Computer Modeling in Engineering & Sciences

Tech Science Press

Doi:10.32604/cmes.2025.061608

ARTICLE

DaC-GANSAEBF: Divide and Conquer-Generative Adversarial Network—Squeeze and Excitation-Based Framework for Spam Email Identification

Tawfeeq Shawly[1], Ahmed A. Alsheikhy[2,*], Yahia Said[3], Shaaban M. Shaaban[3], Husam Lahza[4], Aws I. AbuEid[5] and Abdulrahman Alzahrani[6]

[1]Department of Electrical Engineering, Faculty of Engineering at Rabigh, King Abdulaziz University, Jeddah, 21589, Saudi Arabia
[2]Department of Electrical Engineering, College of Engineering, Northern Border University, Arar, 91431, Saudi Arabia
[3]Center for Scientific Research and Entrepreneurship, Northern Border University, Arar, 73213, Saudi Arabia
[4]Department of Information Technology, College of Computing and Information Technology, King Abdulaziz University, Jeddah, 21589, Saudi Arabia
[5]Department of Information and Communication Technology, College of Computing Studies, Arab Open University, Kuwait Branch, Kuwait City, 13033, Al-Safat, State of Kuwait
[6]Department of Information Systems and Technology, College of Computer Science and Engineering, University of Jeddah, Jeddah, 23218, Saudi Arabia

*Corresponding Author: Ahmed A. Alsheikhy. Email: aalsheikhy@nbu.edu.sa

Received: 28 November 2024; Accepted: 06 February 2025; Published: 03 March 2025

ABSTRACT: Email communication plays a crucial role in both personal and professional contexts; however, it is frequently compromised by the ongoing challenge of spam, which detracts from productivity and introduces considerable security risks. Current spam detection techniques often struggle to keep pace with the evolving tactics employed by spammers, resulting in user dissatisfaction and potential data breaches. To address this issue, we introduce the Divide and Conquer-Generative Adversarial Network Squeeze and Excitation-Based Framework (DaC-GANSAEBF), an innovative deep-learning model designed to identify spam emails. This framework incorporates cutting-edge technologies, such as Generative Adversarial Networks (GAN), Squeeze and Excitation (SAE) modules, and a newly formulated Light Dual Attention (LDA) mechanism, which effectively utilizes both global and local attention to discern intricate patterns within textual data. This approach significantly improves efficiency and accuracy by segmenting scanned email content into smaller, independently evaluated components. The model underwent training and validation using four publicly available benchmark datasets, achieving an impressive average accuracy of 98.87%, outperforming leading methods in the field. These findings underscore the resilience and scalability of DaC-GANSAEBF, positioning it as a viable solution for contemporary spam detection systems. The framework can be easily integrated into existing technologies to enhance user security and reduce the risks associated with spam.

KEYWORDS: Email; spam; fraud; light dual attention; squeeze and excitation; divide and conquer-generative adversarial network—squeeze and excitation-based framework; security

1 Introduction

Email serves as one of the most prevalent communication mediums worldwide, significantly impacting personal, professional, and commercial exchanges. Nevertheless, its extensive use has rendered it a prime target for malicious activities, particularly spam. Spam emails are unsolicited communications typically

dispatched in large quantities, aimed at deceiving, harming, or disrupting recipients. These messages often contain phishing links, malware, or fraudulent schemes, which present considerable risks to individuals, businesses, and governmental entities. The rise in spam has posed a continuous challenge for many years. In 2003, spam constituted nearly 80% of all email traffic globally. Despite advancements in technology and the implementation of stricter regulations, spam continues to be a widespread concern, resulting in billions of dollars lost annually due to decreased productivity and security breaches. Email spam represents not only financial strain but also significant threats to personal privacy and emotional health. In addition to financial losses, spam emails can result in identity theft, psychological distress, and violations of confidential information. For instance, phishing schemes may deceive individuals into disclosing sensitive information, while frequent exposure to spam can lead to frustration and diminish trust in digital communication platforms. Traditional rule-based and filter-based approaches, although initially effective, have become inadequate in addressing the growing sophistication of spammers.

1.1 Email Spams Perspective

In email, spam refers to junk messages or unsolicited ones that are sent by either humans or a set of computers [1–3]. These emails contain viruses, malware, or unwanted links that can be a threat and dangerous to users [1–3]. In general, spam emails are sent in bulk to a group of recipients [1]. Nowadays, spam comes through text messages and social media channels or applications [1,2]. This spam is dangerous and annoying, especially for people who use emails in their daily life and business. Spammers try to deceive people into believing the wrong things to steal their sensitive data or blackmail these recipients [1,2]. Spam messages are driven by educational, financial, or commercial motivations [1]. Recently, spam messages have enclosed numerous subjects, such as online degrees, monetary services, medications, and adult content [1]. Unfortunately, some recipients fall into the tricks of spammers and end up losing their sensitive data or being blackmailed [1]. Despite technological advancements, spam emails can be sent to many addresses daily, which makes them a real threat to the economy [4–8].

Spam emails were started in 1978 by Gary Thuerk [1]. In 2003, spam messages represented nearly 80% to 85% of the total messages that were exchanged worldwide [1]. In 2020 and 2021, spam messages dropped from nearly 300 billion to around 120 billion due to solutions that were developed and installed [1]. Nevertheless, spam rates are still high and cost billions of dollars annually [1]. Spammers use various methods to send spam messages, such as malware, educational, antivirus alerts, fraud, and marketing emails [1]. Currently, several solutions have been implemented to detect and fight spam messages. These solutions use filters [6–9]. However, blocking spam messages completely is impossible since spammers come up with different methods to override the deployed filters. Handheld devices, such as mobile phones have become widely used in daily lives to send and receive Short Messages Services (SMS) and Email messages [4] and SMS spam issues increase daily as well. In 2013, SMS profit in the commercial enterprise field varied between nearly 11% to 25% [4]. This leads to an increase in annoying bulk messages, especially in business, more specifically in advertisements. In addition, mobile phone users receive spam calls to hack their devices and steal sensitive information to blackmail these users for money.

In business, companies and advertisers send either SMSs or Emails to customers to promote or sell their services or products [4,5]. Spammers use this technique to alter customers and get their data if possible. Regarding Emails, spam is considered the most effective and dangerous way to assault users, such as ruining reputations or leading consumers to lose confidence in some products or services [4]. According to these reasons, various solutions based on cutting-edge technologies have been developed and proposed by experts to identify spam Emails [3–6]. Spammers, also called scammers, send Emails that seem to be from legitimate sources, such as big known companies, service providers, or banks to phish users to fraud them later if

possible [7–10]. Email service depends on one-to-many methods to send instant messages from a source to different recipients using various protocols [4,6].

1.2 Technical Perspective

In the digital revolution, Email spam messages are one of the most significant problems since these messages affect individual users, industrial companies, businesses, and public and private sectors. The spam messages cause high resource utilization; thus, a need for practical solutions arises [4–7,9]. Numerous methods using machine learning and deep-learning tools have been implemented [11–15]. The main aim is to identify spam messages and block them to protect users by grouping messages into their safe types and spam [12–14]. These approaches provide more sophisticated methods since extracting features can be obtained using various tools, while the traditional rule-based solutions provide limited protection as the deployed techniques of sending spam messages are growing and getting advanced day by day [15–18]. Developed solutions from deep learning are learned from utilized labeled datasets to leverage pattern recognition to build robust solutions that can identify safe or spam messages of new Emails by analyzing contents, senders' information, embedded Uniform Resource Locators (URLs), and subject headers [19–23]. Naïve Bayes, Support Vector Machine (SVM), Random Forest (RF), and Neural Networks (NNs) are the most used topologies for spam prediction [8,9]. However, these tools depend on datasets to be trained to reach high outcomes, which means the attributes and variety of data play a significant impact on the achieved findings. Furthermore, finding relevant attributes and useful information from email messages is crucial and can be achieved using numerous ways.

Protecting recipients from spammers improves whole throughput and user experience as well [12]. Machine Learning (ML) and Deep Learning (DL) approaches provide various advantages for identifying spam messages, such as deep capabilities to filter and separate harmful Emails and minimize the effort by users to sort Emails [24–27]. In addition, ML and DL solutions can adapt to evolving methods of spammers. Nowadays, Email platforms play a crucial role in business and the economy as it is considered the main point of contact and communications between customers and vendors. These solutions can leverage the capabilities of pattern recognition through extensive training trials to provide a promising result for blocking harmful messages and providing a good experience for recipients. Furthermore, Natural Language Processing (NLP) plays a main role as well [9–12]. Currently, Email has become the main target for scammers since many users have no experience and easily fall into the traps or tricks of these scammers as spam scammers earn approximately 355$ million annually [7–10]. The developed methods to identify spam and harmful messages depend on two approaches: behavior pattern-based and semantic pattern-based [9–13]. However, these two kinds are associated with some limitations as spammers develop new tools daily. As there is no practical solution that provides a full solution to detect spam messages, it is of prodigious concern for users and businesses to build a proper approach to identify safe messages from spam to improve security, block malicious content, and enhance the accuracy level to surpass the existing methods.

Despite advancements in technology and the implementation of stricter regulations, spam continues to be a significant problem, resulting in annual losses of billions of dollars due to decreased productivity and security vulnerabilities. While traditional rule-based and filter-based approaches were initially effective, they have become inadequate in addressing the growing sophistication of spammers. The emergence of Machine Learning (ML) and Deep Learning (DL) technologies has transformed the landscape of spam detection. These approaches utilize sophisticated pattern recognition and feature extraction techniques to improve detection precision. Nevertheless, issues such as imbalanced datasets, substantial computational demands, and the continual evolution of spam tactics hinder their overall effectiveness. This research aims to tackle these challenges by introducing an innovative, adaptive framework for spam detection. Significant progress

has been made in spam detection; however, current methodologies reveal several critical shortcomings: 1. Dataset imbalance: numerous techniques do not adequately tackle the fundamental imbalance between legitimate and spam messages, resulting in skewed classification outcomes. 2. Generalization across datasets: the majority of models are tested on a singular dataset, which raises questions regarding their performance on previously unseen data. 3. High computational costs: sophisticated deep learning models, such as Long Short-Term Memory (LSTM) and Bidirectional Encoder Representations from Transformers (BERT), demand substantial computational resources, rendering them unsuitable for real-time applications. 4. Evolving spam techniques: conventional machine learning and certain deep learning models find it challenging to keep pace with the rapidly changing spam tactics, which restrict their long-term efficacy. Current trends highlight the importance of integrating various deep learning architectures to mitigate individual limitations. For example, hybrid methods that merge attention mechanisms with convolutional neural networks (CNNs) or generative adversarial networks (GANs) have demonstrated the potential to improve feature extraction and overall robustness. Nevertheless, challenges such as mode collapse in GANs and the inadequate exploration of attention mechanisms continue to be insufficiently addressed. The historical context of spam, highlighted by its peak of 80% of global email traffic in 2003, emphasizes the seriousness of the problem. However, the current challenge is to counteract increasingly sophisticated and adaptive spam strategies. These contemporary techniques take advantage of technological advancements, rendering traditional detection methods less effective. To address these challenges, this research presents the Divide and Conquer-Generative Adversarial Network Squeeze and Excitation-Based Framework (DaC-GANSAEBF) as a solution to the identified challenges in spam email detection. This innovative approach leverages deep learning techniques by incorporating several sophisticated elements, including Generative Adversarial Networks (GAN), Squeeze and Excitation (SAE) modules, and a Light Dual Attention (LDA) mechanism. The framework is designed to effectively identify intricate patterns within textual data, resulting in enhanced accuracy and resilience.

1.3 Research Problem

Although numerous spam detection techniques exist, none provide a thorough solution to the increasingly advanced nature of spam strategies. Conventional machine learning approaches, including Naïve Bayes, Random Forest, and Support Vector Machines, frequently encounter difficulties with feature extraction and imbalances within datasets. On the other hand, current deep learning solutions, while exhibiting greater accuracy, are hindered by issues such as substantial computational demands, mode collapse during Generative Adversarial Network training, and challenges in generalizing across varied datasets. These shortcomings underscore the necessity for a more resilient and adaptable spam detection system that can effectively tackle these deficiencies. Therefore, governments and organizations look for practical solutions to protect users and differentiate between legitimate messages and spam. In Saudi Arabia, the government has launched dedicated authorities for cyber security to set rules and regulations to protect its assets and citizens as well [28,29]. These authorities are the Communications, Space, and Technology Commission (CST) and the National Cybersecurity Authority (NCA). One of these rules and regulations is to place and use Email gateway solutions to secure Email platforms, which can be found under categories 4.15 in [28] and 2.4 in [29]. Hence, this paper suggests a model to secure Emails based on Artificial Intelligence technologies: DaC-GANSAEBF and encompasses a dedicated Light Dual Attention (LDA) to get features from scanned words and improve the prediction's accuracy.

1.4 Research Motivation and Contributions

The idea of this article stems from a promising vision in Saudi Arabia, which is Vision 2030. This vision was approved and appeared in 2016, and it aims to improve the economy of the country, provide sustainable resources, enhance the quality of life for its citizens and residents, and protect assets from unauthorized access or attacks. Therefore, this study seeks to accomplish the following objectives:

1. To create an innovative spam detection framework, DaC-GANSAEBF, which combines Divide and Conquer, Generative Adversarial Networks (GAN), Semantic Embedding Analysis (SEA), and Latent Dirichlet Allocation (LDA) methodologies.
2. To tackle significant shortcomings of current methods, such as issues related to dataset imbalance and the ability to generalize across various datasets.
3. To attain leading performance metrics by training and validating the framework on several benchmark datasets.
4. To offer a scalable solution that can be seamlessly integrated into real-world email systems, thereby improving user security.

The primary contributions of this research are outlined as follows:

A. Innovative Framework: The introduction of the DaC-GANSAEBF model, which combines GAN, SEA, and LDA to enhance the detection of spam emails.
B. Enhanced Dual Attention Mechanism: The incorporation of LDA to proficiently capture both spatial and temporal dependencies within email data.
C. Resolution of Dataset Imbalance: The application of a distinctive balancing strategy aimed at enhancing model performance and generalization capabilities.
D. Thorough Evaluation: The execution of comprehensive experiments across four benchmark datasets to assess the model's effectiveness.
E. Benchmark Comparison: The demonstration of the DaC-GANSAEBF model's superiority over current spam detection methods in terms of accuracy, precision, and robustness.

The rest of this paper is constructed as: in Section 1.5, a literature review of several built solutions based on ML or DL approaches is presented. In Section 2, the problem statement, explanation of the applied and used datasets, and the proposed DaC-GANSAEBF method methodology are described. Section 3 provides deep details of the performed experiments, the carried-out simulation setup with the compulsory assets, and the assessment of the evaluated performance indicators, while Section 4 provides a discussion of the attained outcomes. In conclusion, Section 5 presents the conclusion and potential directions for future work.

1.5 Literature Review

The identification of spam emails has been the subject of considerable research, resulting in the development of various solutions that utilize Machine Learning (ML) and Deep Learning (DL) methodologies. This section offers a comprehensive overview of recent progress in this field and examines the shortcomings of current strategies.

Kumar et al. [4] developed a method to identify spam Emails based on various machine-learning technologies, such as Naïve Bayes (NB), Decision Tree (DT), Support Vector Machine (SVM), and Random Forest (RF). The authors concluded that the random forest algorithm achieved the highest results and surpassed other deployed approaches. The Term Frequency-Inverse Document Frequency (TF-IDF) technology was applied to find relations between words to support the identification process. This method contained four major steps, which were (1) assigning a unique token number to each incoming message to replace sensitive data with an identifier that retains all sensitive data to start processing, (2) estimating probabilities,

(3) extracting features, and (4) applying the Naïve Bayesian Classifier to differentiate between spam and legitimate messages. The authors evaluated efficiency and productivity by removing common words, such as to, pronouns, and conjunctions. However, two words: (to and your) are found in most of the spam messages, according to authors in [4]. In addition, the Exploratory Data Analysis (EDA) was utilized by the authors to leverage their developed method. Ten machine-learning approaches were considered to evaluate two performance indicators, which were accuracy and precision on a dataset. 97.6% accuracy was the highest reached result using the random forest algorithm, while the precision was 98.3% for the same algorithm. Unfortunately, no information about how features were extracted was provided or how the probability was calculated as well. On the other hand, the presented method: DaC-GANSAEBF contains two known topologies to identify spam messages after training it on four datasets. It achieved a better accuracy of 98.87%.

Choudhary [5] built a model to classify messages into safe or spam using an analytical-based approach. This approach relied on a Bayes theorem and a naïve Bayesian classifier. In addition, finding IP addresses was included as well. The author used a web application and a machine-learning tool to build the model. This approach required registration from users to keep records for users. The author provided no information about the developed model. Thus, it is difficult to criticize this approach. This model contained a tool called Wordcloud, which is a handy visualization tool to show most of the common words to be analyzed. In addition, the Term Frequency-Inverse Document Frequency (TF-IDF) Vectorizer was deployed along with the naïve Bayesian classifier to extract characteristics and achieved nearly 93% accuracy. In contrast, the proposed DaC-GANSAEBF approach depends on two deep-learning technologies to identify spam messages and protect recipients. Four datasets are applied to train the model and evaluate its performance metrics. DaC-GANSAEBF achieved better accuracy as it reached more than 97% accuracy, which surpasses the developed model in [5].

In [6], Sharma et al. compared various methods to identify spam messages using different sources to collect data. The authors utilized a dataset from Kaggle, and this dataset contains 5572 records and 4 attributes. Firstly, the authors cleaned the used data by removing unneeded information, renamed the remaining data accordingly, and deleted duplicated data. Secondly, labeling these data took place as 0 referred to safe messages and 1 denoted the spam messages. The EDA method was applied to classify types of messages in the dataset and the authors found that about 87% were safe messages and the remaining were spam ones. In addition, the authors added three additional attributes, which were the number of characters, the number of words, and the number of sentences. Furthermore, four statistical parameters were deployed for the added attributes, and these parameters were mean, standard deviation, minimum, and maximum. Moreover, the Word cloud algorithm was used to get the most common spam words to compare with. The authors utilized several machine learning algorithms to evaluate their model for two performance indicators, which were accuracy and precision. The highest attained values for both indicators were reached by the Naïve Bayes (NB) algorithm: 98.3% accuracy and 100% precision. In contrast, DaC-GANSAEBF uses two deep-learning structures to identify spam messages using four datasets. This approach was trained extensively using more than 10,000 iterations and 75 epochs. DaC-GANSAEBF encompasses various tools, such as TF-IDF. DaC-GANSAEBF was evaluated based on numerous performance indicators. The attained outcomes ranged between 96% to nearly 99%. The obtained findings implied that DaC-GANSAEBF outperforms the developed model in [6].

Reddy et al. [7] implemented a model to detect Email spam using machine learning, NLP, and a dataset. The authors applied several methods to preprocess the utilized data, which were tokenization, stop word removal, and stemming. NB, SVM, and RF algorithms were the machine-learning algorithms that the authors deployed. In addition, hyperparameter tuning was used, and four performance indicators were evaluated: accuracy, precision, recall, and F-score. The authors provided no information regarding the

developed method or its flow chart. The authors claimed that 98% were an average achieved accuracy from all considered ML algorithms. On the other hand, the presented approach: DaC-GANSAEBF was trained using four datasets to evaluate its effectiveness according to several performance indicators. DaC-GANSAEBF achieved acceptable output for all performance indicators. More details are presented in Section 4.

Fatima et al. [8] implemented a model to identify spam messages based on two ensemble techniques to extract characteristics. These two techniques were Count-Vectorizer and TFIDF-Vectorizer. Three public datasets were applied to the developed model to train it. Ling Spam, the University of California Irvine (UCI) Machine Learning Repository SMS Spam, and a proposed dataset were the three public datasets that were utilized by the authors. In addition, the authors deployed twelve machine-learning algorithms, such as NB, SVM, and RF after integrating them with the two ensemble techniques to detect spam messages. The authors used tokenization, lemmatization, and stemming tools to preprocess the used public datasets after cleaning and then split these data into two groups: training and testing. The training set had 80%, while the rest were for the testing set. The cleaning process was essential to remove undesired characters, spaces, and digits. The third dataset was created using web scrapping and a personal email. All three datasets were in Comma-separated Values (CSV) format and all messages were labeled either ham or spam. Ham denotes safe messages. The authors got an average accuracy of around 97.6% on all three datasets. In contrast, the presented model, DaC-GANSAEBF, contains two deep-learning topologies as stated earlier. This approach was trained using four public datasets and achieved better accuracy results. Accuracy varied between 97.3% and nearly 98.9% for 7000 iterations. The outcomes achieved by DaC-GANSAEBF proved that it is better than the implemented method in [8].

Bhargavi [12] developed an approach to identify spam messages based on machine learning and deep learning methods. These methods were Naïve Bayes (NB), SVM, and a Convolutional Neural Network (CNN). Unfortunately, no information regarding the internal architecture of the deployed CNN was provided by the authors nor what hyperparameters were applied. The author claimed that CNN reached the highest accuracy of 99.2%. It would be better if additional information was provided to criticize [12]. However, DaC-GANSAEBF was trained on four datasets and reached accuracy between 96% and nearly 99% when the number of applied iterations exceeded 7000 and 80 epochs.

Nooraee et al. [13] developed an optimized method for spam message detection based on deep-learning topologies. The authors utilized the Long Short-Term Memory (LSTM) and Glove Word Embedding (GWE) technologies. This approach used text word vectors to identify spam messages with an average accuracy of around 98.9% on two datasets. One dataset had 5570 records for spam and non-spam messages, while the other one had 5726 records, 1368 of which were spam messages. The authors split the used datasets into two classes: legitimate and spam. Then, preprocessed data, this preprocessed procedure included data cleaning and transformation using stemming, removing stop words, and removing spaces, symbols, and special characters. The stemming procedure reduced different types of words into general types to reflect a full understanding of semantic meaning. The authors represented every word by a token and stored all tokens in a vector. The feature extraction process was performed after removing specific words by converting words into integers or floating points. These numbers were the input for the implemented method. The authors used a middle word method to predict and count the number of words to build their model. The model was trained for 30 periods and a loss function was considered as a metric to be evaluated. In addition, Adam optimizer was deployed with a learning rate of 0.01 and a batch size of 256. In contrast, DaC-GANSAEBF depends on two deep-learning structures to identify spam messages to provide protection for recipients and allow smooth information sharing without the burden of unwanted messages. DaC-GANSAEBF was trained using four datasets and reached a good accuracy between 97% and 99%, which is better than what was achieved by the authors in [13].

Malhotra et al. [16] used machine learning and deep learning methods to detect spam Emails using a dataset. These methods were a Sequential Neural Network, LSTM, and Bidirectional Long Short-Term Memory (Bi-LSTM). The dataset was divided into two sets, which were training and testing. The training set was 80%, while the rest was for the testing. For the preprocessing procedure, tokenization, stemming, lemmatization, and stop word removal processes were applied to prepare the used data. The utilized dataset had 5171 records for safe and spam messages. Spaces, punctuation, symbols, and special characters were removed to provide high-quality data. Since the dataset was imbalanced, the authors resolved by matching the number of records for both sets, which became around 1500 messages. The extracted features operation was conducted by converting inputs into integers for encoding purposes. The authors reached a high accuracy of around 98.5%. In contrast, the proposed approach: DaC-GANSAEBF uses four datasets for training, validation, and testing purposes. In addition, various procedures are involved, and it achieved accuracy between 97% and 99% in some cases, when the number of iterations and epochs increased.

AbdulNabi et al. [17] developed a method to detect spam messages using different deep-learning approaches, which were Bi-LSTM, LSTM, and Bidirectional Encoder Representations from Transformers (BERT). Two datasets were applied, and the authors achieved 98.67% accuracy. In addition, some machine learning techniques were deployed as well.

Since a few implemented methods to identify spam messages used DL technologies, despite the progress made in technology and the establishment of more stringent regulations, spam remains a considerable issue, leading to annual financial losses amounting to billions of dollars due to reduced productivity and security risks. Although traditional rule-based and filter-based methods were once effective, they have proven insufficient in countering the increasing sophistication of spammers. The advent of Machine Learning (ML) and Deep Learning (DL) technologies has significantly altered the spam detection landscape. These methodologies employ advanced pattern recognition and feature extraction techniques to enhance detection accuracy. However, challenges such as imbalanced datasets, high computational requirements, and the ongoing evolution of spam strategies continue to impede their overall efficacy. Considerable challenges remain in overcoming the limitations of current solutions. For example, conventional machine learning techniques such as Naïve Bayes and Support Vector Machines (SVMs) frequently encounter difficulties with imbalanced datasets, which can lead to biased classifications and inadequate generalization across varied datasets. Although sophisticated deep learning models like LSTM and BERT have demonstrated potential for achieving higher accuracy rates, they often entail substantial computational expenses and lack the flexibility to adapt to the continuously changing tactics employed by spammers. This research seeks to address these obstacles by proposing a novel, adaptive framework for spam detection. Significant advancements have been achieved in the realm of spam detection; however, existing methodologies exhibit several notable deficiencies: 1. Dataset imbalance: many approaches fail to adequately address the inherent imbalance between legitimate and spam messages, leading to biased classification results. 2. Generalization across datasets: most models are evaluated on a single dataset, raising concerns about their effectiveness on previously unencountered data. 3. High computational costs: advanced deep learning models, including LSTM and BERT, require considerable computational power, making them impractical for real-time applications. 4. Evolving spam techniques: traditional machine learning and certain deep learning models struggle to adapt to the swiftly changing spam strategies, limiting their long-term effectiveness. Current trends emphasize the necessity of integrating diverse deep-learning architectures to alleviate individual shortcomings. For instance, hybrid approaches that combine attention mechanisms with convolutional neural networks (CNNs) or generative adversarial networks (GANs) have shown promise in enhancing feature extraction and overall resilience. Nonetheless, issues such as mode collapse in GANs and the insufficient exploration of attention mechanisms remain inadequately addressed.

Significant advancements have been achieved in the realm of spam detection; however, existing methodologies exhibit several notable deficiencies: 1. Dataset imbalance: many approaches fail to adequately address the inherent imbalance between legitimate and spam messages, leading to biased classification results. 2. Generalization across datasets: most models are evaluated on a single dataset, raising concerns about their effectiveness on previously unencountered data. 3. High computational costs: advanced deep learning models, including LSTM and BERT, require considerable computational power, making them impractical for real-time applications. 4. Evolving spam techniques: traditional machine learning and certain deep learning models struggle to adapt to the swiftly changing spam strategies, limiting their long-term effectiveness. Current trends emphasize the necessity of integrating diverse deep-learning frameworks to alleviate individual shortcomings. For instance, hybrid approaches that combine attention mechanisms with convolutional neural networks (CNNs) or generative adversarial networks (GANs) have shown promise in enhancing feature extraction and overall resilience. Nonetheless, issues such as mode collapse in GANs and the insufficient exploration of attention mechanisms remain inadequately addressed. The advancement of spam detection systems has been a focal point of ongoing research, leading to the proposal of various methodologies. The proposed DaC-GANSAEBF framework seeks to mitigate these issues by:

1. Employing customized segmentation strategies through Divide and Conquer (DaC) to enhance computational efficiency and effectively manage imbalanced datasets.
2. Incorporating Squeeze and Excitation (SAE) modules that improve feature extraction by selectively enhancing pertinent patterns while diminishing noise.
3. Implementing a Light Dual Attention (LDA) mechanism that captures both spatial and temporal dependencies within the data, allowing for a more detailed analysis of spam patterns.

This comprehensive approach offers a robust solution to the ongoing challenges identified in previous research. The framework aims to proficiently discern complex patterns within textual data, thereby improving both accuracy and robustness.

Table 1 provides overall research studies between these methods regarding topology, solved issues, and outputs.

Table 1: A summary of some deep learning methodologies presented in existing literature

Reference	Methodology	Problem solved	Outcomes
[4] Kumar et al., 2023	Naive Bayes (NB), Decision Trees (DT), Support Vector Machine (SVM), and Random Forests (RF)	Email fraud detection	90% accuracy
[5] Choudhary, 2023	Naive Bayes (NB)	Spam detection	93.2% accuracy
[6] Sharma et al., 2023	Naive Bayes (NB) and Support Vector Machine (SVM)	Spam detection	98.3% accuracy
[7] Reddy et al., 2023	Naive Bayes (NB), Support Vector Machines (SVMs), and Random Forests (RF)	Spam detection	98% accuracy
[8] Fatima et al., 2023	Naive Bayes (NB), Logistic Regression (LR), Extra Tree, Stochastic Gradient Descent (SGD), XG-Boost, Support Vector Machine (SVM), Random Forest (RF), and Multi-Layer Perception (MLP)	Spam detection	98.12% accuracy

(Continued)

Table 1 (continued)

Reference	Methodology	Problem solved	Outcomes
[12] Bhargavi, 2022	Two ML approaches: NB and SVM One DL approach: a CNN	Spam detection	97.5% accuracy
[13] Nooraee et al., 2022	LSTM and Glove Word Embedding (GWE) technologies	Spam detection	98.9% accuracy
[16] Malhotra et al., 2022	ML approaches, a Sequential Neural Network, LSTM, and Bi-LSTM	Spam detection	98.5% accuracy
[17] AbdulNabi et al., 2021	Bi-LSTM, LSTM, and Bidirectional Encoder Representations from Transformers (BERT)	Spam detection	98.62% accuracy
[23] Kumar et al., 2020	Naive Bayes (NB), Decision Tree (DT), AdaBoost, K-Nearest Neighbors, Support Vector Machine (SVM), Random Forest (RF), and Bagging	Spam detection	98% accuracy

2 Materials and Methods

The majority of the existing methodologies for spam detection have employed machine learning techniques, with deep learning being utilized in studies such as [12,16]. In contrast, the two studies referenced as [13,17] relied exclusively on deep learning approaches, achieving accuracies of 98.6% and 98.9%, respectively. This indicates a necessity for a dedicated deep-learning solution that can surpass the results obtained in [13,17]. In response to this need, the present article introduces a deep learning framework named DaC-GANSAEBF, designed to identify spam messages effectively. This framework aims to safeguard recipients and facilitate a seamless information-sharing environment on the Internet. Additionally, it incorporates the LDA technique to extract features and improve the accuracy of predictions.

The components of DaC-GANSAEBF were carefully selected for their unique capabilities in tackling the challenges associated with spam detection, which can be summarized as follows:

- Divide and Conquer (DaC): This algorithm effectively breaks down extensive datasets into smaller, more manageable subsets, allowing the framework to concentrate on localized patterns and enhance computational efficiency. By isolating specific segments, the model minimizes the likelihood of overfitting and promotes improved generalization.
- Generative Adversarial Networks (GAN): GANs bolster the model's capacity to replicate and respond to various spam tactics by producing synthetic examples that closely resemble actual spam patterns. This adversarial training methodology strengthens the framework's resilience against emerging spam strategies.
- Squeeze and Excitation (SAE): SAE modules are essential in highlighting the most relevant features while diminishing the impact of irrelevant data. By recalibrating feature responses on a channel-wise basis, SAE ensures that the model remains attuned to critical spam-related patterns.
- Light Dual Attention (LDA): The LDA mechanism connects spatial and temporal dependencies, allowing the framework to discern complex relationships among email attributes. This dual-attention approach enhances both the interpretability and accuracy of the spam detection process.

2.1 Public Benchmarks

This study employs four publicly accessible benchmarks/datasets sourced from Kaggle, as referenced in [30–33]. The initial dataset [30] comprises 5172 records categorized as spam and non-spam messages. Within this dataset, there are 1500 spam messages, accounting for 29%, while the remaining 3672 messages are classified as safe, representing 71%. In this context, a value of 0 indicates spam and a value of 1 signifies legitimate messages. The dataset consists of four columns, including two integer fields and two string fields. The second dataset [31] contains 2893 records sourced from the Linguist List, a forum dedicated to discussions on linguistic topics, such as job openings, research opportunities, and software-related matters. This dataset includes 2412 legitimate messages and 481 spam messages, where 0 indicates legitimate and 1 denotes spam. The third dataset [32] features 5574 records of English SMS messages, with 4849 classified as safe and 725 as spam. The final dataset [33] totals 5169 records, comprising 672 spam messages and 4497 legitimate messages, which represent 17% and 83% of the total, respectively. A summary of the datasets utilized in this research is presented in Table 2. As indicated in Table 2, all four datasets exhibit an imbalance, with a significantly higher number of legitimate messages, necessitating a resolution to balance the counts of both categories. Fig. 1 provides a visual representation of the four datasets following the adjustment of message types. The adjusted message counts are divided into three groups: training, validation, and testing, with the training group constituting 67%, the validation group 10%, and the testing group 23%.

Table 2: A description of the applied datasets

Message type	The first dataset [30]	The second dataset [31]	The third dataset [32]	The fourth dataset [33]
Legitimate	3672	2412	4849	4497
Spam	1500	481	725	672
Total number of records	5172	2893	5574	5169

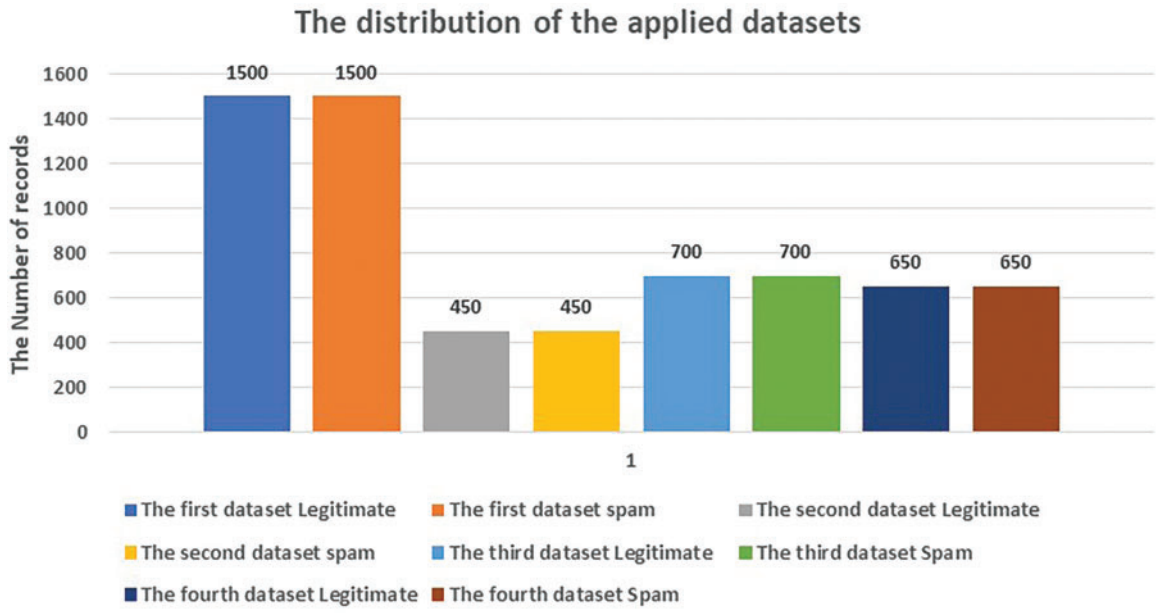

Figure 1: The distribution of the applied datasets after resolving the imbalance problem

2.2 Background of the Developed Technologies

This section provides a background about the utilized technologies in this study. In Sections 2.2.1 and 2.2.2, a brief about the typical spam Email detection and identification and the Divide and Conquer approach are presented, followed by a background of GAN and SAE in Sections 2.2.3 and 2.2.4, respectively. Lastly, Section 2.2.5 describes the existing Dual Attention (DA) model.

2.2.1 Typical Spam Email Identification Mechanism

A standard mechanism for identifying spam messages operates based on a set of established rules and filters. Generally, four key procedures are involved, which can be outlined as follows:

(1) Content filter: each word within the message is examined to identify any potential spam-related terms.
(2) Header filter: the header information of each message is analyzed to detect any spam-associated words.
(3) Blacklist filter: an extensive review is conducted against a commonly used blacklist of sources known for sending spammy content to determine the origin of the messages.
(4) Rule-based filter: this filter identifies users by utilizing stored identification data specific to each service provider, along with parameters defined by the subject line and the organization.

Fig. 2 illustrates a typical spam message detection mechanism utilized in various email platforms, illustrating the sequential processing of emails through various filtering stages. The diagram emphasizes the hierarchical structure of the process, where each filter serves a distinct function in identifying potentially harmful attributes. This methodology is both straightforward and computationally efficient, which contributes to its prevalent use in traditional email systems. However, this mechanism is not without its drawbacks: 1. Static rules: conventional filters depend on fixed rules that can be easily bypassed by spammers who modify keywords or sender details. 2. Limited contextual understanding: the filters are incapable of assessing the contextual relationships among words, rendering them ineffective against more sophisticated spam tactics such as phishing or social engineering. 3. High false positives: genuine emails that contain keywords resembling those found in spam messages are frequently misclassified, resulting in user dissatisfaction. 4. Inability to adapt: these systems cannot learn or adjust to emerging spam patterns without manual updates, rendering them obsolete in the face of rapidly changing spam techniques. These shortcomings highlight the necessity for more sophisticated and adaptive solutions, such as the proposed DaC-GANSAEBF framework, which utilizes deep learning to address the rigid and static characteristics of traditional systems. By incorporating dynamic attention mechanisms and generative networks, DaC-GANSAEBF is capable of analyzing intricate patterns and relationships within email data, thereby mitigating the limitations of standard spam detection systems.

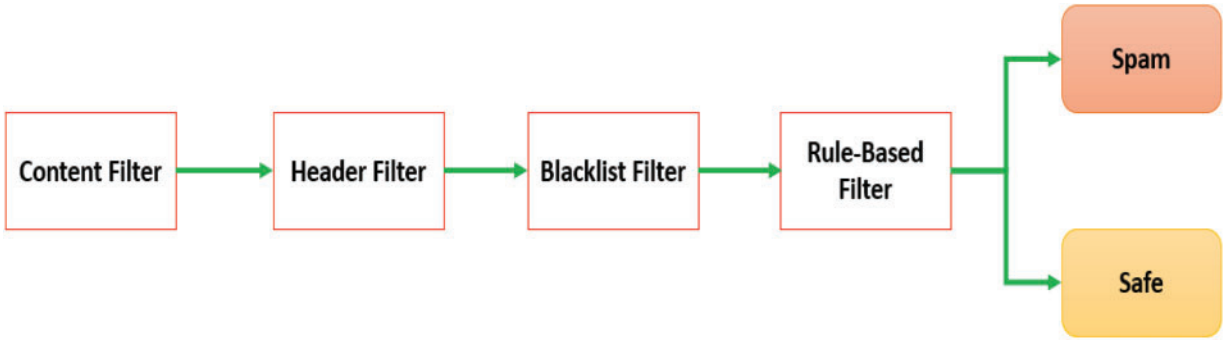

Figure 2: The general spam message detection mechanism

2.2.2 Divide and Conquer (DaC) Algorithm

The Divide and Conquer (DaC) algorithm represents a core computational methodology that addresses intricate problems by decomposing them into smaller, more manageable sub-problems. Each of these sub-problems is resolved independently, and their respective solutions are subsequently merged to yield the final outcome. This method proves particularly advantageous in situations involving extensive datasets or tasks that require significant computational resources, as it mitigates complexity and facilitates parallel processing. In the realm of spam email detection, DaC is instrumental in improving both efficiency and scalability. Conventional spam detection systems often encounter difficulties when handling substantial volumes of email data, as the computational demands grow exponentially with the increase in data size. By utilizing DaC, the proposed DaC-GANSAEBF framework segments email data into smaller portions, processes these portions independently through the Generative Adversarial Network (GAN) and Squeeze and Excitation (SAE) modules, and then consolidates the results into a unified classification outcome.

The significance of the DaC algorithm within the proposed methodology is attributed to its capacity to: 1. Enhance computational efficiency: by enabling the parallel processing of smaller data subsets, the framework effectively minimizes both training and inference durations. 2. Improve model generalization: segmenting the dataset allows the model to concentrate on localized patterns within each section, thereby mitigating the risk of overfitting and enhancing its generalization capabilities across varied datasets. 3. Facilitate dynamic feature extraction: each segment is subjected to independent feature extraction, which allows the model to identify distinct characteristics that may be obscured when analyzing the entire dataset collectively. Numerous studies have validated the effectiveness of DaC in boosting computational performance and accuracy across different fields. For example, DaC was implemented as a strategy for big data classification, resulting in substantial reductions in processing time while preserving high accuracy levels. Likewise, in the realm of image processing, DaC has been employed to partition high-resolution images into smaller patches, leading to more efficient processing and enhanced detail retention.

In the DaC-GANSAEBF framework, the algorithm serves as a fundamental element, facilitating the effective management of email data on a large scale. The framework segments email content into smaller components, processes each segment using GAN and SAE modules to extract relevant features, and integrates the intermediate outcomes through a lightweight dual attention mechanism to generate the final prediction. This methodology guarantees that even the most nuanced features within the email data are captured and analyzed proficiently, thereby enhancing the model's overall performance. By employing the Divide and Conquer algorithm, the proposed framework not only tackles the computational difficulties associated with conventional spam detection techniques but also enhances the system's overall accuracy and adaptability.

2.2.3 Generative Adversarial Network (GAN)

Generative Adversarial Network (GAN) is a category of neural network that comprises two primary elements: the Generator (G) and the Discriminator (D) [34,35]. The role of the generator is to create synthetic data to deceive the discriminator, which in turn, endeavors to differentiate between genuine and fabricated data. The generator is fed input from a singular source that consists of synthetic data, whereas the discriminator obtains input from two distinct sources, one being synthetic data and the other being authentic data. Fig. 3 presents a general schematic representation of a standard GAN.

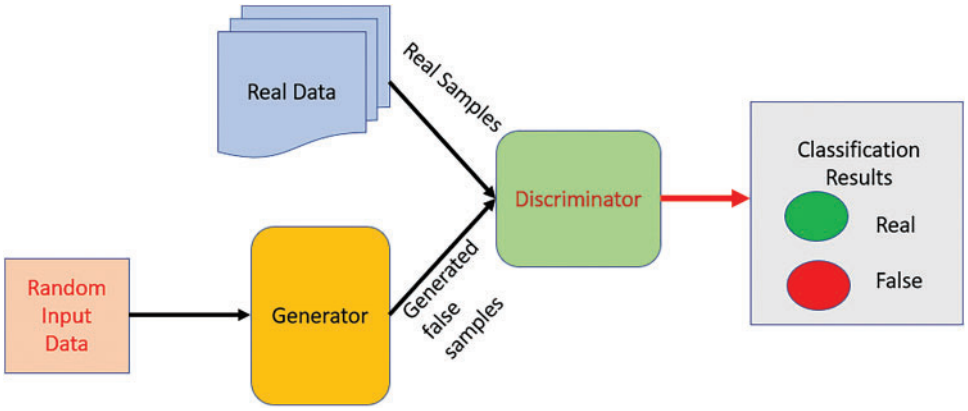

Figure 3: The typical structure of GAN

2.2.4 Squeeze and Excitation Network (SAE)

The Squeeze and Excitation (SAE) network was implemented in 2018 and encompasses two main units, which are a squeeze network and an excitation network. The squeeze network includes a convolutional neural network layer and a global average pooling layer to cumulative spatial features into a channel feature and the excitation network acquires instance weights from the squeezed feature to generate a new weight for every channel [36]. The reweight process is performed by adding a constraint control variable that imposes the original weights to be recalculated. In addition, an activation function is applied in the SAE block. This activation function is the Rectified Linear Unit (ReLU). Fig. 4 depicts a general structure of the SAE network.

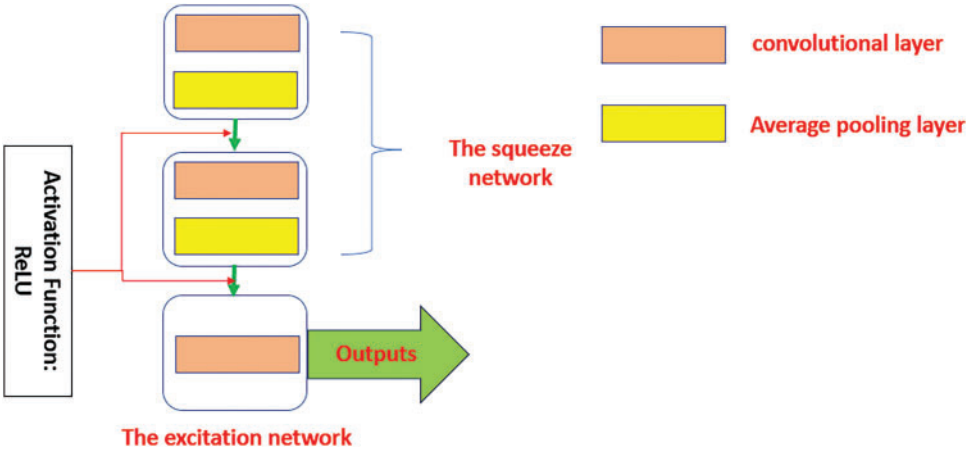

Figure 4: The typical structure of the SAE network

The SAE network is deployed in this study to emphasize the extracted features from the GAN components to advance the performance indicators. This emphasis operation is achieved by integrating the SAE network into the discriminator, so it becomes one of its internal elements. In addition, the original SAE network architecture is customized by adding additional layers, a fully connected layer, and a Sigmoid activation function. Section 2.3.1 provides a comprehensive explanation of the developed SAE network and its integration into the proposed architecture.

2.2.5 Dual Attention Module (DA)

Earlier versions of the Convolutional Neural Networks (CNNs) concentrated on maximizing the depth and width [37–39]. This feature allowed the extracted features to be treated equally at all levels. Consequently, this process causes a lack of flexibility in all extracted feature maps. Hence, a dual attention mechanism resolves this issue by paying more attention to the useful extracted data [37]. In addition, efficient computing resources can be easily allocated. Fig. 5 depicts a general structure of the DA module. Interested readers can refer to [37–39] for additional information. The internal architectures of the spatial and channel attentions are omitted in Fig. 5; however, readers are advised to [37–39] for more information.

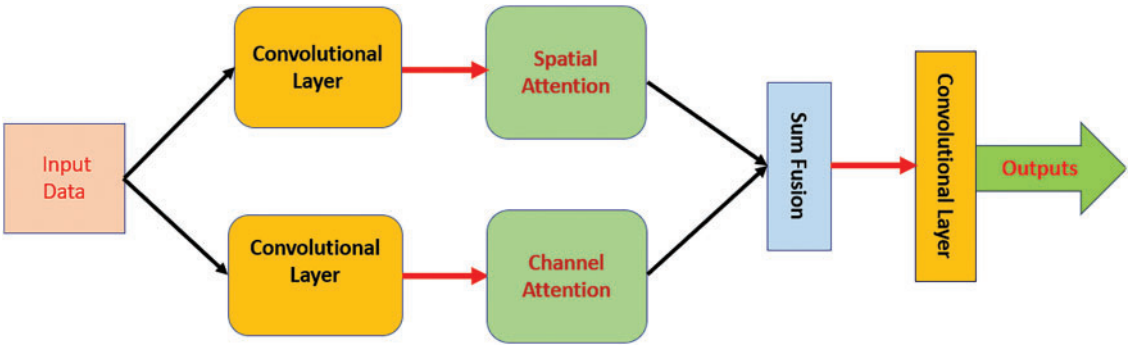

Figure 5: The general structure of the DA module

A typical dual attention module works to determine the important features from the inputs and assign proper resources through two components, which are the global and local mechanisms. These two mechanisms produce attention weights using certain relations between neighborhood data through pairs of queries and keys.

2.3 The Proposed Methodology

A full detail of the presented solution: DaC-GANSAEBF is provided in this Subsection. Currently, Artificial Intelligence (AI) technologies are widely deployed in numerous fields due to their capabilities to generate promising outcomes. These technologies can be used to solve complex problems. However, the entire structures of most of the AI solutions are complicated and very hard to understand and demonstrate. AI is less applied in spam detection. Various AI solutions were developed with ML methods for spam detection purposes.

2.3.1 DaC-GANSAEBF

As stated earlier, the proposed approach, which is DaC-GANSAEBF, stands for Divide and Conquer-Generative Adversarial Network Squeeze and Excitation-Based Framework. This approach uses artificial intelligence-based topologies to identify spam messages as several components are held and integrated into one compact framework. The SAE network is integrated inside the discriminator to improve the extracted features. Fig. 6 illustrates a high overview structure of the developed GAN block, and Fig. 7 depicts an internal architecture of the developed SAE network block. In Fig. 6, the generator contains two convolutional layers with a kernel size of 4 × 4, one average pooling layer of size 2 × 2, and a flattened layer, while the discriminator includes five convolutional layers of kernel size 5 × 5, one max-pooling layer of size 3 × 3, and the developed SAE network block. In addition, the outputs from the discriminator if fed into another

convolutional layer of kernel size 3 × 3. In Fig. 7, H refers to height, W denotes the width, and C represents a channel number. In this study, the total number of applied channels (c) is 8. Furthermore, three fully connected layers of different sizes are deployed in Fig. 7. Each fully connected (FC) layer is associated with its number of neurons, where the kernel size and the number of neurons are different. The first fully connected layer has 5 × 5 with 200 neurons, the second fully connected layer includes 150 neurons with a kernel size of 4 × 4, and the last fully connected layer contains 100 neurons with size of 3 × 3. Fig. 8 shows a high overview of the proposed DaC-GANSAEBF approach. As depicted in Fig. 7, the proposed DaC-GANSAEBF method is composed of three main components, where every stage is associated with its unique colors for simplicity. In addition, every level of layers in Figs. 6–8 denotes a sophisticated resolution since every level is seen as a box with its dedicated size. This size, also referred to as the dimension, is expressed by three letters: H × W × N. H represents the height, W indicates the width, and N expresses the number of applied neurons. The presented DaC-GANSAEBF approach uses a linear projection procedure to project data into a constant dimensional space. After that, a patching agent of size 16 is applied. This agent allows every patch to be coupled with three parameters, which are Query (Q), Key (K), and Value (V). These parameters are generated and acquired during the training stage. All values for all pairs of Q and K are determined by a dot product to decide the relations between spatial data if applicable. Then, the data are sent into a Multi-Layer Perceptron (MLP) of three Gaussian error Linear Units (GeLU), where each unit involves a convolutional layer of kernel size 3 × 3 and a downsampling element. Every unit of GeLU combines the linear and sigmoid activation functions to set non-linearity by assigning 0 and 1 to inputs. In Fig. 7, GAP denotes a Global Average Pooling layer.

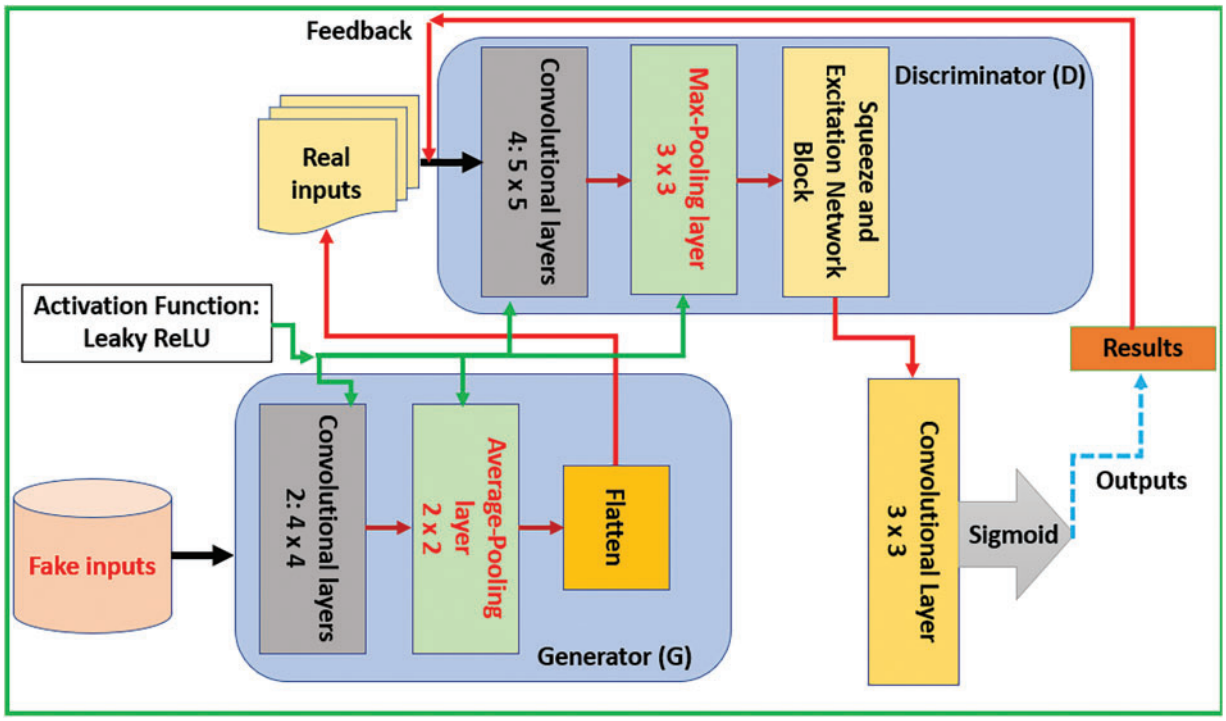

Figure 6: The general architecture of the implemented GAN block

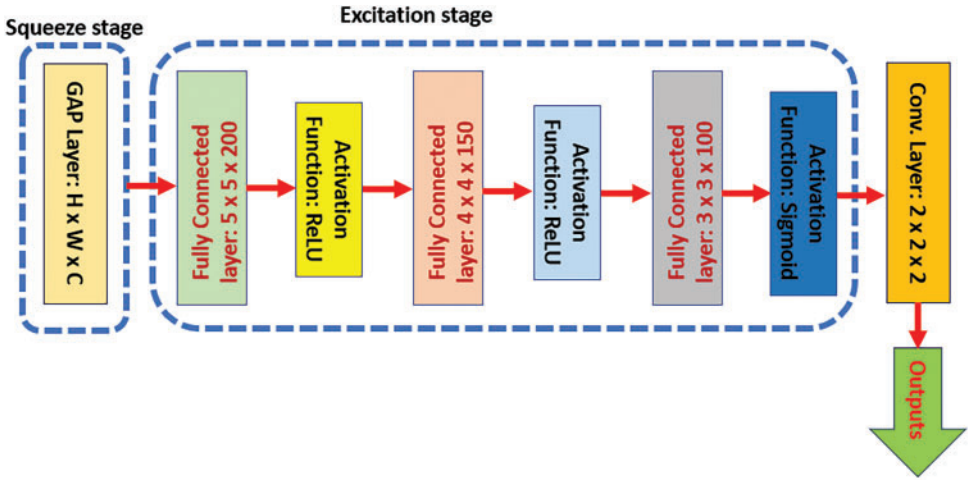

Figure 7: The internal structure of the implemented SAE network

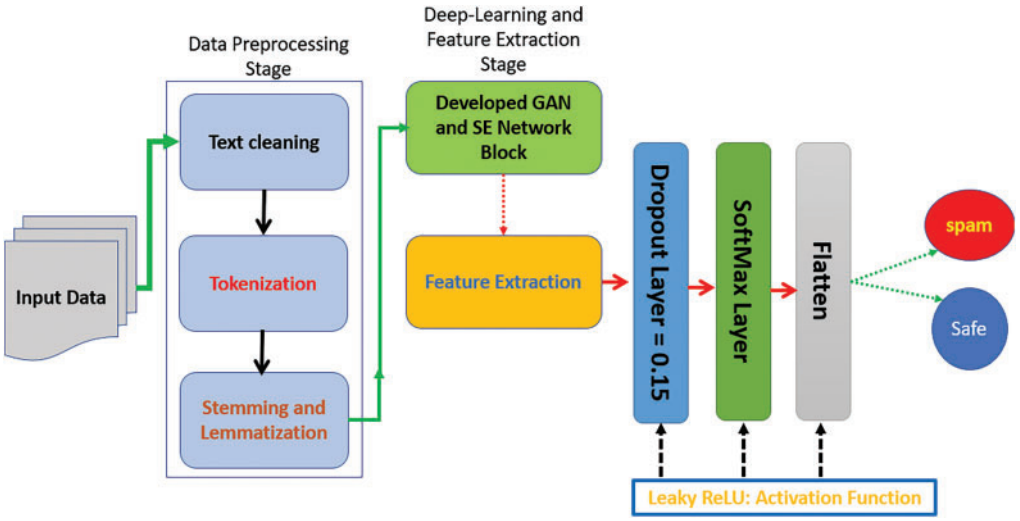

Figure 8: The flow chart of the DaC-GANSAEBF approach

Eq. (1) represents a mathematical expression of GELU's output.

$$GeLU = \frac{i}{2} \times \left[1 + \tanh\left(\sqrt{\frac{2}{i}} \times \left(i + 0.044715 \times i^3\right)\right)\right] \tag{1}$$

where i denotes inputs. In addition, Gated Position-Sensitive Axial Attention (GPSAA) and the Local-Global training (LoGo) methods are employed in the presented DaC-GANSAEBF model to enhance accuracy by increasing the data segmentation processes. Furthermore, GPSAA determines the relationship between the feature extraction process and the effectiveness of the calculations that are carried out by GPSAA. In this study, LoGo is applied to get local and global features, and a value of 0.25 for a decay factor is employed to refresh the model if no features are obtained for 0.25 s. Table 3 lists the utilized hyperparameters inside the presented DaC-GANSAEBF model and their assigned numerical values.

Table 3: Hyperparameters of DaC-GANSAEBF

Name of the hyperparameter	Value
Learning rate (L)	0.1, 0.001
Batch size	24, 16, 10, and 8
Dropout	0.15
Optimizer	Adam
Regression weight	0.01
Momentum	0.75
Activation functions	ReLU, Leaky ReLU, and Sigmoid
Number of iterations	5000, 7000, and 10,000
Number of epochs	40, 65, 90
Loss function	Binary cross entropy
Embedding size	150
Number of embedding layers	4

2.3.2 Light Dual Attention Network (LDA)

In this article, we propose a new Light Dual Attention (LDA) network module to leverage the prediction's ability and accuracy through learning and capturing the dependencies between words among the spatial and temporal channels. In addition, this LDA structure contains a new cross-attention module to capture co-occurring attention feature maps and learn a generated semantic representation through various channels. Lastly, a discrepancy in the generated feature maps is minimized by using the deployed loss function as shown in Table 3. Moreover, both attention mechanisms are incorporated together through a fused gate. The developed cross-attention module determines the cross-correlations between feature maps of all input data using the pairwise query and key in both spatial and temporal aspects to support the presented approach to produce the co-occurrence maps. This operation gives the approach the required robustness in its identification and isolation process to capture all potential spam messages and increase accuracy as well. To increase the performance of the presented method, considering different feature maps and their roles in capturing dependencies between words is critical and crucial as well. The developed LDA serves this purpose. In this study, we use spatial and temporal-wise statistics on every channel to rescale all generated feature maps and model dependencies. From each applied channel, the extracted feature map is achieved from all pairs of queries and keys with the same weights. Due to the different values of pairs and different data, the LDA module can detention discriminative features and dependencies. In every attention module, the GAP layer is deployed to obtain statistical information from the extracted features. Then, the fusion gate aggregates and concatenates both results into one compact output. This fusion gate structure contains a convolutional layer of size $3 \times 3 \times 2$, one ReLU activation function, and two fully connected layers with 50 neurons in each layer. In addition, normalization is performed in this stage. Fig. 9 illustrates a block diagram of the developed LDA module, which resides in the feature extraction block in Fig. 8.

2.4 Prediction Methodology

As shown in Fig. 8, the presented DaC-GANSAEBF approach encompasses two main components: the first component is shown in blue color, which denotes the preprocessing stage and the second component is represented in green color, which denotes the deep-learning stage, which contains the characteristics extraction in orange color. In the preprocessing stage, three main operations take place, which are text

cleaning, stemming, and lemmatization. The text cleaning procedure is a very crucial operation since it removes unwanted letters or characters, such as spaces, punctuation, stop words like a, about, and so on, and special symbols and characters. Then, the stemming operation takes place, and this procedure chops the beginning and end of a word into its root form or forms. In MATLAB, the stemming operation is done using a built-in function called a text normalization technique and it requires a dedicated toolbox. The third operation, which is lemmatization is used to return a word to its original form, such as the word "depends" is converted to "depend". This operation uses the same built-in function as in the stemming procedure.

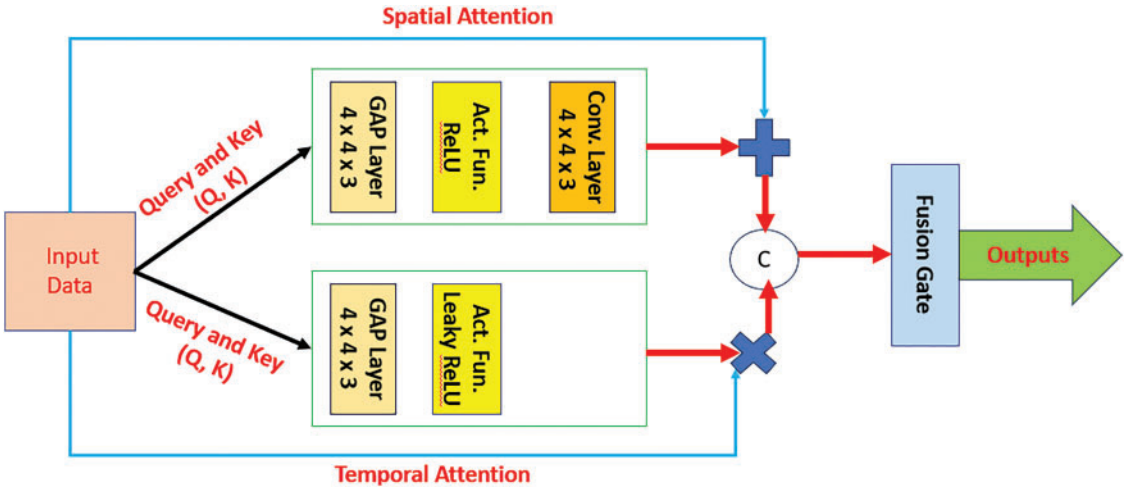

Figure 9: The internal architecture of the developed LDA network

In the deep-learning and feature extraction phase, Figs. 5 and 6 explain in deep detail what occurs. Inside the developed SE network, the data comes from the feature-extracted maps from the max-pooling layer via the Global Average Pooling (GAP) layer to reduce the size of the extracted dimensions of the feature maps to a single dimension of one channel only. Eq. (2) demonstrates a mathematical expression for the GAP procedure.

$$X = GAP(i) \tag{2}$$

X refers to the characteristics that are being squeezed and i represents the input. Then, the squeezed characteristics are passed through three fully connected layers of different sizes to find the dependencies between channels and generate new weights. Two activation functions: *ReLU* and Sigmoid are applied as well. Eq. (3) shows a mathematical expression for the generated new weights:

$$W_{new} = \gamma\left(FC3\left(ReLU\left(FC2\left(FC1\left(i\right)\right)\right)\right)\right) \tag{3}$$

where W_{new} represents the excited new weights, $FC1$, $FC2$, and $FC3$ denote the three fully connected layers, respectively, $ReLU$ is the deployed activation function, and γ is the sigmoid activation function. The excited map feature and the original one are used together to produce a new weight as shown in Eq. (4)

$$Y = W_{new} \times i \tag{4}$$

A sample of the obtained final weighted for five sentences is shown in Table 4.

Table 4: Sample of the evaluated rescaled weights

ID	*Y*
1	0.678
2	0.453
3	0.198
4	0.334
5	0.887

The feature extraction procedure mainly relies on three operators: Count Vectorizer, TF-IDF Vectorizer, and word Embedding. These three operators are applied to convert words to integers or floats. The target is to capture as many words as possible. The first operator: Count Vectorizer is deployed to assign each word an ID to be related to its count during either the training or testing stage. The second operator: TF-IDF is applied to reduce the count of words that appear multiple times and gives a score to every interested word. Then, each score is normalized to a value between 0 and 1. The third operator, which is the word embedding, is utilized to convert any word to a vector to keep track of its position and perform mathematical operations. The results of the word embedding operator are used to calculate the probability of identifying messages as either safe or spam. A mathematical expression for calculating the probability is shown in Eq. (5) as follows:

$$P\left(\frac{A}{B}\right) = \frac{P\left(\frac{B}{A}\right) \times P(A)}{P(B)} \tag{5}$$

A and B represent spam and safe words in a document that is being scanned. Four statistical analysis parameters are applied to each word, which are mean (Me), maximum (Max), minimum (Min), and standard deviation (std). The total number of extracted characteristics depends mainly on the total number of words in the scanned document.

2.5 The Evaluated Performance Indicators

Eight performance indicators (PIs) are computed to evaluate the proposed DaC-GANSAEBF method, which are True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), accuracy, precision, sensitivity, and F-score. The first four PIs are used to compute the other four PIs as follows.

1. Precision (PN): is computed as displayed in Eq. (6):

$$PN = \frac{TP}{(TP + FP)} \tag{6}$$

2. Sensitivity (SVY): is evaluated as shown in Eq. (7):

$$SVY = \frac{TP}{(TP + FN)} \tag{7}$$

3. Accuracy (ARY): is computed using Eq. (8):

$$ARY = \frac{(TP + TN)}{(TP + TN + FN + FP)} \tag{8}$$

4. *F1-score*: is determined via Eq. (9):

$$F1-score = 2 \times \left[\frac{(PN \times SVY)}{(PN + SVY)} \right] \tag{9}$$

3 Results and Discussion

The presented DaC-GANSAEBF model is evaluated using various extensive conducted experiments. Numerous factors that are believed or thought to affect the performance of DaC-GANSAEBF are counted in these experiments using four datasets.

3.1 Experimental Setup

The proposed method: DaC-GANSAEBF was trained, tested, and evaluated on a hosting machine with the MATLAB platform. The used machine runs with Windows Pro 11, which is installed on a Central Processing Unit (CPU) of an Intel Core I7 8th Generation, 16 GB of Random Access Memory (RAM), and 2 GHz. This CPU contains 4 sockets.

3.2 Predicting Results

The presented method was trained and tested using 5940 messages of legitimate and spam, while the remaining were for validation purposes. Fig. 10 shows that 4158 legitimate and spam messages were utilized for training. The total number of legitimate messages was 2079 and the total number of spam messages was the same. Table 5 lists the total number of characters in each set of training, validation, and testing, and Fig. 10 illustrates a distribution of the utilized message among all three sets. The number of legitimate and spam messages in each set is even.

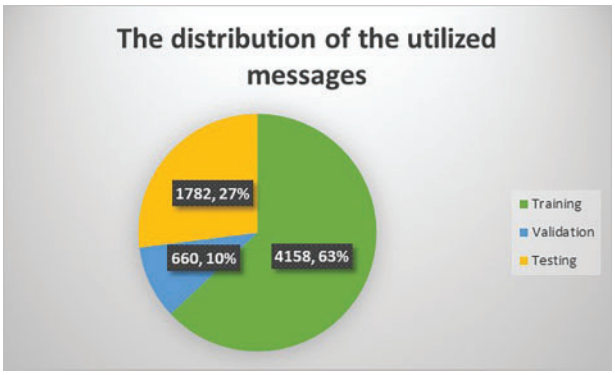

Figure 10: The distribution of messages in each set

Table 5: Sample of the evaluated rescaled weights

Set	Value
Training	45,401
Validation	673
Testing	3895

The experiments conducted were performed to compute the minimum, average, and maximum values for each considered PI of accuracy, precision, sensitivity, and F1-score and these results are shown in Fig. 11 for all applied iterations.

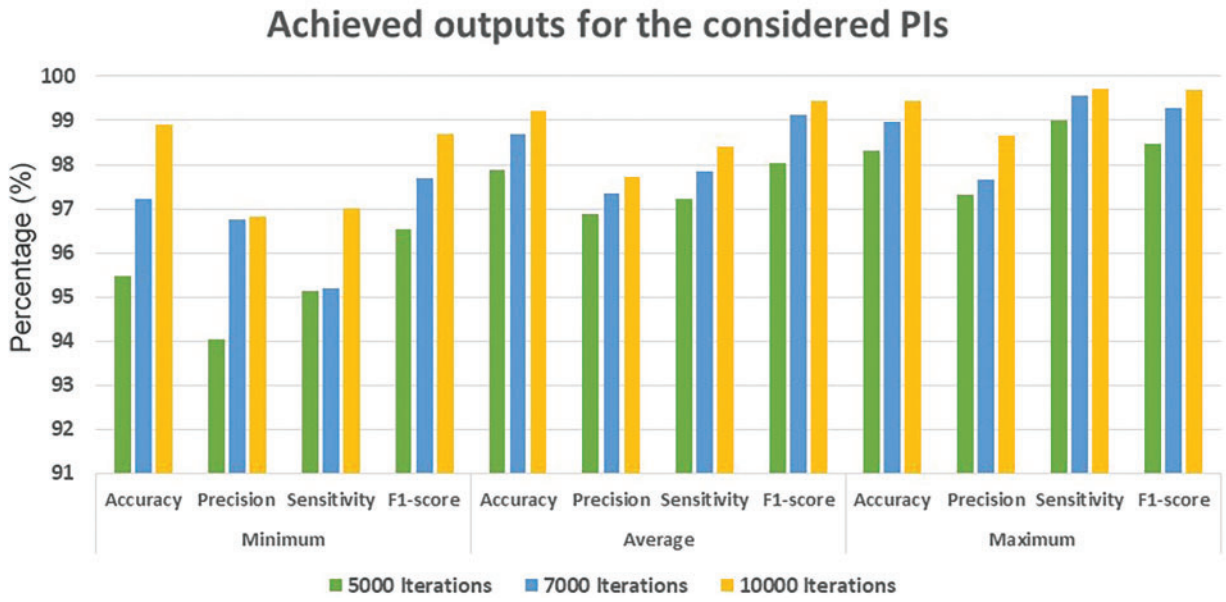

Figure 11: The results obtained from the DaC-GANSAEBF approach

In Fig. 11, for the 5000 iterations, 95.46% was the minimum achieved accuracy, 97.89% was the average value, and 98.32% was the maximum reached accuracy by the presented approach. The obtained Precision values were 94.03%, 96.89%, and 97.31%, while the sensitivity values were 95.13%, 97.21%, and 97.31%. The last PI, which is the F1-score, achieved 96.53%, 98.02%, and 98.46%. The second experiments conducted were performed using 7000 iterations and accuracy outcomes were 97.22%, 98.69%, and 98.97%. The F1-score reached values of 97.7%, 99.12%, and 99.28%. Lastly, in the third experiment at 10,000 iterations, accuracy outputs were 98.91%, 99.23%, and 99.44%. In the same iterations, other considered PIs reached values between 96.81% and 99.68%. The F1-score indicator approached almost 100% in these iterations. In the same figure, the accuracy and F1-score were increased significantly as the number of applied iterations doubled. On the other hand, increasing the number of iterations requires additional computational resources and hardware resources as well. This requirement harmed the total running time, which is also known as the total elapsed time. To accommodate this requirement, all 4 CPUs were used. However, the hosting machine got hot after nearly 45 min. Therefore, pausing the operations was performed every 30 min for an hour to cold the machine. The accuracy increased nearly by 4% from 5000 iterations to 10,000 iterations. Moreover, precision and sensitivity increased slightly in 5000 and 7000 iterations. The results of the identification operation for the testing set are shown in Fig. 12. DaC-GANSAEBF correctly classified 874 legitimate out of 891 messages and identified 866 spam messages out of 891 properly. In addition, 29 messages were improperly identified as legitimate, while these messages were spam. Fig. 12 implies that the achieved accuracy of the testing set was 97.64% when 7000 iterations were applied. However, this percentage increased by 1.5% when the number of iterations went to 10,000 iterations.

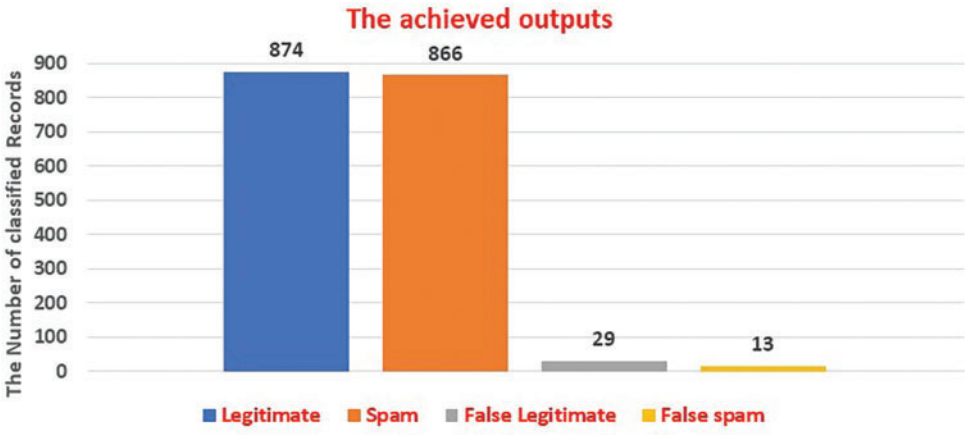

Figure 12: The identification results achieved on the testing dataset

Since the proposed DaC-GANSAEBF approach includes numerous operations and layers, thus, it is crucial to analyze its complexity, which includes the execution time from beginning to identify messages in seconds, the total number of employed parameters, and the number of Floating-Point Operations per Second (FLOPS). The analysis results are listed in Table 6. These results show that the presented approach requires a huge time to identify messages into their right classes and the total number of its parameters is quite massive since less than 79 million parameters are applied, and its number of FLOPS ranges between 55 and 57 million. The execution time for a document to be scanned and classified was nearly 44 s. In addition, the values of the number of parameters and FLOPS are expected because of the internal structures and the operations that are performed internally.

Table 6: The analysis of the complexity of DaC-GANSAEBF

Execution time	FLOPS (M)	Number of Parameters (M)
44 s	56.23	78.02

The achieved chart of the binary cross entropy is depicted in Fig. 13. This chart shows that the best validation value of that indicator occurred at Epoch 8 at a value of 0.00965. In addition, the training and testing values intersected at Epoch 6 as shown in the same graph. Fig. 14 illustrates the fit results of a sample of legitimate messages using all three sets: training, validation, and testing. Only one sample is shown due to space limitations. Figs. 13 and 14 imply that DaC-GANSAEBF worked well and generated exquisite findings. In addition, a few errors occurred, and these errors are illustrated in orange vertical lines in Fig. 13. Furthermore, the produced Receiver Operating Characteristic (ROC) curves for the legitimate and spam messages are depicted in Fig. 15. Both curves reached more than 80% of the areas under the curves. These areas are considered small; however, a high value was reached when the number of applied iterations was 10,000.

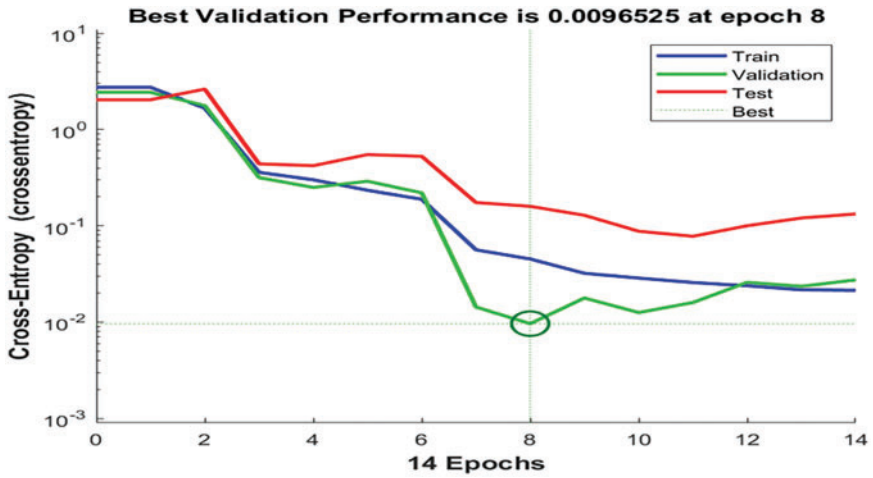

Figure 13: The obtained cross-entropy chart

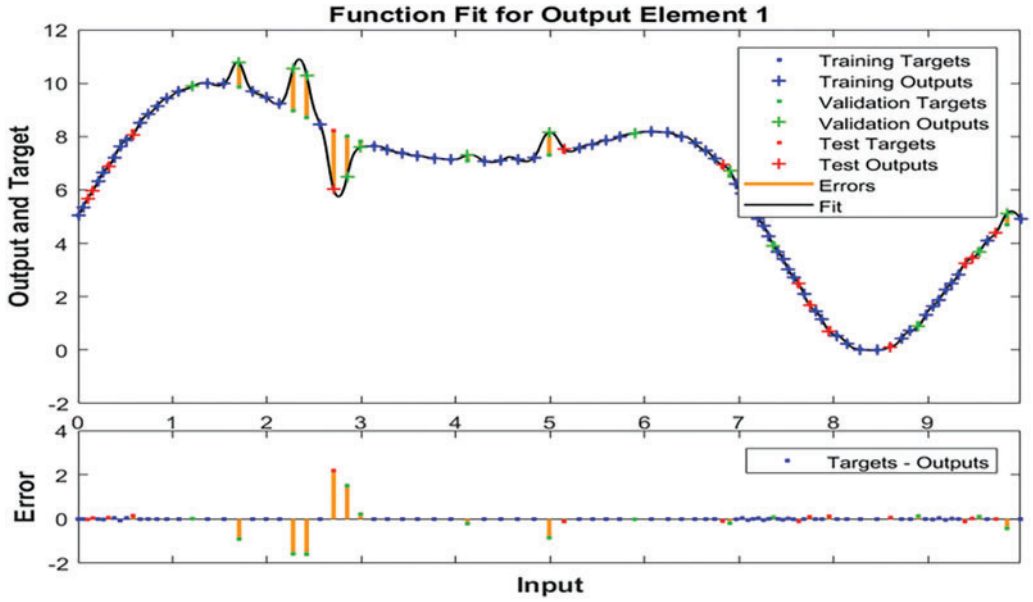

Figure 14: The sample result of the applied fit function

3.3 Comparative Assessment

In this research, it is essential to conduct a comparative analysis between the proposed DaC-GANSAEBF method and several existing deep learning techniques developed by P. Bhargavi, M. Nooraee and H. Ghaffari, P. Malhotra and S. K. Malik, as well as I. AbdulNabi and Q. Yaseen. This analysis will focus on three key aspects: (1) the number of datasets utilized, (2) the technologies employed, and (3) the performance metrics achieved. The highest results reported by these authors will be taken into account. The findings of this comparative analysis are summarized in Table 7, where all top scores are highlighted in bold. The analysis indicates that the DaC-GANSAEBF method achieved superior results across all performance indicators, with the exception of precision, where the method by M. Nooraee and H. Ghaffari attained a perfect score

of 100%. Consequently, DaC-GANSAEBF outperforms all other existing deep-learning methodologies. For clarity, Fig. 16 illustrates two performance indicators: accuracy and F1-score. It is evident from Fig. 16 that the methods developed by P. Malhotra and S. K. Malik and I. AbdulNabi and Q. Yaseen recorded the lowest accuracy rates of 98.5% and 98.65%, respectively, while the methods by P. Bhargavi, M. Nooraee and H. Ghaffari, along with the proposed approach, surpassed 99%. Furthermore, the proposed method excels in both accuracy and F1-score metrics.

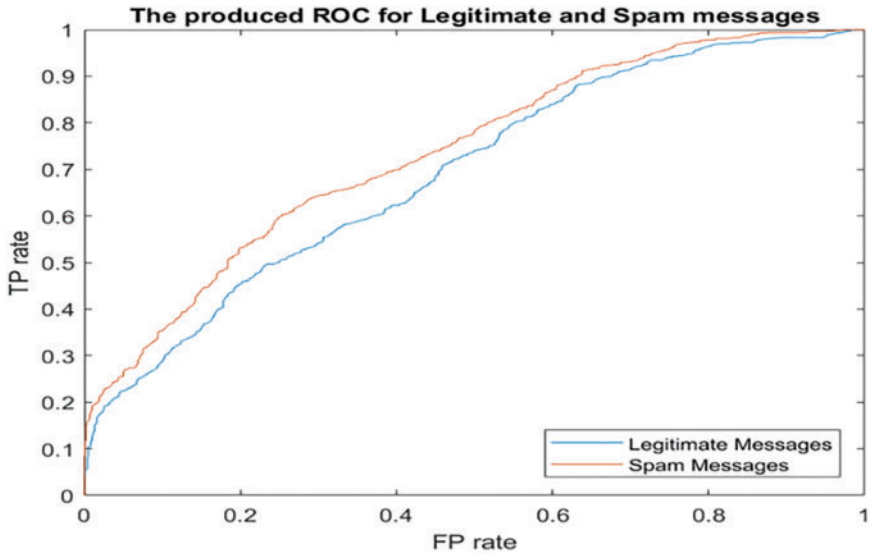

Figure 15: The generated ROC curves for both types of messages

Table 7: Comparative analysis results

Reference	The number of applied datasets	Applied technology	ARY	PN	SVY	F1-score
[12] Bhargavi, 2022	N/A	Naïve Bayes (NB), SVM, and a Convolutional Neural Network (CNN)	99.02%	N/A	N/A	N/A
[13] Nooraee et al., 2022	2	LSTM and Glove Word Embedding (GWE) technologies	99.42%	**100%**	96%	98%
[16] Malhotra et al., 2022	1	ML approaches, a Sequential Neural Network, LSTM, and Bi-LSTM	98.5%	98%	96%	98%

(Continued)

Table 7 (continued)

Reference	The number of applied datasets	Applied technology	ARY	PN	SVY	F1-score
[17] AbdulNabi et al., 2021	2	Bi-LSTM, LSTM, and Bidirectional Encoder Representations from Transformers (BERT)	98.62%	N/A	N/A	98.66%
DaC-GANSAEBF	4	GAN and SE network	**99.43%**	98.67%	**99.71%**	**99.68%**

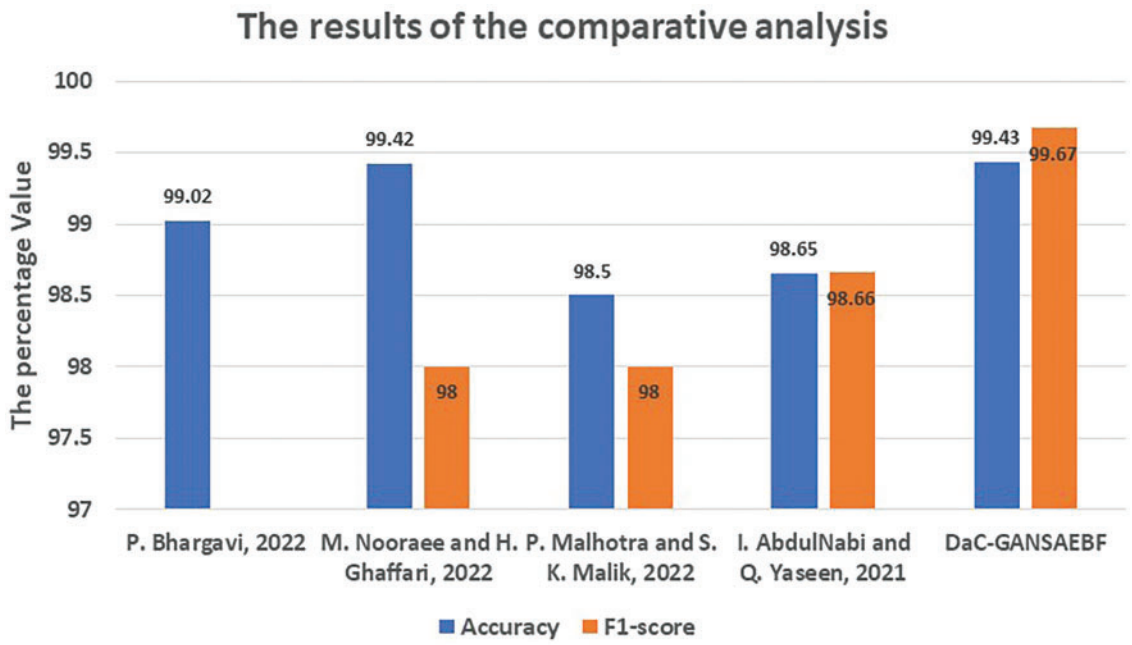

Figure 16: Results of the comparison analysis

The framework exhibits remarkable accuracy and robustness as a trade-off; however, its computational demands necessitate careful evaluation. Specifically, the GAN and SAE components, which are crucial for achieving optimal performance, result in heightened resource usage.

4 Discussion

The findings confirm the efficacy of DaC-GANSAEBF, showcasing its exceptional performance in relation to critical metrics. The framework attained an average accuracy of 98.87% across four publicly accessible datasets, markedly surpassing the performance of current methodologies. These results highlight the capability of DaC-GANSAEBF to transform spam detection approaches in practical applications.

The practical significance of DaC-GANSAEBF is found in its effortless incorporation into current email systems. By enhancing detection precision and reducing false positives, the framework fosters increased user confidence and efficiency. Its modular architecture guarantees scalability, rendering it appropriate for both small businesses and large corporations.

The comparative analysis has been broadened to encompass insights into emerging technologies, including:

- Large Language Models (LLMs): these models demonstrate exceptional proficiency in comprehending intricate text patterns. Nevertheless, their substantial computational demands and latency issues restrict their effectiveness in real-time spam detection.
- Graph Neural Networks (GNNs): by utilizing relational data structures, GNNs offer significant insights into the interdependencies among email attributes.

Their incorporation with frameworks like DaC-GANSAEBF could further improve detection capabilities. This expanded discussion situates DaC-GANSAEBF within the wider context of cutting-edge technologies, emphasizing its distinctive advantages and potential areas for improvement.

Various parameters were tuned and adjusted to analyze and investigate how these parameters affect the proposed approach. Hence, a series of experiments were conducted for different values of parameters.

4.1 Impact of Tuning the Applied Settings

As stated earlier, Table 3 lists the values of all hyperparameters that were applied in this study. Some of these hyperparameters were adjusted and modified, which were Dropout, Optimizer, Regression weight, and the number of embedding layers to monitor the performance and notice how it responds to these changes. The value of Dropout was doubled, the applied Optimizer was removed, the value of deployed Regression weight was modified from 0.01 to 0.25, and the number of applied embedding layers was adjusted from 4 to 6. All considered PIs were monitored and their responses were closely analyzed. Some PIs were negatively and positively impacted. All obtained findings are listed in Table 8 when 8000 iterations were deployed. DaC-GANSAEBF still generated great accuracy results since these results were between 98.91% and 99.71%. However, removing the optimizer had a big negative impact as the accuracy went down to 94.22%. This implies that the Adam optimizer is a crucial element and cannot be excluded or removed. In addition, the execution time was evaluated as well, and its values were slightly varied.

Table 8: Comparative analysis results

Name of hyperparameter	Accuracy	Precision	Sensitivity	F1-score
Dropout	99.71%	98.9%	99.03%	99.32%
Optimizer	94.22%	91%	91.01%	92.49%
Regression weight	98.91%	99.11%	99.01%	98.29%
The number of embedding layers	99.23%	99.1%	98.9%	99.56%

4.2 The Conducted Statistical Analysis

In this research, we performed a statistical evaluation regarding the accuracy and F1-score between DaC-GANSAEBF and three famous random neural networks: AlexNet, Visual Geometry Group-19 (VGG-19), and Recurrent Neural Network (RNN) on the testing set. The statistical evaluation was conducted based on the Wilcoxon signed-rank test and the evaluation results are illustrated in Fig. 17. In this evaluation, 5%

was the allowance p-value and the number of applied iterations was 8000. The results in Fig. 17 show that the presented approach, which is DaC-GANSAEBF, outperforms all other neural networks in accuracy and F1-score. AlexNet got the lowest F1-score value at 94%, while its accuracy was the highest value compared to the other two neural networks. The p-values obtained from the datasets affirmed the statistical significance of the performance enhancements observed, thereby confirming that the noted advantages are not due to random fluctuations.

4.3 Challenges and Limitations

The framework demonstrated encouraging outcomes; however, several challenges were faced throughout the research process: 1. Significant computational demands: the combination of GAN and LDA modules necessitated considerable computational resources, especially during the training phase. This constraint hindered the exploration of additional datasets and configurations within the limits of available resources. 2. Mode collapse in GAN training: the GAN experienced instances of mode collapse during training, resulting in the generator producing repetitive outputs. To mitigate this issue, further regularization techniques and hyperparameter adjustments were required. 3. Complexity in feature extraction: achieving a balance between computational efficiency and the need for thorough feature extraction proved challenging, particularly when managing extensive datasets.

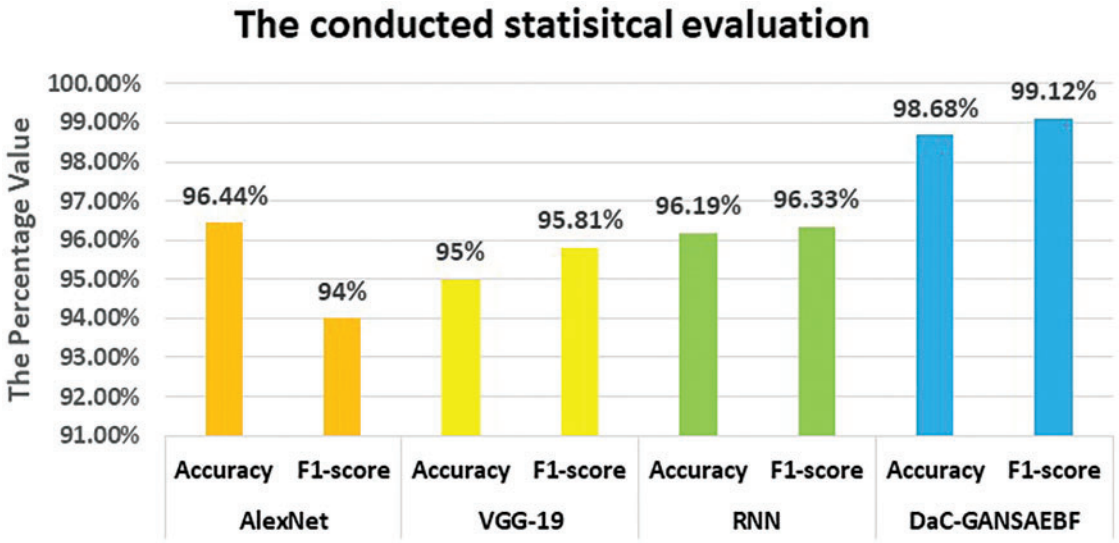

Figure 17: Findings of the statistical analysis evaluation

Despite its achievements, the study is not without limitations: 1. High resource requirements: the proposed framework requires significant computational power, which may not be practical for implementation in environments with limited resources. 2. Narrow dataset range: although four public datasets were utilized, incorporating additional datasets could enhance the validation of the framework's robustness. 3. Lack of real-time testing: the framework has yet to be evaluated in real-time email systems, where issues related to latency and integration may emerge.

For the trade-off, the framework demonstrates exceptional accuracy and robustness; however, its computational requirements require careful consideration. In particular, the GAN and SAE components,

which are essential for optimal performance, lead to increased resource consumption. Future efforts will aim to minimize these demands while maintaining the framework's effectiveness.

5 Conclusion

This article presents a new AI model to identify spam messages to protect recipients and secure Email platforms. The proposed model is DaC-GANSAEBF, and it encompasses two deep-learning topologies, which are GAN and SE networks. DaC-GANSAEBF is built using two main stages and was evaluated using four public datasets from Kaggle. All the findings achieved by DaC-GANSAEBF were promising and exquisite. In this study, four performance metrics, also known as PIs, were mainly considered and evaluated using DaC-GANSAEBF under various conditions. In addition, some data was collected from a personal Email account to support the approach and make its outcomes robust and valid. Since we deal with texts, numerous preprocessing processes are necessary to clean the data and eliminate any unwanted texts. All experiments conducted and their outputs proved that DaC-GANSAEBF is a trusted approach and can be effectively used since its results are good enough and satisfy its objectives. In addition, DaC-GANSAEBF was compared with other developed works in literature and its results surpassed other designs. However, DaC-GANSAEBF suffers from two major limitations, which are (1) the required execution time and (2) the total number of its internal parameters. Some modifications can be considered to minimize these limitations. Yet, these modifications could impact the accuracy negatively and reduce it by 8%. These remarkable findings indicate that the proposed methodology effectively meets the research goals established in the study. The framework successfully attained its key objectives as outlined below: 1. Creation of an innovative framework: the DaC-GANSAEBF was effectively executed, integrating DaC, GAN, SEA, and LDA components into a unified and efficient spam detection system. 2. Mitigation of dataset imbalance: a strategy was implemented to address the class imbalance, promoting equitable performance in the classification of both spam and legitimate emails. 3. Enhanced performance metrics: the model recorded an average accuracy of 98.87% across four benchmark datasets, exceeding the performance of current leading methods in terms of precision, recall, and F1-score. 4. Generalizability and scalability: the framework exhibited resilience across various datasets, highlighting its applicability for real-world scenarios in extensive email systems.

To further advance this research, the following recommendations are suggested for future investigations: 1. Enhancement of computational efficiency: creating streamlined versions of the framework, potentially by refining the GAN architecture or implementing quantization methods, to improve its applicability for real-time scenarios. 2. Validation with broader datasets: evaluating the framework against a wider array of datasets, including proprietary or sector-specific collections, to better understand its generalization potential. 3. Implementation in real-time environments: incorporating the framework into active email systems to assess its performance in practical settings, focusing on aspects such as latency, scalability, and user engagement. 4. Integration of sophisticated techniques: investigating the application of transformers or other advanced neural network models to improve feature extraction and classification accuracy. 5. Examination of adversarial resilience: analyzing the model's robustness against adversarial threats that may exploit vulnerabilities in the detection process.

In summary, the DaC-GANSAEBF framework marks a notable progression in the realm of spam email detection, effectively addressing significant shortcomings of current methodologies and paving the way for future advancements in this vital field. By leveraging the findings and tackling the limitations highlighted in this research, subsequent studies can further improve the efficacy and relevance of spam detection systems.

Acknowledgement: This project was funded by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, Saudi Arabia under Grant No. (GPIP: 71-829-2024). The authors, therefore, acknowledge with thanks DSR for technical and financial support.

Funding Statement: This project was funded by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, Saudi Arabia under Grant No. (GPIP: 71-829-2024). The authors, therefore, acknowledge with thanks DSR for technical and financial support.

Author Contributions: Conceptualization, Ahmed A. Alsheikhy, Yahia Said, and Tawfeeq Shawly; data curation, Shaaban M. Shaaban and Abdulrahman Alzahrani; formal analysis, Ahmed A. Alsheikhy and Husam Lahza; funding acquisition, Ahmed A. Alsheikhy, Tawfeeq Shawly and Husam Lahza; investigation, Ahmed A. Alsheikhy, Abdulrahman Alzahrani and Aws I. AbuEid; methodology, Ahmed A. Alsheikhy, Yahia Said, Abdulrahman Alzahrani and Tawfeeq Shawly; supervision, Ahmed A. Alsheikhy; validation, Yahia Said and Husam Lahza; writing—original draft, Ahmed A. Alsheikhy, Tawfeeq Shawly, Shaaban M. Shaaban and Abdulrahman Alzahrani; writing—review and editing, Tawfeeq Shawly and Aws I. AbuEid. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data used in the preparation of this article were obtained from Kaggle and are available at: https://www.kaggle.com/datasets/venky73/spam-mails-dataset. Available Online (accessed on 20 December 2023). https://www.kaggle.com/datasets/mandygu/lingspam-dataset. Available Online (accessed on 15 December 2023). UCI Machine Learning, https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset. Available Online (accessed on 20 December 2023). https://www.kaggle.com/datasets/shalini2810/input-file. Available Online (accessed on 21 December 2023).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report in this study.

References

1. Kirvan P, Awati R, Teravainen T. What is email spam and how to fight it? [cited 2023 Dec 23]. Available from: https://www.techtarget.com/searchsecurity/definition/spam#:~:text=Email%20spam%2C%20also%20known%20as,a%20large%20list%20of%20recipients.
2. What is spam Email? [cited 2023 Dec 25]. Available from: https://www.cisco.com/c/en/us/products/security/email-security/what-is-spam.html.
3. Stouffer C. How to get rid of spam emails. [cited 2023 Dec 25]. Available from: https://us.norton.com/blog/how-to/spam-spam-go-away.
4. Kumar A, Kumar S, Kumar K, Naib DBB. E-mail fraud detection. Int J Emerg Sci Eng. 2023;11(9):1–7. doi:10.35940/ijese.B7797.0811923.
5. Choudhary J. An analytical study on an email spam detection. Int J Creat Res Thoughts (IJCRT). 2023;11(5):M297–301.
6. Sharma A, Arjun N. Spam detection using machine learning techniques. Int J Res Publ Rev. 2023;4(7):2478–88.
7. Reddy A, Reddy KH, Abhishek A, Manish M, Dattu GVS, Ansari NM. Email spam detection using machine learning. J Surv Fish Sci. 2023;10(1):2658–64.
8. Fatima R, Fareed MMS, Ullah S, Ahmad G, Mahmood S. An optimized approach for detection and classification of Spam email's using ensemble methods. Wirel Pers Commun. 2024;139(1):347–73. doi:10.1007/s11277-024-11628-9.
9. Wankhade NV, Keole RR, Mahore TR. Paper on spam email detection with classification using machine learning. Int J Innov Res Technol. 2022;9(2):1055–9.
10. Ahmed N, Amin R, Aldabbas H, Koundal D, Alouffi B, Shah T. Machine learning techniques for Spam detection in email and IoT platforms: analysis and research challenges. Secur Commun Netw. 2022;2022(5):1862888. doi:10.1155/2022/1862888.
11. Panwar M, Jogi JR, Mankar MV, Alhassan M, Kulkarni S. Detection of spam email. Am J Innov Sci Eng. 2022;1(1):18–21. doi:10.54536/ajise.v1i1.996.
12. Bhargavi P. Spam email detection using machine learning and deep learning techniques. Int J Res Publ Rev. 2022;3(11):1349–52.

13. Nooraee M, Ghaffari H. Optimization and improvement of spam email detection using deep learning approaches. J Comput Robot. 2022;15(2):59–67.
14. Narendra Kumar Rao B, Partheeban P, Naseeba B, Raju HP. ML approaches to detect email spam anamoly. In: 2022 International Conference on Data Science, Agents and Artificial Intelligence (ICDSAAI); 2022 Dec 8–10; Chennai, India. p. 1–6.
15. Abhinav C. Spam mail detection using machine learning. Int J Res Appl Sci Eng Technol. 2022;10(6):2327–9. doi:10.22214/ijraset.2022.44315.
16. Malhotra P, Malik SK. Spam email detection using machine learning and deep learning techniques. In: Proceedings of the International Conference on Innovative Computing and Communication (ICICC); 2022 Feb 19–20; Delhi, India. p. 1–20.
17. AbdulNabi I, Yaseen Q. Spam email detection using deep learning techniques. Procedia Comput Sci. 2021;184(2):853–8. doi:10.1016/j.procs.2021.03.107.
18. Anandpara R. Secured mail transformation system using machine learning. Int J Res Appl Sci Eng Technol. 2021;9(VII):1880–6. doi:10.22214/ijraset.2021.36764.
19. Mohey H, Mohsen S. Using machine learning techniques for predicting email spam. Int J Instr Technol Educ Stud. 2021;2(4):19–23. doi:10.21608/ihites.2021.204000.
20. Sethi M, Chandra S, Chaudhary V, Dahiya Y. Email spam detection using machine learning and neural networks. Int Res J Eng Technol. 2021;8(4):349–55.
21. Madhavan MV, Pande S, Umekar P, Mahore T, Kalyankar D. Comparative analysis of detection of email Spam with the aid of machine learning approaches. IOP Conf Ser: Mater Sci Eng. 2021;1022(1):012113.
22. Sultana T, Sapnaz KA, Sana F, Najath J. Email based spam detection. Int J Eng Res Technol. 2020;9(6):135–9.
23. Kumar N, Sonowal S, Nishant. Email spam detection using machine learning algorithms. In: Proceedings of the Second International Conference on Inventive Research in Computing Applications (ICIRCA); 2020 Jul 15–17; Coimbatore, India. p. 108–13.
24. Prasanthi KS, Deepika T, Anudeep S, Koushik MS. An efficient email Spam detection using support vector machine. Int J Innov Technol Explor Eng. 2019;9(2):5258–62. doi:10.35940/ijitee.B9001.129219.
25. Tope M. Email Spam detection using naive Bayes classifier. Int J Sci Dev Res. 2019;4(6):1–7.
26. Pandey P, Agrawal C, Ansari TN. A hybrid algorithm for malicious spam detection in email through machine learning. Int J Appl Eng Res. 2018;13(24):16971–9.
27. Sharma P, Bhardwaj U. Machine learning based Spam E-mail detection. Int J Intell Eng Syst. 2017;11(3):1–10.
28. Communications, Space and Technology Commission. Cybersecurity regulatory framework (CRF) for service providers in the information and communications technology sector. [cited 2023 Dec 31]. Available from: https://www.cst.gov.sa/en/RulesandSystems/CyberSecurity/Documents/CRF-en.pdf.
29. National Cybersecurity Authority. Essential cybersecurity controls (ECC–1: 2018). [cited 2023 Dec 31]. Available from: https://nca.gov.sa/ecc-en.pdf.
30. Garnepudi V. Spam mails dataset. [cited 2023 Dec 20]. Available from: https://www.kaggle.com/datasets/venky73/spam-mails-dataset.
31. Gu M. Ling-spam dataset. [cited 2023 Dec 15]. Available from: https://www.kaggle.com/datasets/mandygu/lingspam-dataset.
32. UCI machine learning. [cited 2023 Dec 20]. Available from: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset.
33. Gupta S. SMS spam detection application. [cited 2023 Dec 21]. Available from: https://www.kaggle.com/datasets/shalini2810/input-file.
34. Striuk O, Kondratenko Y. Generative adversarial neural networks and deep learning: successful cases and advanced approaches. Int J Comput. 2021;20(3):339–49. doi:10.47839/ijc.20.3.2278.
35. Aggarwal A, Mittal M, Battineni G. Generative adversarial network: an overview of theory and applications. Int J Inf Manag Data Insights. 2021;1(1):100004. doi:10.1016/j.jjimei.2020.100004.
36. Jin X, Xie Y, Wei XS, Zhao BR, Chen ZM, Tan X. Delving deep into spatial pooling for squeeze-and-excitation networks. Pattern Recognit. 2022;121(3):108159. doi:10.1016/j.patcog.2021.108159.

37. Huang B, He B, Wu L, Guo Z. Deep residual dual-attention network for super-resolution reconstruction of remote sensing images. Remote Sens. 2021;13(14):2784. doi:10.3390/rs13142784.
38. Kumie GA, Habtie MA, Ayall TA, Zhou C, Liu H, Seid AM, et al. Dual-attention network for view-invariant action recognition. Complex Intell Syst. 2024;10(1):305–21. doi:10.1007/s40747-023-01171-8.
39. Li YX, Tang H, Wang W, Zhang XF, Qu H. Dual attention network for unsupervised medical image registration based on VoxelMorph. Sci Rep. 2022;12(1):16250. doi:10.1038/s41598-022-20589-7.