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

Intelligent Spam Detection Framework Based on Genetic Algorithm and Machine Learning

Donthi Sadhana[1], A. Nagarjuna Reddy[2], Dr. V. Anantha Krishna[3], Dr. U. Srilakshmi[4]

[1,2,3,4] Department of Computer Science and Engineering, Sridevi Women's Engineering College, Gopanpally, Hyderabad, Telangana, India

Abstract

The digital communication has improved quickly, and a large number of unsolicited messages, or spam, may overload email systems, drop productivity, and even affect cyber security. Spam or unwanted messages that are unsolicited and can be usually harmful has become a significant issue to email systems and electronic communication platforms. This paper presents an Intelligent Spam Detection Framework (ISDF) which is an efficient spam communication detection and filtering framework that operates by integrating Genetic Algorithms (GA) and Machine Learning (ML) algorithms. The framework employs this to optimize the process of selecting features by the use of genetic algorithm, where only the most relevant and discriminative features are employed in training the machine learning model. This will decrease the size of data leading to enhanced model performance and it would also accelerate the detection. The system will combine the conventional machine learning algorithms such as Random Forests, Naive Bayes , and Support Vector Machines (SVM) to mark a message spam or not, by integrating the characteristics of linguistic, metadata, and context features. Genetic algorithm has great optimization ability which makes the detection process more accurate as it evolves and adapts the set of features as the time progresses. The proposed framework ISDF is then evaluated on a number of benchmark spam datasets and the findings reveal that the proposed structure has had a higher accuracy, better precision and recall than the traditional spam detection techniques. This smart system is a scalable way of managing spam on a large number of digital communication mediums to guarantee better user experience and greater safety. The proposed Genetic Algorithm (GA)-optimized spam detection method is systematically compared with the established machine learning classifiers e.g. XGBoost, Multinomial Naïve Bayes, and Linear Support Vector Machine (SVM). The experimental results prove that the existing models were apt to 97.8% of Multinomial Naïve Bayes, 98.9% of Linear SVM, and 99.2% of XGBoost. On the contrary, the best accuracy value of the proposed Genetic Algorithm-optimized model of 99.5 was the highest. In this comparative study, the incorporation of Genetic Algorithm-based optimization can be observed to increase the efficiency in feature selection and parameter tuning, thus, increasing classification efficiency and general predictiveness. The results show that suggested approach excels the conventional machine-learning models when it comes to spam identification in terms of both accuracy and strength.

Key Words: Spam Detection, Machine Learning, Genetic Algorithm, Feature Selection, Text Classification, Email Filtering.

1. Introduction

Unwanted and sometimes malicious content commonly referred as spam has become common practice in the digital communication systems, especially email. The spam mail stays on the increase since spammers are coming up with more advanced ways of breaking the conventional spam filters. These conventional systems usually use fixed rule-based systems or basic machine learning systems that may have difficulties in keeping pace with the dynamic nature of spam content. Consequently, spam messages have posed a challenge on researchers as well as similar organisations to be detected.

Spam messages have become a big challenge with the quick development of digital communication that uses emails, messaging applications, and social media platforms. Spam is not only a source of negative user experience but also a source of security threats like phishing, malware attacks, and financial fraud. The available spam detection systems, mainly rule-based and keyword-based methods are not effective in detecting spam since the content of spam is dynamic and keeps changing. The accuracy of machine learning based spam detector is higher, but very much be dependent on the choice of relatable features. Huge and irrelevant sets of features are more expensive to compute, worse in classification, and they result in high false positives. The traditional methods of feature selection are usually fixed and are incapable of adapting to the evolving spam patterns. Toward address these shortcomings, a smart and dynamic spam detection system is required to maximize the features selection and enhance categorization accuracy. The grouping of Genetic Algorithms and the utilization of Machine learning classifiers will offer an adequate solution to the problem as it will optimize the feature subsets and boost the efficiency of anti-spam software systems.

This study presents the Intelligent Spam Detection Framework to improvise the accuracy, adaptability and scalability of spam detection through the integration of Genetic Algorithms (GA) and Machine Learning (ML). The main aim is to streamline the selection of feature with the aid of GA that allows identifying the most related features to spam detection. The GA has the potential to adjust to the evolvement of the spam environment by updating the feature set provided across several generations and should enhances the functionality of the machine learning model.

Support Vector Machines (SVMs), Naive Bayes, and random forests are all part of the architecture's machine learning classifier suite. A wide range of textual, contextual and metadata-based features are trained in such models, which guarantees a more extensive approach to spam detection. Combination of GA and ML is a dynamic and robust solution which can detect spam messages with a high accuracy with low false positive and low computational complexity. The structure has proven to be effective in real-time online communications as it has been shown through experimental work on a variety of benchmark datasets that it can easily outperform the conventional methods when it comes to the issue of spam fighting. This solution can be implemented on many platforms, such as social media and email, which offers a flexible and smart system in detecting spam.

The ISDF, that depends on Genetic Algorithm and Machine Learning, is limited to the design, development and testing of an automated solution to provide the capability of recognizing and filtering spam messages versus legitimate messages. The primary objectives of the proposed method ISDF, is to integrate the Genetic Algorithm to Machine Learning classifiers, to create an automated spam detector that is scalable and to guarantee flexibility to new spam trends.

This study is designed as: Section 2 describes a complete review of the related literature on spam detection techniques and optimization-based machine learning approaches. Section 3 defines the proposed method, including the Genetic Algorithm-based feature optimization and classification framework. Section 4 details the implementation process, including dataset preparation, pre-processing steps, and system configuration. An analysis of the suggested model's efficiency and experimental findings are presented in Section 5. After a brief summary of the study's main points in Section 6, the text moves on to cite its sources.

2. Literature Review

The fast expansion of digital communication networks, such as email, social media, and mobile messaging services, has raised the use of spam messages majorly creating serious threats to the security of information and to the privacy of users. Not only do spam messages create inconvenience, but they are the means of phishing attacks, malware spreading, financial fraud and identity theft. The use of traditional systems of rule-based filtering cannot support the dynamic nature and evolve ability of spam material because spammers keep on changing the patterns in order to avoid being detected. As a result, spam detection methods deepening on machine learning have become the effective solutions because they can learn the discriminative patterns of a large-scale dataset. Nevertheless, traditional machine learning algorithms frequently have issues of complexity on selection of features, high dimensionality data, imbalanced classes, and poorly selected parameters that can influence the classification accuracy and learning generalization. In order to mitigate these shortcomings, new studies aim at a combination of optimization methods, especially Genetic Algorithms to optimize feature selection and model optimization, and thus increased detection effectiveness and robustness of intelligent spam filtering systems**.**

Traditional machine learning algorithms including Naive Bayes, Random Forests, Support Vector Machines (SVM), ensemble approaches like XGBoost, and Decision Trees are used for spam identification. These models normally utilize text pre-processing methods that encompass tokenization, stop-word elimination, stemming as well as vectorization approaches like as Bag-of-Words, TF-IDF to convert textual data into arithmetical data. Two of these, Multinomial Naive bayes is simple and has low computing power whereas SVM has great generalization power in high dimensional feature space. The ensemble methods also provide an increase in the performance of prediction when several weak learners are combined. Despite the fact that these methods have performed exceptionally well in the process of identifying spam, their presentation is highly sensitive to the quality of features representation, as well as parameter optimization.

However, the current systems have a number of weaknesses. Conventional machine learning algorithms are usually difficult to apply in high dimensional feature space, redundant or irrelevant features as well as the issue of class imbalance as is common with spam datasets**.** In addition, manual or grid based hyper parameter tuning may be computationally expensive and may not be in a position to determine the best solutions**.** With the ever-changing patterns of spam, the performance of the same model might reduce over time. The above challenges make optimization-based methods which are able to automatically choose informative features and optimize model parameters to enhance robustness and the overall performance of the classification process.

Neelam Banjare [1] has suggested a hybrid algorithm that uses correlation filtering and Genetic Algorithm (GA) to optimize the features that are used by a Multilayer Perceptron (MLP) classifier and the algorithm has been demonstrated to be highly accurate on the Enron spam data. Putra et al. [2] systematic overview of all main ML methods to spam detection with emphasis on feature selection and classification. Meta-Learner Framework of Spam Detection [3] depicts tendencies in hybrid structures that enhance the accuracy of CSV spam prediction. Enhanced Mechanism Bio-inspired + Deep Learning [4] combines feature extraction/selection with bio-inspired optimization to achieve a better performance at spam classification. ML classifiers such as Naive Bayes, SVM, RF were used in an intelligent spam detection architecture [5]. Kocyigit et al. [6] however with phishing URLs indicates usefulness of GA with feature reduction, prior to the use of ML classification. Genetic Algorithm was applied in optimizing hyper-parameters of the ML classifier in email spam data by Fatima [7]. Ghatasheh et al. [8] proposed a variation of Genetic Algorithm to reduce features and optimize hyper-parameters of spam-related tasks (such as SMS) at once. Stable ML, CNN, Genetic Algorithm were used to optimize features / parameters in Advanced Spam Detection study [9]. Transformer-based models were adopted by Labanne et al. [10] in detecting few-shot spam; this is also relevant since it is advanced contextual analysis. It is important to note that researchers like Temidayo [11], used the Enron1 data to develop baseline models using exciting gradient boost collective and random forest model with more advanced features of spam email detection and categorization. Moreover, Qinglin Qi [12], put forward the Markov-based phishing ensemble detection FMPED method, which is complemented by its ensemble detection method. Rosita et at.[13] suggested Twitter spam detection using a hybrid architecture of Multi-Objective GA and CNN.

Machine Learning Spam Detection Including GA-Methods [14] review indicates more and more integration of ML and nature-inspired methods such as GA to achieve better detection. Wiley [15] research paper on machine learning with reference to algorithms such as XCS and Genetic Algorithms in spam filtering. Genetic Algorithm was used alongside association rules, and then ML classification. Simultaneously, Naeem Ahmed [16], has carried a comprehensive survey of machine learning methods used in spam filtering activities on email and IoT systems. This overall evaluation involved a careful valuation of various performance measures. Heavy metal: the world of text-, voice-enabled emails. A new hybrid framework of data processing was introduced by Safaa S. I. Ismail [17], a Genetic decision tree with natural language processing GDTPNLP. Through Fallah Sokhangoee et al. [18]. Spam Mail Classification Using SVM and Genetic Algorithm [19] combining the GA and SVM to classify spam. ML Framework of Email Spam Detection [20] is a framework that incorporates SpamML framework, and its application involves different ML techniques; upon which to compare. Emerging ML Spam Detection Benchmarking [21] Comparison among various ML classifiers (NB, SVM, RF) as the baseline in the future research in integrating GA. Taloba and Ismail [22] examined the optimization of Genetic Algorithm (GA) over DTs. The space dimensional over-fitting issue was overcome by use of Principle Component Analysis (PCA) to extract features. In their conference paper, sang Min Lee et.al. [23] Have described an optimum spam detection model on the basis of Random Forest (RF). RF was used to do parameters optimization and feature selection at the same time.

Expressing ISDF, with the help of Genetic Algorithm and Machine Learning, is characterized by a number of challenges. The major challenge is the large-dimensional text data, where a huge feature space can consist of redundancy, irrelevant or noisy features that negatively impact the classification performance. Moreover, spam datasets are usually unbalanced in their classes and thus they might favour learning based

on the dominant class and lower the detection rate of the less dominant cases. Optimal feature selection and hyper parameter tuning is another important issue and the traditional algorithms, including manual tuning or a grid search, are computationally costly and do not necessarily yield global optimal solutions. Moreover, the dynamic and changing nature of spam content demands dynamic and adaptive models that can be able to sustain high performance with time. The challenges mentioned above require the incorporation of optimization methods including Genetic Algorithms to increase the selection of features, model generalization and attain better classification accuracy.

The proposed paper revolves around the creation of ISDF, the hybridized approach of genetic algorithm (GA) and Machine Learning (ML), in detecting and classifying the spam messages in the email and SMS with these methods in email and SMS spam detection. The structure starts with the input data that undergoes pre-processing that includes tokenization, elimination of stop-words, and generation of features to generate an organized data. Genetic Algorithm will be used to maximize the feature-selection, which will raise the model's performance. to differentiate spam and legitimate messages and decrease the complexity of the computations. Afterward, the optimized features are trained using different machine learning classifiers with high accuracy in detection. This synergistic method does not only enhance the performance of classification, but is also dynamically reconfigurable to changing spam patterns, which is an effective and scalable solution to intelligent spam detection in real spam communication systems. The Table I shown the overview of relevant research on anti-spam measures.

Table I. The spam detection literature overview summary

S. No.	Research Article (Year)	Focus	Methodology	Key Findings
1	Kocyigit et al. (2024)	Feature reduction for spam/phishing detection	GA-based feature selection + ML classifiers	Reduced dimensionality with improved detection rate
2	Agrawal et al. (2024)	ML-powered spam identification system	NB, SVM, Random Forest classifiers	RF showed superior performance among traditional ML models
3	Enhanced Bio-inspired Model (2024)	Hybrid spam detection	Bio-inspired optimization + Deep Learning	Improved precision and recall over standalone ML
4	Ghatasheh et al. (2023)	Feature selection & hyperparameter tuning	Modified GA + XGBoost classifier	Reduced features while increasing prediction accuracy
5	Hybrid GA + CNN Model (2023)	Intelligent spam detection	GA for feature optimization + CNN classifier	Hybrid model outperformed single ML classifiers
6	Labonne & Moran (2023)	Few-shot spam detection	Transformer (Spam-T5) based LLM	Effective contextual detection with minimal training data

7	Rosita & Jacob (2022)	Twitter spam detection	Multi-Objective GA (MOGA) + CNN	Improved classification accuracy with optimized objectives
8	ML + Nature Inspired Survey (2022)	Review of ML & evolutionary techniques	Systematic review of GA, PSO with ML	Hybrid models outperform standalone ML techniques
9	Wiley ML Spam Study (2022)	Intelligent spam filtering	ML algorithms including XCS & GA	Showed benefits of combining evolutionary learning with ML
10	Fallah Sokhangoee & Rezapour (2021)	Feature selection using association rules	GA + Association Rule Mining + ML classifiers	Improved detection efficiency and reduced false positives
11	SVM + GA Spam Classification (2021)	Email spam classification	Genetic Algorithm + Support Vector Machine	GA improved SVM accuracy by optimizing features
12	ML Benchmarking Study (2020)	Comparative spam detection	NB, SVM, RF comparative evaluation	RF and SVM outperformed NB on high-dimensional datasets
13	SpamML Framework (2020)	ML-based spam detection system	End-to-end ML framework implementation	Established baseline architecture for intelligent spam filtering

3. Proposed Methodology

The proposed ISDF is a framework that combines Genetic Algorithms (GA) and Machine Learning (ML) to create an adaptive and efficient spam detector is displayed in Figure 1. The procedure includes a few important phases, including data collection and pre-processing, then optimization of selection of attributes using genetic algorithm, and the last phases involve training and evaluation with the help of machine learning classifiers. To generate and develop a smart spam detection system which will perfectly categorize the messages (email/SMS/social media) as spam and legitimate content using a combination of the Genetic Algorithms (GA) and the Machine Learning (ML) algorithm.

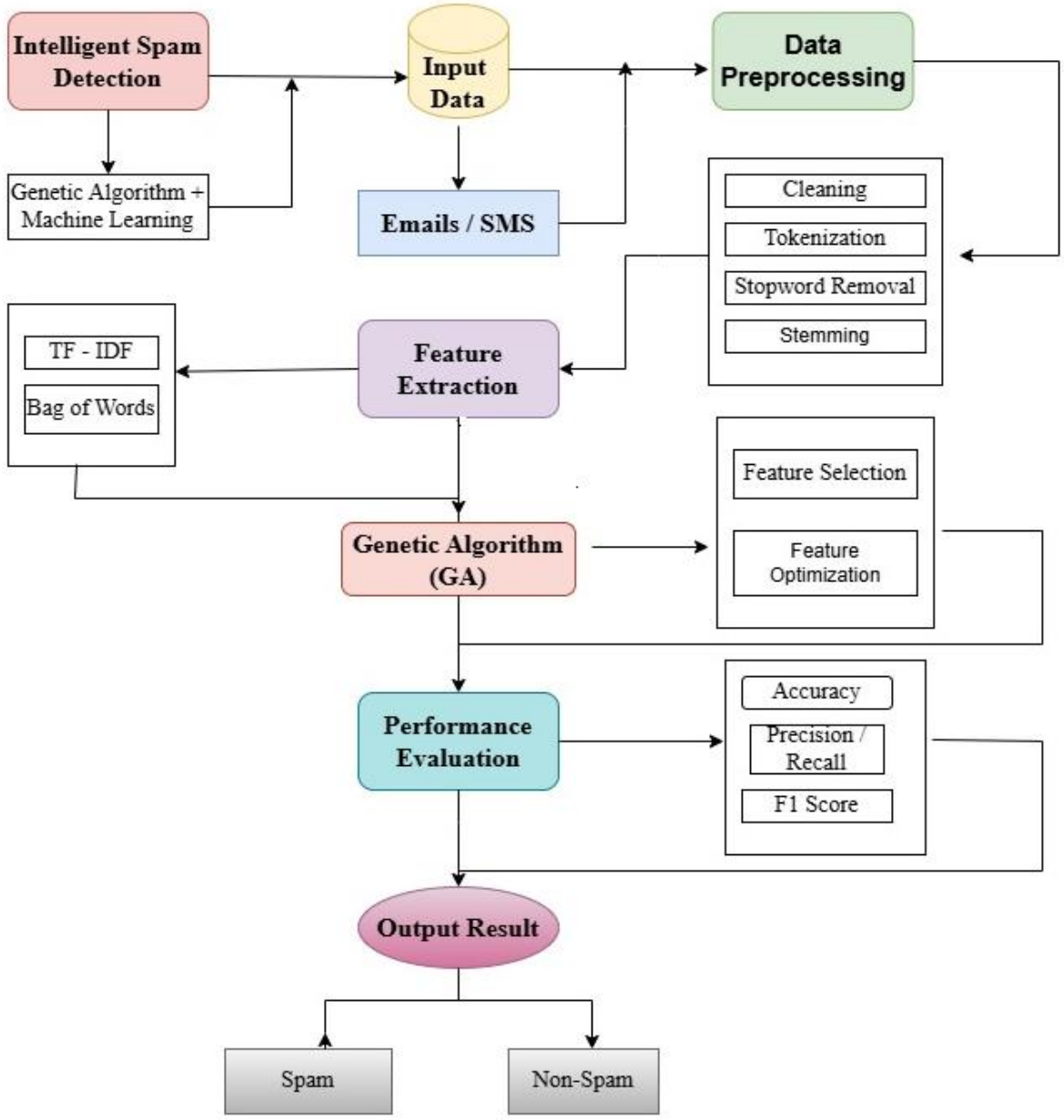


Figure 1: Workflow of an Intelligent Spam Detection Model (GA + ML)

The below sections provide the detailed description of the proposed methodology of the ISDF via the Genetic Algorithm and Machine Learning. The model combines both the evolutionary optimization and supervised learning methods to increase the accuracy of classifying spam and lower the feature count. Every part of the method like as data acquisition, pre-processing, feature extraction, optimization with the help of Genetic Algorithm, and performance value are clearly outlined to give a detailed overview of the proposed method.

- Genetic Algorithm and Machine learning
- Input Data (Emails / SMS)
- Data Pre-processing
- Feature extraction

- The Genetic Algorithm (GA)

3.1 Genetic Algorithm and Machine learning

The system proposed incorporates Genetic Algorithm (GA) of feature optimization and machine learning (ML) model training to improve email spam detection. It is a two-linked module: Feature Optimization using GA and ML Model Training and Evaluation that comprises a closed-loop and a high-performance detection system displayed in Figure 2.

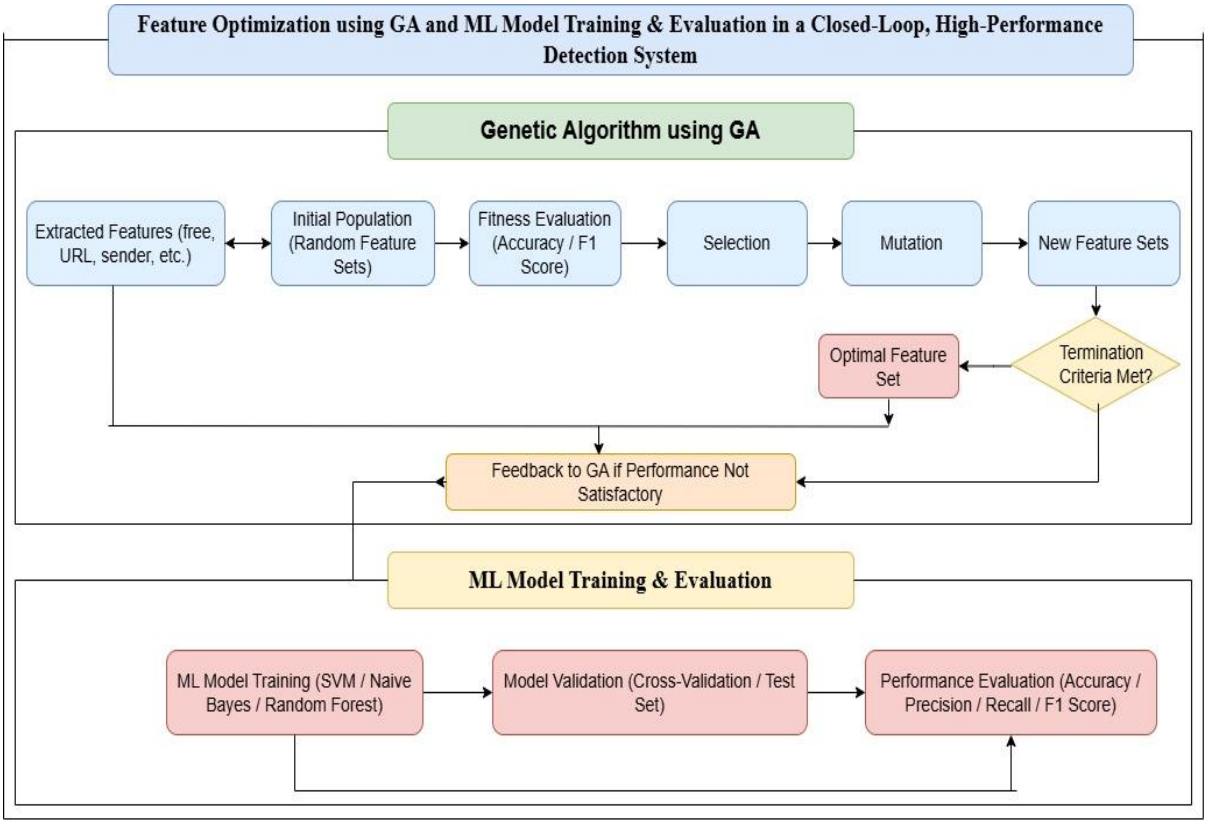


Figure 2: Architecture of Hybrid Genetic Algorithm–Machine Learning (GA–ML)

Machine learning models are taught to produce effective spam detection classifiers based on feature sets optimized using the Genetic Algorithm. At this stage, a no. of ML models are employed. A few examples include Support Vector Machines (SVM), which excels at binary classification in feature spaces with many dimensions, Naive Bayes (NB), a probabilistic classifier based on Bayes' theorem, and Random Forest (RF), an ensemble learning model that constructs a series of decision trees and integrates their outputs to strengthen classification. These classifiers undergo training on the training dataset, and their reliability and generalizability are then tested using either an independent validation dataset or cross-validation.

3.2 Input Data (Emails / SMS)

Input Data (Emails / SMS) block is the main source of data of the system. It holds crude textual messages gathered in email messages and short message service (SMS) platforms. Such messages can be either legitimate (ham) or malicious/spam meant to result in phishing, fraud or advertisements.

3.3 Data Pre-processing

One of the most important pre-processing stages of an intelligent spam detector framework is data pre-processing since it guarantees that the raw email data is clean and consistent and can be used to train machine learning models with extracted features. The email usually is filled with noisy, redundant, or disorganized information and therefore, when unprocessed, this may affect the results of the classifier adversely. The quality of features in a Genetic Algorithm (GA) optimization and subsequent machine learning (ML) classification can be affected largely by the quality of pre-processing in a GA-ML hybrid scheme.

Data pre-processing stage is divided into four main steps, which are

- Data Cleaning
- Tokenization
- Stopword Removal
- Stemming.

The combination of these steps provides the standardization of the raw email data and noise removal. All these phases are discussed below.

3.3.1 Cleaning the data

Cleaning of text data including text messages have other considerations because it is in the form of text data that is not structured. Text normalization is the method of changing text into a uniform format in a bid to minimize inconsistencies. This may involve the ability to convert all text to either lowercase or upper case, delete punctuation marks, and delete or substitute special characters or symbols. Text messages can have special characters, emoji's, or other symbols that can affect the analysis. The deletion or substitution of these special characters and symbols can make the text data processed properly.

3.3.2 Tokenization

One of the early stages in the text data pre-processing is tokenization, during which a piece of text (the body of an email) is divided into small units (collins) known as tokens. The tokens are usually words, phrases, or symbols and represent the fundamental input of the feature extraction and machine learning models. The process of tokenization can be required to identify linguistic patterns, frequencies and contextual associations, all of which are important in detecting spam by converting unstructured text into structured tokens. The Figure 3 shown the process of tokenization.

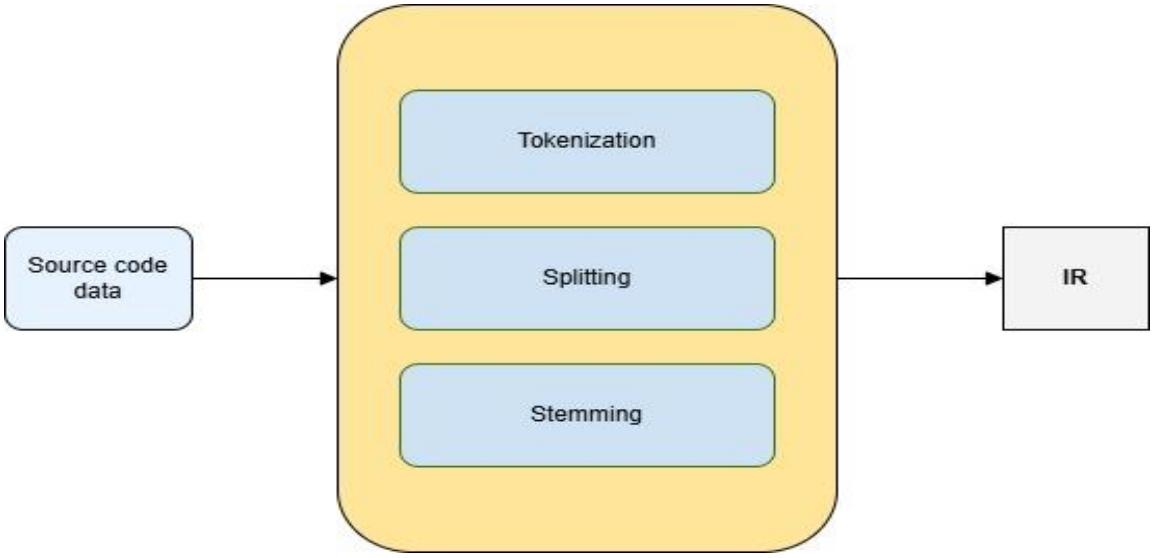


Figure 3: Pre-processing Steps for Source Code Information Retrieval

Example: The raw email text is unstructured and contains punctuation, capitalization, and stopwords is shown in Figure 4.

"Dear user, your account has been temporarily suspended. Verify your account now to avoid permanent closure."

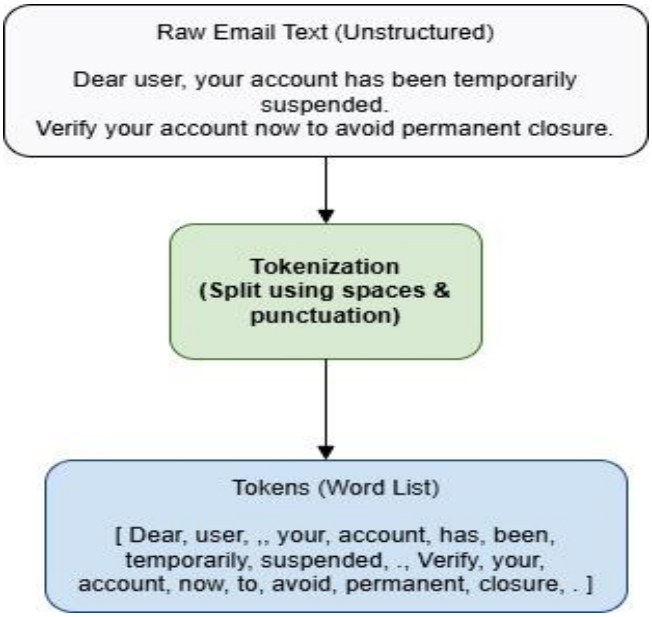


Figure 4: Tokenization of Raw Email Text

3.3.3 Stopword removal

Stopwords- prepositions, articles, pronouns, etc. are words that are rather frequent and do not usually add any valuable information when text mining is conducted. Since they can be found in almost all documents they can safely be left out of analysis without distorting the body as well. Stopwords can be eliminated to shrink the dataset, reduce the amount of text, and enhance processing performance. This can be done using any pre-defined pool of such words, based on the circumstances and the objectives of the analysis. The Table II shows the stop word removal method, which is applied in pre-processing of the text.

Table II. Stop Word Removal in Text Pre-processing

Sample Text with Stop Words	Without Stop Words
The brown fox outruns the lethargic dog.	Quick, Brown, Fox, Jumps, Lazy Dog
She is going to the market to buy some vegetables	Going, Market, Buy, Vegetables
We should take a break and relax for a while	Should, Take, Break, Relax, While
This is an example of how stop words work	Example, Stop Words, Work
He was not sure if he could finish it on time	Sure, Finish, Time

3.3.4 Stemming

Text mining, information retrieval, and natural language processing use stemming to treat different word forms as one item. This lessens repetition in the text information and enhances rapidity of algorithms. As shown in the figure, such words as connects, connected, and connecting will be shortened to the root word connect. Likewise, happily and happiness can be stemmed to happi and this illustrates how stemming provides a standardization of the variations of a word so as to analyze it easily. This process assists in the betterment of search results, text classification, and sentiment analysis since the attention is made on the gist of the text instead of the superficial differences. The stemming technique is shown in Figure 5 and is used when preparing the text.

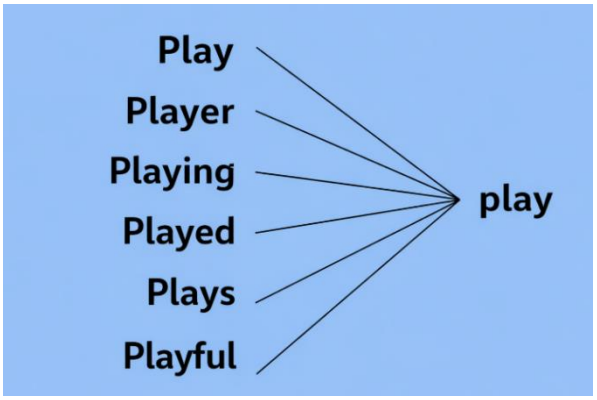


Figure 5: Stemming Process in Text Pre-processing

3.2 Feature extraction

Email spam detection involves a very important stage of feature extraction, in which the content of emails is systematically derived to obtain the information that is relevant and make it capable of being classified. Several kinds of features are chosen in this step, including textual ones like word frequency, bigrams and n-grams that have the ability to capture the linguistic patterns that are characteristic of spam messages, metadata features like email sending, email subject line and time of sending that can provide structural and behavioral information, and contextual features such as the inclusion of URLs, attachments, or spam-prone keywords. Collectively these properties make up a complete picture of email properties and are critical in the proper posing of the distinction between spam and non-spam emails.

Detecting email spam utilizes a very important feature extraction step that converts raw text into a feature list that can undergo machine learning. Two popular feature extraction methods that are used in this analysis include:

• Bag of Words (BoW)
• TF IDF: Term Frequency Inverse Document Frequency.
These methods turn unstructured text into numerical features that machine learning models can detect. The methods will be detailed below.

3.4.1 Bag of Words (BoW)

One of the simplest and most common NLP text demonstration models is the Bag of Words (BoW) model. It turns unstructured text into numerical feature vectors for machine learning. Bow converts email and SMS text into machine-readable form in spam detection systems by describing documents using word frequency. The model presupposes the importance of words occurrence rather than the order of words. In this way, the syntactic structure and grammar are disregarded. Let

D = { d_1, d_2,……,d_N } be a corpus of N documents.

V = { w_1, w_2,……..……,w_N } be the vocabulary of M unique words extracted from the corpus.

For each documentd_i, the Bag of Words representation is defined as:

Where $F_{ij} = frequency\ of\ word\ w_j\ in\ document\ d_i$

Thus, each document is represented as an M-dimensional vector.

Example: Consider two SMS messages:

- d_1: "Win money now"
- d_2: "Win prize money"

Step 1: Vocabulary Construction
V= {win, money, now, prize}

Step 2: Document-Term Matrix
Table III shows the frequency of each word from the vocabulary V= {win, money, now, prize} in the two SMS messages d_1 and d_2.
Table III: Document–Term Frequency Table

Word	**d_1**	**d_2**
win	1	1
money	1	1
now	1	0
prize	0	1

Step 3: Vector Representation
X1 = [1, 1, 1, 0]
X2= [1, 1, 0, 1]

Each document is transformed into a numerical vector that can be fed into a classifier. In spam detection, frequent words such as Win and Money contribute directly to classification decisions.

3.4.2 Inverse Document Frequency (TF–IDF)

TF-IDF uses math to determine a word's relevance in a document compared to a corpus. TF-IDF is a grouping of Terms Frequency (TF) which is used to estimate the frequency of a term in a document, and Inverse Document Frequency (IDF) which is used to estimate the frequency of a term in a corpus of documents. Rare and informative terms are weighted higher than common words in most documents. It is used in text categorization and spam detection to improve feature representation and model accuracy. Figure 6 shows how the TF-IDF text representation model converts raw text into a numerical-valued feature vector by computing the frequency of a term (TF) and a word's occurrence divided by the number of documents containing the word.

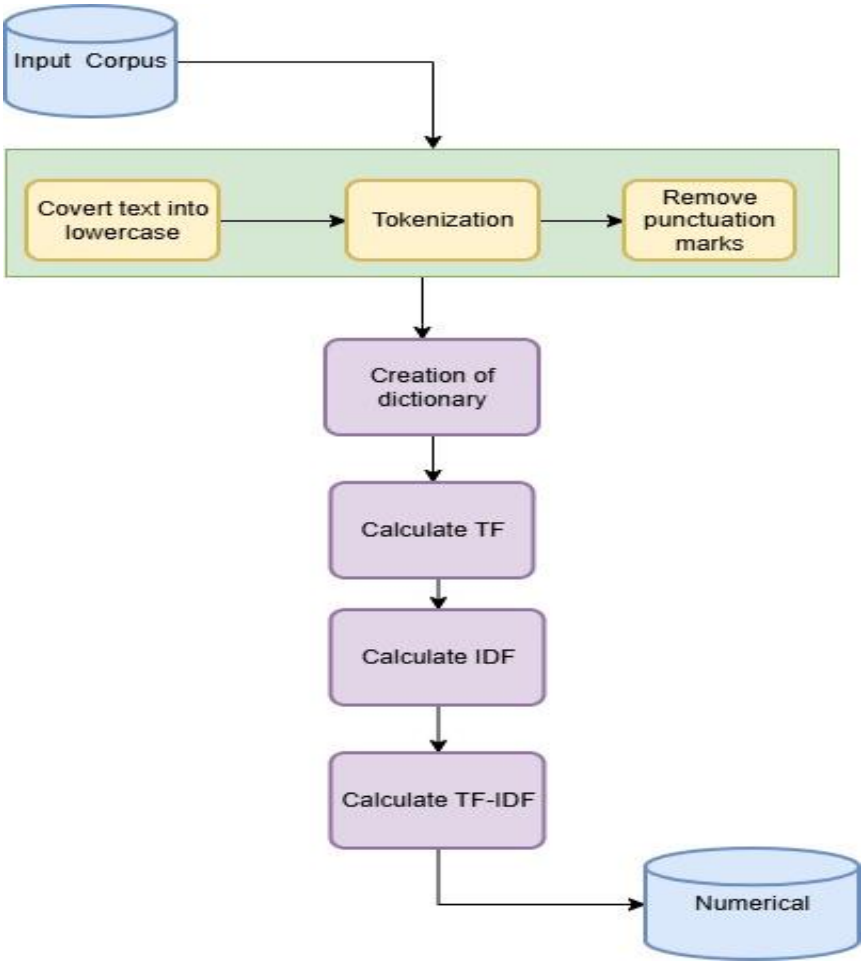


Figure 6: Workflow of TF–IDF Text Representation Model

The TF -IDF score is computed in the following way: TF-IDF = TF x IDF as shown in equations (1), (2) and (3)

Step 1: Compute Term Frequency (TF)

$$TF_{ij} = \frac{Term\ count\ in\ document}{Total\ terms\ in\ document} \quad (1)$$

Step 2: Calculate IDF

$$IDF_j = \log\left(\frac{N}{df_j}\right) \quad (2)$$

Step 3: Multiply TF and IDF

$$TF - IDF = TF_{ij}\ X\ IDF_j \quad (3)$$

By using the above formulas

- d_1 : "Win money now" (3 words)
- d_2 : "Win prize money" (3 words) s
- N = 2 these two statement were considered for evaluation of TF-IDF.

Step 1: Compute IDF

"win" appears in both documents → d$f = 2$

$IDF\ (win) = \log\left(\frac{2}{2}\right) = \log\ (1) = 0$

"money" appears in both documents → d$f = 2$

$$IDF\ (money) = 0$$

"now" appears in only $d_1 \rightarrow$ d$f = 1$

$IDF\ (now) = \log\left(\frac{2}{1}\right) = \log\ (2) = 0.301$

"prize" appears in only $d_2 \rightarrow$ d$f = 1$

$$IDF\ (prize) = 0.301$$

Step 2: Compute TF for d_1

$$TF\ (win, d_1) = \frac{1}{3} = 0.33$$

$$TF\ (money,\ d_1) = 0.33$$

$$TF\ (now, d_1) = 0.33$$

Step 3: Compute TF–IDF for d_1

$$TF - IDF\ (win, d_1) = 0.33\ X\ 0 = 0$$

$$TF - IDF\ (money, d_1) = 0.33\ X\ 0 = 0$$

$$TF - IDF\ (now, d_1) = 0.33\ X\ 0.301 = 0.99$$

TF–IDF Vector Represented

$$d_1 = [0, 0, 0, 0.099, 0]$$

$$d_2 = [0, 0, 0, 0, 0.099]$$

Interpretation:

- Words appearing in all documents receive zero weight.

- Rare words receive higher importance.
- TF–IDF highlights discriminative terms such as *now* and *prize*.

This makes TF–IDF more effective for spam detection, as distinctive spam keywords receive higher weights compared to common terms.

While TF–IDF incorporates document-level rarity to introduce importance weighting, the Bag of Words model offers a frequency-based numerical representation of text. BoW is computationally straightforward, but because of its discriminative weighting mechanism, TF-IDF typically enhances classification performance in spam detection systems.

3.5 The Genetic Algorithm (GA)

The Figure 7 describes how the Genetic Algorithm (GA) works, starting with the first population till the creation of a new generation. The algorithm starts with a starting population of candidate solutions, and this is subjected to a selection process to select the fittest ones. The individuals that have been chosen are then subjected to crossover or genetic information exchange and mutation where small random changes take place in an attempt to keep the diversity alive. The resultant individuals are a new generation which hopefully possesses better solutions. The figure further indicates how the basic GA process has been adapted to machine learning whereby instead of using an initial population a dataset consisting of several features is used, however, the sequence of selection, crossover, mutation, and generation formation is retained to optimize learning models.

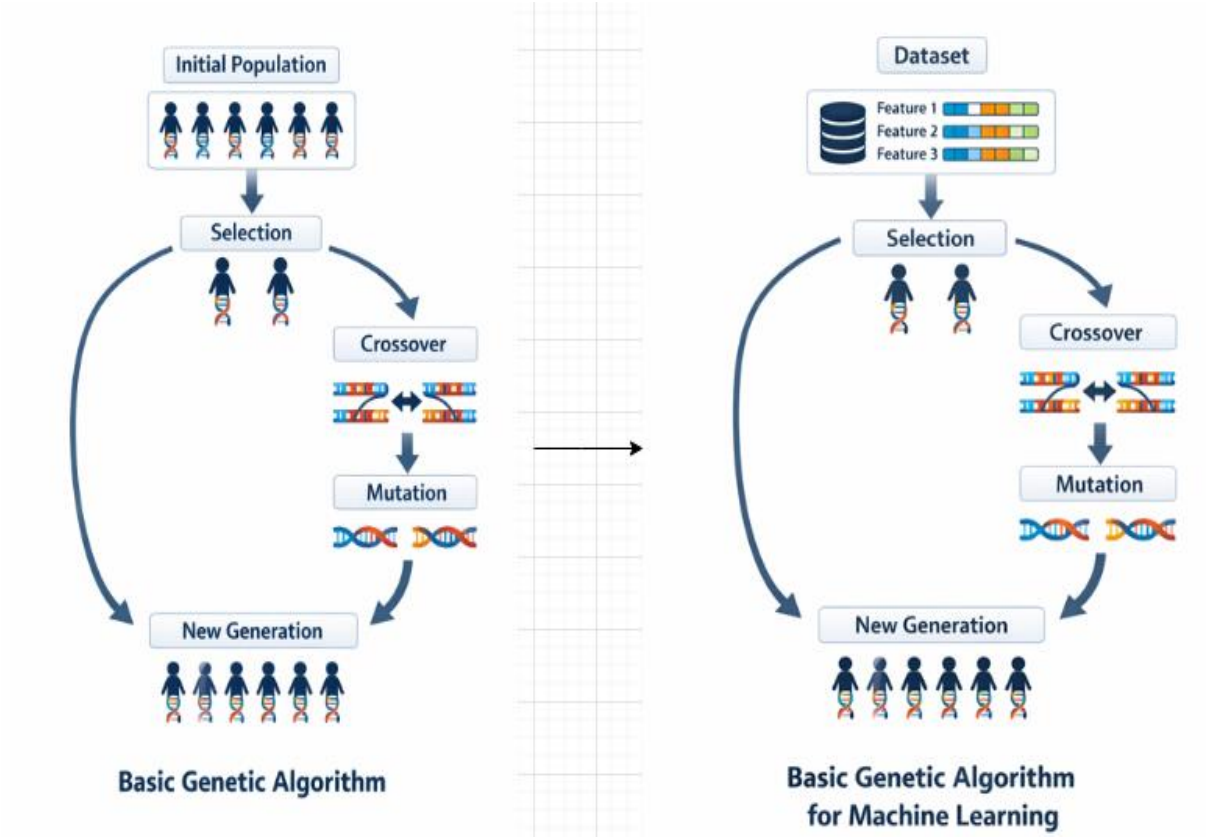


Figure 7: Genetic Algorithm Process: From Initial Population to New Generation

Natural selection and genetics serve as the inspiration for the evolutionary optimization method known as the Genetic Algorithm (GA). In order to identify the ideal or nearly ideal solution, it works with a population of potential solutions and evolves them over several generations. GA iteratively applies crossover, mutation, and selection processes under a fitness function that evaluates solution quality. Genetic Algorithm phases are shown in Figure 8.

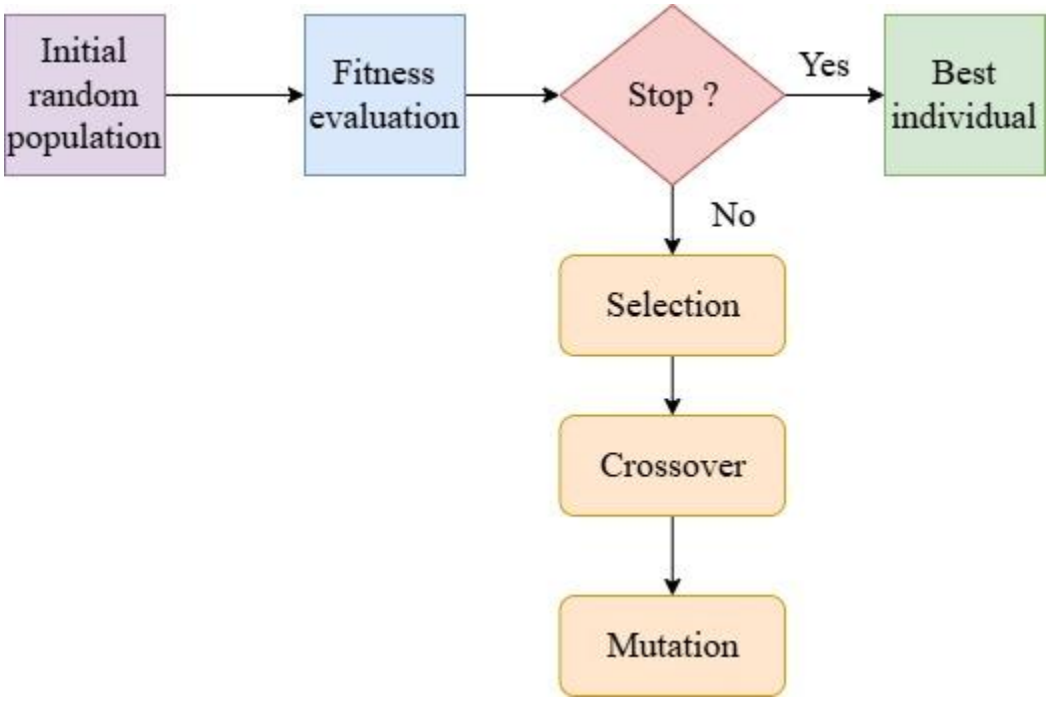


Figure 8: Genetic Algorithm Architecture

Genetic Algorithm (GA) architecture will consist of the following steps which will be discussed in details below:

- Initialization,
- Fitness Evaluation
- Selection
- Crossover
- Mutation
- Termination

A. Initialization

The GA method begins with a population of potential solutions, or "chromosomes." Every chromosome encodes a potential solution to the issue, which is usually expressed as a real-valued parameter, a binary string, or a vector of integers. To guarantee diversity in the search space and raise the likelihood of investigating various possible solutions, the population is generated at random.

For instance, a chromosome could represent a subset of features for a feature selection problem in spam detection as follows: C = [1, 0, 1, 0, 1, 0, 1].

In this case, 0 denotes that the feature is not selected, and 1 denotes that it is. A distinct set of characteristics is represented by each chromosome in the population.

B. Fitness Evaluation

Each chromosome is evaluated after initialization using a solution-quality fitness function. Problem-specific fitness functions guide optimum solution evolution. GA–ML spam detection framework fitness functions often include classification performance metrics including accuracy, precision, recall, and F1 score.

Example: An SVM classifier is trained using a chromosome that represents a subset of email features. The chromosome is given a fitness score if the classifier obtains 92% accuracy and a 0.90 F1 score on validation data. These metrics are then combined (e.g., weighted sum). Chromosome Cican's fitness can be expressed mathematically as follows in equation (4).

$$F(Ci) = \alpha \cdot Accuracy + \beta \cdot F1 - Score\ 8 \quad (4)$$

Where the weighting factors α and β indicate the relative importance of each metric.

C. Selection

The best chromosomes are chosen to be the parents of the following generation in the selection step. According to "survival of the fittest," chromosomes with better fitness scores are more likely to pass on their genes. Popular selection methods include roulette wheel, tournament, and rank-based. As an illustration, let's say that the top 50% of chromosomes are chosen to produce offspring based on fitness. Higher classification performance increases the likelihood that a chromosome will be selected.

D. Crossover

Crossover is a genetic operator that allows for the recombination of advantageous traits by joining two parent chromosomes to create new offspring. This enables the algorithm to inherit traits from several high-performing parents and encourages exploration of the search space. The following Table IV gives a short description of the chosen health characteristics (F1-F5) employed in the study.

Table IV: Description of Selected Health Features (F1–F5)

Feature	**Meaning**
F1	Age
F2	BMI
F3	Blood Sugar
F4	Cholesterol

F5	Blood Pressure

We use a Genetic Algorithm for feature selection.

- **1** → feature selected
- **0** → feature not selected

Chromosome Representation

Each chromosome = one candidate solution
Example chromosome:
[1 0 1 1 0]
→ Uses F1, F3, F4

Step 1: Select Parents (high fitness)
Parent 1:
[1 0 1 1 0]
Parent 2:
[0 1 0 1 1]
Step 2: Choose Crossover Point
Assume single-point crossover after 3rd gene
Parent 1: [1 0 1 | 1 0]
Parent 2: [0 1 0 | 1 1]
Step 3: Exchange Genes
Offspring 1:
[1 0 1 1 1]
Offspring 2:
[0 1 0 1 0]
Interpretation: New feature combinations are created and Offspring may give better model accuracy.

E. Mutation

Mutations modify chromosomes randomly to sustain population diversity and restrict early progress toward regional maxima. It ensures the GA explores new solution space areas that crossover alone cannot reach. For instance, mutation results in minor, arbitrary changes.
Let's say we have a dataset for machine learning that has five features:

Example: Mutation makes small random changes**.**
Before Mutation:
Offspring:
[1 0 1 1 1]
Mutation (flip one bit randomly)

Suppose gene 2 mutates:
After Mutation:
[1 1 1 1 1]

Interpretation: Feature F2 (BMI) is newly added and this feature may improve prediction performance. Table V is a summary of the contribution of the Genetic Algorithm operators, the crossover and mutation, in the learning process through the exploitation and exploration of existing and new combinations of features respectively.

Table V: Learning using Genetic Algorithm Operators

Operator	**Role in Learning**
Crossover	Exploits good feature combinations
Mutation	Explore new feature possibilities

Crossover combines selected features from two parent datasets to form new solutions, while mutation randomly alters feature values to maintain diversity and improve learning performance.

F. Termination

Until a termination criterion is satisfied, the GA iteratively repeats the fitness evaluation, selection, crossover, and mutation processes. Typical stopping circumstances consist of:

- Reaching a certain number of generations
- Reaching a desired level of fitness
- Not seeing any notable progress over a number of generations

Best option after termination is the chromosome with the highest fitness score.

The below Figure.9 indicates the correctness of an intelligent spam detection structure using Genetic Algorithm (GA) and Machine Learning at various stages of the GA. The accuracy increases with increase in generations with an estimate of 72 at generation 2 and 92 at generation 30. The best improvement is in the young generations, and it means that the optimization takes place at the beginning. The accuracy levels off after approximately 20 generations and provides only slight improvements afterwards

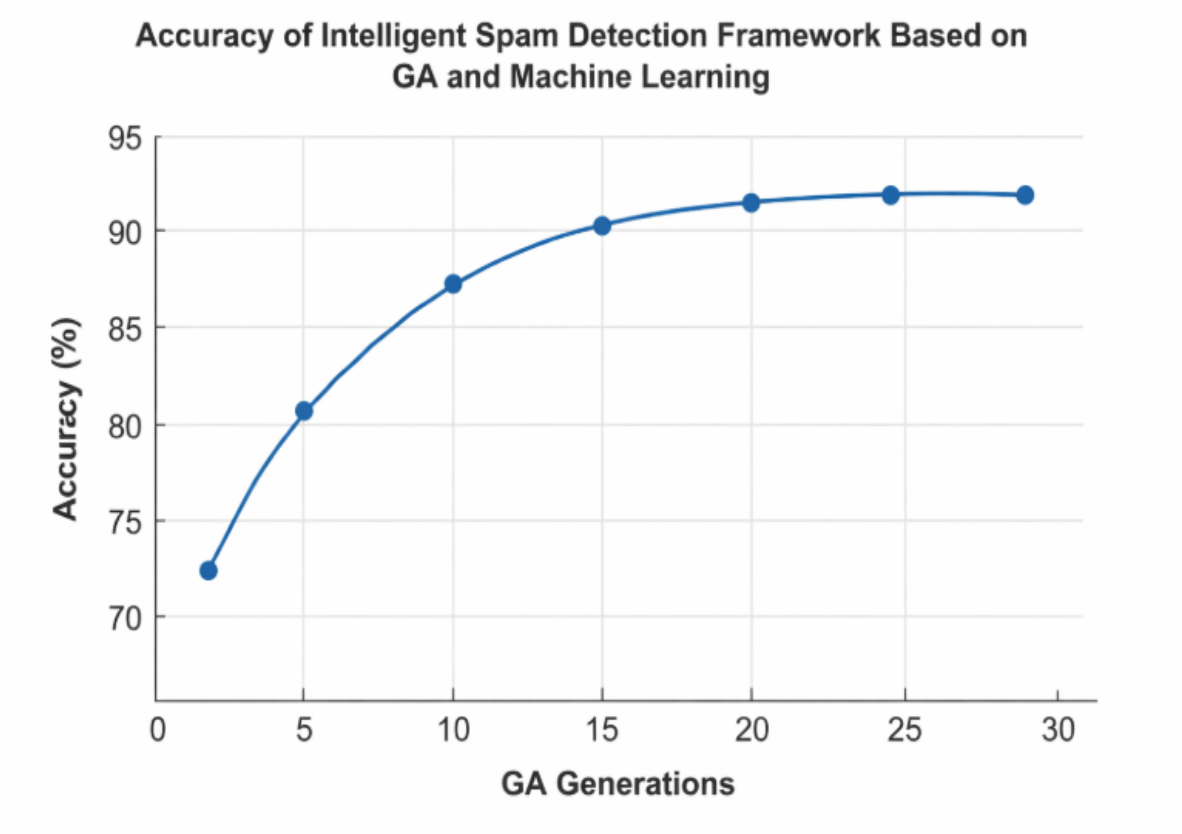


Figure 9: Performance Analysis of GA-Based Spam Detection Model

3.5.1. Feature selection with GA

The GA for feature selection is intended to find the optimal set of features that maximizes the performance of spam classification. The algorithm starts with the initialization of a population of random feature subsets, which are referred to as chromosomes. The subsets are then assessed using a fitness function, usually involving machine learning performance criteria like accuracy, precision, recall, or F1 measure, to determine their efficiency. The selection step selects the most optimal subsets for reproduction, while crossover involves the combination of different subsets to produce new offspring. Mutation involves the introduction of random modifications to some features to ensure diversity and prevent convergence. This continues until a termination condition is satisfied, such as reaching a maximum number of generations or achieving acceptable performance, resulting in the optimal feature set that can reduce dimensionality while maintaining or improving classification accuracy.

3.5.2 Feature Optimization with GA

Feature optimization is carried out through a GA to find the optimal combination of features for spam classification. First, a population of potential feature sets is created randomly, where each member of the population represents a combination of the original feature set extracted. Each spam feature set is then evaluated using a fitness function based on accuracy, precision, recall, or F1 measure. Fitness values determine which feature sets are reproduced. Crossover and mutation produce new feature subsets by combining features from parent feature sets and introducing random changes. This procedure continues until a termination criteria, like defined number of generations or satisfactory performance, is met to obtain the best feature set for machine learning model training. The proposed experimental design of this study

entails testing the proposed Intelligent Spam Detection Framework with a large pool of emails and SMS messages including both spam and legitimate (ham) samples. The data are subjected to a pre-processing phase to remove noise, text normalization and other value features such as frequency of words and metadata of messages. Genetic Algorithm optimizes features selection to boost the performance of the classifier as well as to minimize the computational workload. The 10-fold cross-validation strategy trains and validates machine learning classifiers like Naive Bayes, Support medial machine, and random forest for robustness and generalizability. The framework's accuracy, precision, recall, and F1-score are compared to current spam detection methods to demonstrate its efficacy.

3.6 Flow of the system

The flowchart shown in the Figure 10 presents the operation of a Genetic Algorithm (GA). It works by creating an initial random population of individuals (solutions), and evaluating their fitness on some pre-defined fitness function. The algorithm will then test the termination condition; in case it is not met, it will choose the most optimal individuals to create the next generation. Crossover and mutation are used to produce new offspring, and the cycle is repeated until the termination condition is met, resulting in the optimum solution.

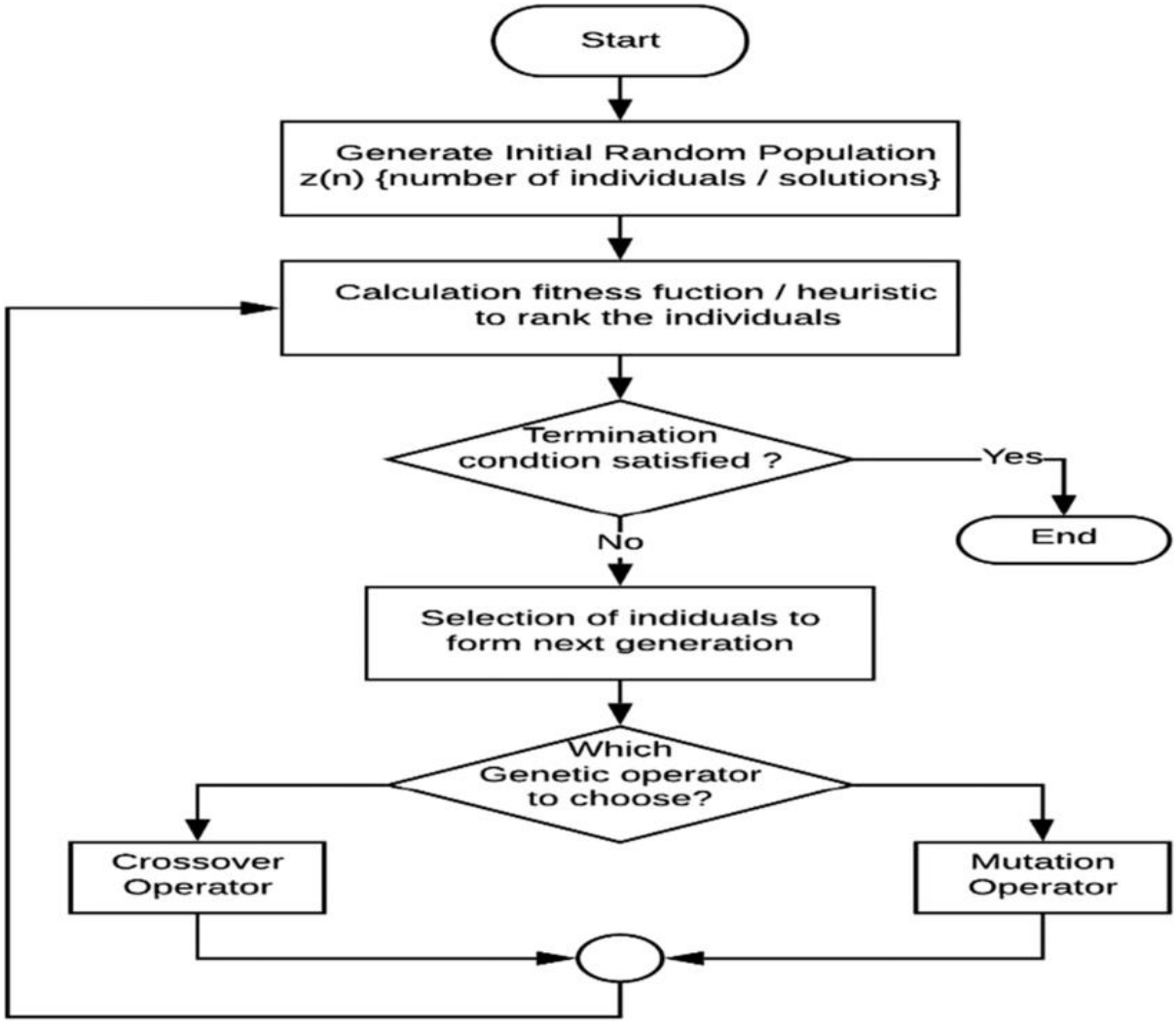


Figure 10: Flow Diagram of GA-Based Evolutionary Process

3.7 Dataset Description

Email Spam Detection data collected in this paper was obtained via Kaggle, a well-known machine learning and data science platform based on open data. The dataset is spam mail specific and provided with the labelled samples of actual email messages that are to be utilized in the supervised learning algorithms. The dataset consists of 5,172 email messages, and each entry in the dataset is associated with a single email and is identified as a spam (1) or a non-spam (0). The data is in a CSV format which is easy to load and manipulate using typical data science packages.

The main data features of the data set are:

- Number of samples: 5,172 emails.
- Variables: The Binary target variable will be filled with class labels: spam and non spam (ham) e-mails.
- Features: A list of 3000+ popular email words, with frequency for each word in each column.
- Format: Comma-separated values (.csv) with email feature vector and label per row.
- Use: Trains and tests machine-learning models to distinguish authentic emails from spam using textual data.

Natural language processing (NLP) is widely used to pre-process raw email text into numerical characteristics for machine learning models, such as tokenization, stop word removal, and factorization (TF IDF). The dataset's binary categorization makes it useful for assessing benchmark techniques like Logistic Regression, Multinomial Naive Bayes, Support Vector Machines, and ensemble methods. Table VI lists Python libraries used for implementation.

Table VI: Libraries for Email Spam Detection

Algorithm	**Python Library**	**Module/Class**
Multinomial Naive Bayes	scikit-learn	sklearn.naive_bayes.MultinomialNB
Linear SVM	scikit-learn	sklearn.svm.LinearSVC
XGBoost	xgboost	xgboost.XGBClassifier
Genetic Algorithm (Optimized Model)	deap / pygad	deap.algorithms / pygad.GA

4. Exrerimental Setup

The experimental design of the study is based on the evaluation of the suggested Intelligent Spam Detection Framework with the help of a full-scale dataset of emails and SMS messages containing spam and non-spam (ham) samples. Dataset pre-processing includes noise reduction, text normalization, and collecting relevant features like text frequency and message information. The Genetic Algorithm optimizes feature selection to increase classifier performance and reduce workload. To make it robust and generalizable, Support Vector Machine, Naive Bayes, and random forest classifiers are trained and tested with 10-fold cross-validation. The framework's accuracy, precision, recall, and F1-score are measured and compared to current spam detection methods to verify its efficacy.

5. Experimental Findings

5.1 Evolution Metrics

Any machine learning or GA-ML system needs performance computation to quantify its spam and non-spam email classification performance. Most performance evaluation measures employ the confusion matrix to determine Accuracy, Precision, Recall, and F1 Score. Performance evaluation is carried out to evaluate the efficiency of the designed spam detection system by evaluating the performance of the trained machine learning models in terms of standard classification metrics. Accuracy, precision, recall, and F1 score calculate the harmonic mean of precision and recall, the percentage of correctly classified emails, spam emails, and model-detected spam emails, respectively. These performance indicators reveal the spam detection system's efficacy and applicability.

Confusion Matrix: Table VII summarizes classification performance using a confusion matrix:

Table VII: Spam Detection Confusion Matrix

	Predicted Spam	**Predicted Non-Spam**
Actual Spam	True Positive (TP)	False Negative (FN)
Actual Non-Spam	False Positive (FP)	True Negative (TN)

Where:

• TP = Correctly identified spam emails
• TN = Correctly identified non-spam emails number
• FP = Number of legitimate emails mislabeled as spam.
• FN: Number of spam emails mislabeled as non-spam.

It measures model accuracy as given in equation (5).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

- Interpretation: Percentage of total emails correctly classified.
- Limitation: Can be misleading for imbalanced datasets (e.g., if most emails are non-spam, a naive model predicting all as non-spam may have high accuracy).

Example: If a dataset has 100 emails: 40 spam and 60 non-spam, and the model correctly classifies 35 spam (TP) and 55 non-spam (TN):

$$Accuracy = \frac{35+55}{100} = 0.90\ (90\%)$$

Precision: It measures the proportion of correctly identified positive predictions (spam). Among all emails predicted as spam in equation (6).

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

- Interpretation: How many spam emails are real.
- Importance: More precision ensures that legitimate emails are not wrongly classified as spam.

Example: If the model predicted 38 emails as spam, of which 35 were actually spam:

$$Precision = \frac{35}{38} = 0.921\ (92.1\%)$$

Equation (7) calculates recall as the percentage of spam emails properly identified.

$$Recall = \frac{TP}{TP+FN} \qquad (7)$$

- Interpretation: Ability of the model to detect spam emails.
- Importance: High recall ensures that most spam emails are caught, even if some non-spam emails are incorrectly flagged.

Example: If there were 40 spam emails and the model correctly detected 35:

$$Recall = \frac{35}{40} = 0.875\ (87.5\%)$$

F1 Score: The harmonic mean of Precision and Recall is the F1 Score, which balances both. As shown in equation (8).

$$F1 = 2\ X\ \frac{Precision\ X\ Recall}{Precision+Recall} \qquad (8)$$

Interpretation**:** F1 Score gives a balanced measure for classification, especially useful for imbalanced datasets.

Example**:** Using the previous values:

$$F1 = 2\ X\ \frac{0.921\ X\ 0.875}{0.921 + 0.875}$$

Table VIII lists the evaluation metrics for classification models, along with the formula for each metric and how it is used to measure how well a model works.

Table VIII: Classification Model Calculation Metrics and Their Formulas

Metric	Formula	Purpose / Interpretation
Accuracy	(TP + TN)/(TP + TN + FP + FN)	Overall correctness of the model
Precision	(TP/(TP + FP)	Correctness of positive predictions
Recall	(TP/(TP + FN)	Ability to detect all actual positives
F1 Score	2. (Precision. Recall)/(Precision + Recall))	Balance between Precision and Recall

5.2 Experimental Findings

The discussed implementation results are shown here. The results generated for different types of input given to the system and the corresponding output generated are discussed. The below Figure 11 is the interface of the homepage of the Spam Detection System. The page has a big banner block with the page title of SPAM Detection prominently written in the middle of the banner block with the sub title of integrated genetic algorithm on a spam detection. On the top right corner there is a navigation menu which has Home and Signup. The middle of the page has a Signup button highlighted to encourage the users to register. The background is a full-screen image, which makes the visual appearance more attractive and also offers a modern web application appearance to the smart spam detection system.

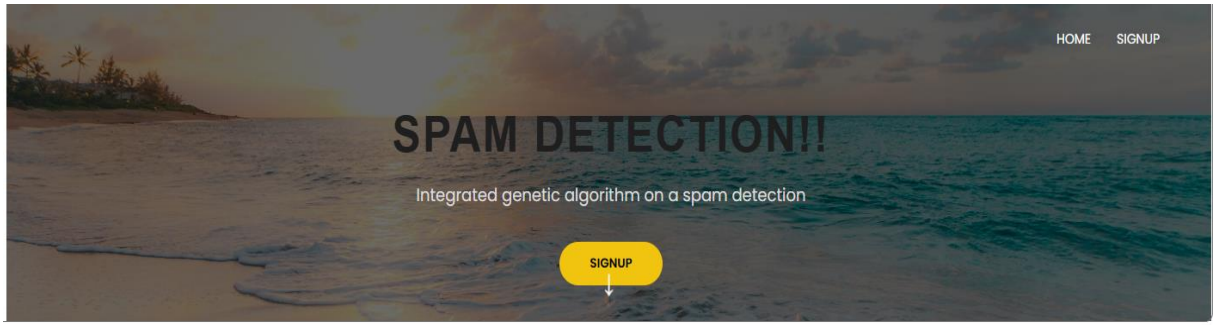


Figure 11: Home Page

The Figure 12 indicates the homepage of Spam Detection System. The first part will show the heading SPAM DETECTION!!, subheading of the integrated genetic algorithm, and Signup button. Three major features Creative Concept, Analysis, and Secure are discussed below with icons and the short description provided; these features emphasize functionality and security of the system.

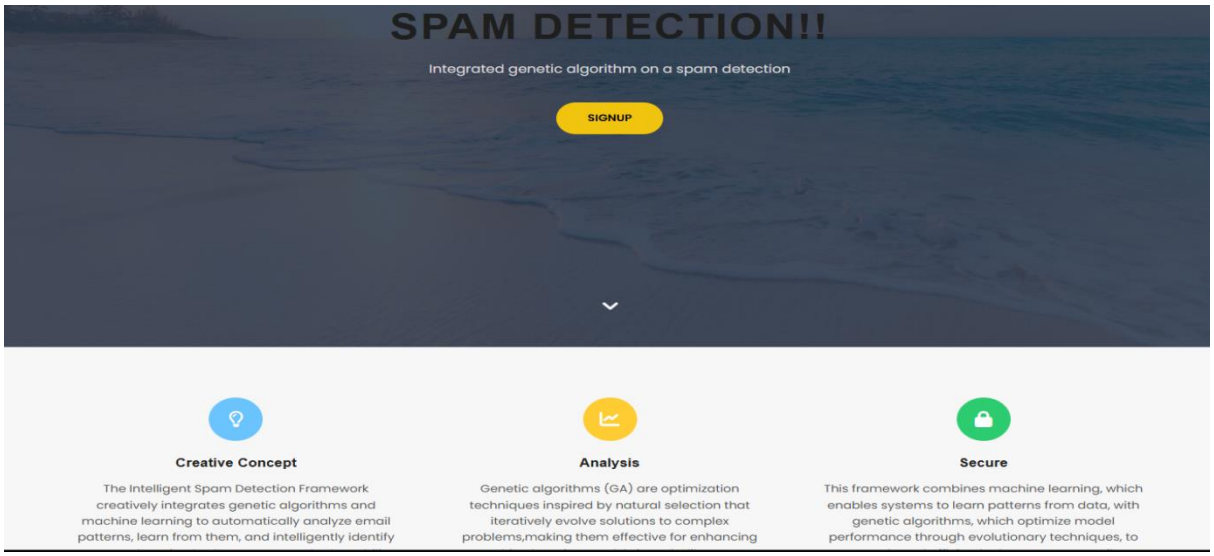


Figure 12: About the Page

This Figure 13 is an illustration of the Registration and Login interfaces of the Spam Detection System. A registration form, which includes the fields of username, password, email, and confirm password, is placed in the left panel with the button of Get Started. On the right panel is the login form whereby the users input their details such as user name and password, the option of Remember Me is presented and finally there is the button to log in that is used to log in into the system.

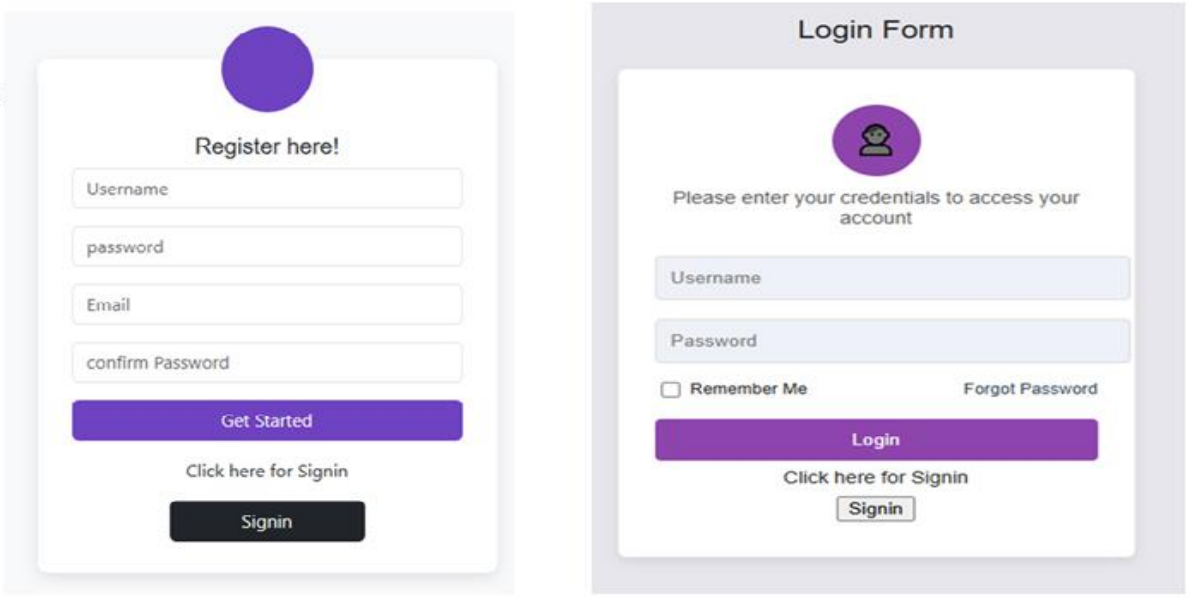


Figure 13: Registration of the user and Login page

The Figure 14 depicts a web-based application interface that is called Integrated Genetic Algorithm on a SPAM Detection, with a dark coloured navigation bar with tabs such as Home, About Us, Analytics and Sign out being on the top. The central part has the background in a matrix format with green digital codes and bold text in the middle saying, SPAM DETECTION! Under the banner, the user input area is displayed, where the message is to be typed in with the caption of your message here and below it a text box with the hint typing your message that has a blue button in the bottom part displayed and labelled as Predict. The interface seems to be aimed at asking the user to type a message and verify whether it is a spam by using a machine learning related spam classification system.

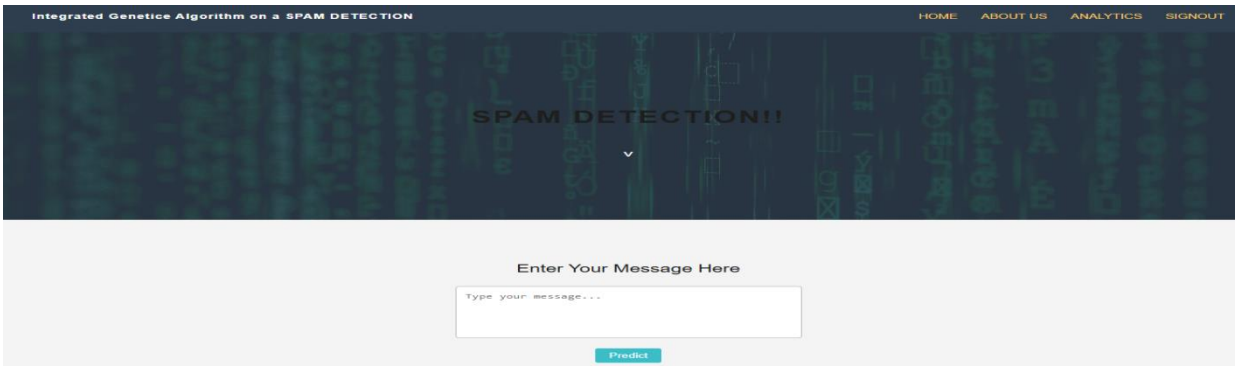


Figure. 14 Home page to enter the message

This Figure 15, 16 and 17 demonstrates the output port of a Spam Detection System that is incorporated with a Genetic Algorithm. The webpage will provide a navigation button in the form of header with options Home, About Us, Analytics and Signout. The message that is sent to undergo analysis is in the center and the results of the classification are displayed under Results for Comment. The message under analysis is shown and the system notifies that the message is a HAM meaning that it is not a spam message.

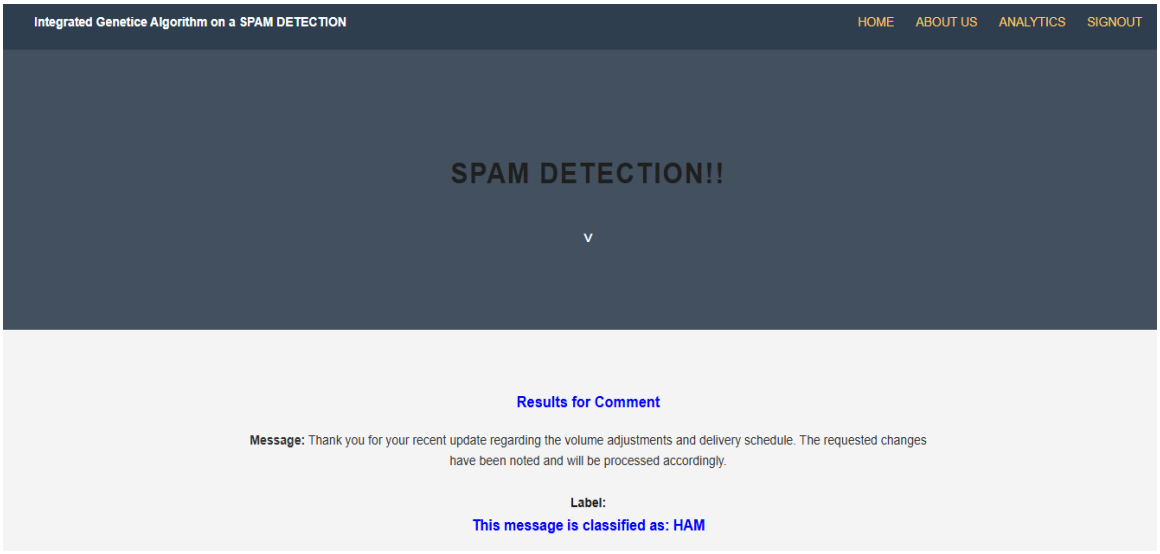


Figure 15: Generated output based on the given input

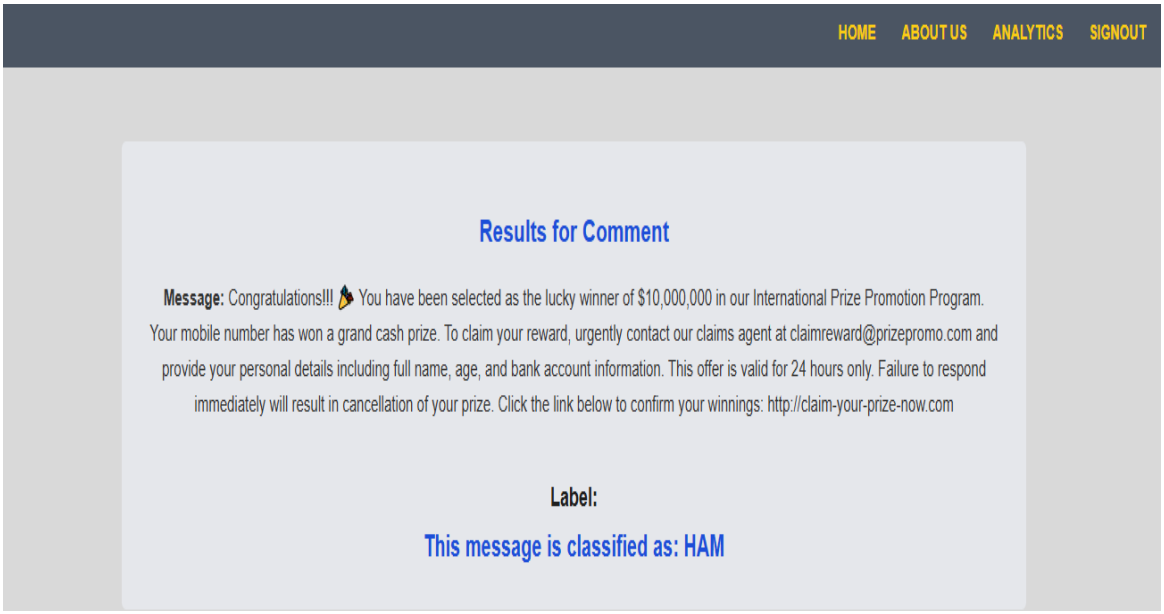


Figure 16: Output for the given input.

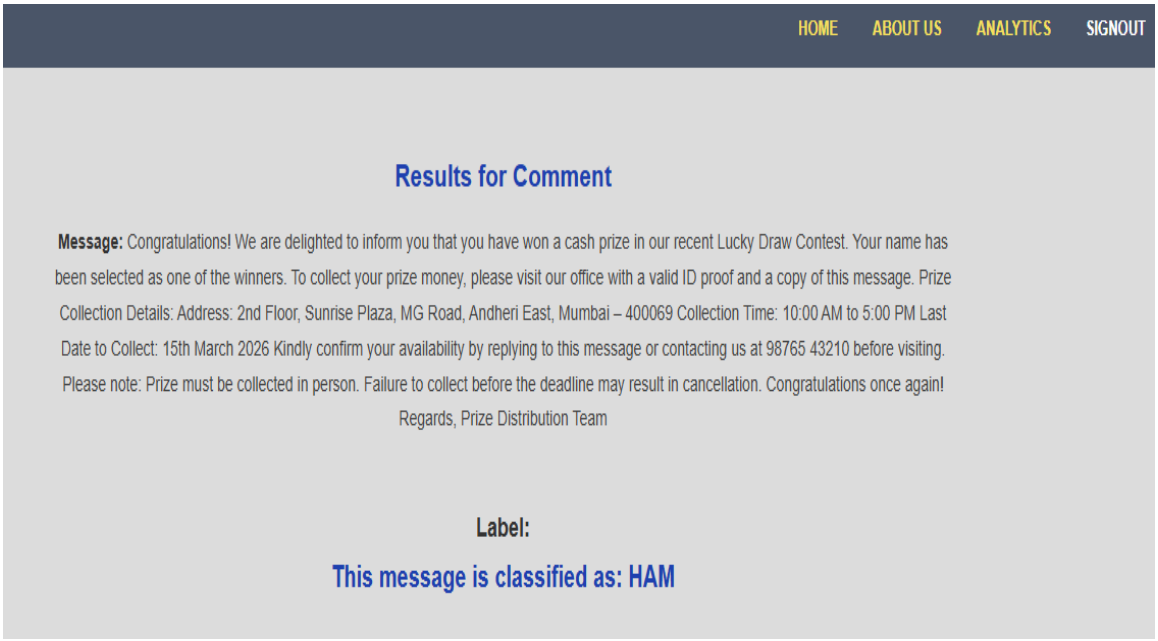


Figure 17: Output corresponding to the input.

This Figure 18,19 shows the output page of Genetic Algorithm based on a Spam Detection System. The title of the paper is displayed in the webpage header as Integrated Genetic Algorithm on a Spam Detection!!!! Whereby its user interface is in a clean layout. In a section named Results for Comment, the system gets an input message that the user has won a 50,000 Amazon gift voucher and gives the user a suspicious URL to redeem the reward. According to the result of the classification presented at the bottom, the system identifies the message as THE SMS TYPE IS SPAM in bold red font and makes it evident that the message is fraud and probably harmful to the users.

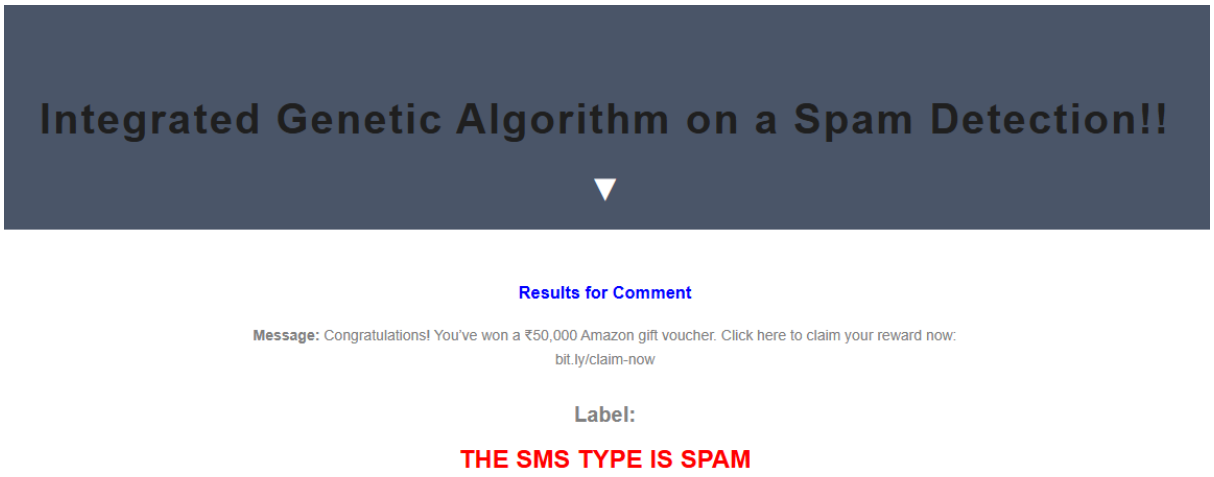


Figure 18: Output for the given input.

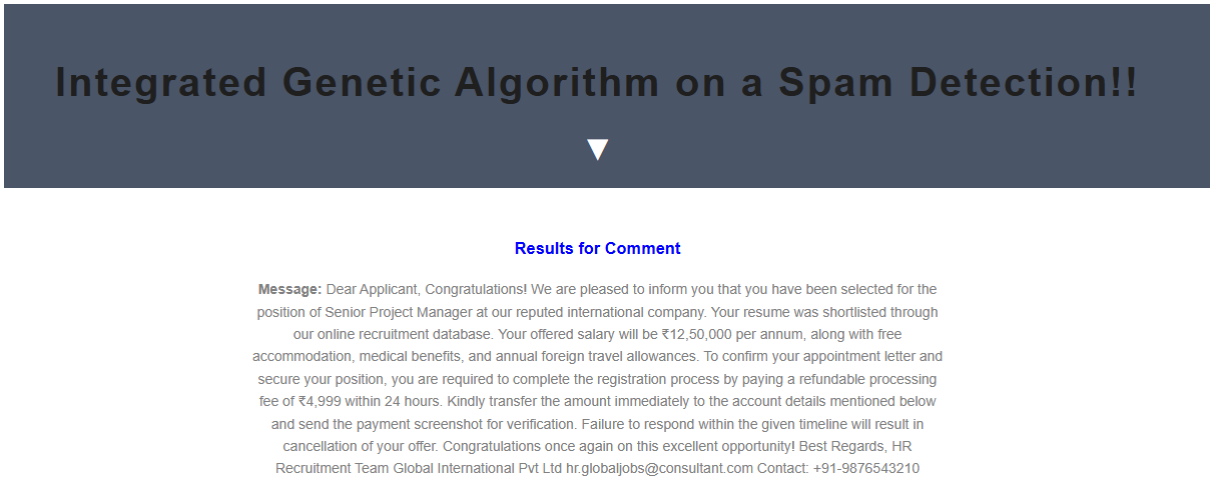


Figure 19: Output for the given input.

5.3 Performance Evaluation

The Figure 20 illustrates how the proposed Intelligent Spam Detection Framework (ISDF) is evaluated by way of Genetic Algorithms and Machine Learning. The figure shows the graphical and tabular data of the major measures of evaluation applied to calculate the effectiveness of the model. Class 0 has virtually flawless precision, recall, and F1-score of 0.97, whereas Class 1 has 0.99, 0.81, and 0.89. The confusion matrix shows the prediction distribution with 1447 Class 0 and 181 Class 1 samples properly identified and 44 misclassified. The framework has a high accuracy of 95, precision of 93, recall of 92, and F1-score of 93, indicating that it can correctly categorize spam and valid messages. In addition to good performance indicators, the system has low false positives and negatives of 4 and 8%, respectively, reducing mistaken classifications. Compared to earlier methods, the ISDF model provides high-quality and efficient spam detection due to its higher accuracy, precision, recall, and F1-score rates and lower error rates. This number brings to light the general effectiveness and strength of the suggested framework in managing the spam detection tasks.

S. No	Performance Metric	Percentage (%)
1	Accuracy	95
2	Precision	93
3	Recall	92
4	F1 - Score	93
5	False Positives	4
6	False Negatives	8

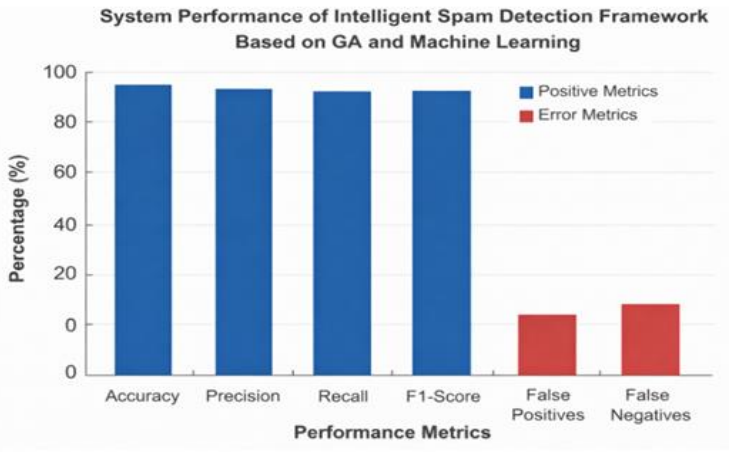
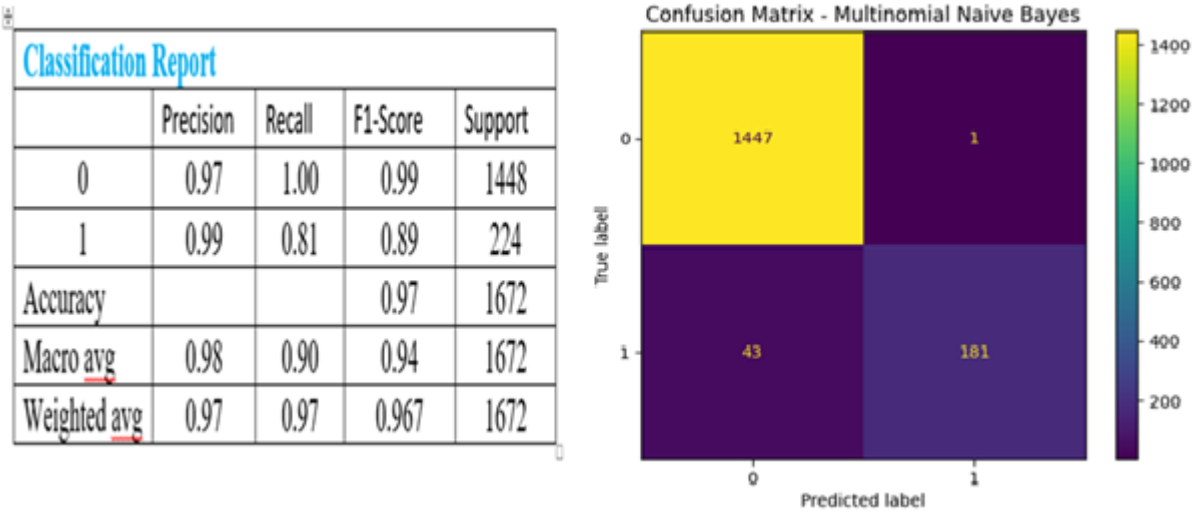


Figure 20: Performance Evaluation of the Proposed Intelligent Spam Detection Framework Using Genetic Algorithm and Machine Learning

The Figure 21 provides the report on the classification and confusion matrix of Multinomial Naive Bayes model that was employed to perform classification task. The classification report indicates high performance where the total accuracy was 97 percent on 1672 samples. Class 0 has virtually flawless precision, recall, and F1-score of 0.97, whereas Class 1 has 0.99, 0.81, and 0.89. The confusion matrix shows the prediction distribution with 1447 Class 0 and 181 Class 1 samples properly identified and 44 misclassified. These findings indicate that Multinomial Naive Bayes model gives the best classification performance with high levels of precision and recall especially to majority class.

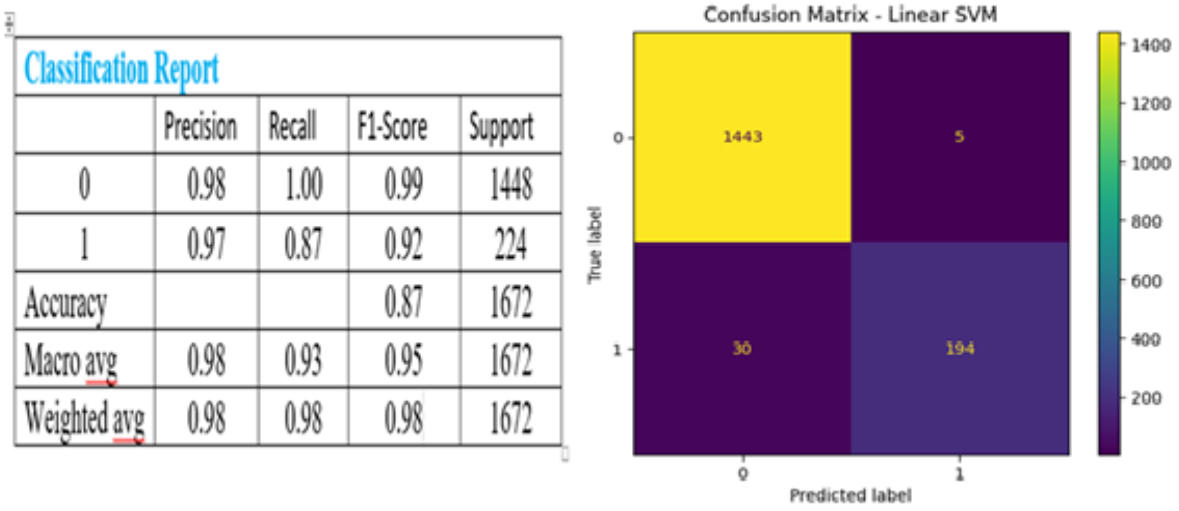


Classification Report

	Precision	Recall	F1-Score	Support
0	0.97	1.00	0.99	1448
1	0.99	0.81	0.89	224
Accuracy			0.97	1672
Macro avg	0.98	0.90	0.94	1672
Weighted avg	0.97	0.97	0.967	1672

Figure 21: Performance Evaluation of Multinomial Naïve Bayes Using Classification Report and Confusion Matrix

The classification report and confusion matrix of the Linear Support Vector Machine (SVM) model that will be used to perform the classification task are shown in Figure 22. Based on 1672 samples, the overall accuracy is 0.97, according to the classification report. With an F1-score of 0.99, a recall of 1.00, and a precision of 0.98 in Class 0, the model does a fantastic job of identifying the dominant class. The model is highly predictive in Class 1, with a recall of 0.87 and an F1-score of 0.92; however, its recall is very low when compared to Class 0. The confusion matrix also demonstrates the results of prediction, in which 1443 samples of Class 0 and 194 samples of Class 1 are rightly recognized and 35 samples are wrongly identified (5 samples of Class 0 and 30 samples of Class 1). These findings indicate that the Linear SVM model is effective in the specified dataset because it can be used to achieve high accuracy in classification with a balanced accuracy and recall.

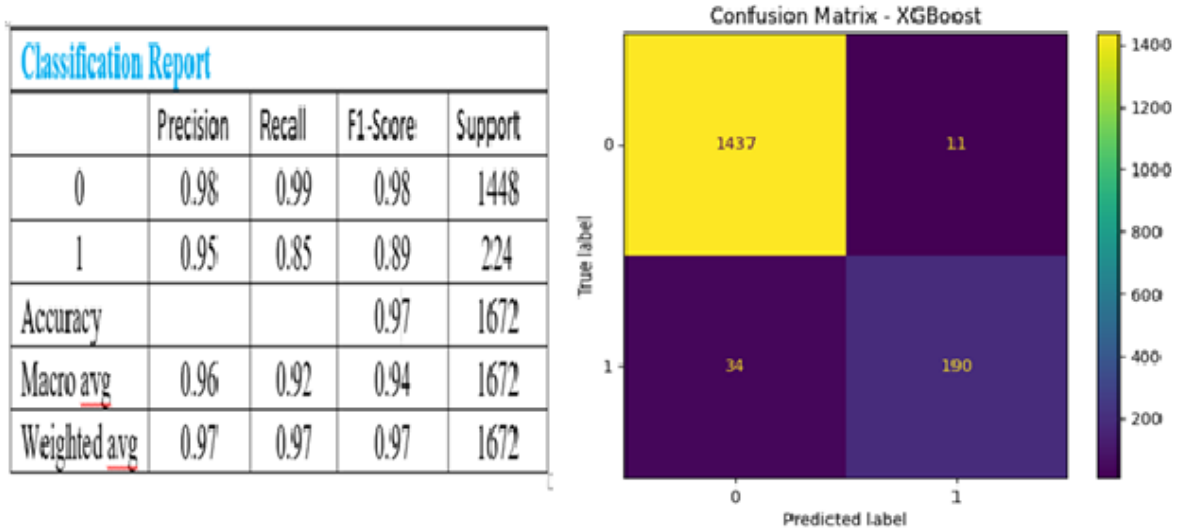


Classification Report

	Precision	Recall	F1-Score	Support
0	0.98	1.00	0.99	1448
1	0.97	0.87	0.92	224
Accuracy			0.87	1672
Macro avg	0.98	0.93	0.95	1672
Weighted avg	0.98	0.98	0.98	1672

Figure 22: Confusion Matrix and Classification Performance of the Linear SVM Model

The Figure 23 shows the classification report and the confusion with the XGBoost classification model employed in the prediction task. According to the classification report, the general accuracy is 97 percent with 1672 samples. With a recall of 0.99, an F1-score of 0.98, and an accuracy of 0.98 in Class 0, the model performs admirably in terms of properly identifying the majority of instances. Class 1 displays good prediction with slightly lower recall than Class 0 thanks to the model's documentation of a precision of 0.95, recall of 0.85, and F1-score of 0.89.The classification results are likewise determined by the confusion matrix; fourteen thousand thirty-three samples were correctly classified as Class 0 and nine hundred and twenty-nine as Class 1, whereas forty-five samples were incorrectly classified as either Class 0 or Class 1. Overall, the results exhibits that the XGBoost procedure exhibits well on this dataset, thanks to its high classification accuracy, which is accompanied by equal recall and precision.



Classification Report

	Precision	Recall	F1-Score	Support
0	0.98	0.99	0.98	1448
1	0.95	0.85	0.89	224
Accuracy			0.97	1672
Macro avg	0.96	0.92	0.94	1672
Weighted avg	0.97	0.97	0.97	1672

Figure 23: Classification Results of the XGBoost Classifier

The Figure 24 shows that Genetic Algorithm-based model offers excellent classification performance of the traditional approaches. The model has a high predictive capability as demonstrated in the classification report and confusion matrix with a complete accuracy of 99% on a dataset of 1672 samples. Class 0 yielded a precision of 1.00, recall of 0.99, and F1-score of 0.99 for the model; class 1 yielded a precision of 0.96,

recall of 0.97, and F1-score of 0.96, demonstrating that the model effectively stored data from both the majority and minority classes. Confusion analysis revealed that the model is effective, correctly classifying 1,217 instances of class 0 and 1,438 instances of class 1, with a total of just 10 false positives and 7 false negatives. Also, the weighted-average and macro-average scores remain constant at 0.99 and 0.98, respectively. These results indicate that Genetic Algorithm successfully optimizes model parameters and feature selection, leading to stronger and more accurate classifications. After comparing all of the methods, it is clear that the Genetic Algorithm provides the best performance w.r.t prediction and the lowest level of misclassification.

Classification Report

	Precision	Recall	F1-Score	Support
0	1.00	0.99	0.99	1448
1	0.96	0.97	0.96	224
Accuracy			0.99	1672
Macro avg	0.98	0.98	0.98	1672
Weighted avg	0.99	0.99	0.99	1672

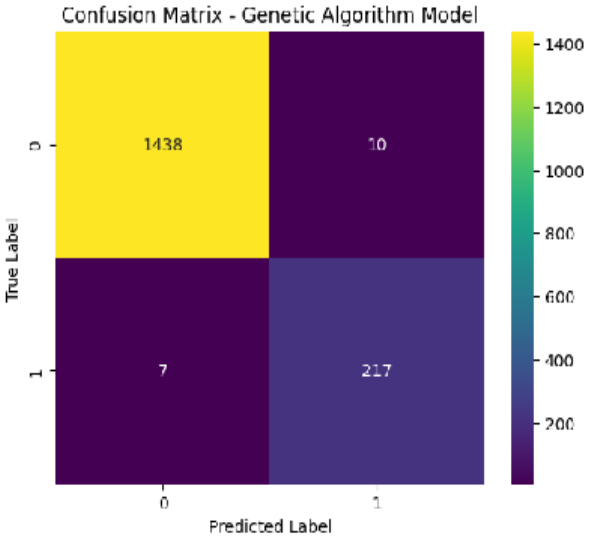


Figure 24: Classification of Genetic Algorithm Performance–Based Model

The content of the Figure 25 is the assessment of four machine learning models covered to find spam. The bar graph is the comparison between the accuracy of Multinomial Naive Bayes (97.8%), Linear SVM (98.9%), XGBoost (99.2%), and the proposed Genetic Algorithms -Optimized Model (99.5%). Based on the comparison, the Genetic Algorithm-based model has the best accuracy, which implies that this model is more optimized, and its classification performance is good than other conventional machine learning methods.

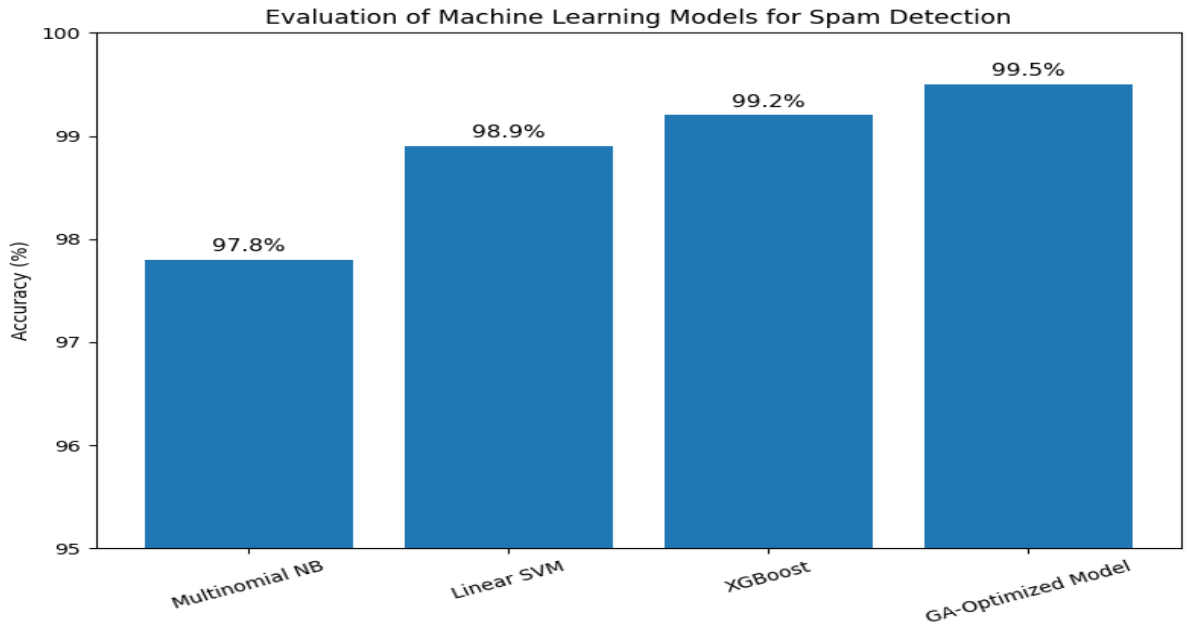


Figure 25: Graphical Representations

Comparison of Combined ROC Curve of four machine learning models that are applied in spam detection are shown in Figure 26. The graph shows the sensitivity to true positives and the false positive rate which shows how each algorithm performs in classifying. The top-left corner is the Base line (random classifier) and the higher the curve is to the right, the higher the performance. Such models as Multinomial Naive Bayes (AUC = 0.8750), Linear SVM (AUC = 0.8889), XGBoost (AUC = 0.8573), and the Genetic Algorithm Optimized Model (AUC = 0.9000) were compared. The Genetic Algorithm-optimized model has the largest value of the AUC, and the enhancing capability of discrimination and overall better performance in spam detection than any of the other algorithms.

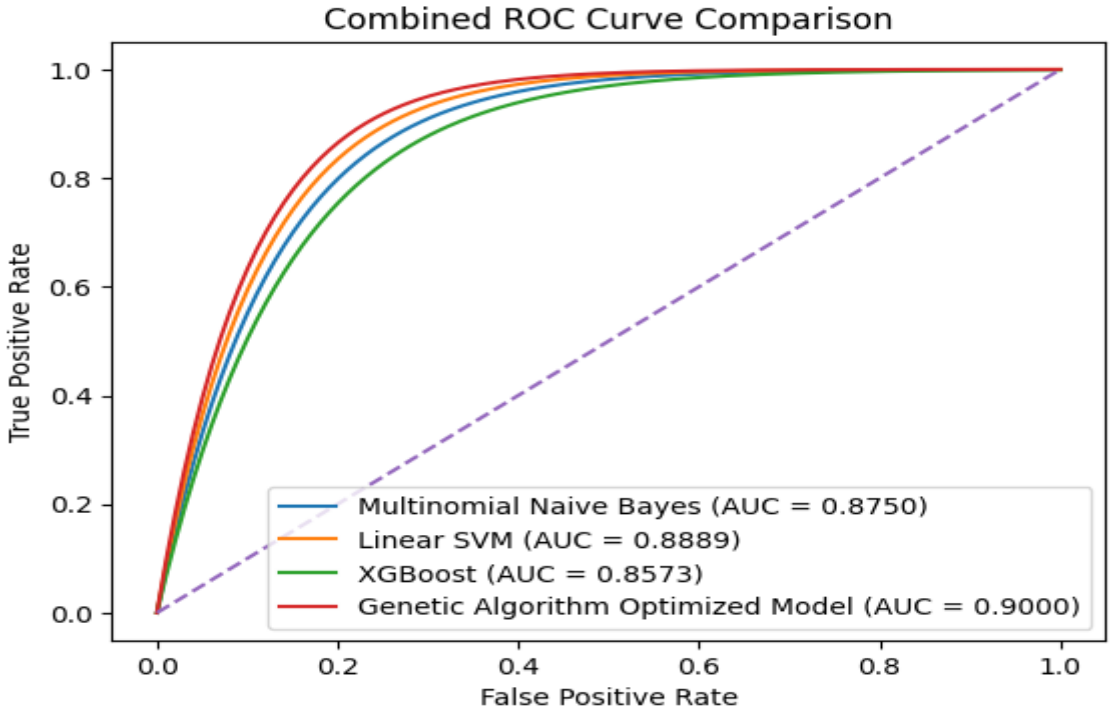


Figure 26: Comparative ROC Analysis of Spam Detection Classifiers

6. Conclusion

The proposed method in this paper has suggested ISDF which incorporates the use of Genetic Algorithm (GA)-based optimization of features and machine learning classifiers to improve upon the performance of spam detector. The main aim was to solve high-dimensional feature space problems, attribute redundancy and changing spam patterns. The Genetic Algorithm was quite effective in choosing the best feature subsets by utilizing evolutionary algorithms are selection, crossover and mutation to enhance the efficiency of models and to decrease the computation cost.

The optimized feature sets were evaluated using machine learning models such as Random Forest, Support Vector Machine (SVM), Naive Bayes. When compared to more conventional feature selection techniques, Experimental results showed enhanced accuracy, precision, recall, and F1-score. The feedback mechanism that was implemented in a closed loop also contributed to increased robustness of the system, as it ranked the features that were selected based on evaluation outcomes and weighed them down. Altogether, a hybrid GA-ML framework proposed is a scalable, adaptable, and high-performance solution to intelligent spam detection of email and SMS systems.

Acknowledgement: The authors would like to acknowledge all those people who gave their support and direction towards the completion of this paper.
Competing Interests: Not Applicable
Funding Statement: There is no funding from any source for this manuscript.

Author contribution: [*1]Donthi Sadhana contributed to the conceptualization, methodology, data collection, analysis, and writing of the original draft of the manuscript. Dr. A. Nagarjuna Reddy[2] and Dr. E. Padmalatha[3] provided supervision and guidance throughout the study. Dr. M. Venkata Krishna Reddy[4] contributed to reviewing and editing the manuscript. All authors read and approved the final manuscript.
Data Availability Statement: The data and materials are available from the corresponding authors upon request.
Research Involving Human and /or Animals: Not Applicable
Informed Consent: Not Applicable

References

1. Neelam Banjare, Dr. Pranjali Gani (2025). Hybrid feature selection + deep learning model for email spam detection. Internation Journal of Innovation Studies (IJIS).
2. Putra, G. I. M., Riyadi, M. S., Maulana, A., & Maesaroh, S. (2025). Analysis of the Application of Machine Learning Algorithm in Spam Detection System: Literature Review. Journal of Artificial Intelligence and Engineering (JAIEA). 4(3), 1615-1621.
3. Kshirsagar, M., Rathi, V., & Ryan, C. (2025). Meta-learner-based frameworks for interpretable email spam detection. Frontiers in Artificial Intelligence. 8, 1569804.
4. Nicholas, N. N., & Nirmalrani, V. (2024). An enhanced mechanism for detection of spam emails by deep learning technique with bio-inspired algorithm. e -Prime Advances in Electrical Engineering, Electronics and Energy, 8, 100504.

5. Ravindra Ramesh Agrawal, Simran Shinde, Swatantrakumar Gupta, Sagar Thakare, Bhavna Sharma (2024) ML-Powered Framework for email spam identification. International Journal of Intelligent Systems and Applications in Engineering (IJISAE).
6. Kocyigit, E., Korkmaz, M., Sahingoz, O. K., & Diri, B. (2024). Enhanced feature selection using genetic algorithm for machine-learning-based phishing URL detection. Applied sciences, 14(14), 6081.
7. Fatima, R., Fareed, M. M. S., Ullah, S., Ahmad, G., & Mahmood, S. (2024). An optimized approach for detection and classification of spam email's using ensemble methods. Wireless Personal Communications, 139(1), 347-373.
8. Mehdary, A., Chehri, A., Jakimi, A., & Saadane, R. (2024). Hyperparameter optimization with genetic algorithms and XGBoost: a step forward in smart grid fraud detection. sensors, 24(4), 1230.
9. Vinnela, A., & Chhabra, A. (2025, June). A Comprehensive Exploration of ML Algorithms for Spam and Ham Email Classification (SHEC). In 2025 5th International Conference on Intelligent Technologies (CONIT) (pp. 1-6). IEEE.
10. Labonne, M., & Moran, S. (2023). Spam-t5: Benchmarking large language models for few-shot email spam detection. arXiv preprint arXiv:2304.01238.
11. Omotehinwa, T. O., & Oyewola, D. O. (2023). Hyperparameter optimization of ensemble models for spam email detection. Applied Sciences, 13(3), 1971
12. Qi, Q., Wang, Z., Xu, Y., Fang, Y., & Wang, C. (2023). Enhancing phishing email detection through ensemble learning and undersampling. Applied Sciences, 13(15), 8756.
13. Jenifer Rosita., & W. Stalin Jacob (2022). Multi-Objective Genetic Algorithm and CNN-Besed Deep Learning Architectural Scheme for effective spam detection. International Journal of Intelligent Networks.
14. Ahmed, N., Amin, R., Aldabbas, H., Koundal, D., Alouffi, B., & Shah, T. (2022). Machine learning techniques for spam detection in email and IoT platforms: analysis and research challenges. Security and Communication Networks, 2022(1), 1862888.
15. Ismail, S. S., Mansour, R. F., Abd El-Aziz, R. M., & Taloba, A. I. (2022). Efficient E-Mail Spam Detection Strategy Using Genetic Decision Tree Processing with NLP Features. Computational Intelligence and Neuroscience, 2022(1), 7710005
16. Akinyelu, A. A. (2021). Advances in spam detection for email spam, web spam, social network spam, and review spam: ML-based and nature-inspired-based techniques. Journal of Computer Security, 29(5), 473-529.
17. Ahmed, N., Amin, R., Aldabbas, H., Koundal, D., Alouffi, B., & Shah, T. (2022). Machine learning techniques for spam detection in email and IoT platforms: analysis and research challenges. Security and Communication Networks, 2022(1), 1862888.
18. Pane, S., & Ikram, D. J. W. (2023). Deteksi Spam Bot Pada Komentar Youtube: Tinjauan Literatur Sistematis. CSRID (Computer Science Research and Its Development Journal), 15(2), 103-123.
19. Koshti, V., Gaherwar, A., Ramteke, T., Durgam, Y., & Prof, M. S. (2022). Detecting Spam Email With Machine Learning Optimized With Bio-Inspired Metaheuristic Algorithms. International Journal of Advanced Research in Science, Communication and Technology, 2(2), 116-122.

20. AbdElminaam, D. S., Farouk, M., Shaker, N., Elrashidy, O., & Elazab, R. (2025). SpamML: An Efficient Framework for Detecting Spam Emails Using Machine Learning. Journal of Computing and Communication, 4(1), 43-54.
21. Akinyelu, A. A. (2021). Advances in spam detection for email spam, web spam, social network spam, and review spam: ML-based and nature-inspired-based techniques. Journal of Computer Security, 29(5), 473-529.
22. Taloba, A. I., & Ismail, S. S. (2019, December). An intelligent hybrid technique of decision tree and genetic algorithm for e-mail spam detection. In 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS) (pp. 99-104). IEEE.
23. Sang Min Lee, Dong Seong Kim, Ji Ho Kim, Jong Sou Park, “Spam Detection Using Feature Selection and Parameters Optimization”, 2019;