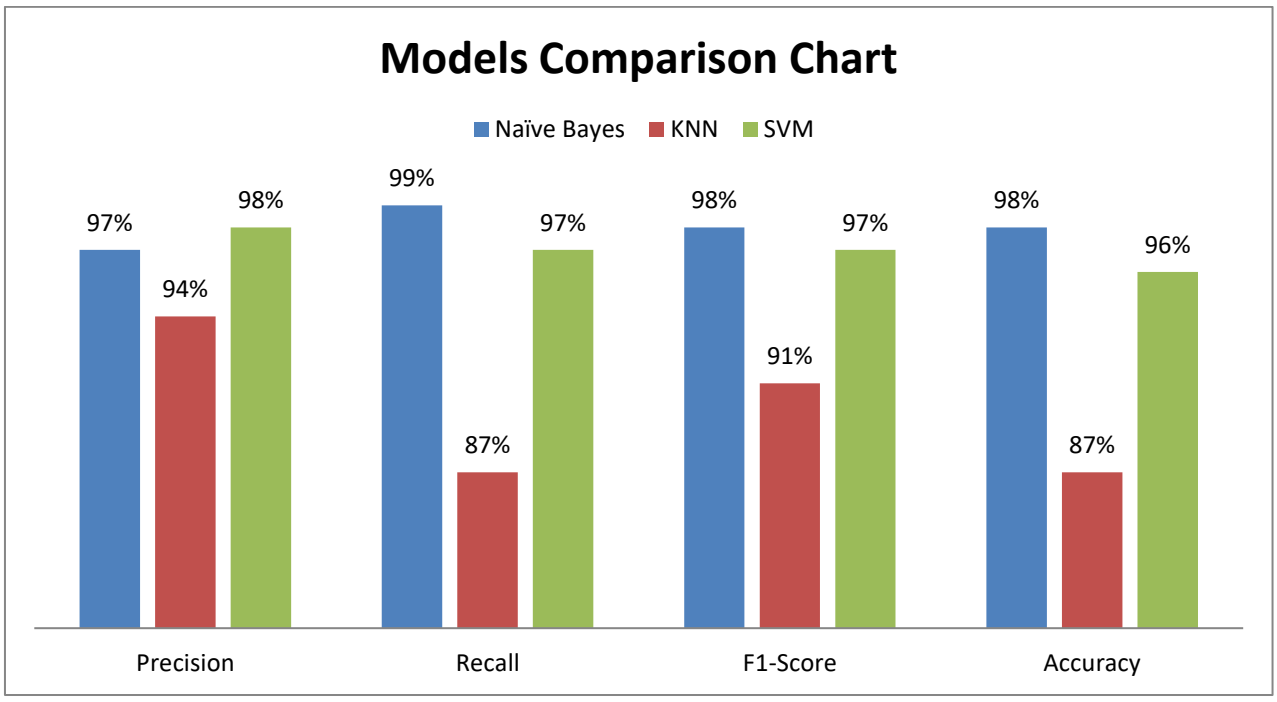
RESULT

The table compares the performance of three machine learning models Naïve Bayes, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) across four standard classification metrics: Precision, Recall, F1-Score, and Accuracy.

Table 5: An Illustration of Performance statistics comparison : Models Comparasion

Model/Metric	Precision	Recall	F1-Score	Accuracy
Naïve Bayes	97%	99%	98%	98%
KNN	94%	87%	91%	87%
SVM	98%	97%	97%	96%

Table 5.



T

Table 4.1 presents a comparative performance analysis of three machine learning models—Naïve Bayes, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM)—used for spam email detection. The evaluation is based on four key metrics: Precision, Recall, F1-Score, and Accuracy. Naïve Bayes demonstrates strong performance across all metrics, with a Precision of 97%, Recall of 99%, F1-Score of 98%, and an overall Accuracy of 98%. The high Recall suggests that Naïve Bayes is particularly effective at identifying spam messages without missing many. Its robust F1-Score further indicates a good balance between Precision and Recall, making it highly reliable for spam classification tasks. SVM also performs impressively, achieving the highest Precision at 98%, along with a Recall of 97%, F1-Score of 97%, and Accuracy of 96%. This implies that SVM is slightly more precise than Naïve Bayes in identifying actual spam messages and maintaining low false positive rates. However, it trails slightly behind Naïve Bayes in Recall and F1-Score, indicating a marginally lesser ability to capture all spam instances.

In contrast, KNN records the lowest performance, with a Precision of 94%, Recall of 87%, F1-Score of 91%, and Accuracy of 87%. These results reflect a higher tendency to miss spam messages and a lower classification reliability overall, likely due to its sensitivity to the local structure of the dataset. The results show that Naïve Bayes emerges as the best model for spam email detection in this comparison, offering a balanced and robust

performance across all metrics. SVM is a close contender, particularly in terms of Precision, while KNN underperforms and may not be ideal for this task.

CONCLUSION

This study Based on the findings from Chapter Four (Implementation and Results), the project concludes that Naive Bayes is well-suited for spam email detection due to its simplicity, efficiency in handling text data, and satisfactory performance metrics. The decision to implement Naive Bayes within the Flask framework proved effective in achieving the project objectives of developing a functional and reliable email filtering system.. The evaluation phase highlighted Naive Bayes' strengths in accurately classifying spam emails while maintaining reasonable computational efficiency. However, it also acknowledged areas for potential improvement, such as enhancing model robustness through additional feature engineering and incorporating more diverse datasets for training.

REFERENCES

1. Agarwal, K., & Kumar, T. (2018). Email spam detection using an integrated approach of naïve bayes and particle swarm optimization. Proceedings of the 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), 685–690. IEEE, Madurai, India.
2. Beaman, C., & Isah, H. (2022). Anomaly detection in emails using machine learning and header information. arXiv preprint arXiv:2203.10408.
3. Chemudupati, S., & Valecha, R. (2024). Phishing Email Detection Through Amygdala Hijack Threats. NEAIS 2024 Proceedings, 8. https://aisel.aisnet.org/neais2024/8/
4. Ewees, A. A., Gaheen, M. A., Alshahrani, M., et al. (2024). Improved ML technique for feature reduction and its application in spam email detection. Journal of Intelligent Information Systems.
5. Ferrag, M. A., Maglaras, L., Moschoyiannis, S., & Janicke, H. (2020). Deep learning for cybersecurity intrusion detection: approaches, datasets, and comparative study. Journal of Information Security and Applications, 50, Article ID 102419
6. Ghatasheh, N., Altaharwa, I., & Aldebei, K. (2023). Modified genetic algorithm for feature selection and hyperparameter optimization: Case of XGBoost in spam prediction. arXiv preprint arXiv:2310.19845.
7. Jamal, S., & Wimmer, H. (2024). An Improved Transformer-based Model for Detecting Phishing, Spam, and Ham: A Large Language Model Approach. arXiv preprint arXiv:2311.04913. https://arxiv.org/abs/2311.04913.
8. Kumar, R., & Singh, P. (2023)..). Evaluating machine learning algorithms for spam detection under class imbalance. IEEE Access, 11, 120345–120359.
9. Lee, J., Park, S., & Kim, H. (2024). Revisiting precision–recall trade-offs in email spam filtering. Applied Intelligence, 54(7), 9123–9140:
10. Mutemi, A., & Bacao, F. (2024). E-Commerce Fraud Detection Based on Machine Learning Techniques: Systematic Literature Review. Big Data Mining and Analytics, 7(2), 419–444.
11. Nasreen, G., Khan, M. M., Younus, M., Zafar, B., & Hanif, M. K. (2024). Email spam detection by deep learning models using novel feature selection technique and BERT. Egyptian Informatics Journal, 26, 100473.
12. Reddy, A., Reddy, K. H., Abhishek, A., Manish, M., Dattu, G. V., & Ansari, N. M. (2023). Email spam detection using machine learning. Journal of Survey in Fisheries Sciences, 10(1), 2658-2664.
13. Swetha, P., & Rao, D. S. (2023). Effective feature selection-based meta-heuristics optimization approach for spam detection. SN Computer Science, 4, 681. https://doi.org/10.1007/s42979-023-02126-z
14. Tusher, E. H., Ismail, M. A., Rahman, M. A., Alenezi, A. H., & Uddin, M. (2024). Email spam: A comprehensive review of optimize detection methods, challenges, and open research problems. IEEE Access.
15. Wani, M. A., ElAffendi, M., & Shakil, K. A. (2024). AI-Generated Spam Review Detection Framework…. Computers, 13(10), 264.

16. Zhang, Y., Li, H., & Chen, X. (2025).. Transformer-based spam filtering: A reproducible evaluation across public corpora.Machine Learning-Based Email Spam Filter. Journal of Big Data, 12(1), 45.