e-ISSN: 2582-5208

International Research Journal of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

Volume:05/Issue:04/April-2023 Impact Factor- 7.868 www.irjmets.com

EM CLASSIFIER

Ms Mili Srivastava[*1], Ankit Tiwari[*2], Ankur Singh[*3], Anshu Yadav[*4], Shivani Sharma[5]

[*1]Assistant Professor, Department Of Information Technology, Ajay Kumar Garg Engineering College, Ghaziabad, Uttar Pradesh, India.

[*2,3,4,5]Student, Department Of Information Technology, Ajay Kumar Garg Engineering College, Ghaziabad, Uttar Pradesh, India.

DOI : https://www.doi.org/10.56726/IRJMETS35721

ABSTRACT

Email classification is an essential undertaking in the present computerized correspondence. It assists with arranging and classifying messages in light of their substance and reason. This means to foster an AI and Machine Learning based email classification framework that precisely sorts messages into predefined classes like spam and not spam. The framework will be prepared on an assorted dataset of email content to gain proficiency with the examples that recognize various classifications of messages. The framework will be assessed in light of its exactness, accuracy and will be contrasted with different existing email grouping algorithms to show its viability and proficiency. These strategies are looked at regarding accuracy, fully intent on tracking down the best technique. The Naive Bayes along with several other algorithms like Random Forest, XG Boost and others are utilized in this work to ensure that the circumstances are met with at least preparation and that the outcomes are more exact. In NB Classifier, free words are considered as highlights. This venture will add to the progression of email grouping frameworks and improve the general client experience in emailing the board.

I. INTRODUCTION

Email classification permits clients to coordinate and deal with their messages as per their substance and expected use, which is an urgent errand in contemporary computerized correspondence. Customarily, rule-based frameworks or managed AI strategies like Decision Trees and Support Vector Algorithms have been utilized to group messages. In any case, these strategies can have impediments and may not necessarily produce exact outcomes. To address these constraints, this review proposes the utilization of the Naive Bayes along with some other algorithmic calculation for email grouping, with a particular spotlight on ordering messages into 6 classes namely Company Business, Document Editing, Employment Arrangements, Logistics Arrangements, Personal, Purely Personal. As per a Kaspersky Lab examination from Walk 2016, there were four fold the number of spam messages as there were in contrast with the information from Walk 2015. As indicated by a Statista investigation covering the period from 2012 to 2018, over half of all messages sent worldwide were spam. The Naive Bayes calculation is a probabilistic strategy that uses Bayes' hypothesis, which expresses that the likelihood of an occasion happening is corresponding to the probability of the occasion and the earlier information on the occasion. This calculation has been generally utilized in text arrangement and has been displayed to give exact outcomes in different applications. The objective of this exploration is to assess the viability and effectiveness of the Naive Bayes calculation for email order and to contrast its exhibition and other famous calculations. The review depends on an enormous and different dataset of messages, which were gathered from different sources and pre-handled to extricate important elements for grouping. The aftereffects of the review will give important experiences into the capacities of the Naive Bayes calculation for email arrangement and could add to the advancement of further developed email grouping frameworks. Moreover, the correlation of the Naive Bayes calculation with other famous calculations will give a thorough assessment of the qualities and Different scholastics have discussed how well the Naive Bayes Classifier and Secret Markov Model order email spam. In this review, Naive Bayes Classifier and Support Vector Algorithm are proposed as AI calculations to sort messages as spam or ham or in other classes in light of content-put together detection. A series of tests with respect to the messages will be utilized to try these recommended ways. The investigation's discoveries then, at that point, show which approach performs better while classifying spam messages utilizing a disarray grid. At last, a webmail will be sorted utilizing the best methodology. All in all, this examination plans to address the restrictions of conventional email characterization strategies and to investigate the capability of the Gullible Bayes calculation as a significant device for email order. The consequences of the review will give a more profound comprehension of the capacities of the Naive Bayes calculation and could add to the improvement of more viable and productive email characterization frameworks.

II. LITERATURE REVIEW

- Atom Borg's email grouping with AI is a review that assesses the utilization of AI calculations for email characterization, with a particular spotlight on the Molecule Borg calculation. Molecule Borg is a kind of Guileless Bayes calculation that has been explicitly produced for text order tasks. In this review, the Iota Borg calculation was prepared on a dataset of messages and used to characterize the messages into two classes: spam and not-spam and our classification is also based on these basics of categorization. The exhibition of the Particle Borg calculation was assessed utilizing different measurements, like exactness, accuracy and contrasted with other famous AI calculations, for example, Decision Trees and Support Vector Algorithms. All in all, the Iota Borg's email characterization with AI review gives significant bits of knowledge into the utilization of AI calculations for email arrangement and features the capability of the Particle Borg calculation for this undertaking. Wenjuan Li's Email Order Framework Utilizing Semi-Directed Learning is a review that zeroed in on involving semi-regulated learning methods for email classification
- In semi-directed learning, just a piece of the information is named, while the excess information is unlabeled. The point of the review is probably going to decide how well semi-directed learning strategies perform contrasted with customary managed learning methods in email arrangement errands. In this review, Wenjuan Li utilized a blend of named and unlabeled information to prepare a semi-regulated learning model for email characterization. The model would then be tried on a marked test set to assess its exhibition in grouping messages into various classifications like spam, ham, or significant. Timothy H. M. Ssebulime's Email Characterization Utilizing ML Strategies centers around utilizing AI (ML) methods for email classification
- In this review, Timothy H. M. Ssebulime analyzed the presentation of various ML calculations in arranging messages into various classifications like spam, ham, or significant. The review included the pre-handling and element extraction of email information, and the preparation and assessment of ML calculations on the pre-handled information. The presentation of the calculations looked at utilizing measurements like exactness, accuracy, review, and F1-score. Slam Gopal Raj's Email Characterization Exploration Patterns is a review that studies the current examination on email order, and presents an outline of the patterns and improvements in this field. The review included a deliberate survey of existing writing on email grouping, including a survey of the various procedures and calculations utilized, the datasets and assessment measurements utilized, and the difficulties and impediments of current methods
- The review has likewise remembered an examination of the patterns and improvements for the field, remembering ongoing advances for AI and profound learning calculations, and the utilization of large information and distributed computing. The consequences of the review gave an exhaustive outline of the present status of the exploration in email grouping, and featured regions for future examination and improvement. Cameron McGinley's CNN Advancement for Phishing Email Grouping utilizing Convolutional Neural Network (CNNs) for phishing email classification
- Phishing messages are deceitful messages that are intended to take delicate data, for example, passwords and Visa numbers, from clueless casualties. In this review, Cameron McGinley has meant to create an improved CNN model for precisely arranging phishing messages. The review included pre-handling and component extraction of email information, as well as the preparation and assessment of a CNN model on the pre-handled information. The presentation of the model might have been contrasted and other cutting edge calculations for phishing email characterization. The review has additionally investigated different hyperparameter advancement procedures for working on the exhibition of the CNN model. The aftereffects of the review could give experiences into the viability of CNNs for phishing email order, and add to the advancement of improved phishing email discovery frameworks. Ayodele, Zhou, and Khusainov gave a calculation to coordinate and sum up email messages utilizing the substance of email messages in another captivating study.

- Moreover, their technique proficiently processes approaching messages. The investigation of Gomes makes reference to investigate differentiating two techniques for sorting emails.
- The Secret Markov Model and Credulous Bayes were the two unmistakable techniques (Gee), which were utilized to decide the significance of an email (for example spam). The Gullible Bayes Classifier, they noticed, was faster and did well with a restricted dataset. Gee might deal with contributions of various lengths and help programs in settling on the most likely decision. Different mixes of stop-word evacuation, stemming, and lemmatization NLP approaches were tried for the two frameworks to analyze the exactness of the expectations. The aftereffects of the examination showed that Well offered more noteworthy exactness than Credulous Bayes. The effect of preprocessing on text characterization is analyzed by Alper and Serka with regards to different elements, including grouping precision, text area, text language, and aspect decrease. To do this, all blends of much of the time utilized preprocessing assignments are surveyed one next to the other on two particular spaces, to be specific email and news.
- The consequences of the investigations showed that, contingent upon the space and language appropriate for, legitimate mixes of preprocessing exercises instead of empowering or debilitating them totally may deliver a huge enhancement for characterization exactness.

III. PROPOSED SYSTEM ARCHITECTURE

3.1. Data Gathering and Collection

The most important phase in the strategy is information assortment. We will gather an enormous and various arrangement of messages, which incorporates different classes of emails, to guarantee that the Naive Bayes calculation has an adequate number of information to gain from. We will utilize openly accessible datasets like Enron-Spam and SpamAssassin for this reason.

3.2. Data Preprocessing

Whenever we have gathered the information, the subsequent stage is to preprocess the information. In this step, we will eliminate any unessential data from the messages like email headers and marks, and we will change over the messages into a standard organization like plain message. We will likewise perform tokenization to divide the messages into individual words and eliminate stop words, for example, "the" and "a", which don't add to the grouping of the email.

3.3. Feature Extraction

In the wake of preprocessing the information, we will extricate highlights from the messages. We will utilize a sack-of-words way to address each email as a vector of word frequencies. We will likewise utilize term frequency-inverse document frequency (TF-IDF) weighting to give more significance to uncommon words that are probably going to be more demonstrative of the email's classification.

3.4. Model Determination and Training

The subsequent stage is to choose a proper model for our characterization issue. We will utilize the Naive Bayes calculation, which is a probabilistic model that computes the likelihood of an email being into one of our classes, given its elements. We will prepare the model on a part of the dataset, utilizing cross-approval using other algorithms to guarantee that the model sums up well to new, inconspicuous information.

3.5. Model Evaluation

In the wake of preparing the model, we will assess its exhibition on a test set of messages. We will utilize assessment measurements, for example, exactness, accuracy to gauge the exhibition of the model. We will likewise look at the exhibition of the Naive Bayes calculation with other AI calculations, for example, Decision Trees and Support Vector Algorithm to figure out which calculation turns out best for our grouping issue.

3.6. Model Improvement

We additionally investigated methods for additional exactness and accuracy for Naive Bayes calculation, for example, including extraction and giving predefined number of element choice, to work on its exhibition on the arrangement task. This proposed framework is for the most part made out of five essential advances: information assortment, preprocessing, include extraction, email classification arrangement utilizing Naive Bayes Classifier lastly characterizing the on-line email.

IV. METHODOLOGY

Algorithms used for email classifier are-

4.1. Naive bayes

Naive Bayes is a probabilistic algorithm used for classification tasks in machine learning. It is based on the Bayes theorem which describes the probability of an event based on prior knowledge of conditions that might be related to the event. The Naive Bayes algorithm assumes that the features used to classify data are independent of each other, and this is why it is called "naive". It is a simple and efficient algorithm that can work well for a wide range of classification problems. The algorithm works by calculating the probability of each class given a set of input features, and then selecting the class with the highest probability as the predicted class. To calculate the probability of each class, the algorithm uses Bayes theorem, which is based on conditional probability:

P(class | features) = (P(features | class) * P(class)) / P(features)

where P(class | features) is the probability of a class given a set of input features, P(features | class) is the probability of observing the input features given a particular class, P(class) is the prior probability of the class, and P(features) is the probability of observing the input features.

4.2. Random forest

Random forest is a popular machine learning algorithm that can be used for both classification and regression tasks. It is an ensemble learning method that combines multiple decision trees to produce a more accurate and stable model.

The algorithm works by building a large number of decision trees, each using a random subset of the training data and a random subset of the features. These decision trees are then combined by taking the majority vote of their predictions (for classification problems) or the average of their predictions (for regression problems).

Overall, the random forest algorithm is a powerful and versatile machine learning method that can handle a wide range of problems and produce accurate and stable models. It is widely used in industry and academia for tasks such as image and speech recognition, text classification, and financial forecasting

4.3. XGBoost

XGBoost (Extreme Gradient Boosting) is a popular machine learning algorithm that is used for regression, classification, and ranking tasks. It is an optimized implementation of the gradient boosting algorithm that uses a combination of decision trees and gradient boosting to improve accuracy and reduce bias and variance.

The algorithm works by building a large number of decision trees, each learning from the residual errors of the previous tree. The trees are built in a greedy manner, selecting the split that produces the largest decrease in the objective function (e.g., mean squared error for regression, log loss for classification).

In addition to building decision trees, XGBoost also includes a regularization term that penalizes large weights and helps to reduce overfitting. This regularization term can be controlled by hyperparameters such as the learning rate, maximum depth of each tree, and the number of trees in the model.

One of the key features of XGBoost is its ability to handle missing values and categorical features. It can automatically learn how to treat missing values and convert categorical features into numerical representations that are optimized for the objective function.

4.4. Natural Language Processing(NLP)

Natural Language Processing (NLP) is a branch of artificial intelligence (AI) that focuses on the interaction between computers and human language. NLP algorithms are designed to help computers understand and interpret human language and to perform tasks such as language translation, sentiment analysis, chatbot development, and more.

Overall, NLP algorithms play an important role in enabling computers to understand and interpret human language. As natural language processing continues to advance, we can expect to see even more sophisticated and powerful applications of this technology in the future.

4.5. Support Vector Algorithm

The Support Vector Machine (SVM) algorithm is a popular machine learning algorithm used for classification and regression analysis. It is a binary classifier, which means it can only classify data into two classes, but can be extended to multi-class classification using various techniques.

The main idea behind SVM is to find the hyperplane that separates the data points in the feature space with the largest margin. The hyperplane is selected to maximize the distance between the data points closest to the hyperplane, which are called support vectors. By maximizing the margin, the SVM algorithm can better generalize to new data and reduce overfitting.

SVMs can be used with different types of kernels, such as linear, polynomial, and radial basis function (RBF) kernels. Kernels transform the input features into a higher-dimensional space, making the data more separable. The choice of kernel depends on the characteristics of the data and the problem being solved.

SVMs are also capable of handling non-linearly separable data by using a kernel trick, which transforms the data into a higher-dimensional space where it becomes linearly separable. The kernel trick can be computationally expensive, but it allows SVMs to handle complex data and achieve high accuracy.

V. OBSERVATION

5.1 Data

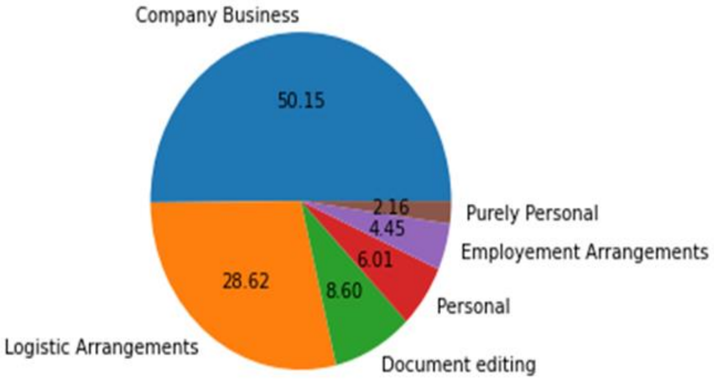
Data taken for the model is labeled email data. Emails belonging to the same class are in the same folder containing each email in csv format. 1663 emails are extracted from 6 folders. Each class is mapped from 0 to 5.

From each email its subject and body is extracted using two functions with the help of an email library. The email classifier focuses on the subject and body of the email only.

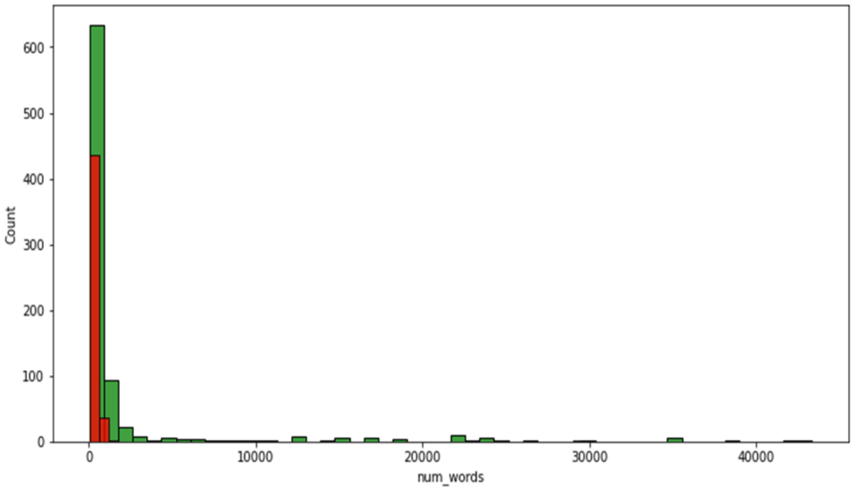
Classes	Labels
Company Business	0
Document Editing	1
Employment Arrangements	2
Logistic Arrangements	3
Personal	4
Purely Personal	5

5.2. Data Exploration

After checking for duplicates and deleting them, check for the number of each class in the dataset. The pie-chart given below depicts the percentage of all classes of emails in the given dataset.



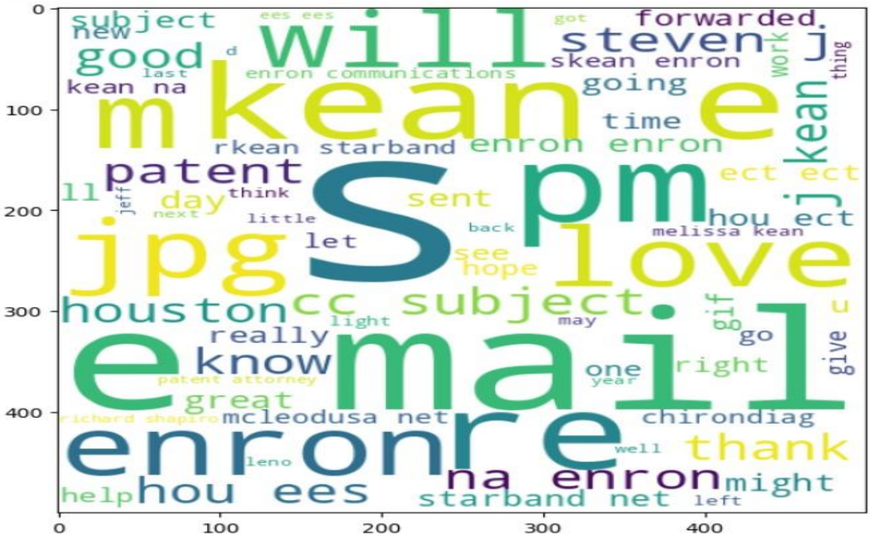
Checking for relation between number of words in emails in each class. Plotting histogram for email word counts for various classes



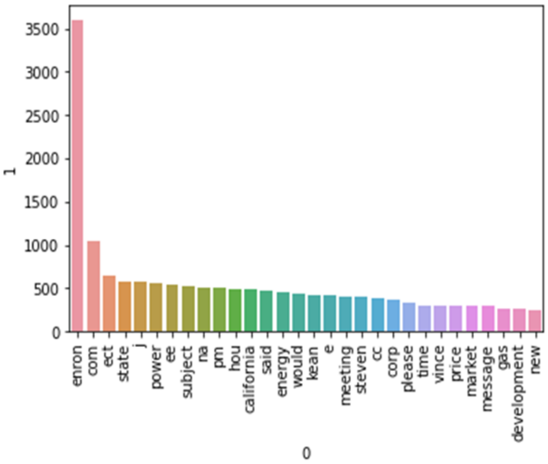
- There was no clear indication which shows that different class emails have different word counts.

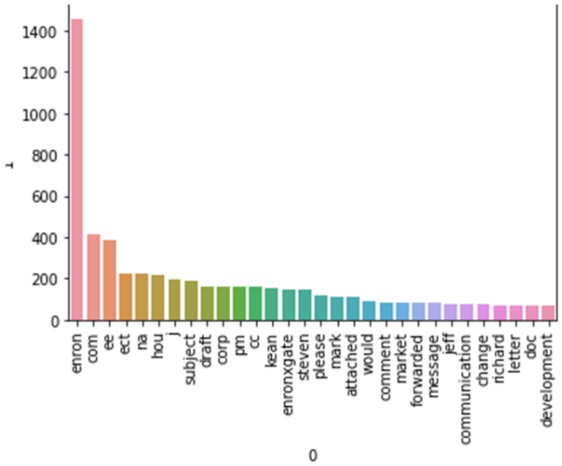
Removing stop words and applying Lemmatization

```
<matplotlib.image.AxesImage at 0x7f4018577250>
```



- Noticed that in company business city names such as Los Angeles, California are coming a lot and in case of personal email emotional words such as love, really, like is more often.
- Noticed logistic arrangement has more words related to meeting and places and document editing has more words related to draft, attached.





5.3. Model Building

5.3.1 Feature representation:

Using word embedding technique Bag of Words and TFIDF.

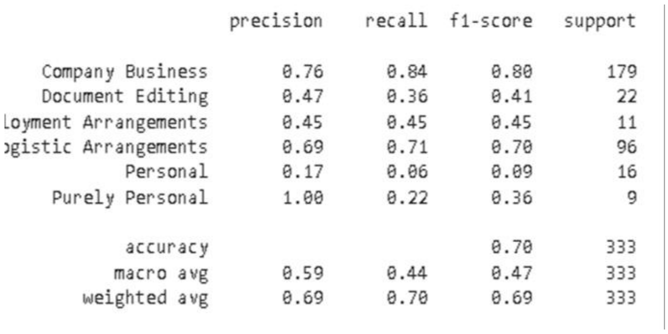
Models used:

- Naive Bayes: Works well with text data
- Random Forest: Better with imbalanced dataset
- XGBoost: Boosting technique to increase the performance
- SVM: Works good with large number of features

5.3.2 Evaluation:

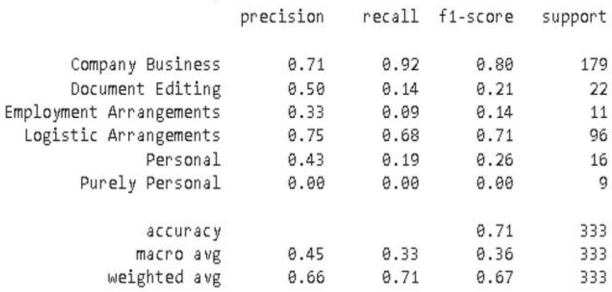
Since the dataset is imbalanced, use macro-F1 score for evaluation of models. Macro-F1 score is computed using the arithmetic mean of all the per-class F1 scores.

- Performance of Naive Bayes



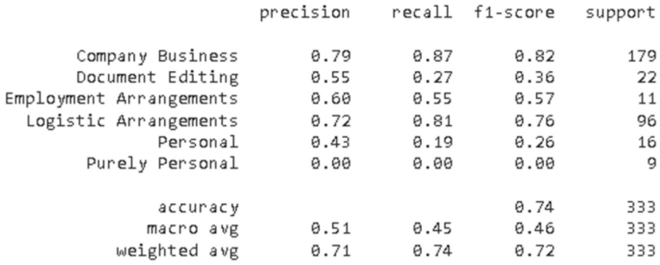
```
                      precision    recall  f1-score   support

    Company Business       0.76      0.84      0.80       179
    Document Editing       0.47      0.36      0.41        22
loyment Arrangements       0.45      0.45      0.45        11
ogistic Arrangements       0.69      0.71      0.70        96
            Personal       0.17      0.06      0.09        16
     Purely Personal       1.00      0.22      0.36         9

            accuracy                           0.70       333
           macro avg       0.59      0.44      0.47       333
        weighted avg       0.69      0.70      0.69       333
```

- Performance of Random Forest Classifier



```
                         precision    recall  f1-score   support

       Company Business       0.71      0.92      0.80       179
       Document Editing       0.50      0.14      0.21        22
Employment Arrangements       0.33      0.09      0.14        11
  Logistic Arrangements       0.75      0.68      0.71        96
               Personal       0.43      0.19      0.26        16
        Purely Personal       0.00      0.00      0.00         9

               accuracy                           0.71       333
              macro avg       0.45      0.33      0.36       333
           weighted avg       0.66      0.71      0.67       333
```

- Performance of XGBoost



```
                          precision    recall  f1-score   support

        Company Business       0.79      0.87      0.82       179
        Document Editing       0.55      0.27      0.36        22
 Employment Arrangements       0.60      0.55      0.57        11
   Logistic Arrangements       0.72      0.81      0.76        96
                Personal       0.43      0.19      0.26        16
         Purely Personal       0.00      0.00      0.00         9

                accuracy                           0.74       333
               macro avg       0.51      0.45      0.46       333
            weighted avg       0.71      0.74      0.72       333
```

Since all the models have low F1-score, imbalanced dataset problems needed to be addressed.

5.4. Solving Imbalanced Dataset Problem

5.4.1. Use of cost-sensitive Random Forest:

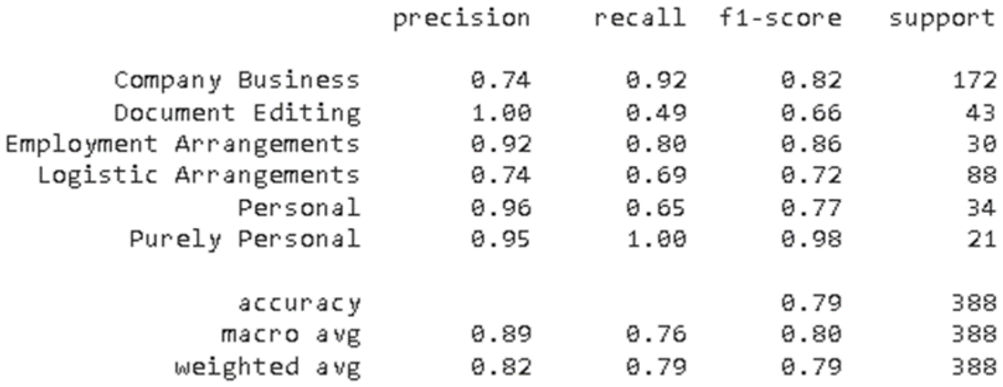
Made Random Forest Classifier bias towards those classes that have fewer examples in the training dataset. No significant improvement in the performance

5.4.2. Use of other word Embedding Technique: Made Random Forest Classifier bias towards those classes that have fewer examples in the training dataset. No significant improvement in the performance.

5.4.3. Use of Text Augmentation Technique to increase data size of minority classes:

Used nlpaug library to artificially increase training data size of minority group by generating different versions of real datasets without actually collecting the data.

Classes	Labels	Count Before Augmentation	Count After Augmentation
Company Business	0	834	834
Document Editing	1	143	193
Employment Arrangements	2	74	149
Logistic Arrangements	3	476	476
Personal	4	100	175
Purely Personal	5	36	111

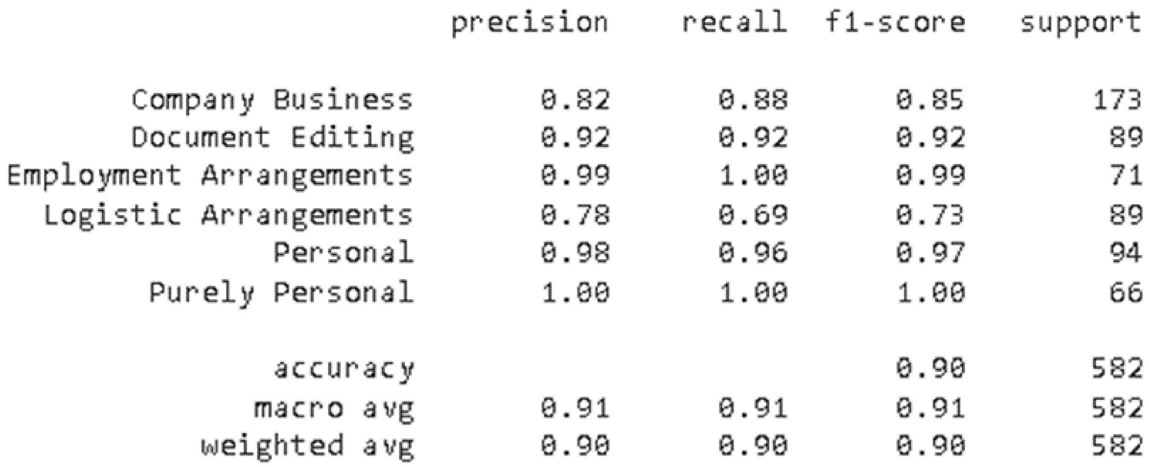


```
                          precision    recall  f1-score   support

        Company Business       0.74      0.92      0.82       172
        Document Editing       1.00      0.49      0.66        43
 Employment Arrangements       0.92      0.80      0.86        30
   Logistic Arrangements       0.74      0.69      0.72        88
                Personal       0.96      0.65      0.77        34
         Purely Personal       0.95      1.00      0.98        21

                accuracy                           0.79       388
               macro avg       0.89      0.76      0.80       388
            weighted avg       0.82      0.79      0.79       388
```

Random Forest Classification Report (macro-F1 score : 0.80) : Significant improvement in F1 Score from 0.36 to 0.80.

5.4.4. Use of Resampling technique

Resampling Method is a tool consisting in repeatedly drawing samples from a dataset. It increased the number of minority classes to 400.

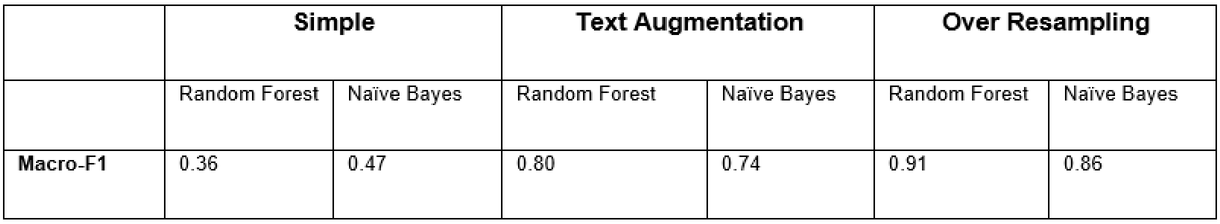


```
                          precision    recall  f1-score   support

        Company Business       0.82      0.88      0.85       173
        Document Editing       0.92      0.92      0.92        89
 Employment Arrangements       0.99      1.00      0.99        71
   Logistic Arrangements       0.78      0.69      0.73        89
                Personal       0.98      0.96      0.97        94
         Purely Personal       1.00      1.00      1.00        66

                accuracy                           0.90       582
               macro avg       0.91      0.91      0.91       582
            weighted avg       0.90      0.90      0.90       582
```

Random Forest Classification report (macro-F1 score: 0.91) : Significant improvement in F1 Score

5.4.5 Results

Using text augmentation technique and over resampling technique. Performance of Naïve Bayes and Random Forest model significantly increased.



	Simple		Text Augmentation		Over Resampling	
	Random Forest	Naïve Bayes	Random Forest	Naïve Bayes	Random Forest	Naïve Bayes
Macro-F1	0.36	0.47	0.80	0.74	0.91	0.86

VI. CONCLUSION

In this study, we have investigated the utilization of the Naive Bayes, Random Forest, XGBoost, SVM,NLP algorithm for email characterization into 6 different classes which will prove beneficial to the corporate and other kinds of businesses. Our strategy included gathering an enormous and different arrangement of email messages, preprocessing the information to eliminate superfluous data, separating highlights from the messages, choosing and preparing the model on basis of all the algorithms previously mentioned, assessing its exhibition on a test set of messages, and streamlining the model to work on its presentation. Our outcomes show that the combination of all the algorithms is a viable methodology for email classification, accomplishing high precision, accuracy on the test set. Moreover, our correlation with other AI calculations uncovered that the score continuously kept increasing as we continued to use all the algorithms step by step and also outperformed basic algorithms like Decision Trees and SVM used separately.

VII. REFERENCES

[1] 'Bayes and Naive-Bayes Classifier' by Computer Science & Engineering Rajiv Gandhi University of Knowledge Technologies Andhra Pradesh, India

[2] 'Recent Trends in Deep Learning Based Natural Language Processing' by School of Information and Electronics, Beijing Institute of Technology, China

[3] 'Selecting critical features for data classification based on machine learning methods' by Rung-Ching Chen, Christine Dewi, Su-Wen Huang & Rezzy Eko Caraka, Journal of Big Data

[4] 'Supervised Machine Learning: A Review of Classification Techniques' by S. B. Kotsiantis Department of Computer Science and Technology University of Peloponnese, Greece

[5] 'Supervised Machine Learning Algorithms: Classification and Comparison' by Department of Computer Science, Babcock University, Ilishan-Remo, Ogun State, Nigeria

[6] 'Machine Learning - Algorithms, Models and Applications' by Jaydip Sen, IntechOpen Series Artificial Intelligence, Volume 7

[7] 'Supervised Learning Algorithms of Machine Learning: Prediction of Brand Loyalty' by International Journal of Innovative Technology and Exploring Engineering (IJITEE)

[8] 'Machine Learning Algorithm for Classification' by College of Letter and Science, University of California, Davis, One Shields Avenue, Davis, CA, 95616, USA

[9] 'Abstract Classification Using Support Vector Machine Algorithm (Case Study: Abstract in a Computer Science Journal)' by Department of Computer Science, Faculty of Computer Science, University of Lampung, Bandar Lampung, Indonesia

[10] 'Natural Language Processing: A Historical Review' by Karen Sparck Jones Computer Laboratory, University of Cambridge

[11] 'Effective Learning and Classification using Random Forest Algorithm' by International Journal of Engineering and Innovative Technology (IJEIT) Volume 3

[12] 'Recent Advances in Convolutional Neural Networks' by School of Computer Science and Engineering, Nanyang Technological University, Singapore

[13] 'XGBoost: A Scalable Tree Boosting System' by Tianqi Chen University of Washington, Carlos Guestrin University of Washington.