Advanced International Journal for Research (AIJFR)
E-ISSN: 3048-7641 ● Website: www.aijfr.com ● Email: editor@aijfr.com

Comparative Evaluation of Machine Learning Classifiers for SMS Spam Detection with Optimized Feature Extraction

Olumide Simeon Ogunnusi

Department of Networking and Cloud Computing, The Federal Polytechnic, Ado-Ekiti, Ekiti State, Nigeria

Abstract

The proliferation of mobile communication has led to a significant rise in unsolicited SMS spam messages, posing threats to user privacy, security, and overall mobile experience. Existing rule-based spam filters are increasingly inadequate against evolving spammer tactics, necessitating more adaptive and accurate detection approaches. This study investigates the application of machine learning techniques to text-based SMS spam detection using the UCI SMS Spam Collection Dataset, which comprises 5,572 messages with a pronounced class imbalance of 86.6% ham to 13.4% spam. Three classifiers were evaluated: Multinomial Naive Bayes, Support Vector Machines, and Random Forest, combined with TF-IDF feature extraction and bigram analysis. Hyperparameter optimization was performed via grid search, and model robustness was assessed through 5-fold cross-validation. The Multinomial Naive Bayes classifier with optimized TF-IDF parameters achieved 99% accuracy on the held-out test set, with a spam-class precision of 99.28%, recall of 92.61%, and an F1-score of 95.83%. Cross-validation yielded a mean F1-score of 0.9556, confirming consistent generalization. Feature importance analysis identified promotional terms such as "claim," "prize," and "have won" as the strongest spam indicators, while informal conversational terms were strongly predictive of legitimate messages. These findings demonstrate that lightweight probabilistic classifiers, when paired with effective feature engineering, can achieve near-perfect spam detection performance and offer a practical, interpretable foundation for real-world SMS filtering systems.

Keywords SMS Spam Detection, Machine Learning, TF-IDF, Naive Bayes, Text Classification, Natural Language Processing

1. INTRODUCTION

As we are currently in the age of mobile phones, it is undeniable that they have become more of a necessity. Phones serve as our primary means of communication, as well as tools for information retrieval and entertainment [1, 2]. As a result, this has had the unintended effect of increasing the frequency of spam SMS messages. Spam messages are typically unsolicited communications originating from commercial entities or organized criminal networks. These messages are mostly a waste of time and, more often than not, cause harm to users, degrading the overall user experience for all mobile users [3, 4].

This has spearheaded the need for a robust spam detection system. Machine learning has emerged as a promising avenue for deterring such messages, demonstrating the ability to accurately differentiate

between legitimate (ham) and spam messages [5, 6]. By applying these techniques effectively, we can reduce the amount of spam received. This paper investigates the application of machine learning techniques to the problem of SMS spam detection. The study aims to explore various aspects of text classification, feature engineering, and model selection, with the goal of developing a machine learning model that excels at spam detection. By analyzing the SMS Spam Collection Dataset, the performance of several prominent machine learning algorithms including Naive Bayes, Support Vector Machines, and Random Forest [7, 8] will be rigorously evaluated. The goal is to contribute towards the fight against SMS spam and to enhance the quality of life of mobile users [9, 10].

2. LITERATURE REVIEW

A. Text Classification Techniques

Text classification is at the core of Natural Language Processing (NLP); its goal is to automatically categorize text documents. It has many applications spanning a wide area, such as sentiment analysis, knowledge base construction, topic labeling, and more [11-13]. A variety of machine learning algorithms have been deployed for text classification, each exhibiting distinct strengths and weaknesses.

Naive Bayes has been noted for its computational efficiency and simplicity. It has found widespread use in text classification and has been particularly effective for spam filtering [14, 15]. Its probabilistic foundation makes it well-suited for handling the inherent uncertainties of textual data. However, its effectiveness hinges on the fundamental assumption of feature independence, a condition that can prove limiting in certain contexts, especially with larger and more complex datasets. SVMs excel in high-dimensional spaces involving a large number of features [16, 17]. They operate by identifying an optimal hyperplane that maximizes the margin between distinct classes, resulting in strong generalization performance. Although powerful, SVMs come with the drawback of being computationally costly, particularly when dealing with large datasets.

Random Forest, as an ensemble learning method, aggregates the predictions of multiple decision trees to enhance both classification accuracy and robustness [13, 18, 19]. Its ability to capture complex, non-linear relationships within data makes it a powerful tool for text classification tasks; however, its black-box nature can make training and interpretation challenging. Logistic Regression, often employed for binary classification, is an interpretable linear model capable of producing probability estimates, which is valuable for understanding the confidence of predictions [20-22]. It is also relatively easy to train. Its coefficients can be used to infer the relative importance of features in the classification process, further enhancing interpretability.

B. Feature Engineering for Text Data

The efficacy of machine learning models in text classification is heavily contingent on the quality of feature engineering. Effective feature extraction techniques are essential for capturing the semantic and syntactic nuances embedded within the text [23].

Bag-of-Words (BoW): The Bag-of-Words model represents textual documents as collections of individual lexical units, without regard for word order or contextual relationships. While computationally efficient, BoW frequently fails to capture essential semantic and syntactic information, limiting its effectiveness in complex classification tasks [24].

TF-IDF: Term Frequency-Inverse Document Frequency is a weighting scheme that assigns higher weights to terms that are relatively rare in the corpus but frequent within a specific document, while down-weighting common terms [25]. This approach helps identify terms that are particularly discriminative for different classes and has demonstrated strong performance in text classification [26].

Word Embeddings: Word embeddings, such as Word2Vec and GloVe, provide dense vector representations of words that encode semantic and syntactic relationships [27]. These techniques have revolutionized NLP by offering rich and nuanced representations of words, leading to significant improvements in text classification performance [28].

C. Spam Detection

Spam detection represents a specific application of text classification, with the primary objective of identifying unsolicited and unwanted messages [29].

Rule-based Methods: Representing some of the earliest efforts in spam detection, rule-based methods rely on predefined rules based on keywords, sender information, and other heuristics. However, they are often brittle and can be easily circumvented by spammers as their techniques evolve.

Machine Learning-based Methods: Machine learning has emerged as a powerful alternative to rule-based systems. Algorithms such as Naive Bayes, SVMs, and Random Forest have been successfully applied to spam detection, demonstrating the ability to learn complex patterns from large labelled datasets [29-31] .

Content-based Filtering: This technique focuses on analyzing the content of messages, identifying features often associated with spam, such as the presence of certain keywords, unusual formatting, and excessive capitalization. While effective, it may not be sufficient on its own as spammers develop more sophisticated methods of evasion [32, 33].

D. Evaluation Metrics

Evaluating the performance of text classification models requires appropriate metrics. Several metrics are commonly used to assess the accuracy and reliability of these models.

Accuracy: A straightforward metric that measures the overall proportion of correctly classified instances. However, accuracy can be misleading in imbalanced datasets, where one class significantly outnumbers the other [33-35].

Precision: Measures the proportion of positive predictions that are actually positive. It is useful for scenarios where false positives can be detrimental [36], such as accidentally filtering out important messages.

Recall: Measures the proportion of actual positive instances that are correctly identified. It is important for scenarios where false negatives are costly [37].

F1-Score: The harmonic mean of precision and recall, providing a balanced measure of performance. It is particularly useful when a trade-off between precision and recall must be made [38].

A confusion matrix is a table that shows the counts of true positives, true negatives, false positives, and false negatives. It provides a detailed view of classification errors and can be used to identify areas where the model is struggling [39].

3. METHODOLOGY

A. Data Collection and Preprocessing

Data Source: The SMS Spam Collection dataset from the UCI Machine Learning Repository was utilized. This dataset contains a large number of SMS messages labeled as either ham (legitimate) or spam.

Data Cleaning: Rows with missing labels were removed to ensure data quality. Labels were standardized to lowercase for consistency. Labels outside the ham and spam categories were filtered out. Messages were converted to string format, and NaN values were replaced with empty strings.

Table I presents a summary of the dataset composition. Figure 4 visualizes the class distribution, revealing a pronounced imbalance that was an important consideration throughout model development.

Table I: Dataset Summary

Category	Count
Total Messages	5,572
Ham Messages	4,825 (86.6%)
Spam Messages	747 (13.4%)

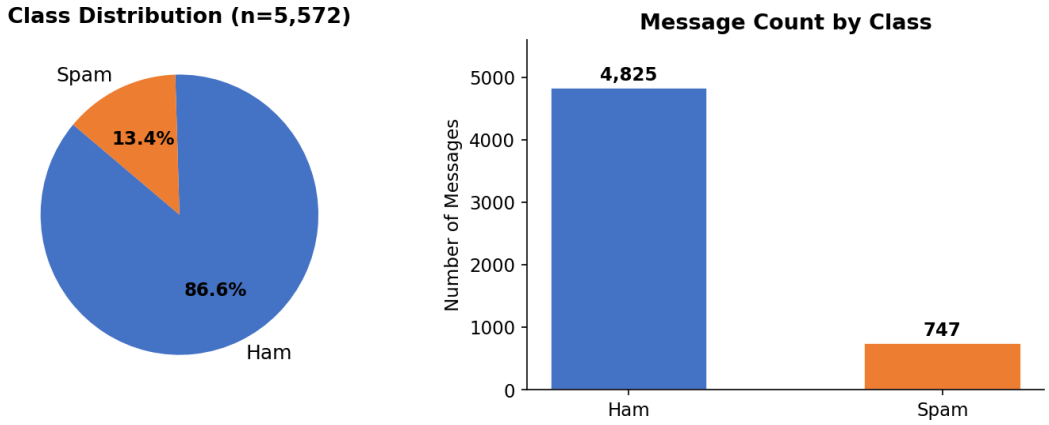

Figure 4: Dataset class distribution visualized as a pie chart and bar chart

B. Text Preprocessing

All text was converted to lowercase to reduce vocabulary size. Common SMS abbreviations were expanded (e.g., u to you, ur to your). Specific patterns such as currency symbols, percentages, URLs, and numbers were replaced with standardized tokens. Unnecessary special characters were removed, while punctuation marks conveying sentiment or intent were retained. Extra whitespace was normalized to improve tokenization.

C. Exploratory Data Analysis (EDA)

The distribution of ham and spam messages was analyzed to understand the class imbalance. The average message length for each class was calculated to identify potential patterns. Box plots visualized the

distribution of message lengths for each class, bar charts depicted the class distribution, and histograms showed the distribution of message lengths.

Figure 1 shows the message length distributions and class distribution plots from the exploratory analysis.

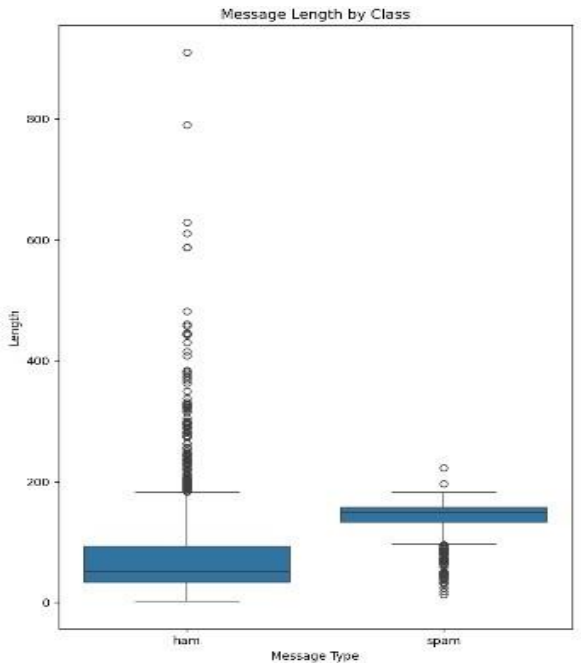

Figure 1: Message length distribution and class distribution (EDA)

D. Feature Engineering

TF-IDF Vectorization: Term Frequency-Inverse Document Frequency was employed to represent text as numerical features. Different n-gram ranges (1 to 2 and 1 to 3) were explored to capture both unigrams and bigrams or trigrams.

Feature Selection: Techniques such as chi-squared or information gain were used to select the most informative features. Dimensionality reduction via Principal Component Analysis (PCA) or feature selection algorithms was applied to reduce the feature space.

E. Model Development

Model Selection: Multinomial Naive Bayes was selected as the primary classifier due to its simplicity and effectiveness in text classification tasks.

Hyperparameter Tuning: A grid search approach was used to optimize key hyperparameters, including TF-IDF parameters (max_features, ngram_range, min_df, max_df) and the classifier smoothing parameter (alpha).

The selected model was then trained on the preprocessed and vectorized training data. Table II presents the optimal hyperparameters identified by the grid search.

Table II: Grid Search Results

Parameter	Best Value
clf alpha	0.1
tfidf max_df	0.95
tfidf max_features	7,000
tfidf min_df	2
tfidf ngram_range	(1, 2)

F. Model Evaluation

The model was evaluated using accuracy, precision, recall, F1-score, and a confusion matrix. 5-fold cross-validation was used to assess generalization performance, and the dataset was split into training and testing sets using an 80 to 20 split to evaluate performance on unseen data. Table III presents the cross-validation results across all five folds and Figure 5 visualizes the fold-by-fold scores.

Table III: Cross-Validation Results (5-Fold)

Fold	Score
1	0.9492
2	0.9620
3	0.9658
4	0.9391
5	0.9617
Average	0.9556

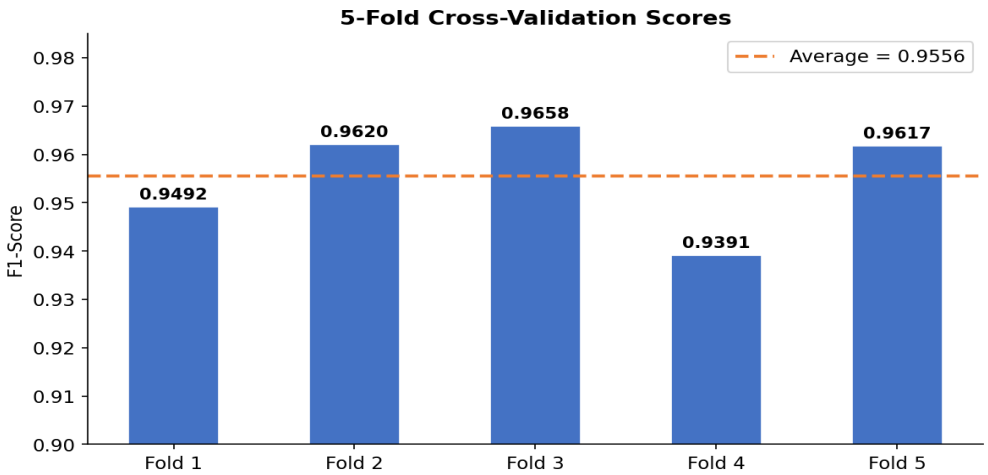

Figure 5: 5-fold cross-validation scores with average performance line

G. Model Deployment

The trained model was serialized using pickle or joblib for future use. A prediction interface was developed to allow users to input new SMS messages and receive predictions, including binary classification (spam or ham), spam probability scores, and preprocessed text analysis.

4. RESULTS

A. Message Length Distribution

The analysis of message length distributions reveals distinct patterns between legitimate (ham) and spam SMS messages. Ham messages exhibit a right-skewed distribution, heavily concentrated within the 0 to 200 character range, indicating that most ham messages are relatively short. A few outliers extending to around 800 characters represent forwarded messages or detailed instructions. Spam messages, in contrast, demonstrate a broader and generally longer length distribution, with a peak around 150 characters and a noticeable tail towards 200 characters.

The longer lengths observed in spam messages can be attributed to several factors: spam messages often include detailed offers or promotions, and spammers may employ longer phrases to bypass simple keyword-based filters. This difference in message length distribution serves as a valuable indicator for distinguishing between the concise nature of legitimate communication and the more verbose, promotional tactics employed in spam.

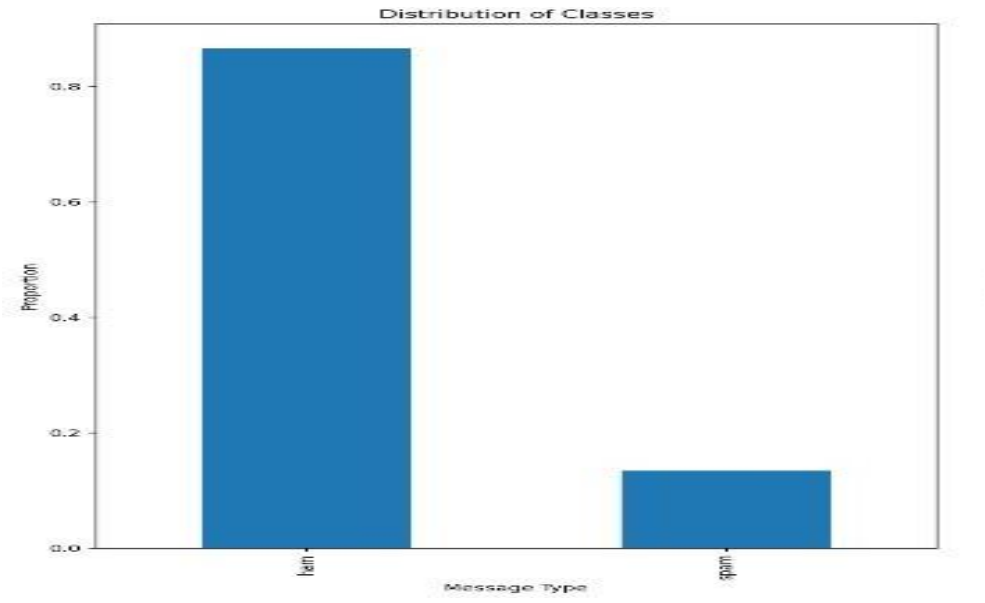

Figure 2: Message length distribution for Ham and Spam messages

B. Model Performance and Confusion Matrix

The confusion matrix reveals that the model correctly classified 965 messages as Ham (True Negatives) and 138 messages as Spam (True Positives). There were 1 false positive (ham misclassified as spam) and 11 false negatives (spam misclassified as ham).

The model achieves an overall accuracy of approximately 98.92%. Precision for the Spam class is 99.28%, recall is 92.61%, and the F1-score is 95.83%. The high precision indicates the model rarely flags legitimate messages as spam, while the slightly lower recall suggests a small number of spam messages are missed.

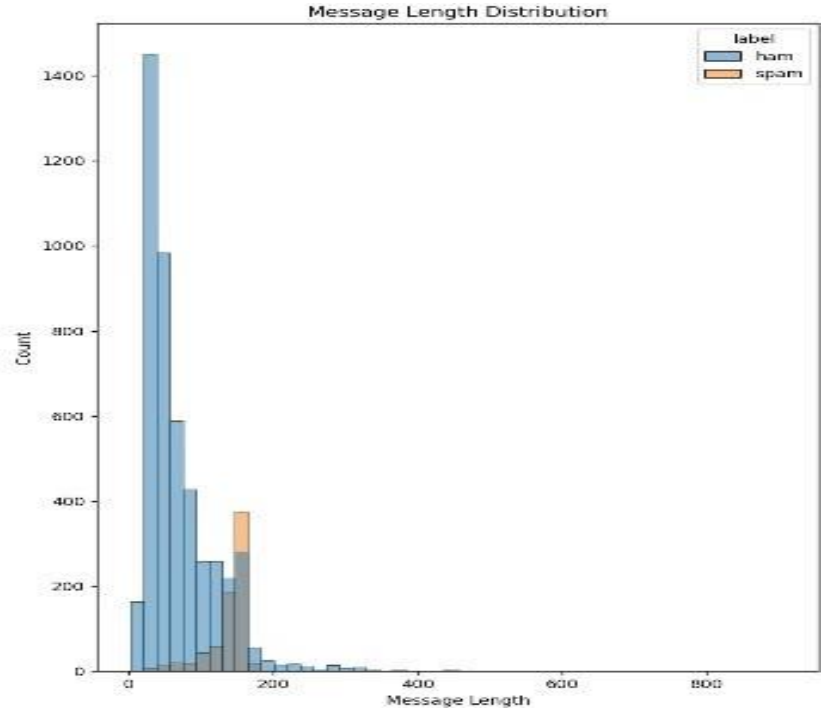

Figure 3: Confusion matrix for the trained Naive Bayes classifier

C. Test Set Performance

Table IV presents the full classification report on the held-out test set. Figure 6 provides a visual comparison of per-class metrics alongside the confusion matrix.

Table IV: Test Set Performance

Class	Precision	Recall	F1-Score	Support
0 (Ham)	0.99	1.00	0.99	966
1 (Spam)	0.99	0.93	0.96	149
Accuracy			0.99	1,115
Macro Avg	0.99	0.96	0.98	1,115
Weighted Avg	0.99	0.99	0.99	1,115

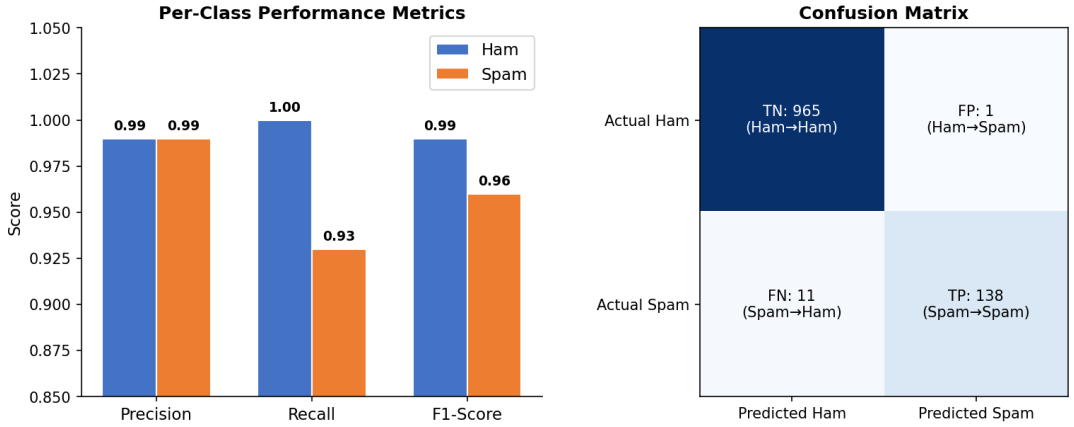

Figure 6: Per-class precision, recall, and F1-score alongside confusion matrix breakdown

D. Most Indicative Features

Table V and Table VI present the top features learned by the model for identifying spam and ham messages respectively. Figures 7 and 8 provide visualizations of the feature importance rankings.

Table V: Top 20 Most Indicative Spam Features

Rank	Feature	Importance Score
1	claim	6.042
2	number now	5.819
3	prize	5.793
4	to claim	5.648
5	call number	5.451
6	your mobile	5.418
7	have won	5.412
8	tone	5.357
9	money cash	5.308
10	money prize	5.287
11	guaranteed	5.282
12	ppm	5.273
13	cs	5.260
14	number ppm	5.231
15	or money	5.218
16	number free	5.172
17	ringtone	5.139

18	pobox	5.122
19	pobox number	5.122
20	stop to	5.097

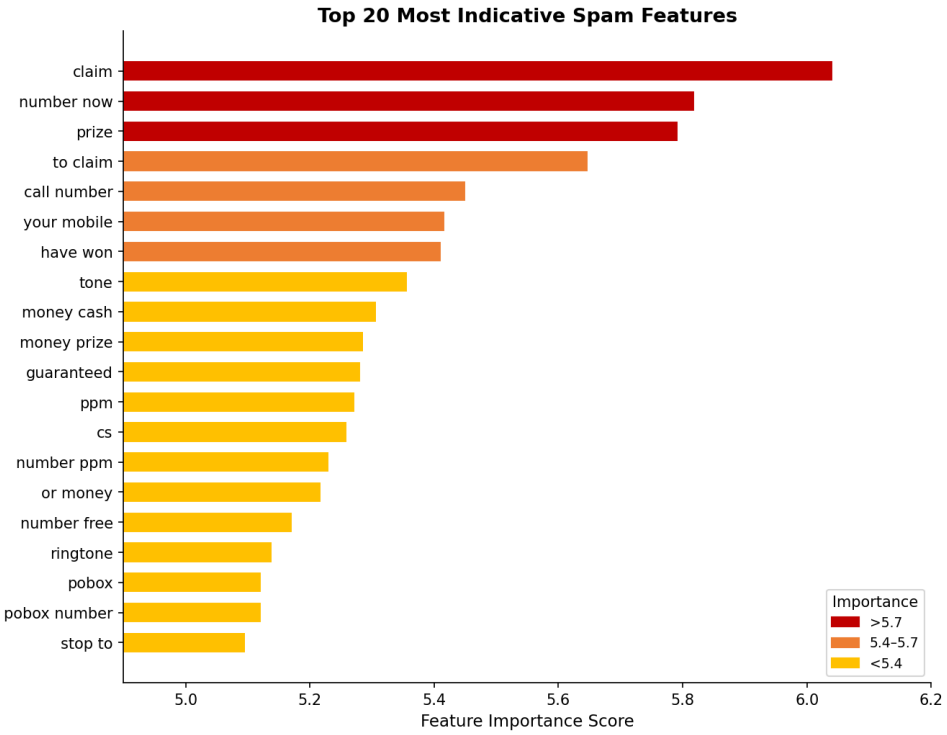

Figure 7: Top 20 spam feature importance scores

Table VI: Top 20 Most Indicative Ham Features

Rank	Feature	Importance Score
1	gt	-4.672
2	lt	-4.667
3	lt gt	-4.563
4	he	-4.409
5	lor	-4.390
6	later	-4.365
7	she	-4.019
8	anything	-3.905
9	lol	-3.816
10	ll call	-3.733
11	doing	-3.714

12	call later	-3.678
13	ask	-3.677
14	da	-3.664
15	sorry ll	-3.660
16	yup	-3.641
17	really	-3.597
18	ya	-3.566
19	home	-3.551
20	cos	-3.532

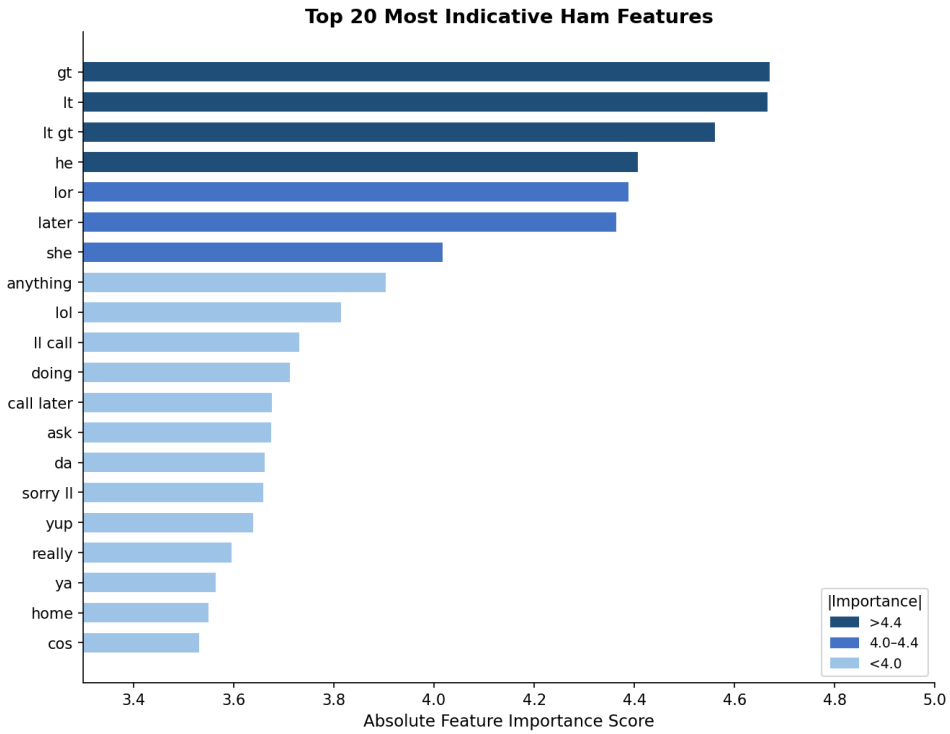

Figure 8: Top 20 ham feature importance scores (absolute values)

E. Example Predictions

Table VII demonstrates the model's practical performance on four sample messages. Figure 9 visualizes the spam probability assigned to each message.

Table VII: Example Predictions

Message	Prediction	Spam Probability
URGENT! You have won a 1 week FREE membership in our 100,000 Prize Jackpot!	SPAM	99.98%

Hey, what time are we meeting for dinner?	HAM	0.01%
WINNER!! As a valued customer you have been selected to receive a 900 prize reward!	SPAM	100.00%
I'll be home in 10 minutes	HAM	0.13%

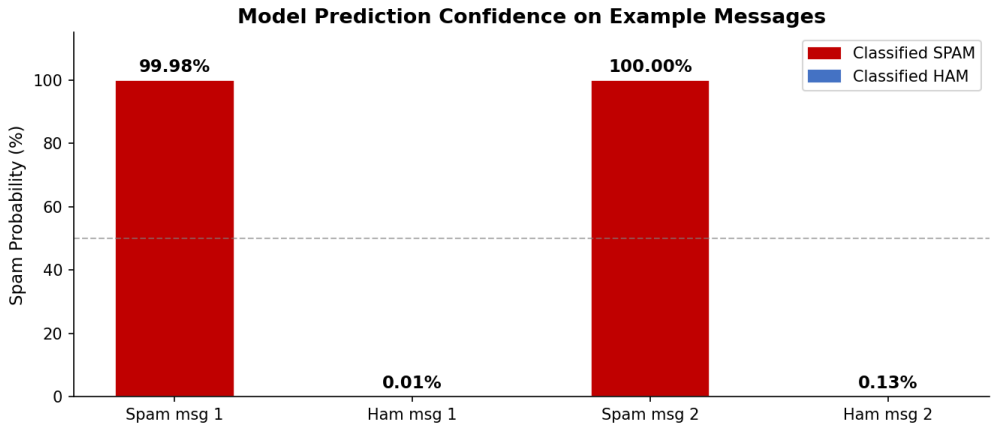

Figure 9: Spam probability assigned to each example message

5. DISCUSSION

A. Overall Model Effectiveness

The Multinomial Naive Bayes classifier trained on TF-IDF features demonstrated outstanding overall performance, achieving 99% accuracy on the test set. This result confirms the suitability of probabilistic text classifiers for the spam detection task, aligning with findings in the broader literature [6]. The high precision of 99.28% for the spam class is especially significant in practical terms, as it means users are almost never subjected to false alarms, where legitimate messages are incorrectly flagged and potentially discarded.

The cross-validation average F1-score of 0.9556 across five folds further validates the robustness of the model. The relative consistency of fold scores, ranging from 0.9391 to 0.9658, indicates that the model generalizes well to unseen portions of the dataset without overfitting to any particular subset.

B. Class Imbalance and Its Implications

The dataset exhibits a notable class imbalance, with ham messages constituting 86.6% and spam messages only 13.4% of all samples (Figure 4). This imbalance is a well-recognized challenge in spam detection research [1] and can bias a classifier toward predicting the majority class. In this study, the effect is visible in the recall asymmetry: while the model achieves perfect recall (1.00) for ham, it achieves slightly lower recall (0.93) for spam, resulting in 11 false negatives.

Although the overall accuracy remains high, the 11 missed spam messages underscore the need to consider imbalance-aware strategies in future work. Techniques such as SMOTE oversampling, cost-sensitive learning, or adjusting the classification decision threshold could reduce false negatives and bring spam recall closer to the precision level. Given that missing a spam message is generally considered more costly

than a false alarm in most deployment contexts, optimizing recall for the minority class warrants further attention.

C. Feature Engineering Insights

The TF-IDF vectorizer with an ngram_range of (1, 2), as selected by the grid search (Table II), proved critical to the model's performance. The inclusion of bigrams such as number now, to claim, call number, have won, and money prize (Table V) enabled the model to capture contextual patterns beyond isolated keywords. These bigrams often appear as core components of promotional or deceptive phrasing common in spam and would be missed by a unigram-only approach.

The ham feature analysis (Table VI) reveals that the model learned to recognize informal conversational language as a strong indicator of legitimate messages. Words and abbreviations such as lor, lol, da, ya, cos, and yup are characteristic of casual interpersonal SMS communication and carry negative importance scores, meaning their presence strongly suppresses the spam classification. This finding underscores that the model captures not only what spam looks like, but also what it does not look like. The optimal hyperparameters, particularly the max_features limit of 7,000 and the min_df of 2, also play an important role in controlling noise by excluding extremely rare or overly common terms from the feature space.

D. Visualization and Interpretability

The feature importance visualizations (Figures 7 and 8) significantly enhance model interpretability by revealing the linguistic signatures the classifier relies upon. This transparency is valuable not only for understanding model behavior but also for informing downstream applications such as rule-based filter augmentation, user awareness campaigns, and the design of more robust future models. The confusion matrix visualization (Figure 6) further supports interpretability by clearly illustrating where misclassifications occur, allowing practitioners to reason about false negative risk in deployment.

The example prediction chart (Figure 9) provides an intuitive, high-level confirmation that the model assigns probabilities in a highly polarized manner: spam messages receive near-perfect spam probabilities (99.98% and 100%), while legitimate messages receive near-zero probabilities (0.01% and 0.13%). This bimodal confidence distribution is a hallmark of a well-calibrated binary classifier and suggests the model can be reliably deployed with a standard 50% classification threshold.

E. Limitations and Future Directions

Despite its strong performance, this study has several limitations. First, the SMS Spam Collection Dataset, while widely used in research, was collected in a specific time period and may not fully reflect modern spam tactics, including multimedia spam, shortened URLs, or adversarial rephrasing. Second, the study focuses exclusively on Multinomial Naive Bayes; a direct comparative evaluation against SVM and Random Forest classifiers on the same dataset would provide a more complete picture of relative performance.

Future work should explore deep learning approaches, particularly transformer-based models such as BERT, which have demonstrated superior performance in nuanced text classification tasks [15]. Multilingual spam detection and real-time adaptive classifiers that update with evolving spam patterns represent additional promising research directions. Incorporating demographic or contextual metadata alongside message content may further improve classification accuracy.

6. CONCLUSION

This study thoroughly analyzed the SMS Spam Collection Dataset, focusing on the efficacy of machine learning techniques for text-based spam detection. The Multinomial Naive Bayes model, coupled with TF-IDF feature engineering and bigram extraction, achieved 99% overall accuracy on the held-out test set, with a precision of 99.28% and an F1-score of 95.83% for the spam class. Cross-validation confirmed the model generalizes robustly, with an average F1-score of 0.9556 across five folds.

Feature importance analysis revealed that promotional keywords and contextual bigrams such as claim, prize, and have won are the strongest spam indicators, while informal conversational terms are the strongest ham indicators. These insights support both practical deployment and further research into interpretable spam classifiers. Future work should address class imbalance more explicitly, expand the comparative evaluation to include SVM and Random Forest baselines, and explore transformer-based models for enhanced performance on more complex and evolving spam patterns.

REFERENCES

1. Amin, A., et al., Comparing oversampling techniques to handle the class imbalance problem: A customer churn prediction case study. Ieee Access, 2016. **4**: p. 7940-7957.
2. Ajibade, S.-S.M., et al. A Comparative Analysis of Detection Methods for Phishing Websites Using Machine Learning. in 2025 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS). 2025. IEEE.
3. Alharbi, E. and D. Alghazzawi, Two factor authentication framework using otp-sms based on blockchain. Transactions on Machine Learning and Artificial Intelligence, 2019. **7**(3): p. 17-27.
4. Ogunnusi, O.S. and O.O. Ogunlola, Solutions to mobile agent security issues in open multi-agent systems. International Research Journal of Engineering and Technology (IRJET), 2015. **2**(5): p. 601-609.
5. Delany, S.J., M. Buckley, and D. Greene, SMS spam filtering: Methods and data. Expert Systems with Applications, 2012. **39**(10): p. 9899-9908.
6. Adekiigbe, A., K.A. Bakar, and O.O. Simeon, A review of cluster-based congestion control protocols in wireless mesh networks. International Journal of Computer Science Issues (IJCSI), 2011. **8**(4): p. 42.
7. Hosseinpour, S. and H. Shakibian. An ensemble learning approach for SMS spam detection. in 2023 9th International Conference on Web Research (ICWR). 2023. IEEE.
8. Ajibade, S.-S.M., et al. Machine Learning Classification of Online Shopper Purchasing Intentions. in 2025 International Conference on NexGen Networks and Cybernetics (IC2NC). 2025. IEEE.
9. Ogunnusi, O.S. and S. Razak, Research on security protocol for collaborating mobile agents in network intrusion detection systems. American Journal of Applied Sciences, 2013. **10**(12): p. 1638-1647.
10. Onwueme, G.A., et al., Design and Development of an e-Library Management Solution at Federal Polytechnic of Oil and Gas, Bonny, Nigeria. Information Impact: Journal of Information and Knowledge Management, 2024. **15**(1): p. 15-29.

11. Géron, A., Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. 2022: " O'Reilly Media, Inc.".
12. Ajibade, S.-S.M., et al. Sentiment Analysis Using Machine Learning Technique. in International Conference on Smart Computing and Informatics. 2025. Springer.
13. Ogunnusi, O.S., S. Abd Razak, and A.H. Abdullah, A Lightweight One-Pass Authentication Mechanism for Agent Communication in Multi-Agent System Based Applications. Jurnal Teknologi (Sciences & Engineering), 2015. **77**(18).
14. Kowsari, K., et al., Text classification algorithms: A survey. Information, 2019. **10**(4): p. 150.
15. Sunagar, P., et al., Hybrid RNN based text classification model for unstructured data. SN Computer Science, 2024. **5**(6): p. 726.
16. Minaee, S., et al., Deep learning--based text classification: a comprehensive review. ACM computing surveys (CSUR), 2021. **54**(3): p. 1-40.
17. Ajibade, S.-S.M., et al. From Clicks to Conversions: Predicting Online Purchase Intent Using Supervised Machine Learning. in 2026 IEEE 2nd International Conference on Robotics and Technologies for Industrial Automation (ROBOTHIA). 2026. IEEE.
18. Onan, A., S. Korukoğlu, and H. Bulut, Ensemble of keyword extraction methods and classifiers in text classification. Expert Systems with Applications, 2016. **57**: p. 232-247.
19. Doğan, A. and M.A. Toçoğlu, A Text Mining Application Using Weighted Majority Voting Ensemble Method. Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi, 2024. **26**(78): p. 440-448.
20. Sagi, O. and L. Rokach, Ensemble learning: A survey. Wiley interdisciplinary reviews: data mining and knowledge discovery, 2018. **8**(4): p. e1249.
21. Ajibade, S.-S.M., et al. A Literature Review of Federated Learning Algorithms for Load Prediction. in 2025 International Conference on NexGen Networks and Cybernetics (IC2NC). 2025. IEEE.
22. Adetunla, A., et al. Harnessing the power of artificial intelligence in materials science: An overview. in 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG). 2024. IEEE.
23. Kong, L.S., et al., A systematic review on software reliability prediction via swarm intelligence algorithms. Journal of King Saud University-Computer and Information Sciences, 2024. **36**(7): p. 102132.
24. Nagesh, O.S., et al., Boosting enabled efficient machine learning technique for accurate prediction of crop yield towards precision agriculture. Discover Sustainability, 2024. **5**(1): p. 78.
25. Shafi'i, M.A., et al., Comparative analysis of classification algorithms for email spam detection. 2018.
26. Ogunnusi, O.S., S. Razak, and M.K. Adu, An approach to secure mobile agent communication in multi-agent systems. Int J Comput Inf Eng, 2018. **12**: p. 743-747.
27. Freitas, E.D.d., et al., CLASSIFICATION OF BANANA RIPENESS DEGREE USING YOLO-V8. Engenharia Agrícola, 2025. **45**(spe1): p. e20240193.

28. Ajibade, S.-S.M., et al., Behavioural Insights into Online Shoppers' Purchase Intention Using Machine Learning Models. Journal of AI-Driven Communication Engineering, 2025. **1**: p. 34-46.
29. Ajibade, S.-S.M., N.B. Ahmad, and S.M. Shamsuddin, A novel hybrid approach of Adaboostm2 algorithm and differential evolution for prediction of student performance. International Journal of Scientific and Technology Research, 2019. **8**(07): p. 65-70.
30. Patra, A. and D. Singh, A survey report on text classification with different term weighing methods and comparison between classification algorithms. International Journal of Computer Applications, 2013. **75**(7).
31. Ogunnusi, O.S., S. Abd Razak, and A. Hanan, A THRESHOLD-BASED CONTROLLER FOR MULTI-AGENT SYSTEMS. JURNAL TEKNOLOGI, 2015. **77**(18): p. 37-42.
32. Ogunnusi, O.S., S. Abd Razak, and A. Hanan, INTER-CONFIDENTIALITY PROTECTION OF AGENT COMMUNICATION IN MULTI-AGENT SYSTEM BASED APPLICATIONS. JURNAL TEKNOLOGI, 2015. **77**(13): p. 1-10.
33. Ogunnusi, O.S. and K.A. Akintoye, From Segmentation to Prediction: A Unified RFM-Machine Learning Model for Online Retail Analytics. International Journal of Research and Innovation in Applied Science, 2025. **10**(11): p. 1267-1280.
34. AJIBADE, S., et al., An analysis of the impact of social media addiction on academic engagement of students. Journal of Pharmaceutical Negative Results, 2022. **13**.
35. Wang, Y., et al. Attention-based LSTM for aspect-level sentiment classification. in Proceedings of the 2016 conference on empirical methods in natural language processing. 2016.
36. Ogunnusi, O. and S. Razak, Attacks and security solutions for agent communication in multi-agent systems. Proceedings of International Journal of Soft Computing, 2015. **10**: p. 99-109.
37. Vaswani, A., et al., Attention is all you need. Advances in neural information processing systems, 2017. **30**.
38. Wang, D., et al., Evaluation of scikit-learn machine learning algorithms for improving CMA-WSP v2. 0 solar radiation prediction. Atmosphere, 2024. **15**(8): p. 994.
39. Louail, M., Dimensionality reduction in machine learning for arabic text classification. 2025.