International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Framework for Spam Toxicity Level Detection using Machine Learning Models

K Aditya Shastry[1], Akshatha G C[2], Naresh E[3,*], Karthik D U[4], Mohan Kumar T G[5]

Submitted: 26/01/2024 **Revised**: 04/03/2024 **Accepted**: 12/03/2024

Abstract: Electronic mails are a technique of conveying a message to an individual or group through the internet. When emails first became popular, they could only be used for institutional and scientific research. However, as technology has advanced to the point where it can reach every individual on the planet, everyone now can have an email address in his or her name. As technology advances and more people begin to use email, billions of emails are sent in the name of promotions, ads, and spam. It is challenging to manage emails, and users nowadays are having difficulty locating their essential email. In this work, a framework is proposed for classifying the emails into multiple classes (bills, promotions, personal, spam, and OTP) so that the relevant emails can be identified by the users easily. The framework consists of supervised machine learning (ML) models such as "random forest (RF)", "support vector machine (SVM)", "naïve bayes (NB)", "k-Nearest Neighbour (k-NN)", and "decision tree (DT)" for classifying mails. A web application was also developed for the same purpose. Results demonstrated that the RF and k-NN classifiers outperformed the other classifiers based on accuracy.

Keywords: *Machine Learning, Spam, Toxicity, Classification.*

1. Introduction

Emails have become an indispensable part of modern life. Nowadays, everything from product advertisements to university verification, delivering bills and pursuing positions to getting a traditional conversation is probable via mail. Numerous m ails are being generated as time goes on. Today, many people have a habit of constantly surfing their emails [1].

Email classification is a technique used to separate emails and organise them into groups so that they can be easily identified. When objects are classified as opposed to jumbled and dumped, identification is always quick and simple. Similar to how books are organised in libraries based on the author, genre, or favourites, an email classification system can be used. Email classification is typically used to eliminate spam emails that are useless to users and could endanger their personal data. [2].

Email was primarily utilised for research purposes as sending mail was expensive. However, nowadays every third person on earth has an email address. According to estimates, there will be approximately 4.3 billion email subscribers by the end of 2025, and 3.13 million emails will be sent worldwide every second. It can be difficult to manage such a large quantity of emails. Therefore, classifying them might aid the users to identify relevant mails [3].

[1,4,5]*Nitte Meenakshi Institute of Technology, Bengaluru – 560064, India*
[2] *M S Ramaiah Institute of Technology, Bengaluru-560054, India*
[3] *Department of Information Technology, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India.,*
** Corresponding Author Email: naresh.e@manipal.edu*

Even though several well-known email service providers have experimented with their own ways for classifying emails and have their own systems in place, the classification of emails is still a common practise. For instance, Gmail, the most popular email service provider, groups emails based on the sender, which may also be used as a classification method. Gmail has divided emails into personal, promotional, and social emails. Outlook Mail, a product of Microsoft, is the next most popular email service provider. The other email service providers are working to improve email classification. Although several researchers have discovered a method for identifying spam emails, there is still work to be done for classifying the mails into different classes. This has made the domain of email classification an active research area [4].

Keeping these points in mind, in this work, we have developed a web-based framework using ML for the effective classification of mails into 5 classes of mails related to bills, promotions, OTPs, spam and personal. For this purpose, the ML models such as RF, k-NN, SVM, NB, and DT models were used for classification. Later, these models were embedded into the web-based application.

2. Related work

Email marketing strategy research and application is not a recent subject. Researching email is an area of ongoing learning, and every year, new approaches and improved algorithms for email classification and boosting email reach to users emerge. Numerous research papers on email classification have been published over the years, and academics have each described a special way to put email classification approaches into practice [3, 5]. Most

publications discuss the use of DL and ML approaches to classify emails. To classify emails, ML and DL algorithms are used to a variety of datasets, each of which results in a unique approach and level of accuracy [6,7].

Numerous classification techniques, including Logistic regression, SVM, NB, DT and RF, are used in most works on email classification to classify spam emails [8,9]. To increase classification performance, some studies employ hyper parameter tunings and improved versions of the aforementioned techniques. In addition, neural networks and hidden Markov models are being used for email classification [1,4]. Only a small number of publications have attempted to address multiclass email classification as a necessary component of a better email service [10,11]. In our work, the multiclass classification of emails has been implemented using ML techniques.

Semi-supervised algorithms were also employed in classification [11]. Depending on the requirements of the researcher, diverse datasets were employed. Some of them made use of the from, to, date of the email, subject, and body of the email, among other email components. They used the email's components in accordance with the user's needs [2].

Researchers have employed a variety of pre-processing approaches, mostly NLTK tool kit with natural language processing (NLP) techniques, to improve accuracy [12]. These techniques include “bag of words”, “word tokenization”, “word vectorization”, “stop word” removal, keyword identification, “lexicology”, and “TF-IDF”, to mention a few. The authors have demonstrated how well several techniques performed at classifying emails using the training data set and have recommended the top method to employ.

Regarding the research they have conducted, researchers have offered suggestions for their own future research and study areas. The flaws in the emails are one of the main worries [10]. When emails are written, it might be challenging to categorize them since occasionally the mail is, for instance, categorized as class A while considering that a similar piece of mail will be classified as class B. [13,14]. Analyzing the context of the letter, which can be challenging at times, and classifying it appropriately is one of the main issues. Some have also stated that the issue is with the dataset because it is challenging to obtain a suitable email dataset for the model's training. Even after extensive modifications to the algorithm and pre-processing methods, the aforementioned factors continue to adversely impact the classification accuracy. Additionally, researchers have briefly discussed the multiclass email categorization and its advantages in their works [15].

3. Framework and system design

3.1. Machine learning

In this work, ML approach was chosen for classifying emails into multiple classes. is the utilization and development of computer systems that can learn and adjust without being given explicit instructions, by analyzing data patterns and making predictions via statistics and algorithms. ML can be classified into three categories: “reinforcement learning”, “unsupervised learning”, and “supervised learning”.

For our approach we have identified supervised algorithms as best choice for classification purpose based upon the data set identified for training and testing purpose. Some of the supervised ML algorithms identified for the solution are RF, DT, SVM, NB, and k-NN.

- K-Nearest Neighbor (k-NN): An algorithm for supervised classification is K-nearest neighbors. To forecast how a test sample point will be classified, this algorithm uses certain data points and a data vector that have been divided into several classes. A new point is classified using the k-NN using a similarity metric, which can be Euclidian proximity. Equation (1) illustrates the Euclidean distance and names its neighbors.

$$dist(x,y)(a,b) = \sqrt{(x-a)^2 + (y-b)^2} \quad (1)$$

Where x, y, a, b represent the records and $dist$ signifies the distance between these records [16].

- Random Forest (RF): Many DTs are constructed throughout the training stage of the RFs, utilized for categorization, regression, and other tasks. The RF output for classification tasks is the class chosen by most trees. Based on the training data, several DTs are built, and the one receiving the most votes is chosen as the optimal splitting strategy for the classification task. It can be thought of as a generalized improvement to the DT algorithm [17].

- Support vector Machine (SVM): Prominent supervised learning algorithms include the SVM, which is used in ML approaches to solve categorization issues. Decision points served as the primary inspiration for SVMs. The SVM algorithm's primary purpose is to draw a line or decision boundary. The SVM algorithm produces a hyperplane that can categorize fresh samples. In two dimensions, a "hyperplane" is a line that separates a plane into two halves, with each category represented on one side [18].

- Naïve Bayes (NB): The algorithm is employed in supervised learning. The Bayesian classifier uses interdependent occurrences and calculates the likelihood that an occurrence that has already happened could predict an occurrence that will

happen in the future. Based on the Bayes theorem, which presumes that characteristics are independent of one another, NB was developed. The NB algorithm can be applied as a method for categorizing emails into multiple classes. When performing Bayesian classification, which is a method for multiple classification, the phrases that identify the email messages class could be used as occurrences. The Probabilistic computations are shown in equations (2) and (3).

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \qquad (2)$$

$$P(A|B) = \sum P(B|A)P(A) \qquad (3)$$

Where, P(A | B) represents the conditional probability of event A occurring provided event B has already occurred [19].

- Decision tree: A decision-tracking tool (DT) uses a tree-like architecture to depict options and their potential outcomes, including efficiency, resource costs, and probability occurrence outcomes. It's one technique to show a conditional control algorithm. DT algorithm is generally more preferred in classification techniques due to its approach of classifying the data based on identifying the correlation between the dimensions in the data. This approach of splitting the classes can give more accurate value and with better efficiency for classification [20].

Also, it's very important to look after some of the key concepts of data pre-processing techniques that have been used which helps in improving the performance of classification.

3.2. Data pre-processing

Data pre-processing in general is a method to clean the data from noise, outliers, missing values, and other unwanted information. Manifestation of these inputs in the dataset disrupts the algorithm performance but altering the calculations done by the algorithms across the data. This results in obtaining the un-desired output that is deviated from the actual output. There exist different methods for pre-processing the information [21]. Some of the well-known pre-processing techniques employed in our work are described as follows:

- "Stop words removal": Textual format of the data usually have some words which have no meaning when they are used alone. Those words are repeatedly used in between the key words for sentence formation and to deliver the precise meaning. Whilst training the model usage of such words can reduce the performance of the algorithms. Hence removing these words is prescribed whilst training the model. Hence this is one of the approaches of information pre-processing [22].
- "TF-IDF": TF-IDF stands for "term frequency-inverse document frequency", as well as it is a measure, used in the areas of information retrieval and ML, that can quantify the importance or relevance of string representations in a document amongst a collection of documents (corpus) [23].

3.3. Design of the proposed work

Fig.1 illustrates the design of the proposed work. This method is used to train the model over the data set. Every component in the design has significance in constructing the model for classification purpose. Every part of the design is explained as follows:

- "Input data": It signifies the data which is collected to provide solution to the problem related to the dataset. It represents the email information used for classification purpose. The data has attributes such as email text, and corresponding tag used for the purpose of training.
- Data pre-processing: Here the input data is prepared for training the algorithm. Irrelevant "outliers", "noise", "null values", "stop words" are removed and the words with importance used for classification purpose is utilized for training the model. The stopword removal technique, word vectorization is carried out along with TF-IDF is used to form matrix of key words. It is used for pairing the similar words from the test data.
- Processed information: It is the output information obtained after applying the mentioned pre-processing techniques that results in development of sparse matrix with the value assigned to keywords used to find out similar keywords which can be grouped for classifying the emails into desired class.
- ML techniques: The processed data is taken as input and trained diverse supervised machine learning algorithms. Initially the data is split into train and test data which is mandatory to train the model and test to know the ability of the model to classify the emails. Algorithms mentioned previously are trained one after the other by using the input data. Performance based on the performance metrics are recorded for each algorithm and later all the algorithm performances are evaluated by comparing amongst them. The one with the better accuracy and better performance in classifying the emails is

determined and further analysis and investigation is carried out.

- Check accuracy: This module of the design deals with the evaluation of the algorithms individually over the performance evaluation metrics and understand the performance of each algorithm based on different parameters which is later utilized to compare as parameters to identify the best usable approach for classification purpose with respect to the data used for training the model.

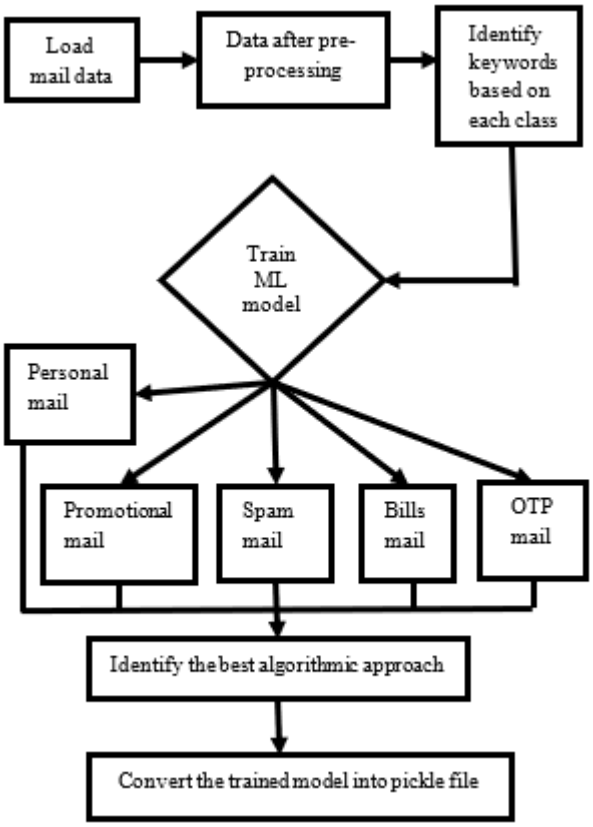

Fig. 1. Design of the proposed email classification system.

- Performance Evaluation: There are numerous metrics used to assess the performance. Certain significant metrics used in this work are mentioned below [24]:

  - The number of authentic records that were accurately identified as legitimate is represented by the TP (True Positive) indicator.
  - The number of records that are labelled as legitimate but are not authentic is known as FN (False Negative).
  - TN (True Negative) is the proportion of records deemed to be false that are actually false.
  - FP (False Positive): is the number of genuine instances categorized as not authentic.
  - Accuracy - The proportion of records successfully categorized by the classifier is known as accuracy. It is computed as shown in equation (4).

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FN + FP)} \quad (4)$$

  - True Positive Rate (TPR) - It quantifies the proportion of authentically altered imagery (positives), that are accurately identified. It is computed as shown in equation (5).

$$TPR = \frac{TP}{(TP + FN)} \quad (5)$$

  - True Negative Rate (TNR) - Also referred to as Specificity, True Negative Rate (TNR) calculates the proportion of actual images (negatives) that are accurately categorized as such. Equation (6) shows the TNR.

$$TNR = \frac{TN}{(TN + FP)} \quad (6)$$

  - False Positive Rate (FPR): This statistic shows the proportion of genuine negative images that are incorrectly labelled. Equation (7) shows the FPR computation.

$$FPR = FP/((FP + TN)) = 1 - TNR \quad (7)$$

  - Error rate: This statistic shows the proportion of images incorrectly categorized. It is calculated in accordance with equation (8).

$$Error\ rate = ((FN + FP))/((TP + TN + FN + FP)) \quad (8)$$

- "Multiclass confusion matrix": The confusion matrix for 2*2 matrix is simple and evaluating the matrix is easy. But for multiple class classification the matrix size depends upon the number of classifications performed. If n classes are classified, then the confusion matrix built will be of the order n*n [24].

The model or the algorithm that has proven better among the rest of the algorithms with respect to the performance evaluation is identified and is trained separately and compressed into a transferable file called the pickle file which is used to test the real time data over the user interface. Here we have used flask library of python for user interface creation which helps in analyzing the test data and the accuracy of classification.

In our work, best evaluated algorithm is used for classifying the emails. It is tested on by delivering the email in real time to understand the ability of the algorithm to classify the emails into necessary class. The algorithm evaluation is compressed into pickle file which is later used to build the user interface used to read the received email and classify the email accordingly.

The approach proposed is to classify the emails into five groups as personal emails, promotional emails, OTP (one time password) emails, bills and payments and spam emails. This methodology can be useful and more efficient in today's situation. As several emails are delivered every day maintaining the emails that are very important for the user is a tedious task. Hence classifying the emails in the designed method could assist in optimizing the emails they're by providing advantage for the users to find his/her emails easily.

4. Proposed work

The implementation is completed in accordance with the design plan. The information is first loaded, and then different pre-processing methods are applied to it in accordance with the Figure 4. Every one of the information sets in the data are continually subjected to the word vectorization and TF-IDF sparse matrix construction processes. Fig.2 illustrates how data pre-processing comes to an end once the dataset has been thoroughly examined and the necessary keywords have been obtained and vectorized.

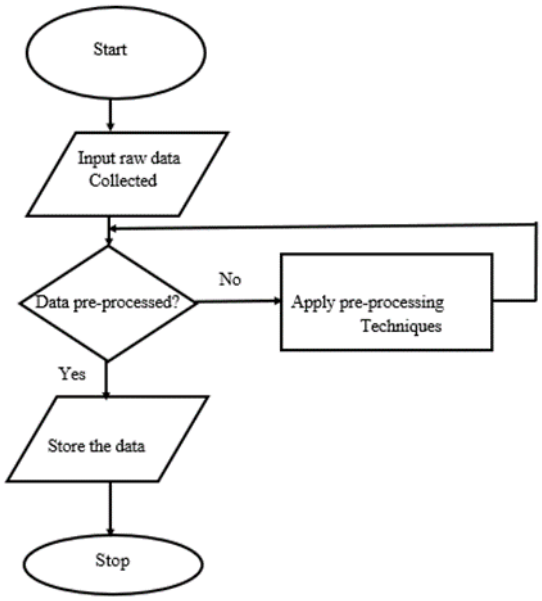

Fig. 2. Data pre-processing

After all pre-processing is finished, the processed data is divided into train and test data. Every algorithm is trained individually using the train data, and the algorithms trained using the test data are tested concurrently. Following a preliminary test, the algorithms are assessed using the relevant performance evaluation tools, as shown in Fig.3.

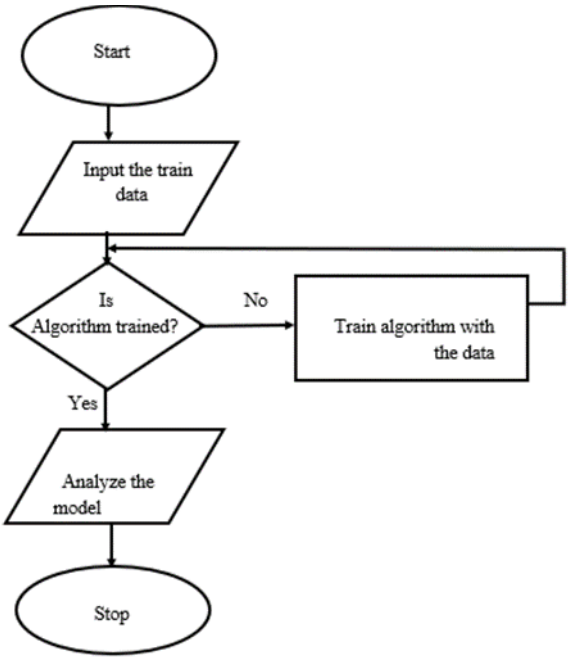

Fig. 3. Training ML model

5. Experimental setup and results

Evaluation of the individual's effectiveness for email classification through testing is crucial. Testing is the method of evaluating a solution to assess its effectiveness in relation to the problem by using numerous types of inputs with diverse elements. In this case, the assessment is done in two stages. The test data that was produced previously during the train-test split.

The second type of testing is done with real emails that have been read, analyzed, and categorized. The classified email is put in the appropriate class, and the result is shown in the Flask-created web interface. Emails sent using Gmail in real-time were used for the second type of testing. After completing the Gmail-provided multiple authentication processes, the messages are retrieved. The mail is further examined, given the appropriate classification, and forwarded to the group to which it belongs. For proper identification, the final product is shown on the internet with color coding. Threats to user information are denoted by the color red, while mails that are only occasionally or never utilized are denoted by the color orange. Green codes indicate emails of the top importance.

The findings acquired are evaluated utilizing the classification results of all the methods after the techniques have been trained and the trained models have been tested. Researchers have determined that the RF method is the superior strategy for email categorization by contrasting the methods with their performance assessment. According to the training data used to train the systems, it provided superior prediction performance. This process is illustrated in Fig.4.

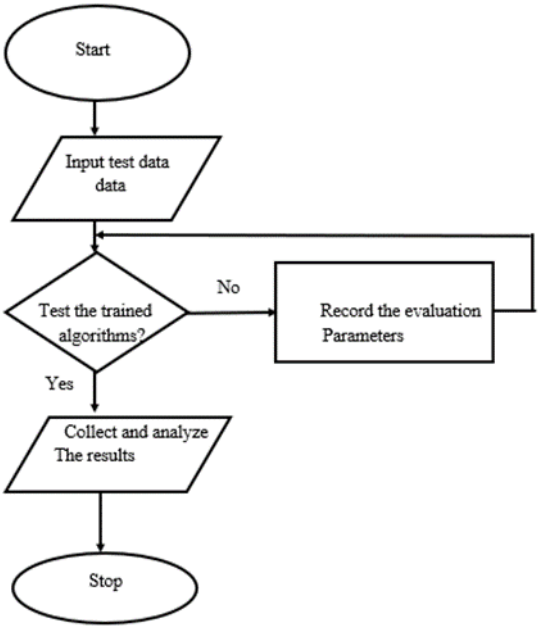

Fig. 4. Testing ML model

The confusion matrix obtained for the classifiers is shown in table 1.

Table 1. Confusion matrix of classifiers

Classifiers	**Actual Classes**	**Predicted Classes**				
		Bills	OTP	Personal	Promotion	Spam
RF	Bills	4	0	0	0	0
	OTP	0	4	0	0	0
	Personal	0	0	10	0	1
	Promotion	0	0	0	4	1
	Spam	0	0	4	1	4
NB	Bills	3	0	0	0	1
	OTP	0	4	0	0	0
	Personal	3	2	6	0	0
	Promotion	0	0	0	5	0
	Spam	1	1	4	3	0
k-NN	Bills	4	0	0	0	0
	OTP	0	4	0	0	0
	Personal	0	0	10	0	1
	Promotion	0	0	0	4	1
	Spam	0	0	4	1	4
SVM	Bills	4	0	0	0	0
	OTP	0	4	0	0	0
	Personal	0	0	6	0	5
	Promotion	0	0	0	3	2
	Spam	0	0	4	0	5
DT	Bills	4	0	0	0	0
	OTP	0	4	0	0	0
	Personal	0	0	10	0	1
	Promotion	0	0	0	4	1
	Spam	0	0	4	1	4

Following observations can be made from table 1.

The classifiers RF, k-NN, SVM, and DT were able to classify the mails related to bills and OTP accurately.

- The RF and k-NN classifiers incorrectly classified 1 personal mail and 1 promotional mail as spam mails, 4 spam mails as personal mails, and 1 spam mail as promotional mail. Both the classifiers accurately classified 10 personal mails, 4 promotional mails, and 4 spam mails.
- The NB classifier accurately classified the OTP mails. However, it incorrectly classified 1 billing mail as spam, 3 personal mails as bill, and 2 personal mails as OTP.
- The SVM classifier correctly classified 6 personal mails but inaccurately classified 5 personal mails as spam mails. It classified 3 promotional mails correctly but misclassified 2 promotional mails as spam. It classified 5 spam mails accurately but misclassified 4 spam mails as personal mails.
- Finally, the DT classifier was able to classify 10 personal mails, 4 promotional mails, and 4 spam mails accurately. However, it misclassified 1 personal mail as spam, 1 promotional mail as spam, 4 spam mails as personal, and 1 spam mail as professional.

Table 2 shows the classification report of all the classifiers used for email classification.

Table 2. Classification results of ML models

Classifiers	Classes	Precision	Recall	F1-Score	Support
RF	Bill	1.00	1.00	1.00	4
	OTP	1.00	1.00	1.00	4
	Personal	0.71	0.91	0.80	11
	Promotion	0.80	0.80	0.80	5
	Spam	0.57	0.44	0.53	9

NB	Bill	0.43	0.75	0.55	4
	OTP	0.57	1	0.73	4
	Personal	0.60	0.55	0.57	11
	Promotion	0.62	1.00	0.77	5
	Spam	0.00	0.00	0.00	9
k-NN	Bill	1.00	1.00	1.00	4
	OTP	1.00	1.00	1.00	4
	Personal	0.71	0.91	0.80	11
	Promotion	0.80	0.80	0.80	5
	Spam	0.67	0.44	0.53	9
SVM	Bill	1.00	1.00	1.00	4
	OTP	1.00	1.00	1.00	4
	Personal	0.60	0.55	0.57	11
	Promotion	1.00	0.60	0.75	5
	Spam	0.42	0.56	0.48	9
DT	Bill	1.00	1.00	1.00	4
	OTP	1.00	1.00	1.00	4
	Personal	0.71	0.91	0.80	11
	Promotion	0.80	0.80	0.80	5
	Spam	0.67	0.44	0.53	9

Fig.5 shows the comparison of the ML models with respect to accuracy.

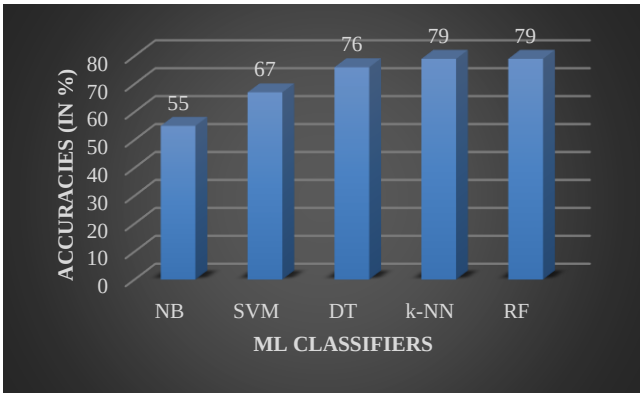

Fig. 5. Performance of ML models with respect to accuracy

As depicted in Fig.5, the k-NN and RF models exhibited comparable performance with an accuracy of 79%. They performed with accuracy improvements of 24% over NB classifier, 12% over SVM classifier, and 3% over the DT model.

Figures. 6, 7, 8, 9, and 10 provide the snapshots of the web application results for classification of mails as personal, bills, OTPs, promotion, and spam.

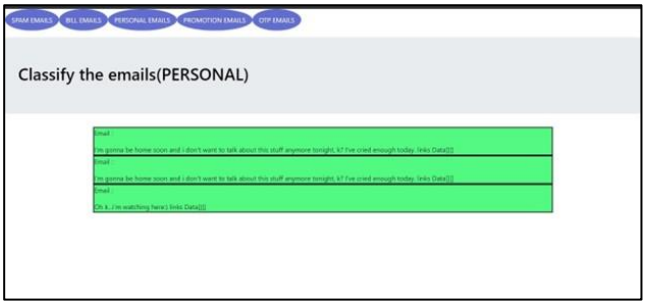

Fig. 6. Classification of personal mails

Personal mails are coded as green which shows the mail importance to the user as depicted in Fig.6.

Fig. 7. Classification of mails related to bills

As depicted in Fig.7, the bills and payment mails are coded as green since they need to be stored for longer period of time.

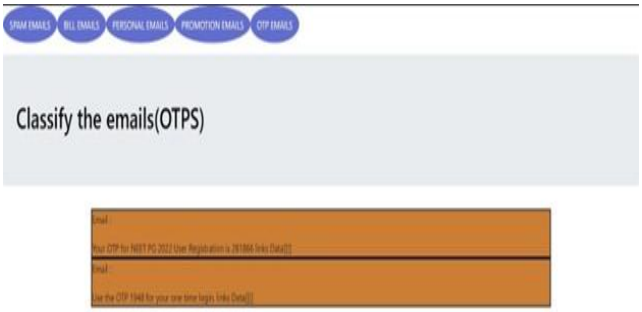

Fig. 8. Classification of OTP mails

As illustrated in Fig.8, OTP are one-time usage and long-term storage. Similar emails might be deleted to save storage space.

Fig. 9. Classification of promotional mails

Advertising is the focus of advertisements. By deleting such unneeded mails that date back a very long time, it also contributes to reduced material usage because these are no longer helpful once the user's interests shift.

Fig. 10. Classification of spam emails

Spam mails pose threat to users. Hence such emails are coded red to warn not to follow the emails.

6. Conclusion and future scope

It was comfortable to use supervised ML techniques for multiclass email classification, and they produced acceptable results about presentation borders. Each computation has demonstrated its importance in the sphere of arranging and has unique drawbacks when describing the messages. After analyzing the display of numerous computations, we were able to improve results by an atypical calculation of the timberland, which revealed a more effective manner of message organization. Nevertheless, it is entirely dependent on the data used to create the model. Better results for the other reported computations might come from a diverse dataset. As technology is continually evolving, solutions could be updated and enhanced by further investigation of the problem, which can be helpful in providing a better and more exact solution that results in the user-friendly product.

For the time being, we have been preparing the data by adding tags to the original email. Additionally, the topic and body can be combined to create the model's display. The decision-making process for communication layout can also be significantly developed by using period and shipper specifics. Currently, we use the email content rather than the topic to determine whether the topic is sufficient for email organization. We found that the subject is not sufficient for message ordering. Additionally, the use of "deep learning" and "neural network" approaches for email order can provide greater accuracy and provide a better solution than controlled AI computations. Additionally, hyper boundary tuning can be used to improve the model's display, which helps it identify the correct mail and organize it properly.

In addition, improvements in "natural language processing" can be utilized to sort communications according to the emails that are most beneficial for the customers and the messages that are generally too much for the clients to handle over the long term. This could also be used as a strategy for classifying communications to streamline the process of collecting them and making them effective. If the anticipated number of message types could be increased while taking the client's requirements into account as the number of emails sent increases gradually.

Author contributions

K Aditya Shastry: Conceptualization, Methodology, Software, Field study **Chandrashekhar B N:** Data curation, Writing-Original draft preparation, Software, Validation., Field study **Manjunatha B A:** Visualization, Investigation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Gomes, S. R., Saroar, S. G., Mosfaiul, M., Telot, A., Khan, B. N., Chakrabarty, A., & Mostakim, M. (2017). A comparative approach to email classification using Naive Bayes classifier and hidden Markov model. 2017 4th International Conference on Advances in Electrical Engineering (ICAEE), 482-487. https://doi.org/10.1109/icaee.2017.8255404.

[2] Raza, M., Jayasinghe, N. D., & Muslam, M. M. A. (2021). A Comprehensive Review on Email Spam Classification using Machine Learning Algorithms. 2021 International Conference on Information Networking (ICOIN), 327-332. https://doi.org/10.1109/icoin50884.2021.9334020.

[3] Li, W., & Meng, W. (2015). An empirical study on email classification using supervised machine learning in real environments. 2015 IEEE International Conference on Communications (ICC), 7438-7443. https://doi.org/10.1109/icc.2015.7249515.

[4] Iqbal, K., & Khan, M. S. (2022). Email classification analysis using machine learning techniques. Applied Computing and Informatics, 630-635. https://doi.org/10.1108/aci-01-2022-0012 .

[5] Junnarkar, A., Adhikari, S., Fagania, J., Chimurkar, P., & Karia, D. (2021). E-Mail Spam Classification via Machine Learning and Natural Language Processing. 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 693-699. https://doi.org/10.1109/icicv50876.2021.9388530.

[6] Crawford, M., Khoshgoftaar, T. M., Prusa, J. D., Richter, A. N., & Al Najada, H. (2015). Survey of review spam detection using machine learning techniques. Journal of Big Data, 2(1). https://doi.org/10.1186/s40537-015-0029-9.

[7] Dada, E. G., Bassi, J. S., Chiroma, H., Abdulhamid, S. M., Adetunmbi, A. O., & Ajibuwa, O. E. (2019). Machine learning for email spam filtering: Review, approaches, and open research problems. Heliyon, 5(6). https://doi.org/10.1016/j.heliyon.2019.e01802.

[8] Risch, J., & Krestel, R. (2020). Toxic Comment Detection in Online Discussions. Algorithms for Intelligent Systems, 85-109. https://doi.org/10.1007/978-981-15-1216-2_4.

[9] Ahsan, M. I., Nahian, T., Kafi, A. A., Hossain, M. I., & Shah, F. M. (2016). Review spam detection using active learning. 2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 1-7. https://doi.org/10.1109/iemcon.2016.7746279.

[10] A Karim, A., Azam, S., Shanmugam, B., Kannoorpatti, K., & Alazab, M. (2019). A Comprehensive Survey for Intelligent Spam Email Detection. IEEE Access, 7, 168261-168295. https://doi.org/10.1109/access.2019.2954791.

[11] Guang Jun, L., Nazir, S., Khan, H. U., & Haq, A. U. (2020). Spam Detection Approach for Secure Mobile Message Communication Using Machine Learning Algorithms. Security and Communication Networks, 2020, 1-6. https://doi.org/10.1155/2020/8873639.

[12] Vinitha, V. S., & Renuka, D. K. (2019). Performance Analysis of E-Mail Spam Classification using different Machine Learning Techniques. 2019 International Conference on Advances in Computing and Communication Engineering (ICACCE), 1-5. https://doi.org/10.1109/icacce46606.2019.9080000.

[13] Raja, P., Sangeetha, K., SuganthaKumar, G., Madesh, R., & Vimal Prakash, N. (2022). Email Spam Classification Using Machine Learning Algorithms. 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), 343-348. https://doi.org/10.1109/icais53314.2022.9743033.

[14] Li, W., Meng, W., Tan, Z., & Xiang, Y. (2014). Towards Designing an Email Classification System Using Multi-view Based Semi-supervised Learning. 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications, 174-181. https://doi.org/10.1109/trustcom.2014.26.

[15] Kumar, N., Sonowal, S., & Nishant. (2020). Email Spam Detection Using Machine Learning Algorithms. 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), 108-113. https://doi.org/10.1109/icirca48905.2020.9183098.

[16] K. Taunk, S. De, S. Verma and A. Swetapadma, "A Brief Review of Nearest Neighbor Algorithm for Learning and Classification," 2019 International Conference on Intelligent Computing and Control Systems (ICCS), 2019, pp. 1255-1260, doi: 10.1109/ICCS45141.2019.9065747.

[17] T. Toma, S. Hassan and M. Arifuzzaman, "An Analysis of Supervised Machine Learning Algorithms for Spam Email Detection," 2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI), 2021, pp. 1-5, doi: 10.1109/ACMI53878.2021.9528108.

[18] V. Vishagini and A. K. Rajan, "An Improved Spam Detection Method with Weighted Support Vector Machine," 2018 International Conference on Data Science and Engineering (ICDSE), 2018, pp. 1-5, doi: 10.1109/ICDSE.2018.8527737.

[19] A. Sumithra, A. Ashifa, S. Harini and N. Kumaresan, "Probability-based Naïve Bayes Algorithm for Email Spam Classification," 2022 International Conference on Computer Communication and Informatics (ICCCI), 2022, pp. 1-5, doi: 10.1109/ICCCI54379.2022.9740792.

[20] I. Čavor, "Decision Tree Model for Email Classification," 2021 25th International Conference on Information Technology (IT), 2021, pp. 1-4, doi: 10.1109/IT51528.2021.9390143.

[21] N. Mirza, B. Patil, T. Mirza and R. Auti, "Evaluating efficiency of classifier for email spam detector using hybrid feature selection approaches," 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), 2017, pp. 735-740, doi: 10.1109/ICCONS.2017.8250561.

[22] Doulamis, Anastasios D., Ismail, Safaa S. I., Mansour, Romany F., Abd El-Aziz, Rasha M. Taloba, Ahmed I.,(2022), Efficient E-Mail Spam Detection Strategy Using Genetic Decision Tree Processing with NLP Features,Computational Intelligence and Neuroscience, Hindawi, doi:10.1155/2022/7710005.

[23] J. Cui and X. Li, "Content Based Spam Email Classification using Supervised SVM, Decision Trees and Naive Bayes," ICMLCA 2021; 2nd International Conference on Machine Learning and Computer Application, 2021, pp. 1-4.

[24] M. Heydarian, T. E. Doyle and R. Samavi, "MLCM: Multi-Label Confusion Matrix," in IEEE Access, vol. 10, pp. 19083-19095, 2022, doi: 10.1109/ACCESS.2022.3151048.