

Article

Email spam detection powered by deep learning, ensemble learning, and an optimization-based framework

Hidayet Takci

Computer Engineering Dept. Sivas Cumhuriyet University, Sivas, Türkiye; htakci@cumhuriyet.edu.tr

Received: 01 July 2026; Accepted: 05 September 2026; Published: 28 September 2026.

Abstract: This study evaluates a common experimental framework comprising deep learning, ensemble learning, and optimization-based feature selection techniques for spam email detection. To date, knowledge-based and machine learning-based methods have been used for spam detection. However, classical machine learning-based methods can be insufficient in representing and classifying complex spam content. In this study, experiments were conducted on the Enron and SpamAssassin datasets using CNN, GRU, XGBoost, and Random Forest algorithms, a genetic algorithm-based feature selection approach, and TF-IDF, Word2Vec, and embedding-layer representations. The contribution is the cross-comparison of established model, representation, and genetic-algorithm feature-selection combinations on two benchmark datasets rather than a new learning algorithm. Experimental results are presented using metrics such as AUC, F1 Score, and accuracy. The maximum F1-score and AUC value obtained in our study were 0.9997.

Keywords: email spam detection, deep learning, ensemble learning, genetic algorithm

1. Introduction

Emails have become one of today's most important communication channels. However, as email use has become widespread, the volume of spam messages—its unintended use—has also increased. According to reports published in recent years, approximately half of current email traffic is spam [1]. The increase in spam poses threats to information security, user privacy, and the efficiency of corporate resources. Spam emails are not limited to promotional content; they can also include actions targeting users through phishing, malware, phishing attacks, and social engineering techniques [2].

Early studies on spam detection relied on rule-based filtering, blacklists, and content-based static methods. However, because attackers easily bypassed existing static methods, there was a need for data-driven methods. As a result, machine learning-based approaches have gained prominence since the early 2000s, employing techniques such as Naive Bayes [3], support vector machines [4], K-nearest neighbor (KNN), decision trees, and ensemble methods. Studies from this period primarily employed bag-of-words and TF-IDF-based representation models, and it was demonstrated that n-gram features increased spam filtering success [5].

In the following years, non-content features such as sender address, message header, URL structure, and domain name were also included in the model and were observed to increase spam success [6,7]. In addition to classical feature extraction methods, semantic relationships between words have also been learned since the 2010s, thanks to representation models such as Word2Vec, Glove, and FastText.

Algorithms such as SVMs, Naive Bayes, decision trees, and logistic regression have been widely used in machine-learning-based spam detection [8,9]. Furthermore, ensemble methods such as XGBoost, LightGBM, and CatBoost also provide high performance on spam datasets. However, due to the evolving nature of spam messages, such as their multilingual and context-sensitive nature, classical machine learning algorithms based on statistical features have become insufficient.

In recent years, deep learning architectures [10] and transformer models [11] have achieved significant success in spam detection. Advances in deep learning have led to improved results in spam detection. In this context, CNNs that capture patterns at the character and word levels, and LSTM and GRU models that capture long dependencies, have appeared in the literature [12]. In addition, BERT and similar transformer-based large

language models have achieved success in context-aware representation and have achieved high success rates [13,14]. These models are particularly successful in phishing detection compared to classical approaches [2].

One of the significant problems in spam detection is the use of adversarial techniques by spammers. Character-level content changes can mislead classifications. Therefore, methods based on semantic content can fail. Adversarial machine learning methods have begun to develop to address such problems. Efforts such as feature extraction and anomaly detection have been employed to address these issues [15].

Spam detection has evolved from rule-based systems to the use of transformative architectures. Spam generators, in turn, deceive spam detection systems with adversarial attacks. Therefore, effective techniques are needed to improve the accuracy of spam detection, in both form and content. This study evaluates these methods within a common experimental framework to compare model families, representations, and genetic-algorithm feature selection across the two datasets. Its novelty lies in this comparative integration of established components rather than in proposing a new classifier architecture.

2. Literature

Spam detection fundamentally encompasses classical machine learning methods, deep learning-based methods, ensemble methods, and optimization-based hybrid methods. Because our work will utilize two different datasets and techniques, the literature is organized into two subsections.

2.1. Literature for the SpamAssassin dataset

Trivedi and Dey [16] achieved a significant performance improvement over classical machine learning algorithms by combining content-based feature extraction and evolutionary feature selection on the SpamAssassin dataset. Optimizing feature subsets played a critical role in this success. The most important conclusion of the study is that even simple models can achieve high performance if content analysis is performed correctly.

In their study on the SpamAssassin dataset, Yahyaouy et al. [17] used a Doc2Vec-based representation. This study achieved over 99% accuracy. It was observed that the document-level representation provides a significant advantage in spam detection.

Chogare and colleagues [18] tested different data preprocessing strategies on the SpamAssassin and Enron datasets. Their findings indicated that proper stemming and lemmatization steps improved performance. An accuracy of over 99% was achieved with Random Forest.

Ratmele et al. [19] combined Octave CNN, capsule networks, and a multi-threaded attention mechanism in a single framework and achieved over 99% accuracy on the SpamAssassin dataset. Kepler-based optimization was used in the study, and its success was demonstrated.

Mamathashree and Sheethal [20] applied Naive Bayes and other models using the TF-IDF representation to seven datasets, including the SpamAssassin dataset. They also applied genetic algorithms and PSO to the models, and in particular, the MNB model optimized with GA

achieved 100% accuracy. The most important outcome of the study is that simple models with hyperparameter optimization achieve results competitive with deep learning models.

Rojas-Galeano [21] investigated the zero-shot learning performance of pre-trained large language models on SpamAssassin. Models such as GPT-4 and Flan-T5 produced F1 scores in the 90–95% range using only prompt-based methods. This demonstrates that large language models are a powerful alternative to traditional classification models.

2.2. Literature for Enron dataset

The Enron dataset is a widely used dataset for email spam detection. This section compiles studies that have used the Enron dataset to detect spam.

Candan et al. [22] compared individual classifiers, ensemble classifiers, and deep learning models for detecting Turkish and English spam emails. They used the Enron dataset for English emails and achieved around 99.9% accuracy with machine learning algorithms using the TF-IDF representation.

Alrammahi et al. [23] combined the Enron email dataset and the SMS spam collection dataset and established a feature-extraction, clustering, and classification chain. They used statistical features in addition to TF-IDF features to cluster using K-Means and Fuzzy C-Means. The labels obtained from the clustering

output were used to feed supervised classifiers such as KNN, NB, SVM, and XGBoost. The use of clustering algorithms in preprocessing positively affected the classification performance.

Kshirsagar et al. [24] proposed an interpretable meta-learning framework on four datasets, including Enron-Spam, SpamAssassin, and TREC 2007. Approximately 98% accuracy was achieved on the Enron-Spam dataset using the Random Forest algorithm with the TF-IDF representation. Furthermore, high F1 scores and AUC values were obtained with deep learning models such as GRU/BiLSTM and the GloVe representation. In the final stage of the study, a meta-learner was used, which increased classification accuracy.

Poobalan et al. [25] used the Enron dataset in their study and first performed tokenization, stemming, and stop-word removal on the textual data, then extracted TF-IDF features, and then compared the BiLSTM model with LR, CNN, RF, and RNN methods. Experimental results show that the BiLSTM model achieves classification accuracy between 98% and 99%.

Ayo et al. [26] combined a correlation-based deep learning approach with a fuzzy inference system and tested their method on email spam datasets, including the Enron dataset. Correlation-based Feature Selection (CFS) and rule-based genetic search were used to select the most important features. According to the performance data for the proposed method, the F1-score was around 96%, and the accuracy was 94%.

2.3. Overall evaluation

One of the factors that influences the success of spam detection in datasets is accurate content analysis. The statistical summary of the content provides useful features for spam detection. It has been frequently used, particularly in traditional spam detection.

Because spam datasets contain textual content, accurate language processing steps have been helpful in their detection. After the data preprocessing step, they are appropriately vectorized and processed using representation methods. In this regard, TF-IDF, Word2Vec, Embedding Layer, FastText, and Glove representations have been frequently used. Spam content is represented at the document, character, and word levels, which positively impacts performance.

Because the outputs of these representation methods are multidimensional, feature selection methods are used to reduce dimensionality. Dimensionality reduction methods such as genetic algorithm-based ranking or PCA are frequently used. The impact of feature selection on performance has also been observed at this stage.

High classification success has been achieved by combining deep learning models and deep learning-based representations with meta-learners. CNN, GRU, RNN, and BiLSTM are frequently preferred models in deep learning.

One key finding in the literature is that ensemble learning methods using TF-IDF representations perform as well as deep learning models. Simple models, with the right adjustments, can achieve results as successful as complex models.

Model architecture isn't the only factor affecting success in spam detection; feature extraction, data transformation, data balancing, and optimization also influence the results. Transformational models, in particular, have yielded strong results in the literature, while metaheuristic optimization methods have also shown strong performance.

Despite the high accuracy achieved on spam email datasets, performance can degrade under adversarial attacks. According to a study by Hotoğlu [27], under well-designed attacks, accuracy in some models can drop from 99% to 40%.

3. The proposed method

This study aims to examine and compare the effects of deep learning models, ensemble learning models, representation methods, and feature selection on spam datasets. This comparison is used to identify suitable components for a spam detection system. In this context, the following subtopics are discussed:

- Deep learning component: CNN, GRU
- Ensemble learning component: XGBoost, Random Forest
- Optimization component: Genetic algorithm-based feature selection
- Data representation: TF-IDF, Word2Vec

3.1. Deep learning component

Spam detection studies require the ability to model both semantic patterns and long-term dependencies. In this context, CNNs were chosen for their ability to capture semantic patterns, and GRUs for their ability to model long-term dependencies.

The CNN model learns distinctive features using multiple filters to uncover sequential relationships at the n-gram level. Because spam emails exhibit specific word patterns, commercial and advertising content, and template-based structures, the CNN algorithm is well-suited.

The GRU model is one of the recurrent neural network variants and uses fewer gates than the LSTM algorithm. Its ability to learn sequential relationships stems from its recurrent neural network architecture. It was chosen for its success in modeling the long, complex content of spam emails. The present study does not compare GRU directly with LSTM and therefore does not establish general superiority over LSTM.

3.2. Ensemble learning component

Machine learning models are classical methods and fall short in increasingly complex datasets. Ensemble learning models are one of the approaches that emerge when machine learning models fail. XGBoost and Random Forest algorithms were chosen in this study to detect spam patterns and achieve high-performance spam detection.

Random Forest is a model composed of multiple decision trees and is successful in capturing complex patterns. XGBoost, on the other hand, yields high sensitivity and AUC scores, particularly on imbalanced datasets, thanks to its gradient-boosting approach. Both of these methods are evaluated for high-accuracy detection by capturing word patterns in spam.

3.3. Optimization component

In the proposed study, genetic algorithms are used to optimize feature selection. Genetic algorithms use binary chromosomes to represent whether each feature is selected. The initial population consists of randomly generated 0-1 sequences, and each individual represents a feature subset. The ROC-AUC value is used as the fitness function. New generations are generated in the population using selection, crossover, and mutation operators. The genetic algorithm is used to search for a high-performing feature subset in the high-dimensional feature space; the present experiments do not establish global optimality or immunity from local optima.

A genetic algorithm-based approach is intended to reduce redundant features and thereby may reduce computational cost or susceptibility to noise in high-dimensional spam data. Because these properties and overfitting are not measured separately in the reported experiments, the study evaluates the method only through its observed effect on accuracy, F1-score, and AUC. Genetic algorithms were chosen in this study as a feature selection method.

3.4. Data embedding

The embedding layer, Word2Vec, and TF-IDF methods were used to represent the data. The embedding layer and Word2Vec were chosen for their suitability for deep learning models, while TF-IDF and Word2Vec were chosen for their suitability for ensemble learning models.

The embedding layer is a representation method that converts words and symbols into low-dimensional, meaningful vectors. This layer learns the semantic relationships between words and transforms discrete inputs into a continuous space. It was chosen because it yields successful results in deep learning models.

The Word2Vec representation models the semantic proximity of words in a vector space. Trained with the Skip-gram or CBOW architecture, the representation method can learn expressions, phrases, financial data, and persuasive language structures found in spam messages. The method's output is suitable for both deep learning and ensemble learning models.

The TF-IDF representation moves words into a new space based on a function of the word's frequency in the document and its frequency in the entire corpus. Although it is a sequence-

independent model, it can still distinguish messages by frequency. Although the TF-IDF representation is old, it gives high results when used with methods such as XGBoost or Random Forest.

4. Experimental results

4.1. Datasets

Spam detection in emails was performed using two different datasets. The first is the SpamAssassin dataset, and the other is the Enron dataset, both of which contain text.

The SpamAssassin dataset is a dataset created by the Apache SpamAssassin project. Repackaged versions of the dataset were later published on Kaggle. The Kaggle version of the dataset was used in this study [28].

The Enron email dataset is a real-world dataset created from Enron Corporation's corporate correspondence between 2000 and 2002 and is widely used in spam detection studies. This study used the modified Enron Spam Dataset published on Kaggle [29]. The dataset consists of 5,172 emails: 1,500 classified as spam and 3,672 as raw.

4.2. Experimental design

CNN and GRU are based on deep learning, while XGBoost and RandomForest are based on ensemble learning. This factor influenced the selection of the representation method. Embedding layers and Word2Vec were used as representation methods for deep learning-based methods, while TF-IDF and Word2Vec were used for ensemble learning-based models. Because TF-IDF is a sequence-independent model, it is insufficient for learning semantic and sequential relationships in deep learning models and is not suitable for use. Word2Vec is suitable for both types of models.

During the experimental studies, each algorithm was run both without and with feature selection. This allowed us to measure the effect of feature selection on classification performance. Within the experimental studies, the effects of the learning method type, representation method, and feature selection on classification were investigated. The results were analyzed separately for each dataset.

Because the class distribution was unbalanced in both datasets, metrics such as F1-score and ROC-AUC were preferred for evaluation.

The reported implementation details do not specify the train/validation/test split, random seed, number of repeated runs, preprocessing sequence, model architecture and training hyperparameters, representation-to-model feature mapping, GA population/operator/stopping parameters, matched hyperparameter-tuning budgets, the partition used to compute GA fitness, or whether representation fitting and GA feature selection were restricted to the training partition. Accordingly, the comparisons below are interpreted descriptively and do not establish statistical significance, leakage-free validation, or universal model superiority.

4.3. Experimental results for the SpamAssassin dataset

In this section, experiments are conducted on the SpamAssassin dataset in accordance with the experimental design, and the results are reported. The evaluation process is based on common performance metrics such as accuracy, F1 Score, and ROC AUC. The experiments aim to compare the effects of model-representation combinations on spam detection performance and to determine when GA-based feature selection provides an advantage. The results are summarized in Table 1.

Experimental results obtained on the SpamAssassin dataset are shown in the table. The representation method and feature selection strategy are associated with differences in the reported model performance. While the use of an embedding layer generally yielded higher performance in CNN and GRU models, the Word2Vec representation led to lower performance in some combinations. In XGBoost and Random Forest without GA, TF-IDF and Word2Vec produced similar accuracy and F1-score point estimates, while AUC varied more across the GA configurations. For XGBoost, the GA configurations produced higher reported point estimates for all three metrics with both representation settings, while the corresponding changes were mixed for CNN, GRU, and Random Forest.

While the results are similar, comparisons based on F1-score and AUC values will be useful for comparing models. A comparative evaluation of the model, representation method, and genetic algorithms in terms of AUC and F1-score on the SpamAssassin dataset is presented in Figure 1.

The model that yielded the highest AUC was the Random Forest algorithm, which uses a Word2Vec representation and features selected via a Genetic Algorithm. No model family uniformly outperformed the

other across accuracy, F1-score, and AUC: CNN with an embedding layer without GA and XGBoost with TF-IDF and GA shared the highest reported accuracy and F1-score, while Random Forest with Word2Vec and GA produced the highest AUC. Accordingly, GA-based feature selection was associated with metric- and model-dependent changes rather than a uniform improvement.

Table 1. Comparison of spam detection performances of different model–representation–feature selection combinations on the spamassassin dataset

Model	Embedding	Feature selection	Acc.	F1	AUC
CNN	Embedding layer	NA	0.9965	0.9947	0.9974
CNN	Word2vec	NA	0.9931	0.9894	0.9914
CNN	Embedding layer	GA	0.9853	0.9772	0.9991
CNN	Word2Vec	GA	0.9758	0.9623	0.9975
GRU	Embedding layer	NA	0.9896	0.9842	0.9989
GRU	Word2vec	NA	0.9844	0.9762	0.9956
GRU	Embedding layer	GA	0.9853	0.9780	0.9976
GRU	Word2Vec	GA	0.9732	0.9583	0.9917
XGBoost	TF-IDF	NA	0.9922	0.9881	0.9908
XGBoost	Word2Vec	NA	0.9913	0.9867	0.9881
XGBoost	TF-IDF	GA	0.9965	0.9947	0.9967
XGBoost	Word2vec	GA	0.9939	0.9907	0.9994
Random Forest	TF-IDF	NA	0.9862	0.9786	0.9816
Random Forest	Word2Vec	NA	0.9862	0.9785	0.9802
Random Forest	TF-IDF	GA	0.9853	0.9772	0.9796
Random Forest	Word2vec	GA	0.9853	0.9772	0.9996

NA: no feature selection, GA: genetic algorithm

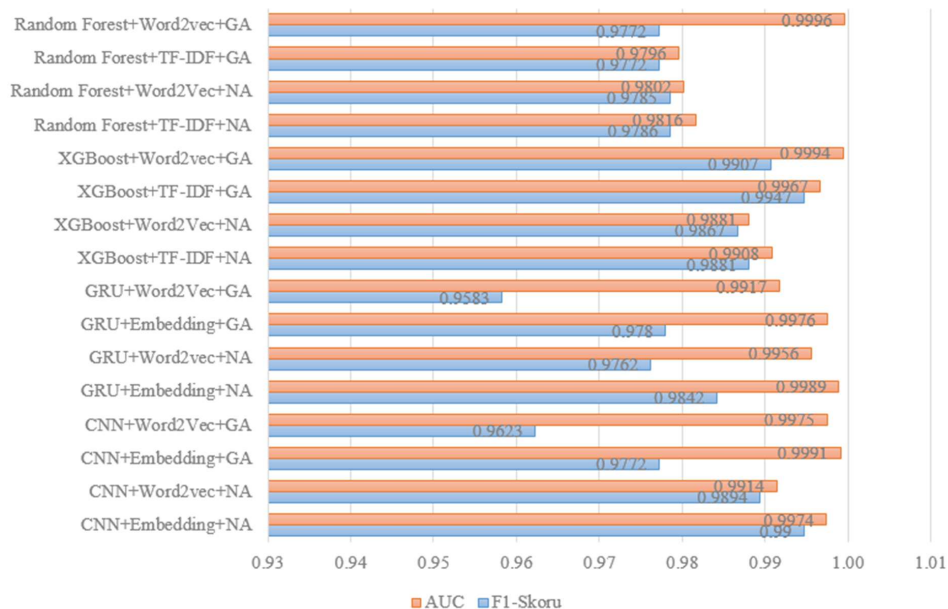


Figure 1. Comparison of the model, representation method, and genetic algorithms based on AUC and F1-score on the SpamAssassin dataset

4.4. Experimental results for Enron dataset

The experiments performed on the SpamAssassin dataset were also performed on the Enron dataset in the same way, and the comparisons are summarized in Table 2.

According to the results in the table, all models demonstrated very high performance, producing very similar accuracy, F1-score, and AUC values. In particular, the XGBoost, Random Forest, and CNN

models consistently achieved accuracy values close to 0.999 across different representation methods, showing uniformly high performance on the reported Enron evaluation. The use of the genetic algorithm produced only small, mixed changes across the reported metrics.

Table 2. Comparison of Spam Detection Performances of Different Model–Representation–Feature Selection Combinations on the Enron Dataset

Model	Embedding	Feature selection	Acc.	F1	AUC
CNN	Embedding layer	NA	0.9994	0.9994	0.9994
CNN	Word2vec	NA	0.9992	0.9992	0.9994
CNN	Embedding layer	GA	0.9991	0.9991	0.9995
CNN	Word2Vec	GA	0.9992	0.9992	0.9995
GRU	Embedding layer	NA	0.9982	0.9982	0.9994
GRU	Word2vec	NA	0.9980	0.9981	0.9994
GRU	Embedding layer	GA	0.9991	0.9991	0.9995
GRU	Word2Vec	GA	0.9979	0.9979	0.9994
XGBoost	TF-IDF	NA	0.9997	0.9997	0.9995
XGBoost	Word2Vec	NA	0.9997	0.9997	0.9997
XGBoost	TF-IDF	GA	0.9997	0.9997	0.9994
XGBoost	Word2vec	GA	0.9997	0.9997	0.9995
Random Forest	TF-IDF	NA	0.9997	0.9997	0.9995
Random Forest	Word2Vec	NA	0.9997	0.9997	0.9995
Random Forest	TF-IDF	GA	0.9997	0.9997	0.9995
Random Forest	Word2vec	GA	0.9995	0.9995	0.9995

NA: no feature selection, GA: genetic algorithm

Due to the imbalanced dataset in accordance with the experimental design, the models are compared using F1-score and AUC in the graph (Figure 2).

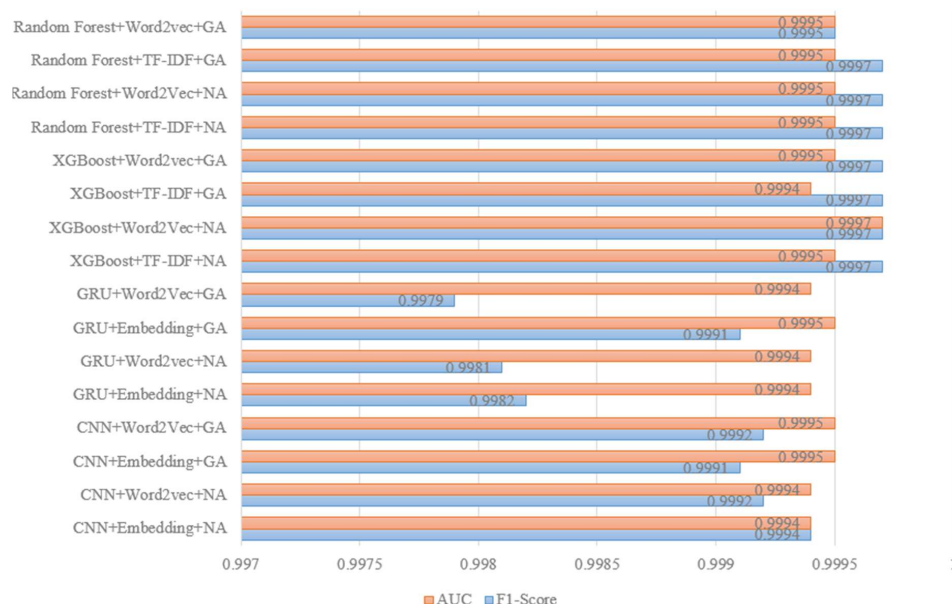


Figure 2. Comparison of the model, representation method, and genetic algorithms based on AUC and F1-score on the Enron dataset

All models yielded high classification performance on the Enron dataset. The classification point estimates from ensemble learning methods were slightly higher than those from deep learning-based models, but the reported differences are very small and no repeated-run uncertainty or statistical test is provided. Among the deep learning models, the CNN had higher reported point estimates than the GRU. The results obtained with XGBoost and Random Forest in ensemble learning models were very similar.

4.5. Comparison with previous studies

Recent studies on spam emails highlight the use of classical machine learning algorithms, ensemble learning algorithms, deep learning models, and large-scale language models. Additionally, natural language processing-based feature extraction, selection, transformation, hyperparameter optimization, and preprocessing steps were utilized.

According to statistics obtained from studies conducted over the last 10 years, accuracy between 0.88 and 0.98, F1-score between 0.82 and 0.98, and AUC between 0.90 and 0.99 were achieved on the SpamAssassin dataset. On the Enron dataset, accuracy between 0.85 and 0.98, F1-score between 0.80 and 0.98, and AUC between 0.88 and 0.99 were achieved. Mamathashree and Sheethal [20] reported 100% accuracy, but because the datasets are unbalanced, the F1-score and AUC are crucial. The maximum F1 score given in the study was 0.99.

In our study, the best reported accuracy, F1 score, and AUC values across the evaluated configurations for the SpamAssassin dataset were 0.9965, 0.9947, and 0.9996, respectively, and for the Enron dataset, they were 0.9997, 0.9997, and 0.9997, respectively. These values are at the upper end of the literature ranges reported above. The present results show that model and representation choices are associated with performance, while GA feature selection has a mixed, metric-dependent effect. Because the experimental protocols and tuning budgets of the cited studies are not reported as matched to those used here, these cross-study values should not be interpreted as a controlled proof of superiority.

5. Discussion

Spam detection has been a field of study that has attracted researchers' attention for many years. Due to the increasing complexity of spam content and the increasingly deceptive tactics of spammers, classical methods can be insufficient for some spam patterns. Therefore, there is a need for machine learning methods, augmented with ensemble and deep learning models, that better decipher patterns in data. The natural language content of spam messages necessitates preprocessing steps. Furthermore, redundancy in textual data is a significant problem. Based on observations of the models' characteristics and the nature of the data, an experimental framework was developed in this study, and experiments were conducted.

The experiments first considered the spamassasin dataset, followed by the Enron dataset. In the experiments conducted on the Anaconda platform, each model/algorithm was evaluated using different representation methods and feature selection techniques. Thus, four experiments were conducted for each model/algorithm. The experimental results were obtained as accuracy, F1-score, and AUC values. CNN and GRU were used as deep learning models. The textual data were transformed into a format suitable for deep learning architectures. In the ensemble learning phase, XGBoost and Random Forest were chosen. Embedding and Word2Vec were used as representation methods in deep learning models, while Word2Vec and TF-IDF were used in ensemble learning models. This difference stems from the differences in the models. Word2Vec is compatible with both. Furthermore, a genetic algorithm was used for feature selection, and all experiments were run with or without it.

A review of past studies reveals that numerous techniques have been used in spam detection, ranging from rule-based to knowledge-based to machine learning systems. However, the present results do not show that a single method family is uniformly superior; performance depends on the model-representation combination and the selected evaluation metric. The engineering contribution is therefore a comparative evaluation framework for established components across two datasets, not a new learning algorithm or a demonstrated adversarially robust architecture.

6. Conclusion

Experimental results showed that the best reported spam detection values were at or above the literature ranges summarized in this study. The best reported F1 score was 0.9947 in the SpamAssassin dataset and 0.9997 in the Enron dataset. The best reported AUC value was 0.9996 in the SpamAssassin dataset and 0.9997 in the Enron dataset. On the SpamAssassin dataset, CNN with an embedding layer without GA and XGBoost with TF-IDF and GA shared the highest reported F1 score of 0.9947, while Random Forest with Word2Vec and GA yielded the highest AUC of 0.9996. These results show that different model-representation combinations lead on different metrics; they do not establish that character-level representations are superior, because the

embedding granularity and adversarial robustness were not evaluated. Furthermore, XGBoost with Word2Vec without GA yielded the highest AUC of 0.9997 on the Enron dataset. Considering all the results, feature selection using genetic algorithms does not consistently increase classification accuracy; its effect is model-, representation-, and metric-dependent.

As the volume of spam e-mail increases daily, the importance of effectively filtering it has become even greater. Classical methods can be insufficient for evolving spam content. Both the increasing complexity of spam messages and spam generators' techniques for bypassing spam detection tools necessitate new solutions.

To address the current challenges, this study evaluates deep learning architectures, ensemble learning models, representation methods, and genetic algorithm-based feature selection within a common framework. The high detection rates achieved through these studies hold promise in combating spam emails. The reported results support high performance on the evaluated datasets but, given the unreported split and tuning details and the absence of repeated-run uncertainty, do not establish statistical superiority, adversarial robustness, or deployment performance. As spam generators evolve their strategies, new technologies should be employed to counter them in real time, or even before they do.

Conflicts of Interest: The author declares that there are no conflicts of interest related to this work.

Data Availability: The data supporting the findings of this study are available from the author upon reasonable request.

Funding Information: This research received no external funding.

References

- [1] Moorthy, J. (2026, January 27). 23 email spam statistics to know in 2026. Mailmodo. <https://www.mailmodo.com/guides/email-spam-statistics/>
- [2] Atawneh, S., & Aljehani, H. (2023). Phishing email detection model using deep learning. *Electronics*, 12(20), Article 4261.
- [3] Sahami, M., Dumais, S., Heckerman, D., & Horvitz, E. (1998). A Bayesian approach to filtering junk e-mail. In *Learning for text categorization: Papers from the AAAI Workshop* (pp. 55–62). AAAI Press. (Technical Report WS-98-05).
- [4] Drucker, H., Wu, D., & Vapnik, V. N. (1999). Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, 10(5), 1048–1054.
- [5] Androutsopoulos, I., Koutsias, J., Chandrinou, K. V., & Spyropoulos, C. D. (2000). An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 160–167). Association for Computing Machinery.
- [6] Gómez Hidalgo, J. M., Cajigas Bringas, G., Puertas Sanz, E., & Carrero García, F. (2006). Content based SMS spam filtering. In *Proceedings of the 2006 ACM Symposium on Document Engineering* (pp. 107–114). Association for Computing Machinery.
- [7] Carreras, X., & M'arquez, L. (2001). Boosting trees for anti-spam email filtering. In *Proceedings of the 3rd Conference on Recent Advances in Natural Language Processing (RANLP 2001)* (pp. 58–64).
- [8] Guzella, T. S., & Caminhas, W. M. (2009). A review of machine learning approaches to spam filtering. *Expert Systems with Applications*, 36(7), 10206–10222.
- [9] Gattani, G., Mantri, S., & Nayak, S. (2023). Comparative analysis for email spam detection using machine learning algorithms. In R. Agrawal, C. K. Singh, A. Goyal, & D. K. Singh (Eds.), *Modern electronics devices and communication systems: Select proceedings of MEDCOM 2021* (Lecture Notes in Electrical Engineering, Vol. 948, pp. 11–21). Springer.
- [10] Saleem, S., Islam, Z. U., Hasan, S. S. U., Akbar, H., Khan, M. F., & Ibrar, S. A. (2025). Spam email detection using long short-term memory and gated recurrent unit. *Applied Sciences*, 15(13), Article 7407.
- [11] Jamal, S., Wimmer, H., & Sarker, I. H. (2024). An improved transformer-based model for detecting phishing, spam and ham emails: A large language model approach. *Security and Privacy*, 7(5), Article e402.
- [12] Altwaijry, N., Al-Turaiki, I., Alotaibi, R., & Alakeel, F. (2024). Advancing phishing email detection: A comparative study of deep learning models. *Sensors*, 24(7), Article 2077.
- [13] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems* (Vol. 30, pp. 5998–6008).
- [14] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (Vol. 1, pp. 4171–4186). Association for Computational Linguistics.

- [15] Penaloza Rumie, J. J., Sarkar, S., & Karem, A. (2025). From adversarial attacks to robust classifiers—A study in social media spam detection. In *SoutheastCon 2025* (pp. 1007–1012). IEEE.
- [16] Trivedi, S. K., & Dey, S. (2019). A modified content-based evolutionary approach to identify unsolicited emails. *Knowledge and Information Systems*, 60(3), 1427–1451.
- [17] Hnini, G., Fahfouh, A., Riffi, J., Mahraz, M. A., Yahyaouy, A., & Tairi, H. (2021). Spam filtering based on PV-DBOW model. *International Journal of Data Analysis Techniques and Strategies*, 13(4), 302–316.
- [18] Ghogare, P. P., Dawoodi, H. H., & Patil, M. P. (2024). Enhancing spam email classification using effective preprocessing strategies and optimal machine learning algorithms. *Indian Journal of Science and Technology*, 17(15), 1545–1556.
- [19] Ratmele, A., Dhanare, R., & Parte, S. A. (2025). Octave convolutional multi-head capsule nutcracker network with oppositional Kepler algorithm based spam email detection. *Wireless Networks*, 31(2), 1625–1644.
- [20] Mamathashree, G. M., & Sheethal, P. P. (2025). Email spam detection. *International Research Journal of Modernization in Engineering Technology and Science*, 7(8), 2881–2886.
- [21] Rojas-Galeano, S. (2024). *Zero-shot spam email classification using pre-trained large language models* [Preprint]. arXiv. <https://arxiv.org/abs/2405.15936>
- [22] Candan, E. N., K"uç"ukilhan, R., & Eroğlu, A. (2025). Spam mail detection in Turkish and English languages: A holistic study of AI-based techniques including individual, ensemble and hybrid approaches. *Necmettin Erbakan "Universitesi Fen ve M"uhendislik Bilimleri Dergisi*, 7(2), 189–205.
- [23] Alrammahi, A. A. H., Sari, F. A. O., Muhammad, Z. A., Kadhim, M. N., Al-Shammmary, D., & Ibaida, A. (2025). Enhancing spam detection with advanced feature extraction and unsupervised clustering. *International Journal of Information Technology*, 17, 5109–5119.
- [24] Kshirsagar, M., Rath, V., & Ryan, C. (2025). Meta-learner-based frameworks for interpretable email spam detection. *Frontiers in Artificial Intelligence*, 8, Article 1569804.
- [25] Poobalan, A., Ganapriya, K., Kalaivani, K., & Parthiban, K. (2025). A novel and secured email classification using deep neural network with bidirectional long short-term memory. *Computer Speech & Language*, 89, Article 101667.
- [26] Ayo, F. E., Ogundele, L. A., Olakunle, S., Awotunde, J. B., & Kasali, F. A. (2024). A hybrid correlation-based deep learning model for email spam classification using fuzzy inference system. *Decision Analytics Journal*, 10, Article 100390.
- [27] Hotoğlu, E. (2024). *A comprehensive analysis of adversarial attacks on spam filters* [Master's thesis, Hacettepe University]. Hacettepe University Open Access Repository.
- [28] Bayes. (n.d.). *Emails for spam or ham classification SpamAssassin* [Data set]. Kaggle. <https://www.kaggle.com/datasets/bayes2003/emails-for-spam-or-ham-classification-spamassassin>
- [29] Cukierski, W. (2015). *The Enron email dataset* [Data set]. Kaggle. <https://www.kaggle.com/datasets/wcukierski/enron-email-dataset>



© 2026 by the authors; licensee PSRP, Lahore, Pakistan. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).