International Journal of Women in Technical Education and Employment
ISSN: 2811-1567. Volume 6 – Issue 2. December 2025
https://fpiwitedjournal.federalpolyilaro.edu.ng

A Modified K-Nearest Neighbor Classifier for E-Mail Spam Detection

Jumoke Soyemi & Mudasiru Hammed
Department of Computer Science, The Federal Polytechnic, Ilaro
[1]jumoke.soyemi@federapolyilaro.edu.ng; mudasiru.hammed@federalpolyilaro.edu.ng

ARTICLE HISTORY
Received: Nov 25, 2025
Revised: Dec 15, 2025
Accepted: Dec 02, 2025

Abstract
Electronic mail (e-mail) is one of the most effective and common communication and information exchange methods used to promote products and services. Irrespective of the many benefits e-mail spam offers, it still poses a significant threat in the current Internet ecosystem, leading to loss of revenue by organizations, as well as insecurity and privacy threats to individual users. Therefore, several methods have been devised to counter and reduce spam, especially e-mail classification and filtering methods. Nevertheless, a number of the current solutions have proven to be limited in their ability to differentiate between valid e-mails and spam messages. To overcome these limitations, the paper suggests a customized K-Nearest Neighbor (KNN) algorithm to improve the performance of spam detection. The experimental findings show that the refined KNN algorithm is more accurate in identifying and classifying legitimate and spam e-mails than the traditional methods.

Keywords: E-mail spam, K-Nearest Neighbor, Cross Validation, Euclidean Distance

Citation
Soyemi, J & Hammed M. (2026). A Modified K-Nearest Neighbor Classifier for E-Mail Spam Detection. *International Journal of Women in Technical Education and Employment,* 6(2), 115--123

Introduction
Electronic mail (e-mail) has been one of the most effective, cheap, and common ways of communication for personal and business communications (Basavaraj and Prabhakar, 2010). E-mail has become a prevalent form of communication in the modern digital world, serving individuals, organizations, and governments alike, owing to its convenience and speed with which it can transmit information (Basavaraji and Prabhakar, 2010; Zhang, 2024). Nonetheless, e-mail has become extremely popular, thus a good target of unsolicited and unwanted messages referred to as spam. The number of spam e-mails flooding the inbox of users often results in resource wastage, storage issues, and security risks (Sahil et al., 2013).

Ordinarily, spam refers to unsolicited e-mail messages that the recipients do not request and may contain unsolicited advertising or malicious information (Kumar, Rana, and Mehta, 2012). Spam is ubiquitous within the international e-mail infrastructures because it is virtually free to deliver millions of messages at the same time (Kumar, Rana, and Mehta, 2012). Not only does it decrease the productivity of its users, but also endangers privacy and cybersecurity, having been used to commit phishing, malware distribution, and social engineering attacks, and there is no end to the research of efficient detection and filtering methods (Asliyuksek, Tonkal, and Kocaoglu, 2025).

To overcome spam, many machine learning and artificial intelligence methods have been used to categorize and filter e-mails as legitimate and spam. Common algorithms that have been utilized include Neural Networks, Support Vector Machines (SVM), k-Nearest Neighbor (KNN), and Naive Bayes (NB) (Raihen et al., 2024; Sharma, & Kaur, 2016; Guzella and Caminhas, 2009; Rungsawang, Taweesirawate and Manaskasemsak, 2011; Awad and Elseuofi, 2011; Basavaraju, M., & Prabhakar, 2010). Nevertheless, further development of spam strategies has shown that most of these strategies are not absolutely relevant and require enhanced strategies that are capable of keeping

up with the evolving content patterns and advanced evasion strategies.

Recent studies indicate that there has been continuous development in spam detection. As an example, the classification performance of machines based on comprehensive machine learning models has been proven to be very high when it comes to spam versus legitimate e-mails, and Support Vector Machines have high classification rates in comparative studies (Ahmad, 2024; Tursher et al., 2024; Ali & Abdullah, 2025). The combination of semantic feature engineering and word embeddings has also been suggested as a hybrid method to improve the accuracy of classifiers (Mohammed and Ahmed, 2024). Stacking models have demonstrated important gains in detection accuracy, which highlights the importance of stacking a sequence of base classifiers (Al-augby et al., 2025). In addition, new models like CNN-based models, which have been adjusted to multilingual spam classification, have recorded a higher accuracy of over 99% (Sankaine, Ndia & Kaburu, 2025). Also, efficient machine learning frameworks and optimal feature engineering methods are being studied to detect spam email more efficiently, which points to a significant advancement of the area (Akeel et al., 2025).

In spite of these developments, it is necessary to consider better methods of classification that would balance accuracy, computation speed, and flexibility to change spam content. Specifically, the k-Nearest Neighbor (KNN) algorithm can optimally be adapted or improved to demonstrate better performance compared to baseline models because it can be adjusted to achieve the best neighborhood size to classify. This research suggests a better KNN technique, which will be used to detect and classify legitimate and spam e-mails. The KNN model has been modified in order to provide better accuracy, strength as well as flexibility that will be used to provide greater burden of spam detection in practical email systems.

Materials and Methods

Feature Selection and Extraction

Feature selection is a significant aspect that leads to the accurate detection of spam using email. In this paper, a collection of well-chosen content-based and URL-based features was used to determine the difference between legitimate e-mail messages and spam. The features were selected according to their relation to typical features of spam and their performance in previous works.

The features that have been chosen are:

- ✓ *Average number of characters per word in the e-mail message text, considering the standard e-mail line length limit of 78 characters, which is often adhered to in legitimate e-mail formatting*
- ✓ *Existence of shortened URLs that are commonly used in spam email to mask malicious or fraudulent links*
- ✓ *Number of URLs that have IP addresses, since spam messages frequently contain IP-based links rather than domain name links.*
- ✓ *Limit on the number of dots in the domain name of any URL, as too many dots tend to be used with phishing and spam URLs.*
- ✓ *The number of words in the body of the e-mail.*
- ✓ *Number of hyperlinks included in the message.*
- ✓ *E-mail subject/title number of words.*
- ✓ *Redundancy of content, measured as repeated or duplicated words and phrases within the e-mail body*

After receiving an e-mail message, a feature extraction module reads the contents of the e-mail message and calculates the values of the features that have been selected. These features are then extracted and fed into the proposed modified k-Nearest Neighbor (KNN) classifier used in spam detection.

Modified k-Nearest Neighbor (KNN) Classification Algorithm

The suggested spam detection methodology is based on a refined k-Nearest Neighbor (KNN) algorithm, which is aimed at enhancing the accuracy of the classification process by relying on a clearly defined set of features and a set of optimized decision criteria

that are optimized. The algorithm follows two primary steps known as training and filtering (classification).

Stage 1: Training Phase

During the training step, the feature values are obtained using an annotated set of authentic e-mail messages and spam messages. The training feature set is the following:

- ✓ *Mean number of characters per word in the e-mail text.*
- ✓ *Quantity of URLs with IP addresses.*
- ✓ *Limit on the number of dots in the domain of a particular URL.*
- ✓ *Number of words in the e-mail body.*
- ✓ *Volume of links in the message.*
- ✓ *Word count in the subject/title.*
- ✓ *Level of redundancy of e-mail content.*

These values of features are stored and applied to determine the reference patterns of legitimate and spam messages in the KNN classifier.

Stage 2: filtering (Classification) Phase.

- ✓ *During the filtering phase, the messages in incoming e-mail are handled in the following manner:*
- ✓ *The values of features of the getting e-mail are calculated by the same feature extraction process as in the training.*
- ✓ *The feature set is modified to compare the extracted feature vector with the training feature set by the modified KNN algorithm.*
- ✓ *In case the message value of the feature is not within the acceptable range or neighborhood that is represented by the valid training samples, the message is identified as spam.*
- ✓ *Otherwise, it is considered as legitimate e-mail.*

The change allows the classifier to more effectively distinguish between spam and legitimate messages since it highlights similarities and differences among features and difference to known patterns that are legitimate. The values of these features are stored and utilized in order to determine the reference pattern of legitimate and spam mail in the KNN classifier.

Architecture of Email Spam Detection System

The architecture of the email filtering is depicted in Figure 1, while Figure 2 is a flowchart showing the flow of information in the system

The following are the key components of the architecture:

- ✓ E-Mail Input Module - incoming e-mail messages are received here;
- ✓ Feature Extraction Module - examines message content and identifies useful features;
- ✓ Altered KNN Classification Module - assigns e-mail messages to spam or legitimate according to the features extracted;
- ✓ Decision Module - directs the messages to either the spam or the inbox.

This scalable architecture guarantees performance, ease of integration, and compatibility with already in place e-mail systems.

International Journal of Women in Technical Education and Employment
ISSN: 2811-1567. Volume 6 – Issue 2. December 2025
https://fpiwitedjournal.federalpolyilaro.edu.ng

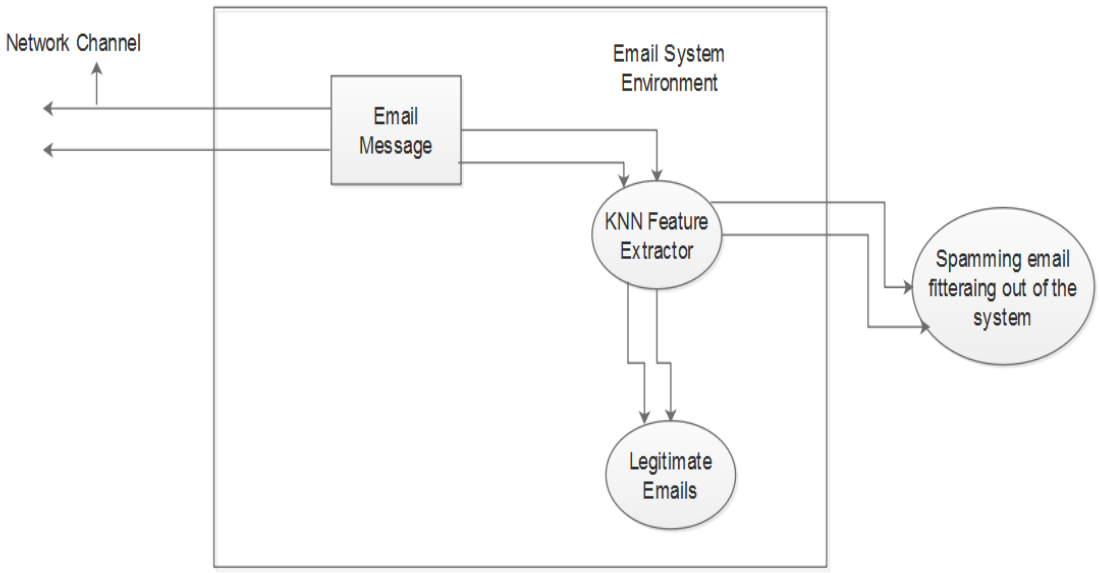

Figure 1: Architecture for E-mail Spam Classification System

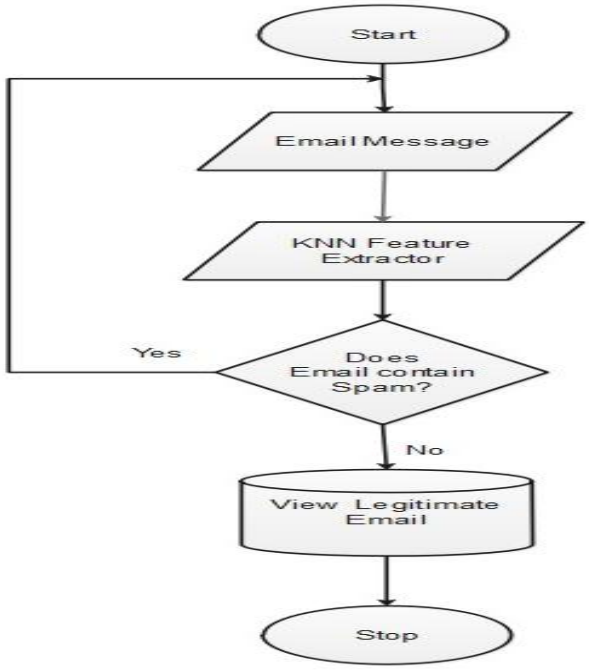

Figure 2: Flowchart of email spam detection system

Model Formulation and Classification Framework

The K-Nearest Neighbor (KNN) algorithm is a supervised machine learning method that categorizes a query example on the basis of the closeness of the query feature vectors to the labeled training samples. Each of the training features is extracted and given a set of training features. The KNN algorithm is used to compute the classification of a query point X by locating the k nearest training samples in the feature space, usually through a distance measure like Euclidean distance.

The algorithm calculates the distance between the query point X and every training sample for each

feature vector. The k nearest neighbors vote majorly, and the decision made is to assign X the class label Y. This can be mathematically derived as:

$$Y = \arg\max_{c \in C} \sum_{i=1}^{k} \delta(c, y_i) \quad (1)$$

Cross Validation

The study uses cross-validation to come up with the best k of the k-Nearest Neighbor (KNN) classifier because the selection of k can be a major concern in terms of accuracy of the classification and stability of the model. The process implemented consists of dividing the e-mail data set into a constant number of subsets or disjoint and randomly selected elements (folds). With a set value of k, the modified KNN model is trained using all folds with a minus one, whereas the remaining fold is utilized in the validation. The classifier then makes its estimates of the class labels of the validation samples, and an error measure suitable for the classification error is determined. This is done repeatedly, whereby every fold is used as the validation set once.

Upon completion of each cycle of validation, the average classification error of each fold is calculated to have a good estimate of the performance of the model using the given value of k. This is repeated with various values of k, and a value that gives the least average validation error is chosen as the best k to be used in the spam detection model. The proposed method will provide better generalization and less overfitting, as well as greater resilience of the customized KNN classifier to differentiate between legitimate and spam e-mails.

Distance Metric

The k-Nearest Neighbor (KNN) classifier is used in this paper to classify a query e-mail instance based on the closeness between its attributes and those of the training samples by calculating the distance between the feature vectors of the two. A proper measure of distance is a necessary condition for effective classification since it directly affects the choice of closest neighbors.

The Euclidean distance measure is used in this study because it is not complicated, and it is also useful in the context of determining similarity in the numerical feature spaces. Given a query point
Where X and Y are training samples and Y(y1,y2,...y n) and x1,x2...x n, Euclidean distance is calculated as follows:

$$d(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (2)$$

Where n denotes the number of features extracted. The modified KNN algorithm uses the calculated distance values to determine the k nearest neighbors, the class labels of which determine the query e-mail as spam or legitimate.

Distance Weight

The k-Nearest Neighbor (KNN) classifier is based on the idea according to which similar data are more likely to be similar in the feature space and have similar classifications. In order to further boost the level of classification, this paper uses a distance-weighted KNN method whereby closer neighbors have a stronger impact on the classification of a query sample than far neighbors.

Distance weighting has the advantage that all the k nearest neighbors are given a weight (w) proportional to the distance between the query point and the neighbor. As a result, nearest neighbors are given more weight in the eventual decision, with far neighbors having less weight. The weighting function to be applied in this research is given as:

$$w_i = \frac{1}{d(X, X_i) + \epsilon} \quad (3)$$

K-nearest neighbor predictions are based on an intuitive approach of the objects that are close in distance and are potentially similar. That is, the closest points among the KNN affect the outcome of the query point by introducing a set of weights (W). The formula is shown in equation 3, where d(X,Xi) is the Euclidean distance between the query point X and the closest ith

nearest neighbor point Xi, and e is a small positive constant that prevents division by zero.

This weighted contribution of the k nearest neighbors is added to the final label of the query e-mail, thus enhancing the strength and correctness of spam classification.

K-Nearest Neighbor Predictions

The k-Nearest Neighbor (KNN) algorithm is a prediction algorithm that relies on the performance of the k nearest training samples in the feature space to predict a query point in the feature space. When applied to regression problems, the predicted value of a query instance is calculated as an average of the results (target values) of the k nearest related cases.

Mathematically, the predicted value y^ for a query

point X is given by:

$$y = \frac{1}{K}\sum_{i=1}^{k} y_i$$

$$\hat{y} = \frac{1}{k}\sum_{i=1}^{k} y_i \quad (4)$$

where $y_i$represents the outcome (e.g., class label or continuous value) of the i^{th}nearest neighbor.

KNN is normally used in classification problems where majority voting among immediate neighbors is used to form a predicted class. In this work, KNN is modified to serve classification functions, although distance weighting is added to it in order to enhance the accuracy of prediction.

Results and Discussion

Result

In this paper, a set of features was used as a means of successful e-mail spam detection. The most important characteristics were the number of characters used per word on an average e-mail message (which is usually limited by the standard 78 characters), the use of shortened URLs, the number of URLs that include IP addresses, and the proper usage of dots in the domain names in the URLs. These were features that helped distinguish legitimate e-mails and spam. Python 3.5.0 was used in the implementation process of the e-mail spam detection system with the modified k-Nearest Neighbor (KNN) algorithm as the main classification procedure. The proposed system was tested using labeled training data on sample e-mail messages of well-known providers, such as Yahoo and Gmail.

The system was able to track spam messages with high accuracy, as shown in Figure 3. The implementation shows the ability of the system to scan the e-mails received, extract the features, and correctly classify messages as spam or legitimate. Figure 3 indicates that the system detects and reports spam messages in real time; hence, the system can be practically applicable. The high precision of the study confirms the efficiency of the feature selection strategy and the adapted KNN algorithm to improve spam detection accuracy in relation to the conventional methods. These findings suggest that the model is effective in separating spam content with high reliability, thereby enhancing the security of e-mails and user experience.

Figure 3: An e-mail spam detection system

Discussion

In this study, it is shown that the suggested modified k-Nearest Neighbor (KNN) algorithm is much more accurate and reliable at e-mail spam detection as opposed to using the conventional KNN-based filters. Although such classical spam-filtering methods as Naive Bayes, Support Vector Machines (SVM), and neural networks have demonstrated competitive results, they may demand sophisticated training schemes, high levels of feature engineering, and a significant number of computer resources (Sahami et al., 1998). Conversely, the classical KNN algorithm is simple and effective yet highly constrained in several aspects, such as sensitivity to noise, equal weighting of all features irrespective of importance, and expensive to run during classification, which are not appropriate in large-scale e-mail systems that are run in real time (Metsis et al., 2006).

To overcome these issues, the KNN variant provided in this paper is the modified one, which uses domain and email-relevant features as well as a two-step training and filtering process. The features chosen (average characters in a word, the count of URLs that have IP addresses, the number of dots in the URL domains, content redundancy, and the frequency of hyperlinks) are highly related to known spam features. The previous studies confirm that spam e-mails usually have abnormal lexical patterns and the overuse of hyperlinks and irregular domain structures, and high content redundancy (Almeida et al., 2011; Guzella and Caminhas, 2009). Using these features that are domain-aware, the revised KNN algorithm will recognize better discrimination of legitimate and spam e-mails.

Additional improvement in classifications is achieved by the addition of distance weighting and the application of Euclidean distance measures. The use of the inverse distance of the neighbor to the query point further has the effect of reducing the impact of the outlier and enhancing that of the more predictive feature vectors, which in turn improves the overall performance of the classification (Cover & Hart, 1967; Zhang, 2007). Moreover, cross-validation is applied to optimize the choice of the parameter (k), which alleviates the risk of overfitting and underfitting that is inherent to the traditional KNN classifiers.

One of the contributions of this work that can be mentioned is the introduction of training-based thresholding in the filtering stage. This rule-based improvement by setting feature values ranges based on the training set supplements the distance-based classification to allow deterministic filtering which is useful in identifying structural anomalies in e-mails. This composite system is consistent with research results on hybrid e-mail filtering design which also ensures its effectiveness (Delany et al., 2005; Cover and Bryl, 2008).

In addition, the system architecture formulated below illustrates a slim pipeline of feature extraction, training and classification, which will be applicable in the practice deployment in real life e-mail systems. The updated KNN takes advantage of lazy learning unlike many machine learning models, which need retraining during updates of the feature space, which is important to recognize, unlike other lazy learners, the updated KNN adapts (without full retraining) to dynamic environments as previously found (Aha et al., 1991).

In general, the suggested changes to the KNN algorithm led to a higher spam rate detection through the integration of statistical pattern recognition, rule-based filtering, and the optimization of distance metrics. The methodology is more resistant to changing spamming strategies and still retains the simplicity and interpretability of KNN as a popular baseline classifier. The findings emphasize the relevance of hybrid and feature-adaptive algorithms in solving modern spam detection challenges, which have become increasingly important with spammers inventing new ways to circumvent more traditional content-based filters.

These findings support the significance of hybrid and feature-adaptive algorithms in terms of contemporary spam-detection challenges, particularly at a time when spam-detection attackers are still finding ways to evade conventional content-based filters.

Conclusion

This research paper concludes that despite its effectiveness and popularity as a medium of communication and exchange of information, the ubiquitous issue of e-mail spam is a major obstacle to the usefulness and user-friendliness of e-mail. The suggested modified k-Nearest Neighbor (KNN) algorithm proves to be a better spam-detecting and classifying algorithm than the traditional spam-filtering algorithms. The results of the implementation show that the KNN method with altered modifications, which is based on the well-chosen features and optimization of classification methods, has better accuracy and strength in spam detection. This paper signifies the possibility of customizable machine learning adjustments to improve e-mail security and the necessity of further investigation of adaptive spam filtering.

References

Aha, D. W., Kibler, D., & Albert, M. K. (1991). Instance-based learning algorithms. *Machine Learning*, 6(1), 37–66.

Ahmad, S. M. (2024). Spam classification using machine learning and deep learning (Doctoral dissertation, Dublin Business School).

Akeel, M., Butt, K. K., Javed, K., Tariq, M., & Yousaf, M. (2025). Email spam detection using machine learning with optimized feature engineering and classification techniques. *Journal of Computing & Biomedical Informatics*, 10(01).

Al-augby, S., Alyasiri, H., Abdulkadhim, F. G., & Oleiwi, Z. C. (2025). A stacked ensemble classifier for email spam detection via an evolutionary algorithm. *Mesopotamian Journal of CyberSecurity*, 5(2), 657-670.

Ali, A. A., & Abdullah, A. A. (2025). Email Spam Detection: A Novel Hybrid Approach Using Machine and Deep Learning Techniques. *International Journal of Intelligent Engineering & Systems*, 18(7).

Almeida, T. A., Hidalgo, J. M. G., & Yamakami, A. (2011). Towards an evaluation of anti-spam filters. *Expert Systems with Applications*, 36(7), 10206–10222.

Asliyuksek, H., Tonkal, O., & Kocaoglu, R. (2025). A comparative evaluation of a multimodal approach for spam email classification using DistilBERT and structural features. *Electronics*, 14(19), 3855

Awad, W. A., & Elseuofi, S. M. (2011). Machine learning methods for spam e-mail classification. *International Journal of Computer Science & Information Technology (IJCSIT)*, 3(1), 173–184.

Basavaraju, M., & Prabhakar, R. (2010). A novel method of spam mail detection using text based clustering approach. *International Journal of Computer Applications*, 5(4), 15–25.

Blanzieri, E., & Bryl, A. (2008). A survey of learning-based techniques of email spam filtering. *Artificial Intelligence Review*, 29(1), 63–92.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21–27.

Delany, S. J., Cunningham, P., Doyle, D., & Zamolotskikh, A. (2005). Generating estimated probabilities from case-based reasoning classifiers for ranking. In *Proceedings of the 6th International Conference on Case-Based Reasoning* (pp. 61–75). Springer.

Guzella, T. S., & Caminhas, W. M. (2009). A review of machine learning approaches to spam filtering. *Expert Systems with Applications*, 36(7), 10206–10222.

Kumar, N. S., Rana, D. P., & Mehta, R. G. (2012). Detecting e-mail spam using spam word associations. *International Journal of Emerging Technology and Advanced Engineering*, 2(4), 222–226.

Metsis, V., Androutsopoulos, I., & Paliouras, G. (2006). Spam filtering with naive Bayes—Which naive Bayes? In *Proceedings of the Third Conference on Email and Anti-Spam (CEAS)* (pp. 27–28).

Mohammed, C. N., & Ahmed, A. M. (2024). A semantic-based model with a hybrid feature engineering process for accurate spam detection. *Journal of Electrical Systems and Information Technology*, 11(1), 26.

Raihen, M. N., Rana, S., Akter, S., & Kadir, M. A. (2024). Efficient email spam detection using machine learning techniques: A comparative analysis of classification models. *International Journal of Intelligent Computing and Information Sciences*, 24(4), 1–15.

Rungsawang, A., Taweesiriwate, A., & Manaskasemsak, B. (2011). Spam host detection using ant colony optimization. In *IT Convergence and Services: ITCS & IRoA 2011* (pp. 13–21). Springer.

Sahami, M., Dumais, S., Heckerman, D., & Horvitz, E. (1998). A Bayesian approach to filtering junk e-mail. In *Learning for Text Categorization: Papers from the AAAI Workshop* (pp. 55–62). AAAI Press.

Sahili, P., Dishant, G., Mehak, A., Ishita, K., & Nishtha, J. (2013). Comparison and analysis of spam detection algorithm. *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, 2(4), 1–7.

Sharma, R., & Kaur, G. (2016). E-mail spam detection using SVM and RBF. *International Journal of Modern Education and Computer Science*, 8(4), 57.

Sankaine, L., Ndia, J. G., & Kaburu, D. (2025). An English-Swahili email spam detection model for improved accuracy using convolutional neural networks. *Mesopotamian Journal of CyberSecurity*, 5(2), 590–605.

Tusher, E. H., Ismail, M. A., Rahman, M. A., Alenezi, A. H., & Uddin, M. (2024). Email spam: A comprehensive review of optimize detection methods, challenges, and open research problems. *IEEE Access*.

Zhang, H. (2007). Exploring conditions for the optimality of naive Bayes. *International Journal of Pattern Recognition and Artificial Intelligence*, 21(1), 43–60.