Knowledge Engineering and Data Science (KEDS)
Vol 5, No 1, December 2022, pp. 41–52

pISSN 2597-4602
eISSN 2597-4637

A Comparison of Machine Learning Models to Prioritise Emails using Emotion Analysis for Customer Service Excellence

Mohammad Yasser Chuttur [1], *, Yashinee Parianen [2]

Department of Software and Information Systems, University of Mauritius
2nd floor Phase II building University of Mauritius Reduit MU, 80837, Mauritius
*[1] y.chuttur@uom.ac.mu *; [2] yashinee.parianen@umail.uom.ac.mu*
** corresponding author*

ARTICLE INFO

Article history:
Received 22 July 2022
Revised 2 August 2022
Accepted 15 August 2022
Published online 7 November 2022

Keywords:
Emotion Analysis
Machine Learning
Email Prioritization
Customer Service

ABSTRACT

There has been little research on machine learning for email prioritization for customer service excellence. To fill this gap, we propose and assess the efficacy of various machine learning techniques for classifying emails into three degrees of priority: high, low, and neutral, based on the emotions inherent in the email content. It is predicted that after emails are classified into those three categories, recipients will be able to respond to emails more efficiently and provide better customer service. We use the NRC Emotion Lexicon to construct a labeled email dataset of 517,401 messages for our proposal. Following that, we train and test four prominent machine learning models, MNB, SVM, LogR, and RF, and an Ensemble of MNB, LSVC, and RF classifiers, on the labeled dataset. Our main findings suggest that machine learning may be used to classify emails based on their emotional content. However, some models outperform others. During the testing phase, we also discovered that the LogR and LSVC models performed the best, with an accuracy of 72%, while the MNB classifier performed the poorest. Furthermore, classification performance differed depending on whether the dataset was balanced or imbalanced. We conclude that machine learning models that employ emotions for email classification are a promising avenue that should be explored further.

I. Introduction

The problems caused by email overload in organizational settings have been well documented in [1][2][3]. On average, a mid-size organization may receive thousands of emails per day, and employees often struggle to respond to queries in a timely fashion [3]. In many cases, employees adopt a ‘last come first serve strategy’, where emails that are received the last are responded to first. Others may adopt a ‘first come first serve strategy’ and respond to emails that came in first and then proceed to other emails. When it comes to customer service excellence, however, such a strategy, does not represent the most effective way to address the important customer concerns. Customers with complaints or urgent issues are treated with the same priority level as those with no complaints or minor issues. In other words, frustrated and dissatisfied customers who need to be prioritized are disregarded. Instead, a more effective approach to better serve customers would have been to give more attention to customers who urgently need a response than a customer who is more likely to wait for an answer [4]. However, to do so will require that recipients manually filter out all new incoming emails and select only those emails that need to be attended in priority.

With recent technological advancements and widespread adoption of smartphones, however, a rising number of people use email extensively. Manual filtering of emails, thus, is not only laborious but also a non-productive task. Employees are overwhelmed with emails and often face many difficulties when doing manual email triage to determine which emails are to be treated with higher priority. According to [5], people's psychological resources are depleted by work-related emails, especially incoming work-related emails, leading to experiences of job overload, compulsive use,

https://doi.org/10.17977/um018v5i12022p41-52

stress, and work-family imbalance. Email overload has direct negative consequences on employee productivity and must be addressed.

In various contexts, emotion detection from written text, such as emails, may be used to improve work performance and customer relationships [6]. Emotion indicates the psychological state, which is impacted by the discernment of someone's surroundings, health, and intent [7], and email contents are often filled with emotional cues. Through automatic emotion analysis, it is possible to obtain valuable information on how a specific audience feels about a given product, person, or service offered by a business. In other words, automated emotion detection systems can be employed by businesses to track and recognize emotional reactions to their goods and services. For instance, in power marketing, the user's feelings from speech data have been analysed for improved customer service [8]. In other cases, customer service agents can use automated anger detection systems in customer care emails to recognize unhappy consumers more quickly and take the necessary prompt actions to boost customer retention rates [9]. Without measures that track customer emotions, businesses risk-averse consequences on their reputation and related financial impacts, such as the loss of clients [10].

Emotion analysis differs from sentiment analysis, categorizing textual data as positive, neutral, or negative. Instead, emotion analysis provides information about an individual's feelings or emotions through a series of "emotional connotations" like joy, sadness, or anger. Many proposed emotion models are reported in [11][12][13]. Each of those emotion models proposes a list of emotions that humans express. A popular emotion model is the wheel of emotions defined by Robert Plutchik [14]. As shown in Figure 1, the wheel of emotions lists several emotions that an individual usually expresses. Each emotion can have different intensity, as illustrated by different wheel cones. Robert Plutchik also noted that individuals could express one or more of eight primary emotions, as shown in Table 1.

Following the reasoning that frustrated customers will express primarily negative emotions, it should be possible for machine learning to detect email contents with negative content and classify them as high priority compared to emails, which contain neutral or positive emotions. To date, however, not much attention has been given to the use of emotions to classify emails according to

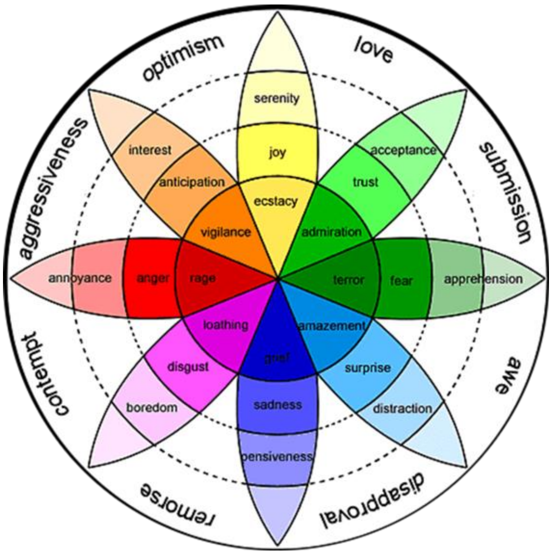


Fig. 1. The "wheel of emotions" by Robert Plutchik

Table 1. Robert Plutchik eight basic emotions

Positive emotions	**Polar opposite emotions**
Joy	Sadness
Anticipation	Surprise
Trust	Disgust
Fear	Anger

different priority levels. Instead, more attention has been given to spam detection.

For instance, the works of [15][16][17][18] demonstrate machine learning techniques for email spam detection. A hybrid approach to spam detection is further found in the work of [19] and [20], and [21] evaluated the use of semantic features for spam detection in emails. In addition, a detailed review of spam detection techniques can be found in the works of [22][23][24][25]. Filtering spam emails targets unwanted emails but does not set any priority scheme for emails [15]. As stated by [26], there is a clear distinction between spam detection and email prioritization. The prioritization of emails aims at personalizing non-spam emails by estimating their relevance. Wang [26] also states that email prioritization can be split into two main groups depending on the targeted outcome: action prediction and priority label prediction, both of which require a classification task. To the researchers' knowledge, research on using machine learning and emotion analysis for email prioritization is scarce. One such research can be found in [27]. The authors used Naïve Bayes to categorize several emails according to their importance. [27] hypothesized that assigning different weights to selected terms from email contents makes it possible to calculate the overall importance or priority of these emails. However, the authors did not report any implementation results.

In this study, we investigate the possibility of using machine learning to analyse the emotions expressed in emails to set a priority ranking to different emails. It is posited that customers will send emails containing different expressed emotions, which, when detected, can further help classify those emails into three main groups: high priority, neutral, and low priority. Our work contrasts with previous studies in that most works on email classification have focused on spam detection. The main contributions of this work are as follows. We create a labelled dataset of emails using emotions from the NRC Emotion Lexicon. There is currently no email dataset labelled with emotions. We then devise a novel algorithm to assign three levels of priorities, namely high, low, and neutral to the messages in our dataset. Once the priority labels are assigned, we subject our dataset to some pre-processing stages. We then train, test, and compare different supervised machine learning models for their ability to correctly classify different email messages according to the three priority levels set for this study.

The rest of the paper is organized as follows. In section II, we provide details on our proposed methodology to use emotions and machine learning to classify emails according to three levels of priorities. In section III, we present and discuss the results obtained. Moreover, in section IV, we conclude our work with some future recommendations.

II. Method

This study aims to evaluate the efficacity of machine learning to prioritize emails based on the emotional contents of the texts within. The general process flow for our proposal is depicted in Figure 2.

A. Data Acquisition

No publicly accessible email dataset is labelled with emotions like happiness, sadness, or anger. Hence, a labelled dataset will have to be created for this study. To this end, the Enron email dataset is selected because it is a large email datasets that has already been used in several related studies such as [19], [20], [28], [29], and [30]. The Enron email dataset at https://www.cs.cmu.edu/~./enron/ includes 517,401emails sent by Enron Corporation employees. The "Federal Energy Regulatory Commission" collected it as part of its inquiry into Enron's downfall. The dataset is saved as a csv file and obtained from Kaggle.

B. Data Cleaning and Pre-processing

The process of data cleaning aims to eliminate irrelevant contents from the dataset. In the context of this project, irrelevant content refers to any part of the email that is not valuable when the learning algorithm assigns a class to the email. Not only will data cleaning make the task of classification easier for the classification model, but it may also significantly reduce the processing time in the training stage. As stated by [20], data pre-processing is essential to yield a better outcome. Data pre-processing aims at curtailing noise and can help tackle the dimensionality curse reported by [31] and [32].

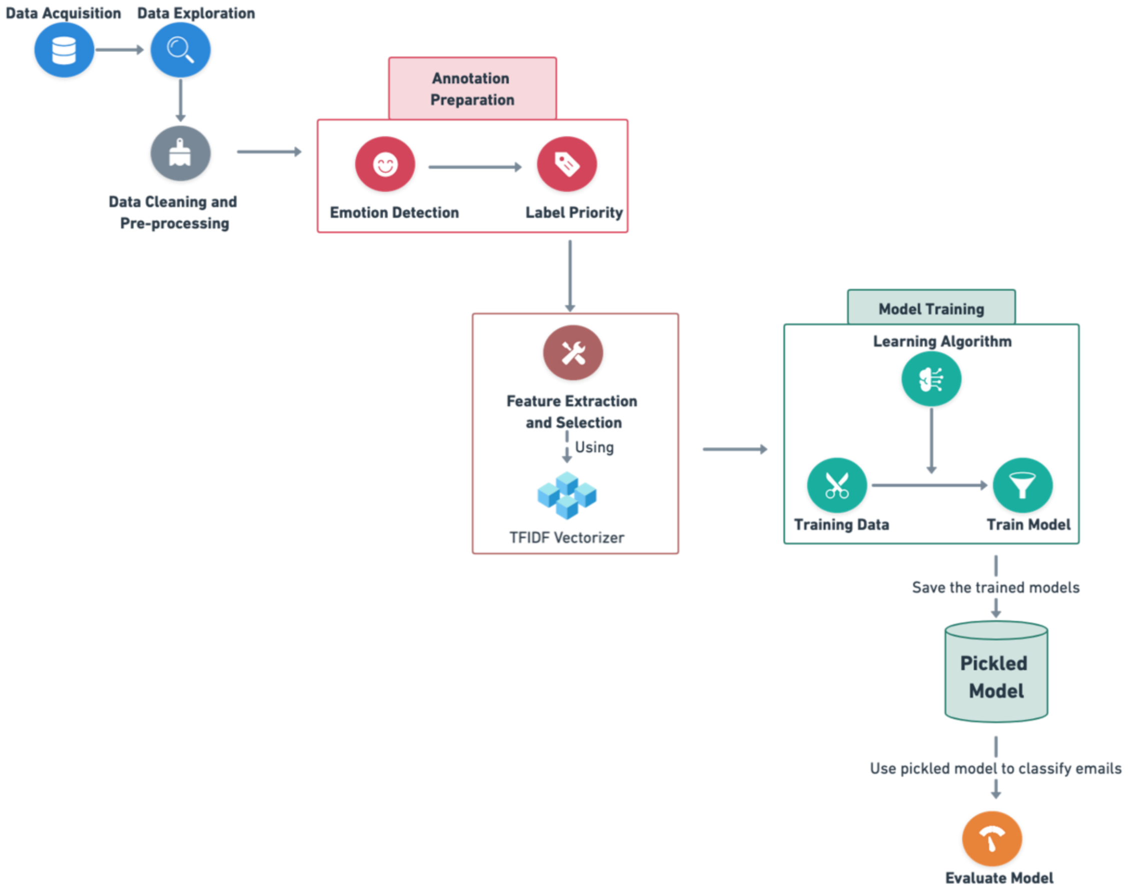


Fig. 2. General process flow for email prioritization based on emotions and machine learning

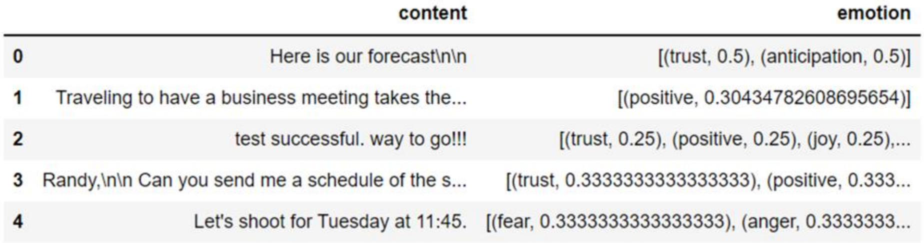
For data cleaning, duplicate and irrelevant fields were removed from the raw dataset. As for data pre-processing, the following was applied to the cleaned email dataset: lower casing, noise removal, stop words removal, and tokenization. The curse of dimensionality constraint is dealt with by including text normalization and lemmatization techniques in the pre-processing phase to help in dimensionality reduction. The steps have been curated and adapted from [19] and [20].

C. Annotation and Priority Labeling

Annotation preparation is a crucial step as the emails in the dataset must be labelled with their relevant emotions to enable the use of supervised machine learning. It was reported by [20] that lexicon labelling provides clear and uniform results. Several existing sentiment lexicons have been employed in developing different systems and algorithms. Some examples are VADER, AFINN, and Sentiment140. In this study, the NRC Word-Emotion Association Lexicon at https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm is used for the emotion detection process since it is a list containing words based on different emotions.

It should be noted that the NRC Word-Emotion Association Lexicon provides multiple emotions, which is associated with a polarity (positive/negative number) weight based on the contents of an analysed text contents. Once labeled, each email is tagged with a priority label according to the emotion detected. The pseudocode for assigning the labels “High Priority”, “Low Priority”, and “Neutral” is as follows.

```
START
   Calculate weight sum of good emotions ('anticipation', 'trust', 'joy', 'positive',
      'surprise')
   Calculate weight sum of bad emotions ('anticipation', 'surprise', 'anger', 'disgust',
      'fear', 'sadness', 'negative')
   `If weight sum good_emotion > weight sum bad_emotion Then return ‘Low Priority”
```



	content	emotion
0	Here is our forecast\n\n	[(trust, 0.5), (anticipation, 0.5)]
1	Traveling to have a business meeting takes the...	[(positive, 0.30434782608695654)]
2	test successful. way to go!!!	[(trust, 0.25), (positive, 0.25), (joy, 0.25),...
3	Randy,\n\n Can you send me a schedule of the s...	[(trust, 0.3333333333333333), (positive, 0.333...
4	Let's shoot for Tuesday at 11:45.	[(fear, 0.3333333333333333), (anger, 0.3333333...

Fig. 3. NRC emotions and polarity weights

Table 2. TD-IDF hyperparameters

Hyperparameter	Description
“$\boldsymbol{max\_df = 0.90}$”	Set a threshold to ignore words with document frequency greater than 0.90
“$\boldsymbol{min\_df = 2}$”	Set a threshold to ignore words with document frequency lower than 2
“$\boldsymbol{max\_features = 1000}$”	To consider the top 1000 features in the corpus
“$\boldsymbol{stop\_words = stop\_words}$”	To remove the words from the stop words list
"$\boldsymbol{ngram\_range = (1,2)}$"	To get features composed of single tokens.

```
    Else If weight sum bad_emotion > weight sum good_emotion Then return ‘High Priority”
    Else return “Neutral”
END
```

An example of emotion polarity weights obtained for different messages that can be obtained from the NRC lexicon is shown in Figure 3.

D. Feature Extraction and Selection

Machine learning algorithms are unable to work directly on raw text. Hence, feature extraction methods, otherwise known as vectorization, are conducted to transform text to numerical data, more specifically into a vector of features using Term Frequency-Inverse Document Frequency (TF-IDF), which was initially designed for text categorization [33].

TF-IDF classifiers use frequency feature vectors as input and assess the weight of the features/words by using both TF and IDF. Term Frequency (TF) is the number of times a term appears in a text and Inverse Document Frequency (IDF) assesses a term’s significance [34]. The formulas used to calculate the TF and IDF are given by (1) and (2).

$$T(t) = \frac{Frequency\ of\ term\ t}{Total\ number\ of\ terms} \tag{1}$$

$$IDF(t) = log_e \frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ t\ in\ it} \tag{2}$$

TF-IDF classifiers rely on a computational statistical approach that works by filtering the features by weighting and rating each unigram and N-grams based on the number of times certain words appear in the text [35]. In this study, TF-IDF is used to execute this conversion as recommended by [18][19][20][35]. Table 2 provides some more details on the hyperparameters used for the *TfidfVectorizer* available in python.

E. Model Training

In this step, the vectors generated during the feature extraction phase are used to train and test the machine learning models selected for this study. The dataset is uniformly and randomly split into 80% train set and 20% test set. We shall train and test the performance of the following popular machine learning models: SVM, NB, LogR, and RF. Those classifiers have been chosen for their reported good performance scores as reported in [35][36][37][38][39]. As recommended by [40], we will also investigate whether an ensemble method may yield better performance than the selected machine learning algorithms alone. Stacking is an ensemble method which learns to integrate the predictions from several machine learning models optimally. Here, the MNB, LSVC and RF model will be stacked to build a new ensemble model. The ensemble method choses the best classification model to use on the test set after each one has been evaluated on the training set. The main goal of ensemble method is to integrate the outputs of several classifiers to build a strong one [41].

F. Model Evaluation

The selected machine learning models will be trained and tested on the Enron email dataset labelled with the NRC lexicon. For evaluation purposes, the accuracy and F1-score obtained for each model will be used to compare the performance of the implemented algorithms. Accuracy refers to the ratio of correctly categorized data to the overall classifications. The formula used to calculate accuracy is:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3}$$

F1-score, alternatively termed as F-measure is the “harmonic mean” of the Precision and Recall. In other words, F1-score indicates which percent of positive predictions observed were correct.

$$F1-score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \tag{4}$$

Precision, in this study concerning the Neutral class, refers to the number of cases where the expected and actual results are both Neutral.

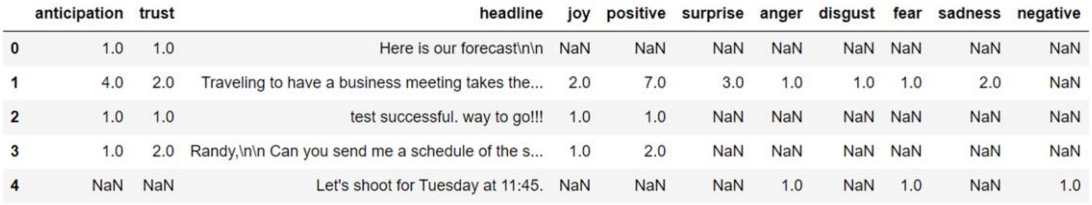
$$Precision = \frac{TP}{TP+FP} \tag{5}$$

Recall, in the context of this study with respect to the Neutral class refers to the capacity of the model to predict the emails of the Neutral class.

$$Recall = \frac{TP}{TP+FN} \tag{6}$$

III. Results and Discussions

We used Python 3.9.2, Jupyter notebook, and the Anaconda distribution to implement our proposed email prioritization approach. Table 3 lists the different python libraries we used to execute some of the main processes described in Section 2.



	anticipation	trust	headline	joy	positive	surprise	anger	disgust	fear	sadness	negative
0	1.0	1.0	Here is our forecast\n\n	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	4.0	2.0	Traveling to have a business meeting takes the...	2.0	7.0	3.0	1.0	1.0	1.0	2.0	NaN
2	1.0	1.0	test successful. way to go!!!	1.0	1.0	NaN	NaN	NaN	NaN	NaN	NaN
3	1.0	2.0	Randy,\n\n Can you send me a schedule of the s...	1.0	2.0	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	Let's shoot for Tuesday at 11:45.	NaN	NaN	NaN	1.0	NaN	1.0	NaN	1.0

Fig. 4. NRC raw emotion scores results

Table 3. Python libraries used

Library	Purpose
pandas	Transform data in tabular format
NRCLex	Measure emotional affect from a body of text
nltk	For stopwords removal, tokenization
spacy	For lemmatization
numpy	Calculate average score
scikit learn	Import selected machine learning classification models
keras	For deep learning
Imbalance learn	To import sampling modules
BeautifulSoup	To remove html tags from emails
string	For noise removal
pickle	To save and load trained machine learning models

A. Calculating Raw Emotion Scores for Annotation and Priority Labeling

Once we obtained the Enron email dataset, as explained in Section IIA, we cleaned the data and applied several pre-processing operations as described in Section IIB. We then used the "top_emotion" module from NRCLex to view the highest polarities from the email text for training our machine learning models. A snapshot of the resulting email messages and the associated emotions is shown in Figure 4.

The "raw_emotion_scores" module from NRCLex was used to obtain the polarities of the different emotions. The results were then transformed into a Pandas DataFrame and the array of the different polarities were classified according to each emotion using the "pandas.DataFrame.form_records" module.

The score obtained for each emotion set was then used to decide on the polarity label (high, low, neutral) to assign to each email message according to the algorithm described in Section IIC. The resulting dataset was then inspected for data distribution. Figure 5 shows the results of the size of classes of the complete dataset and of the dataset after removing duplicates.

As observed, the pre-processing phase and priority labels were applied to two groups of the Enron email datasets. In one group, we kept all the records but in the second group, we removed all duplicate messages. We could see that both data groups were imbalanced, which can further influence the classification performance. In other words, the classifiers may try to improve the accuracy of the larger class to the detriment of the smaller classes.

The data was further sampled to balance the dataset as recommended by [29] to address the issue

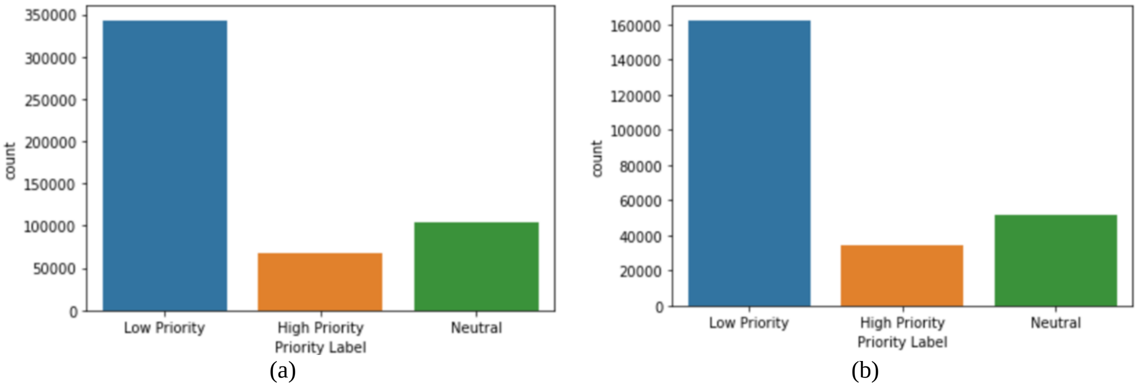


Fig. 5. (a) Bar chart showing the size of classes of the complete dataset (b) Bar chart showing the size of classes of the dataset after removing duplicates.

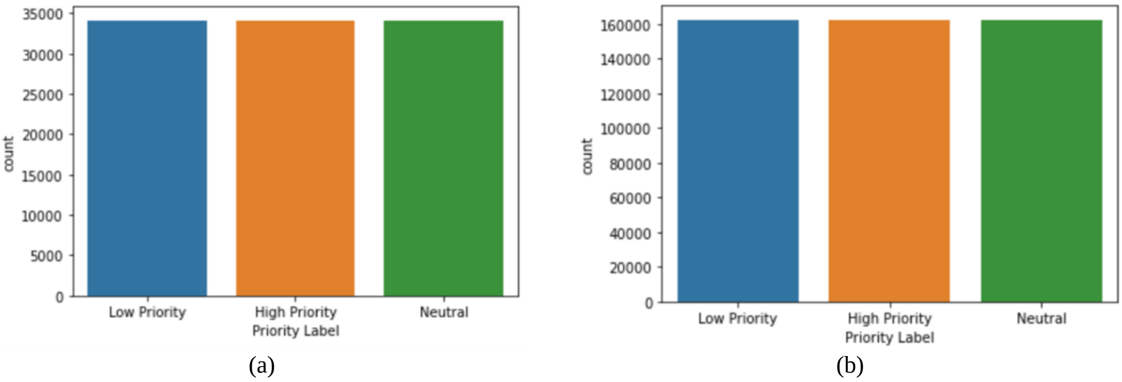


Fig. 6. Class distribution of the dataset with no duplicate after (a) undersampling and (b) oversampling

of the classifier biasing towards the majority class. The sampling method used was random oversampling**,** where data from the “minority class” were duplicated randomly, and random undersampling, where data from the “majority class” were randomly removed. The same sampling techniques were applied to both the complete or full dataset and the dataset with duplicates removed. Figure 6 shows the dataset distribution for the dataset with no duplicate after undersampling and oversampling, respectively. More after, a similar balanced class distribution was obtained for the entire dataset.

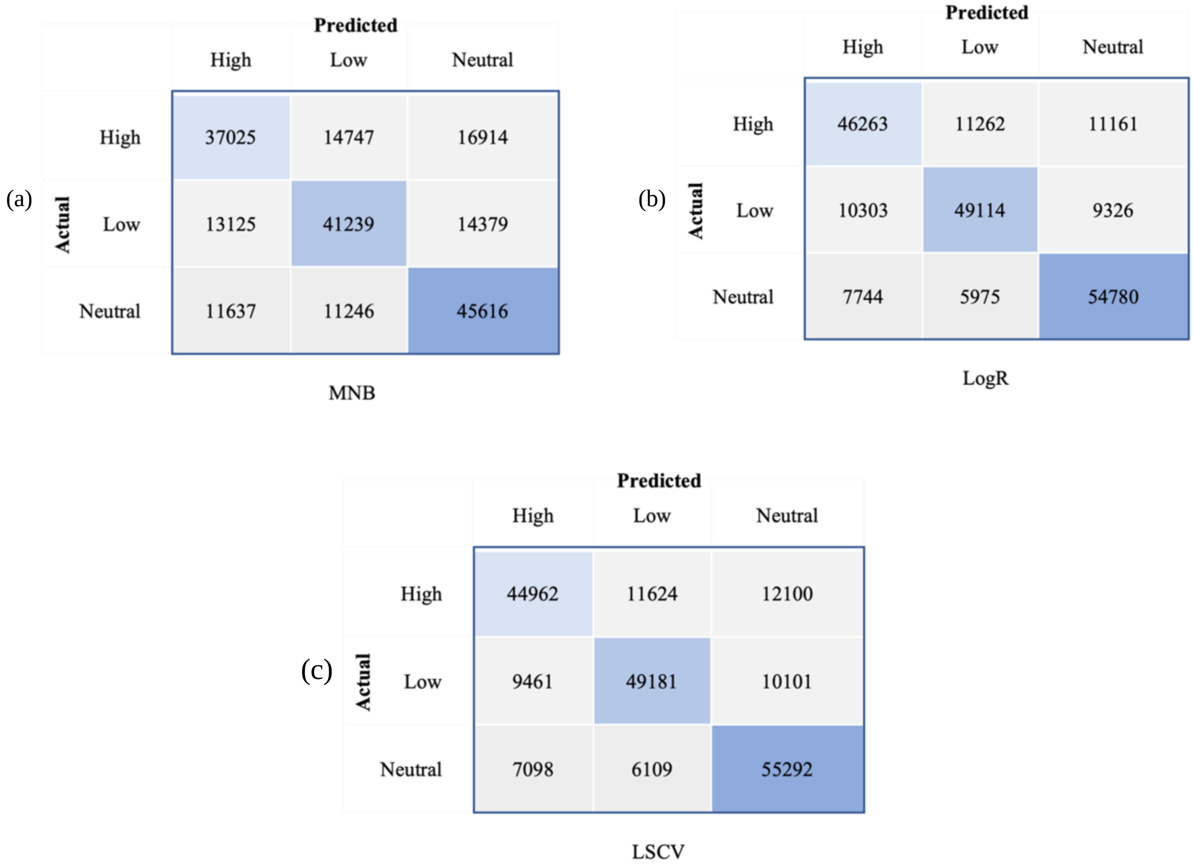
B. Feature Extraction and Selection

For feature extraction, the "TfidfVectorizer()" function from "SciKit Learn" module has been employed. The lemmatized text is fitted into the TfidfVectorizer. The main purpose of this approach was to improve the computation and training processes. Once the TF-IDF representation of the dataset is generated, the dataset was split into 80% train set and 20% test set using sklearn’s “train_test_split” function. The feature vectors generated by the TfidfVectorizer are then used as input to train the ML classification models.

As mentioned earlier, the following classifiers are used to fit the training data: NB, SVM, LogR and RF. Thus, the inbuilt classes, namely MultinomialNB, LinearSVC, LogisticRegression, and RandomForestClassifier from the "SciKit Learn" library are used to train the models on the dataset, both before and after the removal of duplicates, and evaluate whether the performance on a larger data set is improved.

C. Model Training and Evaluation

In python, we used the ***sklearn***’s “***train_test_split***”, feature to split our dataset uniformly and randomly into 80% train set and 20% test set. The feature vectors generated by the *TfidfVectorizer* and the labeled datasets were used as input to train all the ML classification models selected. The vectorizer and models were then pickled using the ***pickle*** python library to enable



(a)

		Predicted		
		High	Low	Neutral
Actual	High	37025	14747	16914
	Low	13125	41239	14379
	Neutral	11637	11246	45616

MNB

(b)

		Predicted		
		High	Low	Neutral
Actual	High	46263	11262	11161
	Low	10303	49114	9326
	Neutral	7744	5975	54780

LogR

(c)

		Predicted		
		High	Low	Neutral
Actual	High	44962	11624	12100
	Low	9461	49181	10101
	Neutral	7098	6109	55292

LSCV

Fig. 7. Confusion Matrix for (a) MNB, (b) LogR, and (c) LSCV classifier for full oversampled testing set (balanced dataset)

Table 4. F1-macro average score for the full dataset with and without duplicate (imbalanced dataset)

Dataset		**MNB**	**LSVC**	**RF**	**LogR**	**Stacking**
Full Dataset	Training	0.42	0.67	1	0.68	1
	Testing	0.42	0.66	0.92	**0.67**	0.93
Full Dataset Duplicate Removed	Training	0.43	0.67	1	0.68	0.99
	Testing	0.43	0.66	0.68	**0.67**	0.74

Table 5. Accuracy Score on Training and Testing Set (balanced dataset)

Dataset		**MNB**	**LSVC**	**RF**	**LogR**	**Stacking**
Full Dataset (Over sampled)	Training	0.60	0.73	1	0.73	1
	Testing	0.60	0.73	0.99	0.73	0.99
Full Dataset (Under sampled)	Training	0.60	0.73	1	0.73	1
	Testing	0.60	0.72	0.89	0.72	0.90
Duplicate removed (Over sampled)	Training	0.60	0.72	1	0.73	0.97
	Testing	0.59	0.72	0.96	0.72	0.75
Duplicate removed (Under sampled)	Training	0.60	0.73	1	0.73	0.98
	Testing	0.60	0.72	0.74	0.72	0.76

saving and loading of the classifiers. We then obtained the training and testing classification score for different datasets and models when classifying emails into different priority categories using emotions. The relevant confusion matrix was generated for each model to calculate the corresponding TP, TN, FP, and FN values. The F1-Score and overall accuracy for each model and the corresponding dataset were calculated from those values. The confusion matrix for the MNB, LogR, and LSVC classifier corresponding to the full oversampled testing set are shown in Figure 7. Similar confusion matrices were obtained for the other datasets.

We used different performance scores to match the dataset used. For an imbalanced dataset, F1-score gives a more representative idea of the performance of a classifier model, whereas, for balanced datasets, we used the accuracy metric. We also prefer to consult the macro average for the F1-Score as this metric treats all classes equally. The classification performance scores obtained for the full imbalanced dataset with and without duplicates are shown in Table 4. Table 5 provides the accuracy results for all the models for the balanced datasets with and without duplicates.

The performance scores for the RF and Stacking classifiers are seen to exhibit model overfitting, with a perfect 100% score in training but a reduced performance score for the testing set. Similarly, as seen in Table 5, the RF and stacking classifiers obtained 100% accuracy on the training set for all the balanced datasets. However, depending on the dataset, it drops between 72% and 99%, creates a misleading sense of obtaining high accuracy, which can be mostly attributed to model overfitting. In other words, both the RF and stacking models overfit the training set at the expense of an inferior performance on the testing set. To recall the Stacking model was built using the MNB, LSVC and RF classifiers. Therefore, it is safe to assume that the output of RF classifier in the stacking model has resulted in overfitting and hence fails to perform well with the new dataset.

In contrast, the performance scores obtained for the other models, i.e., MNB, LSCV and LogR appear to be more reliable. For the imbalanced datasets (Table 4), the LogR classifier gives a slightly better performance score of 0.67 compared to MNB and LSVC. Overall, all the models gave close performance scores during their training and testing phases.

Likewise, for the balanced datasets (Table 5), the LogR classifier is again seen to provide a good classification performance score. Maximum accuracy of 0.73 close to the LSVC classifier across the balanced datasets, was observed, making both LogR and LSVC as the two most suitable priority classifiers for emails using emotions. Since the MNB classifier gave the worst performance for both the balanced and imbalanced datasets, we deduce that this type of task is not the most suitable model.

In general, therefore, it is found that machine learning models are good candidates for classifying emails into different priority levels based on emotional content in the email. Previous studies have mostly focused on using machine learning techniques for spam detection. This study used the NRC Emotion Lexicon to label an otherwise unlabeled email dataset. The best performance score obtained is good but not good enough to be deployed in a real organization setting. Several improvements can still be made to obtain a better-performing email prioritizing solution to the email overload problem. For instance, as discussed in [12], other emotion models can be used for the data labeling step. Using lesser emotion categories could also increase accuracy, as observed by [6]. Last but not least, as investigated by [42], other machine learning models like RNN can be evaluated for their performance in detecting emotions in email contents.

IV. Conclusion

Email overload is a growing organizational problem that has been overlooked. For businesses, this represents a considerable loss in productivity and poor customer service and increasing psychological stress imposed on employees. The efficacity of four machine learning models namely MNB, LSVC, RF, LogR, and an Ensemble of MNB, LSVC, and RF classifiers were evaluated to address this problem, for their performance in prioritising messages from the Enron email dataset. The dataset was labelled using the NRC emotions lexicon and following several experiments on both imbalanced and balanced datasets, it was discovered that supervised machine learning could be used to detect emotions in email contents and assign priorities to emails accordingly. It was also noticed that data balancing influenced the classification performance and that the RF and the Ensemble methods tended to overfit the data. In parallel, it was found that the LogR and LSVC classifiers gave the best classification score while the MNB classifier performed the poorest. However, the highest performance scores obtained from this study are not good and considered good enough to be effective in a real-life organizational setting. Thus, there is a need for more research into the use of emotions in email content when setting up a priority reply list. In future works, it is recommended that other deep learning models and alternative emotion lexicons be tested for the possibility of achieving better performance scores. In addition, the principle discussed in this paper considered email content written in the English language only. The same techniques may not work well for other written languages, which may require other considerations for text cleaning and pre-processing. In this case, further research is warranted.

Declarations

Author contribution

All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare no known conflict of financial interest or personal relationships that could have appeared to influence the work reported in this paper.

Additional information

Reprints and permission information are available at http://journal2.um.ac.id/index.php/keds.

Publisher’s Note: Department of Electrical Engineering - Universitas Negeri Malang remains neutral with regard to jurisdictional claims and institutional affiliations.

References

[1] B. Graf and C. H. Antoni, “The relationship between information characteristics and information overload at the workplace-a meta-analysis,” *European Journal of Work and Organizational Psychology*, vol. 30, no. 1, pp. 143–158, 2021.

[2] B. Mannion, “Information Overload,” *Risk Management*, vol. 69, no. 4, pp. 26–29, 2022.

[3] R. Kong, H. Zhu, and J. A. Konstan, “Learning to ignore: A case study of organization-wide bulk email effectiveness,” *Proceedings of the ACM on Human-Computer Interaction*, vol. 5, no. CSCW1, pp. 1–23, 2021.

[4] T. Ravichandran and C. Deng, “Effects of Managerial Response to Negative Reviews on Future Review Valence and Complaints,” *Information Systems Research*, 2022.

[5] E. Russell, S. A. Woods, and A. P. Banks, "Tired of email? Examining the role of extraversion in building energy resources after dealing with work-email," *European journal of work and organizational psychology*, vol. 31, no. 3, pp. 440–452, 2022.

[6] Z. Halim, M. Waqar, and M. Tahir, "A machine learning-based investigation utilizing the in-text features for the identification of dominant emotion in an email," *Knowledge-Based Systems*, vol. 208, p. 106443, Nov. 2020.

[7] Z. Shao, R. Chandramouli, K. P. Subbalakshmi, and C. T. Boyadjiev, "An analytical system for user emotion extraction, mental state modeling, and rating," *Expert Systems with Applications*, vol. 124, pp. 82–96, Jun. 2019.

[8] X. Li and R. Lin, "Speech Emotion Recognition for Power Customer Service," in *2021 7th International Conference on Computer and Communications (ICCC)*, 2021, pp. 514–518.

[9] S. Angel Deborah, T. T. Mirnalinee, and S. M. Rajendram, "Emotion analysis on text using multiple kernel gaussian...," *Neural Processing Letters*, vol. 53, no. 2, pp. 1187–1203, 2021.

[10] M. Haberzettl and B. Markscheffel, "A Literature Analysis for the Identification of Machine Learning and Feature Extraction Methods for Sentiment Analysis," in *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*, Sep. 2018, pp. 6–11.

[11] Y. Chuttur and L. Pokhun, "An Evaluation of Deep Learning Networks to Extract Emotions from Yelp Reviews," in *Progress in Advanced Computing and Intelligent Engineering*, Springer, 2021, pp. 55–67.

[12] L. Pokhun and M. Y. Chuttur, "Emotions in texts," *Bulletin of Social Informatics Theory and Application*, vol. 4, no. 2, pp. 59–69, 2020.

[13] V. Ahire and S. Borse, "Emotion detection from social media using machine learning techniques: a survey," in *Applied Information Processing Systems*, Springer, 2022, pp. 83–92.

[14] R. Plutchik, "The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice," *American scientist*, vol. 89, no. 4, pp. 344–350, 2001.

[15] A. A. Alurkar *et al.*, "A proposed data science approach for email spam classification using machine learning techniques," in *2017 Internet of Things Business Models, Users, and Networks*, Nov. 2017, pp. 1–5.

[16] S. R. Gomes *et al.*, "A comparative approach to email classification using Naive Bayes classifier and hidden Markov model," in *2017 4th International Conference on Advances in Electrical Engineering (ICAEE)*, Sep. 2017, pp. 482–487.

[17] E. G. Dada, J. S. Bassi, H. Chiroma, S. M. Abdulhamid, A. O. Adetunmbi, and O. E. Ajibuwa, "Machine learning for email spam filtering: review, approaches and open research problems," *Heliyon*, vol. 5, no. 6, p. e01802, Jun. 2019.

[18] F. Jáñez-Martino, E. Fidalgo, S. González-Martínez, and J. Velasco-Mata, "Classification of Spam Emails through Hierarchical Clustering and Supervised Learning," *arXiv:2005.08773 [cs]*, May 2020, Accessed: Dec. 12, 2020.

[19] S. Liu and I. Lee, "Email Sentiment Analysis Through k-Means Labeling and Support Vector Machine Classification," *Cybernetics and Systems*, vol. 49, no. 3, pp. 181–199, Apr. 2018.

[20] R. S. H. Ali and N. E. Gayar, "Sentiment Analysis using Unlabeled Email data," in *2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, Dec. 2019, pp. 328–333.

[21] N. Saidani, K. Adi, and M. S. Allili, "A semantic-based classification approach for an enhanced spam detection," *Computers & Security*, vol. 94, p. 101716, Jul. 2020.

[22] N. Ahmed, R. Amin, H. Aldabbas, D. Koundal, B. Alouffi, and T. Shah, "Machine learning techniques for spam detection in email and IoT platforms: analysis and research challenges," *Security and Communication Networks*, vol. 2022, 2022.

[23] R. Mansoor, N. D. Jayasinghe, and M. M. A. Muslam, "A comprehensive review on email spam classification using machine learning algorithms," in *2021 International Conference on Information Networking (ICOIN)*, 2021, pp. 327–332.

[24] I. Amin and M. K. Dubey, "Hybrid ensemble and soft computing approaches for review spam detection on different spam datasets," *Materials Today: Proceedings*, 2022.

[25] P. Garg and N. Girdhar, "A Systematic Review on Spam Filtering Techniques based on Natural Language Processing Framework," in *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 2021, pp. 30–35.

[26] B. Wang, "Personalized Broadcast Message Prioritization," Thesis, Applied Sciences: School of Computing Science, 2018. Accessed: Jan. 10, 2021.

[27] S. Choudhari, N. Choudhary, S. Kaware, and A. Shaikh, "Email Prioritization Using Machine Learning," *SSRN Journal*, 2020.

[28] E. M. Bahgat, S. Rady, and W. Gad, "An Email Filtering Approach Using Classification Techniques," in The 1st International Conference on Advanced Intelligent System and Informatics (AISI2015), November 28-30, 2015, Beni Suef, Egypt, Cham, 2016, pp. 321–331.

[29] N. Chhaya, K. Chawla, T. Goyal, P. Chanda, and J. Singh, "Frustrated, Polite, or Formal: Quantifying Feelings and Tone in Email," in *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, New Orleans, Louisiana, USA, Jun. 2018, pp. 76–86.

[30] E. M. Bahgat, S. Rady, W. Gad, and I. F. Moawad, "Efficient email classification approach based on semantic methods," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 3259–3269, Dec. 2018.

[31] M. A. Naser and A. H. Mohammed, "Emails classification by data mining techniques," *Journal of Babylon University: Pure and Applied Sciences*, Vol. 22, No 2, 2014.

[32] X. Fang and J. Zhan, “Sentiment analysis using product review data,” *J Big Data*, vol. 2, Dec. 2015.

[33] Xiao-Lin Wang and Cloete, “Learning to classify email: a survey,” in *2005 International Conference on Machine Learning and Cybernetics*, Aug. 2005, vol. 9, pp. 5716-5719 Vol. 9.

[34] B. Ray, A. Garain, and R. Sarkar, “An ensemble-based hotel recommender system using sentiment analysis and aspect categorization of hotel reviews,” *Applied Soft Computing*, vol. 98, p. 106935, Jan. 2021.

[35] P. Barnaghi, P. Ghaffari, and J. G. Breslin, “Opinion Mining and Sentiment Polarity on Twitter and Correlation between Events and Sentiment,” in *2016 IEEE Second International Conference on Big Data Computing Service and Applications (BigDataService)*, Mar. 2016, pp. 52–57.

[36] V. L. Miguéis, A. Freitas, P. J. V. Garcia, and A. Silva, “Early segmentation of students according to their academic performance: A predictive modelling approach,” *Decision Support Systems*, vol. 115, pp. 36–51, Nov. 2018.

[37] Q. Umer, H. Liu, and Y. Sultan, “Emotion Based Automated Priority Prediction for Bug Reports,” *IEEE Access*, vol. 6, pp. 35743–35752, 2018.

[38] T. Bokaba, W. Doorsamy, and B. S. Paul, “Comparative Study of Machine Learning Classifiers for Modelling Road Traffic Accidents,” *Applied Sciences*, vol. 12, no. 2, Art. no. 2, Jan. 2022.

[39] K. Y. Win, N. Maneerat, S. Choomchuay, S. Sreng, and K. Hamamoto, “Suitable Supervised Machine Learning Techniques For Malignant Mesothelioma Diagnosis,” in *2018 11th Biomedical Engineering International Conference (BMEiCON)*, Nov. 2018, pp. 1–5.

[40] C. K. Hiramath and G. C. Deshpande, “Fake News Detection Using Deep Learning Techniques,” in *2019 1st International Conference on Advances in Information Technology (ICAIT)*, Jul. 2019, pp. 411–415.

[41] D. K. Renuka, T. Hamsapriya, M. R. Chakkaravarthi, and P. L. Surya, “Spam Classification Based on Supervised Learning Using Machine Learning Techniques,” in *2011 International Conference on Process Automation, Control and Computing*, Jul. 2011, pp. 1–7.

[42] M. B. Abbas and M. Khan, “Sentiment Analysis for Automated Email Response System,” in *2019 International Conference on Communication Technologies (ComTech)*, Mar. 2019, pp. 65–70.