Formosa Journal of Computer and Information Science (FJCIS)
Vol. 5, No. 1 2026: 111-124

Comparative Analysis of Traditional Machine Learning and Sequential Deep Learning Models for Spam Email Classification

Harliana[1]*, Hartatik[2], Achmad Alvi Yudanuari[3]
[1,3]Universitas Nahdlatul Ulama Blitar
[2]Universitas AMIKOM Yogyakarta
Corresponding Author: Harliana harliana@unublitar.ac.id

A R T I C L E I N F O

Keywords: Spam Classification, TF-IDF, Logistic Regression, RNN, LSTM

Received : 18, November
Revised : 20, January
Accepted: 22, March

A B S T R A C T

This study compares the performance of traditional machine learning methods and sequential deep learning models for text-based spam classification. The primary issue addressed is the lack of consistent, fair evaluation across these approaches due to variations in datasets, preprocessing techniques, and experimental settings across previous studies. To overcome this limitation, this research proposes a controlled comparative evaluation framework by employing a unified dataset, standardized preprocessing procedures, consistent data splitting, and identical evaluation metrics. The dataset used consists of 5,572 messages with an imbalanced class distribution; therefore, oversampling was applied to the training data to mitigate bias. The evaluated models include TF-IDF-based Logistic Regression as the baseline, as well as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRUs) as deep learning models.

DOI prefik: https://doi.org/10.55927/fjcis.v5i1.16502
ISSN-E: 830-3040
https://journal.formosapublisher.org/index.php/fjcis

INTRODUCTION

Spam email classification remains a fundamental problem in text classification, continually evolving as digital communication grows rapidly. Email is widely used not only for personal and professional communication but also as a primary medium for various malicious activities, including spam, phishing, and malware distribution (Qazi et al., 2024). Modern spam has become increasingly sophisticated, characterized by linguistic variation, manipulation of textual structures, and content obfuscation designed to resemble legitimate messages. This increasing complexity poses significant challenges for traditional rule-based approaches, which are inherently limited in their ability to adapt to dynamic and unstructured text patterns. As a result, rule-based methods are becoming less effective in handling the evolving nature of spam detection tasks (Aubaid et al., 2024).

To address these challenges, various machine learning-based approaches have been developed. Traditional methods such as Logistic Regression and Support Vector Machine (SVM) have demonstrated computational efficiency and stable performance across a wide range of text classification tasks, particularly when using statistical feature representations such as TF-IDF (Hasan et al., 2022). However, these approaches have a fundamental limitation: they cannot capture contextual relationships and sequential dependencies within text. This limitation arises from the use of bag-of-words representations, which disregard word order and fail to preserve the inherent sequential structure of textual data (Naulak, 2023).

As an alternative, deep learning approaches based on sequential models, such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTMs), and Gated Recurrent Units (GRUs), have been developed to address these limitations (Bansal et al., 2022). These models are designed to capture both short- and long-term dependencies in textual data, enabling a more effective understanding of linguistic context than traditional methods (Mao et al., 2023). However, empirical studies suggest that the superiority of deep learning models is not always consistent, particularly when applied to datasets with short text formats such as SMS or brief email messages (Guleria et al., 2025). Furthermore, sequential models typically require larger datasets and higher computational resources (Bansal et al., 2022), which may limit their practicality and make them less optimal in certain application scenarios.

A key issue identified in the literature is the lack of consistent and fair comparative evaluation between traditional machine learning methods and deep learning models (Al-augby et al., 2025). Many existing studies employ different datasets, preprocessing techniques, class imbalance handling strategies, and evaluation metrics, resulting in heterogeneous experimental settings (Al-augby et al., 2025). These inconsistencies make it difficult to perform direct comparisons and limit the ability to draw robust conclusions regarding the relative effectiveness of each approach. Consequently, there remains no clear consensus on which method performs best in spam classification tasks, particularly for short-text data with imbalanced class distributions.

In addition, the issue of class imbalance is often overlooked or handled inconsistently, despite being a common characteristic of spam datasets (Aguiar et al., 2024). This imbalance can introduce bias in model performance, particularly when relying on accuracy as an evaluation metric, which may lead to misleading conclusions (Altalhan et al., 2025), Therefore, appropriate handling strategies are required to ensure reliable evaluation. Techniques such as oversampling must be applied carefully, especially within the training data, to mitigate class imbalance while avoiding potential data leakage that could compromise the validity of experimental results (Agrawal & Goyal, 2022).

Based on the identified research gaps, this study proposes a structured and consistent comparative evaluation framework for spam email classification. The proposed framework integrates TF-IDF-based text representation with Logistic Regression, a classical baseline model. In addition, sequential deep learning models, including Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTMs), and Gated Recurrent Units (GRUs), are employed to capture contextual patterns in textual data. To address class imbalance, an oversampling strategy is applied exclusively to the training data to preserve the validity of the evaluation process (Agrawal & Goyal, 2022). Furthermore, the framework ensures a fair comparison by utilizing a unified dataset, consistent preprocessing procedures, and standardized evaluation metrics across all models (Al-augby et al., 2025).

Through this approach, the study not only compares model performance but also provides a more objective analysis of the trade-offs among computational efficiency, text representation capability, and performance stability across different approaches. Accordingly, the main contribution of this research lies in providing a fair and controlled empirical evaluation of traditional machine learning methods and sequential deep learning models for spam classification on short-text data with imbalanced class distributions.

LITERATURE REVIEW

Traditional machine learning approaches such as Naïve Bayes, Logistic Regression, and Support Vector Machine (SVM) have been widely applied in spam classification due to their computational efficiency and ability to handle high-dimensional data. These methods typically rely on statistical text representations, including bag-of-words, n-grams, and TF-IDF, to transform textual data into numerical vectors suitable for model processing. A study by Kowsari et al. highlights that model performance is highly dependent on the quality of feature engineering, where appropriate feature selection can significantly improve classification accuracy (Kowsari et al., 2019). However, these approaches exhibit a fundamental limitation in their inability to capture contextual relationships and word order within text, as the underlying representations treat words independently of their positions. This limitation reduces the effectiveness of traditional models in handling linguistically complex text, such as modern spam messages that involve variations in language and structure. Therefore, there is a need for approaches that explicitly consider the sequential structure of text, as addressed in this study through the use of sequential deep learning models within a unified evaluation framework.

With the advancement of deep learning techniques, models such as Recurrent Neural Networks (RNNs) have been introduced to address the limitations of traditional approaches. RNNs are designed to process sequential data by maintaining information from previous time steps through hidden states, enabling the model to capture sequential patterns in text that TF-IDF-based models cannot effectively represent. However, as reported by Sherstinsky, RNNs suffer from the vanishing gradient problem, which limits their ability to learn long-term dependencies. This limitation negatively affects model performance, particularly when handling text with extended context or complex sentence structures (Sherstinsky, 2020). Therefore, further evaluation is required to assess the extent to which this limitation impacts spam classification tasks, especially in the context of short-text data, which constitutes a primary focus of this study.

To address the limitations of RNNs, advanced variants such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) were developed, incorporating gating mechanisms to regulate the flow of information. LSTM employs a more complex architecture with input, forget, and output gates to retain relevant information over long sequences, whereas GRU offers a simpler structure with fewer parameters, resulting in a more computationally efficient model. According to Sherstinsky, both LSTM and GRU are effective in mitigating the vanishing gradient problem and improving the ability to capture long-term dependencies. However, the increased architectural complexity results in higher computational requirements than those of traditional machine learning models. Moreover, the performance gains achieved by these models are not always significant across all types of datasets, particularly in short-text data with relatively simple patterns (Sherstinsky, 2020). This observation indicates that the superiority of deep learning models is context-dependent and strongly influenced by data characteristics. Therefore, this study evaluates LSTM and GRU under the same experimental conditions as traditional methods, enabling an objective and fair comparison.

In the context of spam detection, several studies have reported that deep learning models, particularly LSTM, can achieve high performance in capturing complex textual variations. For instance, Utomo et al. reported that LSTM achieved an accuracy of up to 99.35% in email spam classification (Utomo et al., 2024). However, such results should be interpreted with caution. High accuracy does not necessarily reflect balanced model performance, especially when applied to imbalanced datasets, where models tend to be biased toward the majority class. Furthermore, variations in datasets, preprocessing techniques, and evaluation metrics across studies make it difficult to directly compare these results with alternative approaches. Therefore, this study not only evaluates accuracy but also employs more representative performance metrics, including precision, recall, and F1-score, within a consistent experimental framework.

On the other hand, studies such as Siddique et al. demonstrate that traditional machine learning methods can still achieve competitive performance, particularly when applied to datasets with limited size or relatively simple text patterns. The primary advantages of these approaches lie in their computational efficiency and performance stability (Siddique et al., 2021). However, their effectiveness depends heavily on the feature engineering process, which is typically manual and may not fully capture complex linguistic variations. This limitation highlights the importance of conducting a direct comparison between traditional methods and deep learning models under identical experimental conditions. Such an evaluation enables a more reliable assessment of which approach is better suited to specific data characteristics.

In addition, Rahimzad et al. reported that although deep learning models such as LSTM and RNN can achieve high performance, the improvement is not always significant when applied to small-scale datasets. This finding indicates that model effectiveness is strongly influenced by both data size and textual complexity (Rahimzad et al., 2021). In other words, no single method consistently outperforms others across all conditions. Therefore, a systematic and controlled comparative evaluation is required to minimize bias in result interpretation and to provide a more reliable assessment of model performance.

Based on the reviewed studies, it can be concluded that a significant research gap persists in the lack of fair and consistent evaluation of traditional machine learning methods and deep learning models for spam classification. In addition, the issue of class imbalance is often handled inconsistently, which may affect the validity of evaluation results. To address these limitations, this study proposes an integrated comparative evaluation approach that employs a unified dataset, consistent preprocessing procedures, and a class-imbalance handling strategy via oversampling applied exclusively to the training data to prevent data leakage. Through this approach, the study aims to provide a more objective empirical analysis of the strengths and limitations of each method in the context of spam classification on short-text data.

METHODOLOGY

This study adopts a comparative experimental approach to evaluate the performance of traditional machine learning methods and sequential deep learning models in text-based spam classification. The entire process is conducted in a structured manner, encompassing data acquisition, preprocessing, dataset splitting, handling class imbalance, feature representation, model development, and performance evaluation. This systematic pipeline is designed to ensure a fair and consistent comparison across all models, enabling a reliable assessment of their respective strengths and limitations under identical experimental conditions.

The dataset used in this study is a publicly available SMS spam dataset consisting of 5,572 messages, with a class distribution of 4,825 non-spam (ham) messages and 747 spam messages. Each data instance is represented as a pair (xi ,yi), where x_i denotes the i-th message text and ($y_i \in 0,1$) represents the class label, with 0 indicating ham and 1 indicating spam. During the preprocessing stage, the raw text x_i is transformed into a cleaned version$x_i = f(x_i)$through a series of text normalization steps. These steps include lowercasing, removal of non-alphabetic characters, whitespace normalization, and stopword removal. The objective of this process is to reduce noise and improve the quality of text representation for subsequent modeling.

The dataset is subsequently divided into training and testing sets using an 80:20 ratio, as defined in Equation (1). This partitioning ensures that the model is trained on the majority of the data while maintaining a separate subset for unbiased performance evaluation.

$$\boldsymbol{D} = \boldsymbol{D_{train}} \cup \boldsymbol{D_{test}} \quad (1)$$

Subsequently, during the deep learning model training process, the training set is further partitioned into training and validation subsets using an 80:20 ratio through a validation split mechanism, as defined in Equation (2). This strategy enables the monitoring of model performance during training and helps prevent overfitting by evaluating the model on unseen validation data.

$$\boldsymbol{D} = \boldsymbol{D_{train}^{fit}} \cup \boldsymbol{D_{val}} \quad (2)$$

Due to the imbalanced class distribution $\left(\left(|D_{ham}| \gg |D_{spam}|\right)\right)$, an oversampling technique is applied to the training data to balance the class proportions. Let (N_h) a denote the number of ham instances and (N_s) denote the number of spam instancesdata spam, The oversampling process adjusts the minority class such that $N_s' = N_h$ resulting in a balanced training dataset, as defined in Equation (3). This approach ensures that the model is exposed to an equal number of samples from each class during training, thereby reducing bias toward the majority class while maintaining the integrity of the testing process.

$$\boldsymbol{D'_{train}} = \boldsymbol{D_{ham}} \cup \boldsymbol{D_{spam}^{upsample}} \quad (3)$$

The oversampling process is applied exclusively to the training data to prevent data leakage into the testing set.

Feature representation is performed using two different approaches. For traditional machine learning models, a TF-IDF-based representation is employed, as defined in Equation (4).

$$\boldsymbol{TF-IDF_{(t,d)}} = \boldsymbol{TF_{(t,d)}}.\boldsymbol{log}\left(\frac{N}{\boldsymbol{df_{(t)}}}\right) \quad (4)$$

In this formulation $t$denotes a term, $d$represents a document, $N$is the total number of documents, and ($df_{(t)}$) indicates the number of documents containing term t. The transformation process produces a feature vector ($x \in \mathbb{R}^n$) where n represents the dimensionality of the feature space.

For deep learning models, each text instance x_i is transformed into a sequence of tokens $x_i \rightarrow (\omega_1, \omega_2, \ldots, \omega_n)$ To ensure uniform input dimensions, all sequences are padded to a fixed maximum length (T = 100), resulting in the representation $x_i = (\omega_1, \omega_2, \ldots, \omega_T)$, where shorter sequences are padded with zeros. This process enables efficient batch processing and ensures compatibility with sequential neural network architectures.

The first model employed in this study is Logistic Regression, which models the probability of the target class using the sigmoid function in Equation (6), as defined in Equation (5). The model estimates the probability that a given input belongs to the spam class based on a linear combination of input features.

$$\boldsymbol{P_{(y=1|x)} = \sigma(w^T x + b)} \quad (5)$$

$$\boldsymbol{\sigma_{(z)} = \frac{1}{1 + e^{-z}}} \quad (6)$$

For the deep learning models, an embedding layer is utilized to map each token into a dense vector representation of dimension 64, denoted as ($e_t \in \mathbb{R}^{64}$) Consequently, each input sequence is transformed into a matrix ($X \in \mathbb{R}^{Tx64}$), where Trepresents the maximum sequence length

The Recurrent Neural Network (RNN) model computes the hidden state at each time step as defined in Equation (7), where the current hidden state is influenced by both the current input and the previous hidden state.

$$\boldsymbol{h_t = tanh\,(W_h h_t - 1 + W_x x_t)} \quad (7)$$

The Long Short-Term Memory (LSTM) model uses gating mechanisms to regulate the flow of information, as defined in Equation (8), where the input, forget, and output gates collectively control the retention and propagation of relevant information across time steps.

$$\boldsymbol{f_t = \sigma\big(W_f[h_t - 1, x_t]\big),\ \ i_t = \sigma(W_i[h_t - 1, x_t]),\ \ o_t = \sigma(W_o[h_t - 1, x_t])} \quad (8)$$

The Gated Recurrent Unit (GRU) model employs a gating-based formulation, as defined in Equation (9), in which the update and reset gates regulate the flow of information and control the integration of past and current inputs.

$$\boldsymbol{z_t = \sigma(W_z[h_t - 1, x_t]),\ \ h_t = (1 - z_t)h_t - 1 + z_t h_t} \quad (9)$$

All models produce an output probability$y = \sigma(z)$ which is subsequently converted into a class label as defined in Equation (10), typically using a threshold (e.g., 0.5) to distinguish between spam and non-spam classes.

$$\boldsymbol{y_{class} = \begin{cases} 1 \ \ if\ y \geq 0,5 \\ 0 \ \ otherwise \end{cases}} \quad (10)$$

The loss function used in this study is binary cross-entropy (Equation (11), which measures the discrepancy between the predicted probabilities and the true class labels in binary classification tasks. This loss function penalizes incorrect predictions more heavily when the predicted probability deviates significantly from the true label.

$$\boldsymbol{\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}[y_i log(y_i) + (1 - y_i) log(1 - y_i)]} \quad (11)$$

Model performance is evaluated using a confusion matrix, which serves as the basis for calculating classification metrics. Based on the confusion matrix, performance is further evaluated using accuracy, precision, recall, and F1-score to provide a comprehensive assessment of classification effectiveness.

All models are assessed under identical experimental conditions, including the same dataset, preprocessing procedures, data splitting strategy, and evaluation metrics, ensuring a fair and controlled comparison. This approach ensures that any observed differences in performance are attributable to the models' inherent characteristics rather than to variations in experimental configuration.

RESEARCH RESULT

The experiments were conducted to compare the performance of TF-IDF-based Logistic Regression with three sequential deep learning models: Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). The dataset consists of 5,572 messages, with a class distribution of 4,825 non-spam (ham) and 747 spam messages, indicating a significant class imbalance where ham messages dominate. Following an 80:20 data split, the test set comprises 1,115 messages, including 966 ham and 149 spam instances. To mitigate bias toward the majority class, oversampling is applied to the training data, thereby balancing the class distribution. Table 1 presents the evaluation results of all models.

Table 1. Model Evaluation Results

Model	Accuracy	Precision Spam	Recall Spam	F1-score Spam
Logistic Regression	0,98	0,93	0,95	0,94
RNN	0,99	0,99	0,92	0,95
LSTM	0,99	0,99	0,94	0,97
GRU	0,99	0,99	0,94	0,97

The evaluation results indicate that all models achieve very high performance, with accuracy ranging from 0.98 to 0.99. The TF-IDF-based Logistic Regression model achieves an accuracy of 0.98, with a spam precision of 0.93, a recall of 0.95, and an F1-score of 0.94. Meanwhile, the RNN, LSTM, and GRU models each achieve an accuracy of 0.99. However, a more detailed analysis reveals notable differences in performance characteristics, particularly in detecting the spam class.

Logistic Regression demonstrates strong performance in identifying spam, with a recall of 0.95, indicating that most spam messages are correctly detected. However, its lower precision (0.93) suggests the presence of false positives, where ham messages are incorrectly classified as spam. Based on the confusion matrix, Logistic Regression produces 11 false positives and 8 false negatives. This indicates that TF-IDF-based models rely heavily on word frequency, making them susceptible to misclassification when spam and ham share similar lexical patterns.

In contrast, the RNN model achieves a very high precision of 0.99, indicating that nearly all predicted spam messages are correct. However, its recall is lower at 0.92, meaning that some spam messages are missed. Numerically, RNN produces only 1 false positive but 12 false negatives. This suggests that RNN exhibits a conservative behavior in spam detection, prioritizing precision over recall, which reduces false alarms but increases the risk of undetected spam.

The LSTM and GRU models demonstrate the best overall performance in this study. Both models achieve a precision of 0.99, a recall of 0.94, and an F1-score of 0.97. According to the confusion matrix, they produce only 1 false positive and 9 false negatives. These results indicate that LSTM and GRU achieve a better balance between precision and recall, making them more effective for spam classification. This advantage stems from the gating mechanisms in LSTM and GRU, which enable the models to retain relevant information across sequences and capture contextual patterns that are not accessible to TF-IDF-based models or simple RNNs.

In addition to metric-based evaluation, analyzing training and validation accuracy curves provides further insight into the models' learning behavior. The accuracy curves for RNN, LSTM, and GRU are presented in Figures 1, 2, and 3, respectively. All models exhibit a rapid increase in accuracy from the first epoch, indicating that the dataset's underlying patterns are relatively easy to learn.

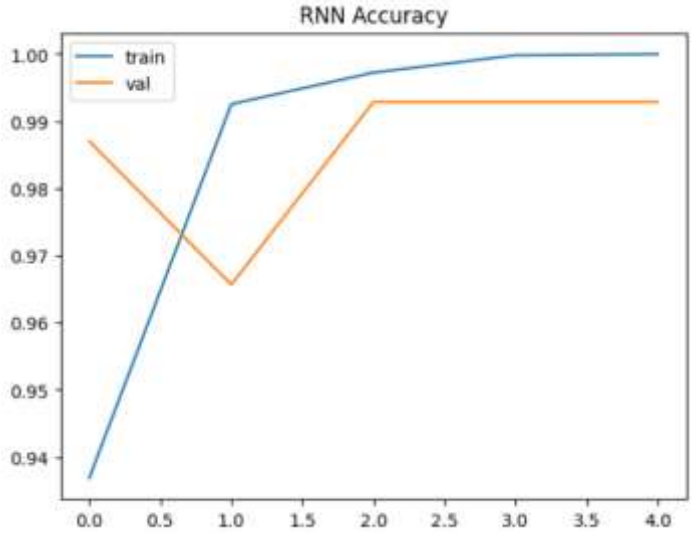

Figure 1. Training and Validation Accuracy of the RNN Model

As shown in Figure 1, the training accuracy of the RNN model increases from approximately 0.93 to 1.00 over five epochs, while the validation accuracy improves from approximately 0.98 to 0.995. However, after the third epoch, a noticeable gap emerges between training and validation accuracies, suggesting mild overfitting. This behavior suggests that the model begins to memorize the training data rather than generalizing effectively to unseen data.

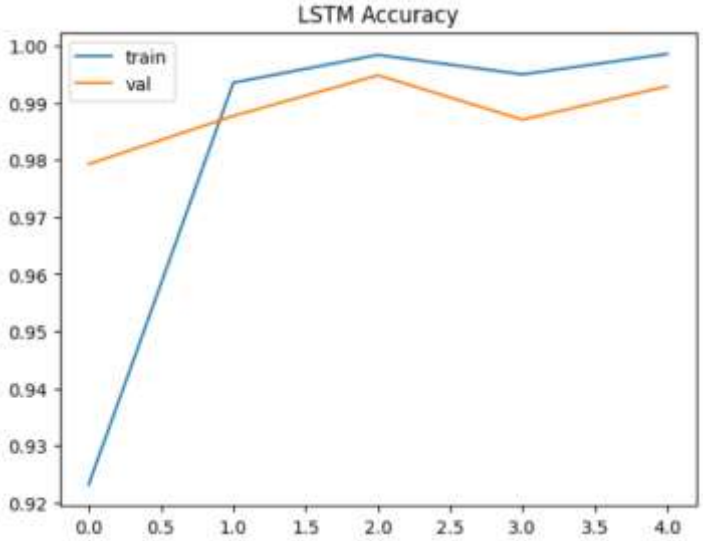

Figure 2. Training and Validation Accuracy of the LSTM Model

In Figure 2, the LSTM model exhibits a more stable learning pattern. The training accuracy increases from approximately 0.92 to nearly 1.00, while the validation accuracy improves to around 0.993 and then stabilizes after the third epoch. This behavior indicates that the model converges relatively early and does not achieve significant performance improvement with additional training epochs. The stability of the validation accuracy suggests that the model generalizes well and is less prone to overfitting compared to the standard RNN

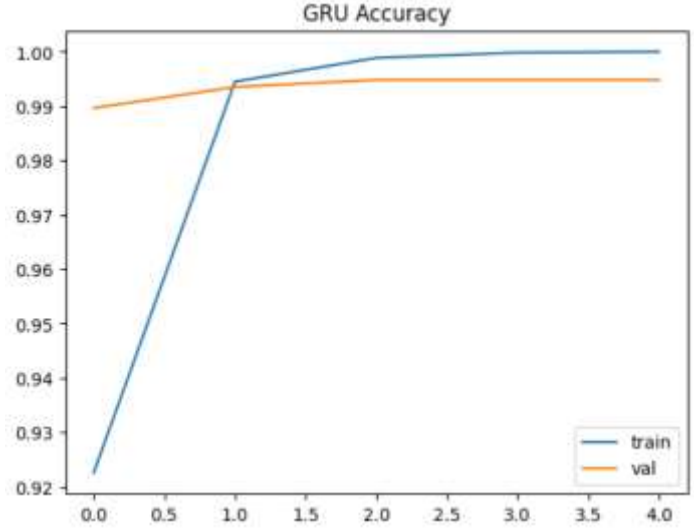

Figure 3. Training and Validation Accuracy of the GRU Model

As shown in Figure 3, the GRU model exhibits a learning pattern similar to that of LSTM, but with slightly better stability in the validation curve. The validation accuracy increases to approximately 0.995 by the third epoch and remains stable throughout the rest of the training. This behavior indicates that the GRU model achieves consistent generalization performance and maintains stability during training, suggesting its effectiveness in learning relevant patterns from the data. Compared to LSTM, the GRU model achieves comparable performance with a simpler architecture, potentially enabling more efficient training and stable convergence.

Overall, deep learning models demonstrate rapid learning and converge within a relatively small number of epochs. However, LSTM and GRU exhibit better stability than RNNs, as indicated by a smaller gap between training and validation accuracy. Although the training accuracy approaches 1.00 across all models, this result should be interpreted with caution, as it may be influenced by oversampling, which introduces duplicate instances in the minority class. Therefore, evaluation on the test dataset is essential to ensure that the models do not merely memorize the training data but can generalize effectively.

A comparison between traditional and deep learning approaches shows that Logistic Regression remains a strong baseline, with performance only slightly lower than that of deep learning models. This suggests that TF-IDF representation remains effective for spam classification, particularly because many spam messages contain explicit lexical patterns. However, deep learning models demonstrate a clear advantage in reducing false positives, decreasing from 11 in Logistic Regression to only 1 in RNN, LSTM, and GRU. This indicates that sequential models are more selective and better at capturing contextual information.

From a contribution perspective, this study does not aim to propose a novel model architecture but rather to provide a structured, controlled evaluation framework. By employing the same dataset, consistent preprocessing procedures, an appropriate oversampling strategy, and standardized evaluation metrics, this study enables a more objective comparison between traditional machine learning methods and deep learning models.

Overall, the experimental results indicate that LSTM and GRU achieve the best performance in spam classification on the given dataset, with F1 scores of 0.97. While RNN also demonstrates strong performance, it has limitations in spam recall. Logistic Regression remains competitive and can serve as a practical alternative in scenarios with limited computational resources. These findings highlight that model selection should not rely solely on accuracy, but also consider the balance between precision and recall, as well as the specific requirements of the application.

DISCUSSION

The experimental results demonstrate that all models achieve high classification performance, with accuracy values ranging from 0.98 to 0.99. However, a deeper examination reveals that accuracy alone is insufficient to distinguish model effectiveness, particularly in the presence of class imbalance. Therefore, the analysis focuses on precision, recall, and F1-score for the minority class (spam), which provides a more reliable assessment of model behavior.

One important observation is the trade-off between precision and recall across different model architectures. Logistic Regression achieves a relatively high recall (0.95), indicating strong capability in identifying spam messages. However, its lower precision (0.93) suggests a tendency to misclassify legitimate messages as spam. This behavior can be explained by the nature of TF-IDF representation, which relies on term frequency and lacks contextual understanding. As a result, messages that share similar lexical patterns with spam may be incorrectly classified.

In contrast, the RNN model exhibits an opposite behavior, achieving very high precision (0.99) but lower recall (0.92). This indicates that the model is more conservative in predicting spam, minimizing false positives at the cost of missing some actual spam messages. From a modeling perspective, this behavior reflects the sequential nature of RNN, where the model captures temporal dependencies but may struggle to retain long-range contextual information due to the vanishing gradient problem.

The superior performance of LSTM and GRU can be attributed to their gating mechanisms, which allow selective retention and forgetting of information. These mechanisms enable both models to effectively capture contextual dependencies within text sequences, thereby improving the balance between precision and recall. As observed, both models achieve an F1-score of 0.97, indicating optimal trade-off performance. Compared to the standard RNN, the improved recall in LSTM and GRU suggests that gating structures play a crucial role in detecting more complex spam patterns.

Another critical aspect influencing model performance is the application of oversampling. While oversampling helps address class imbalance, it also introduces duplicated instances in the training data. This can lead to overly optimistic estimates of training accuracy, as observed in all deep learning models achieving near-perfect training performance. However, the consistency between the training and validation curves for LSTM and GRU indicates that these models exhibit better generalization than RNN, which shows a larger gap and signs of mild overfitting.

From a comparative perspective, the results indicate that traditional machine learning methods, such as Logistic Regression, remain highly competitive, especially considering their simplicity and low computational cost. This finding is consistent with prior studies showing that TF-IDF features are highly effective for text classification tasks with strong lexical signals. However, deep learning models demonstrate a clear advantage in reducing false positives, particularly in real-world spam detection systems, where incorrect filtering of legitimate messages can negatively impact the user experience.

CONCLUSIONS AND RECOMMENDATIONS

This study proposes a structured and controlled comparative evaluation framework to assess traditional machine learning methods and sequential deep learning models for spam classification. By employing a unified dataset, consistent preprocessing procedures, standardized data splitting, and identical evaluation metrics, the proposed framework ensures a fair and objective comparison, effectively addressing the gap in prior studies that lack consistent evaluation settings.

The experimental results demonstrate that all models achieve high performance, with accuracy exceeding 0.98. TF-IDF-based Logistic Regression remains a strong baseline, achieving an F1-score of 0.94 for the spam class. However, deep learning models improve prediction balance, particularly by reducing false positives and increasing precision. Among the evaluated models, LSTM and GRU achieve the best performance, with an F1-score of 0.97, highlighting their effectiveness in capturing contextual patterns in textual data.

Overall, the findings indicate that sequential deep learning models are more effective for spam classification on the evaluated dataset, although traditional methods remain competitive under certain conditions. Therefore, model selection should not be based solely on accuracy, but should also consider the trade-offs between precision, recall, computational complexity, and the specific requirements of the application.

Despite these contributions, several limitations should be acknowledged. First, the experiments are conducted on a single dataset with short-text characteristics, limiting the generalizability of the findings. Second, the use of oversampling on the training data may introduce a risk of overfitting, although no significant indication is observed in validation and testing results. Third, this study does not include more advanced architecture such as transformer-based models (e.g., BERT), which may offer superior performance.

ADVANCED RESEARCH

Future work should explore more diverse datasets and extend the comparative analysis to include transformer-based approaches.

REFERENCES

Agrawal, E. G., & Goyal, S. J. (2022). Survey on Data Leakage Prevention through Machine Learning Algorithms. International Mobile and Embedded Technology Conference (MECON) Survey, 2020–2022. https://doi.org/10.1109/MECON53876.2022.9752047.

Aguiar, G., Krawczyk, B., & Cano, A. (2024). A survey on learning from imbalanced data streams : taxonomy, challenges, empirical study, and reproducible. In Machine Learning (Vol. 113, Issue 7). Springer US. https://doi.org/10.1007/s10994-023-06353-6.

Ahmed, N., Amin, R., Aldabbas, H., Koundal, D., Alouffi, B., & Shah, T. (2022). Machine Learning Techniques for Spam Detection in Email and IoT Platforms : Analysis and Research Challenges. Security and Communication Networks, 2022. https://doi.org/10.1155/2022/1862888.

Al-augby, S., Alyasiri, H., Abdulkadhim, F. G., & Oleiwi, Z. C. (2025). A Stacked Ensemble Classifier for Email Spam Detection via an Evolutionary Algorithm. Mesopotamian Journal of Cybersecurity, 5(2), 657–670. https://doi.org/10.58496/MJCS/2025/039.

Altalhan, M., Algarni, A., & Alouane, M. T. (2025). Imbalanced Data Problem in Machine Learning : A Review. IEEE Access, 13(December 2024), 13686–13699. https://doi.org/10.1109/ACCESS.2025.3531662.

Aubaid, A. M., Mishra, A., & Mishra, A. (2024). Machine learning and rule - based embedding techniques for classifying text documents. International Journal of System Assurance Engineering and Management, 15(12), 5637–5652. https://doi.org/10.1007/s13198-024-02555-w.

Bansal, M., Goyal, A., & Choudhary, A. (2022). A comparative analysis of K-Nearest Neighbor, Genetic, Support Vector Machine, Decision Tree, and Long Short Term Memory algorithms in machine learning. Decision Analytics Journal, 3(November 2021), 100071. https://doi.org/10.1016/j.dajour.2022.100071.

Guleria, P., Frnda, J., & Srinivasu, P. N. (2025). NLP based text classification using TF-IDF enabled fine-tuned long short-term memory : An empirical analysis. Array, 27(July), 100467. https://doi.org/10.1016/j.array.2025.100467.

Hasan, S. U., Ahamed, J., & Ahmad, K. (2022). Analytics of machine learning-based algorithms for text classification. Sustainable Operations and Computers, 3(July 2021), 238–248. https://doi.org/10.1016/j.susoc.2022.03.001.

Kowsari, K., Meimandi, K. J., Heidarysafa, M., & Mendu, S. (2019). Text Classification Algorithms : A Survey. Information, 10(4), 1–68. https://doi.org/10.3390/info10040150.

Mao, S., Member, G. S., & Sejdi, E. (2023). A Review of Recurrent Neural Network-Based Methods in Computational Physiology. IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, 34(10), 1–21. https://doi.org/10.1109/TNNLS.2022.3145365.

Naulak, C. (2023). A comparative study of Naive Bayes Classifiers with improved technique on Text Classification. TechRxiv, 1(May), 0–8. https://doi.org/10.36227/techrxiv.19918360.v1.

Qazi, A., Hasan, N., Mao, R., Mohamed Abo, M. E., Dey, S. K., & Hardaker, G. (2024). Machine Learning-Based Opinion Spam Detection : A Systematic Literature Review. IEEE Access, 12(May), 143485–143499. https://doi.org/10.1109/ACCESS.2024.3399264.

Rahimzad, M., Moghaddam, A., & Hosam, N. (2021). Performance Comparison of an LSTM - based Deep Learning Model versus Conventional Machine Learning Algorithms for Streamflow Forecasting. Water Resources Management, September. https://doi.org/10.1007/s11269-021-02937-w.

Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Physica D, 404, 132306. https://doi.org/10.1016/j.physd.2019.132306.

Siddique, Z. Bin, Khan, M. A., Din, I. U., Almogren, A., Mohiuddin, I., & Nazir, S. (2021). Machine Learning-Based Detection of Spam Emails. Scientific Programming, December. https://doi.org/10.1155/2021/6508784.

Utomo, M. W. S., Murti, H. W., Sujatmoko, Am. W. I., & Sari, A. P. (2024). Deteksi Spam Email Menggunakan Metode LSTM (Long Short Term Memory). JATI (Jurnal Mahasiswa Teknik Informatika), 8(6), 11406–11411. https://doi.org/10.36040/jati.v8i6.11474.