Authorship Identification On Enron Email Dataset

CSC 522 - Team 17

Abhishek Gupta
akgupta3@ncsu.edu
North Carolina State University

Neel Shah
npshah6@ncsu.edu
North Carolina State University

Shaival Shah
sshah35@ncsu.edu
North Carolina State University

Vishwa Gandhi
vgandhi@ncsu.edu
North Carolina State University

1 INTRODUCTION AND BACKGROUND

Emails have become an integral part of our daily lives and a popular means of communication in the digital world. However, with the rise of email-based cybercrimes such as phishing, spamming, and identity theft, users must remain vigilant when using email services. Our project aims to develop a machine learning-based solution that can accurately identify the author of an email. By leveraging advanced algorithms and statistical techniques, we can train our model to detect patterns in the email's content, structure, and metadata, enabling us to identify the sender of an email with high accuracy. This solution can significantly mitigate the risks associated with email-based cyber crimes by allowing users to verify the authenticity of the sender, prevent unauthorized access to user accounts, and minimize the spread of malware through suspicious emails.

1.1 Problem Statement

Our project is a critical task that aims to identify the author of a given piece of text. This problem has significant implications in various domains such as law enforcement, journalism, and forensic analysis. By analyzing the writing style, use of specific words, phrases, and punctuation, machine learning models can be trained to predict the author of an unknown text accurately.

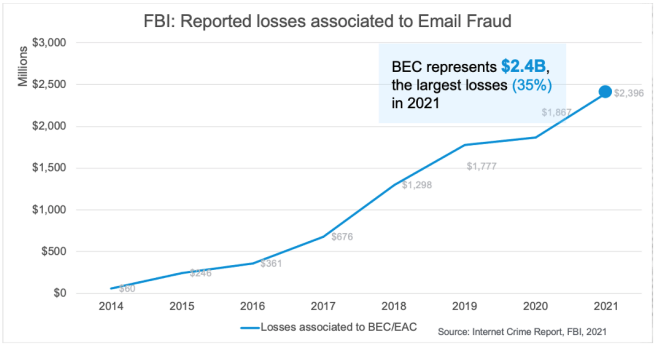

Figure 1: Why EAC is a big problem

Email Account Compromise (EAC) has become a profitable venture for cybercriminals and is rapidly expanding. The FBI reported significant losses due to EAC, as shown in Figure. This highlights the urgent need for heightened awareness and preventive measures to combat EAC attacks and mitigate their detrimental impact on both individuals and businesses.

In our project, we aim to perform author identification on the Enron dataset, which is a vast collection of over 500,000 emails generated by approximately 150 employees of the Enron Corporation. The company collapsed due to fraudulent activities, which makes this dataset particularly interesting and relevant. By analyzing all the emails sent by the employees, we plan to train our model classifier to learn about the specific writing style of each author. This process will involve several pre-processing tasks such as cleaning metadata, lemmatization, and stop word removal, to achieve higher accuracy. The motivation behind our project is to identify the original sender of emails based on their writing style. This approach can help in protecting user privacy, preventing identity theft, and reducing the spread of fake emails. By accurately identifying the author of emails, our proposed solution can contribute to detecting fraudulent activities and mitigating the risks associated with email-based cybercrimes.

1.2 Related Work

The paper titled "Detection of E-Mail Phishing Attacks - using Machine Learning and Deep Learning" by Rathee and Mann [1] is useful in email authentication as it provides an overview of the current state of the art in email-based phishing detection using machine learning techniques. The authors review several different approaches to phishing detection, including feature-based methods (which extract specific features from emails to identify phishing attempts) and content-based methods (which analyze the content of emails to identify phishing attempts). This paper is helpful for identifying relevant features to include in an email authentication model and for understanding the limitations and challenges associated with email-based cybercrime detection.

The paper by de Vel titled "Mining e-mail authorship" [6] is useful in the email authentication project because it deals with the identification of authors of emails. In the paper, de Vel used Support Vector Machines (SVMs) for authorship identification by using various linguistic features such as punctuation, function words, and character n-grams. He also explored the impact of different feature selection methods on the performance of the classifier. This study provides valuable insights into the use of SVMs for authorship identification in emails, which is helpful in our project. The findings of the paper were also useful in the selection of appropriate features for training the model and enhance its accuracy in predicting the authorship of emails.

The paper titled "Stylometric Analysis for Authorship Attribution on Twitter" [7] demonstrates how stylometric analysis can be applied to social media data (specifically Twitter) for authorship attribution. The paper highlights the importance of feature engineering in stylometric analysis, which is helpful for the our project as we aim to extract meaningful features from the emails for authorship identification.

2 PROPOSED METHODOLOGY

2.1 Approach

Initially, we conducted a thorough review of relevant literature on machine learning techniques for authorship and mining email tasks. The existing body of research has identified several algorithms that could be suitable for our project, such as Support Vector Machines, Decision Tree, K-Nearest Neighbors, Naive Bayes, and LSTM. We plan to apply these algorithms to our dataset and analyze their performance on the Enron Corpus, specifically on the features we have extracted. To achieve optimal results, we will spend time fine-tuning these models by optimizing their hyperparameters. Specifically, we will explore the efficacy of below four models: Random Forest Classifiers (RFC), support vector classifier (SVC), multinomial Naive Bayes, and logistic regression along with word2vec.

(1) Random Forest Classifier (RFC):- The RFC approach is a type of ensemble learning algorithm that constructs a multitude of decision trees at training time and outputs the mode of the classes as the prediction. Each decision tree in the RFC is trained on a random subset of features and data, resulting in a diverse set of trees that can capture different aspects of the data. The final prediction of the RFC is based on the majority vote of all the decision trees in the forest. With respect to the context of our project, RFC is used to identify the unique writing style of each author by analyzing various features such as word usage, sentence structure, and punctuation. The novelty of this approach lies in its ability to handle a large number of features and non-linear relationships between the features and the target variable. Furthermore, RFC can handle missing data and noisy data, making it robust to the irregularities and inconsistencies that are common in email data.
(2) Support Vector Classifier (SVC):- The SVC is used to create a boundary, also known as a hyperplane, that separates different classes of data points in a high-dimensional space. In our project, the SVC algorithm is used to classify emails based on the writing style of their respective authors. The novelty of this approach lies in the fact that we used a combination of various text features to train our SVC model, including the frequency of specific words and phrases, as well as punctuation usage. The SVC model was then trained on the training set using various hyperparameters and cross-validation techniques to optimize the model's accuracy. Once the SVC model was trained, we used it to predict the author of previously unseen emails.
(3) Multinomial Naive bayes :- It is a probabilistic machine learning algorithm that uses Bayes' theorem to make predictions. It is commonly used for text classification tasks, such as email authorship attribution, because of its simplicity and efficiency. In our project, the algorithm works by first creating a vocabulary of all the unique words in the dataset. Then, it calculates the frequency of each word in each email and uses this information to create a bag-of-words representation of the text. The algorithm then builds a probabilistic model based on the frequency of each word in each class, where the class represents the author of the email. This model is then used to predict the author of a new email by calculating the probability of each possible author and choosing the one with the highest probability. The novelty of this approach lies in its ability to handle large amounts of text data efficiently and to work well with sparse data. It also has the advantage of being easy to implement and interpret, which makes it a popular choice for text classification tasks.
(4) Logistic Regression + Word2vec:- Logistic regression is a statistical approach used for binary classification tasks, where the goal is to predict a binary outcome (e.g., true/false, yes/no) based on a set of input features. Word2vec is a neural network-based technique that transforms words into vectors in a high-dimensional space, where words with similar meanings are clustered together. In our project, logistic regression is used to predict the author of an email based on various linguistic features, such as word frequency, punctuation, and sentence structure. Word2vec allows the model to capture semantic relationships between words and is used to identify the unique writing style of individual authors. The novelty of combining logistic regression with word2vec for email authorship lies in the ability to capture both semantic and syntactic features of text data.

2.2 Rationale

In the approach we described a short summary of the methods we chose to experiment with. Here, we explain our reasons for selecting these particular methods. Our project involves a classification task where our goal is to predict the author of an email from a group of possible authors. As a result, we have researched and implemented algorithms commonly used for classification. The following paragraphs will explain in detail our evaluation of the appropriateness of the algorithms we intend to use.

(1) Random Forest Classifier (RFC) :- It is a suitable approach for several reasons. Firstly, RFC can handle a large number of features and observations, which is ideal for our dataset that consists of more than 25 features. Secondly, RFC can capture complex non-linear relationships between the features and the target variable, which is beneficial for our task of predicting the authorship of emails. Thirdly, RFC can handle missing values and outliers, which are common issues in real-world datasets. Compared with other possible approaches like multinomial naive Bayes, logistic regression, and SVM, RFC has some advantages and disadvantages. RFC can handle non-linear relationships between features and the response variable, and can work well even with missing data. It is also relatively fast to train and can scale well with large datasets. However, RFC may suffer from instability

and lack of interpretability due to its complex structure and dependence on random subsampling.

(2) Support Vector Classifier (SVC):- It is an appropriate machine learning technique for our project as it works well with high-dimensional data, such as the feature set for our project which initially had more than 25 features. SVCs are known for their ability to handle non-linear data and can handle multiple features simultaneously. Additionally, SVCs only rely on support vectors for hyper-plane computation, which makes it less memory-intensive than other models. Compared to multinomialnb, which assumes that the features are independent, SVC can handle non-linear relationships between features and can work well with high-dimensional data. RFC can also handle high-dimensional data, but it may suffer from overfitting with noisy data, which is common in the Enron dataset. Logistic regression, on the other hand, assumes a linear relationship between features and may not be able to capture the complex relationships that exist in the Enron dataset.

(3) Multinomial Naive Bayes:- It is an appropriate machine learning approach for email authorship attribution due to its ability to handle sparse data and its efficiency in training and prediction. It assumes that the features are independent and follows a multinomial distribution, which is suitable for text data. It works by computing the conditional probability of each word given the author and then multiplying these probabilities to obtain the probability of a given document being authored by a particular person. Compared to other possible approaches MultinomialNB is computationally efficient and has a lower memory requirement. It also works well with small training datasets and is relatively simple to implement. Additionally, as seen previously, MultinomialNB has shown to perform well in natural language processing tasks such as text classification and sentiment analysis.

(4) Logistic Regression + word2vec:- It is a suitable approach for our project as it uses the power of word embeddings to capture the semantic meaning of the text. The approach combines the advantages of logistic regression, such as its simplicity and interpretability, with the benefits of word2vec in understanding the context of words in the text. Additionally, it has a lower risk of overfitting compared to other complex models such as SVC. Compared to SVC, logistic regression has lower computational complexity, making it more suitable for large datasets.It has also shown promising results in previous studies in similar text classification tasks. However, it may not perform as well as random forest or SVC on datasets with a large number of features or noisy data. Overall, logistic regression with word2vec is a viable approach for this dataset, especially for its ability to capture the semantic meaning of the text and its simplicity in implementation and interpretation.

3 EXPERIMENT

3.1 Dataset

The Enron email dataset is a massive collection of emails, created by employees of Enron, one of the largest energy and commodities trading companies in the US. The dataset contains over 500,000 emails that were sent between Enron employees between 1999 and 2002 and is around 1.7 GB in size. The emails cover a wide range of topics, including business strategy, financial performance, personal relationships, and more. This dataset is notable for being one of the largest publicly available collections of emails, and for its role in the Enron scandal, which involved widespread corporate fraud and corruption. In addition to its size and diversity, the Enron email dataset is also notable for its metadata, which includes information such as email timestamps, sender and recipient addresses, and email folders. This metadata can be useful for author identification research, as it can provide additional context for analyzing the language samples. The Enron email dataset has been used extensively in research on author identification, including studies that have explored the effectiveness of various machine learning algorithms and linguistic features for accurately identifying authors of texts. Researchers have also used the dataset to investigate the impact of factors such as email length, frequency, and genre on the accuracy of author identification methods. Overall, the Enron email dataset represents a valuable resource for researchers and practitioners seeking to improve author identification methods.

The Enron Email dataset includes several folders with specific names, such as _sent_mail, _sent, and sent_items, that are located under directories named after various authors. Each folder contains emails sent by that particular author, separated into individual files. The goal of this task was to produce a dataset with folder names as labels and extract features from all the emails in the different sent folders for each author directory.

We selected 20 authors with the highest number of sent emails and extracted all the emails from their sent folders, creating a preliminary dataset that we saved as a CSV file named enron.csv. We used random stratified sampling for all subsequent methods as the distribution of sent emails was uneven among the authors.

To prepare the emails for feature extraction, we took several steps to clean up the data and focus solely on the text. Firstly, we removed all non-alphanumeric characters, such as punctuation marks or special symbols. Secondly, we applied lemmatization to reduce the different inflected forms of words to a single base form, such as converting "walked," "walking," or "walks" to "walk." Lastly, we eliminated stop words, including commonly used words like "the," "of," and "and," which do not contribute much to classification. By applying these steps, we were able to reduce the noise in the data and extract more meaningful data from the emails. This helped us to prepare the data for feature extraction, which is an important process in author identification.

3.2 Hypothesis

The primary question that we try to answer via our project is to determine whether the author of an email can be accurately determined using this dataset or not. We also try to test one more hypothesis. It is widely believed and usually seen that for emotion prediction, TF-IDF is better than CountVectorizer [5]. Therefore, another research goal is to compare the performance of TFIDF Vectorizer and Countvectorizer across various classification models. We hypothesize that TFIDF Vectorizer will outperform Countvectorizer due to its emphasis on the significance of a word's frequency

in the corpus. While seemingly simple, we believe this objective will help lay the foundation for future objectives and research work.

3.3 Experimental Design

In order to validate our hypothesis, we must undertake several crucial steps, ranging from data collection to feature generation, before training our machine learning model. The process we have adopted for this project consists of the following steps:

(1) Data Extraction and Cleaning:- The process of Data Extraction is a crucial step in our processing pipeline. It involves converting unstructured data into a structured format that can be easily processed, facilitating the generation of new features. As discussed in 3.1 Dataset, we have produced a dataset with folder names as labels and extracted features from all the emails in the different sent folders for each author directory. Now we have our data ready to perform cleaning operations on them. Initially, we removed all non-alphanumeric characters, such as punctuation marks or special symbols. Subsequently, we applied lemmatization to transform the different inflected forms of words into a single base form, such as converting "walked," "walking," or "walks" to "walk." Lastly, we removed stop words, including frequently used words like "the," "of," and "and," which don't provide any useful information for author identification. These steps aided in reducing noise in the data and extracting more meaningful information from the emails. This process is essential in author identification as it helps in preparing the data for feature generation.

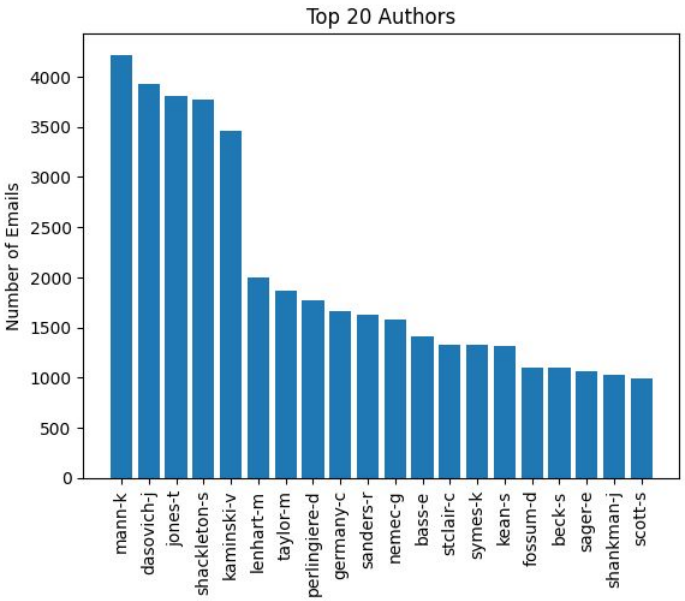

Figure 2: Email distribution among users

(2) Feature Generation:- Stylometric features are essential for email classification. However, email classification differs from document classification in two significant ways: emails are shorter than documents, and there is less variation in topics. Since each user has a unique writing style, there is no single distinctive feature that can differentiate users; rather, it is a combination of features. After conducting a literature review and examining emails, we identified the following feature categories:

- Paragraph based features:
  - Number of Paragraphs
  - Average Sentences per Paragraph
- Word based features:
  - Farewell Words
  - Average Word Length
  - Average Sentence Length
  - Short Word Ratio
  - Most Common Word
  - Freq Most Common Word
- Punctuation based features:
  - freq_punc
  - Last_punc
  - Punc Frequency
  - Punc after Greeting
- Special Character Base:
  - Total Special Character Count
  - Max Occurring Special Char
  - Count of Max Special Char
- Author based Features:
  - Email Length
  - Greeting
  - Subjectivity
  - Polarity
  - Most Common POS
  - Single Sentence

(3) Vectorizing Text:- After completing the above steps, we also test our hypothesis that TF-IDF is better than Countvectorizer. For this, we do feature selection using Select K Best Features while comparing training and validation accuracies.

(4) Model Training:- The data was then split into 80:20 ratio for training and testing. The training data was then trained using MultinomialNB and Random Forest classifiers.

4 RESULTS

4.1 Model Selection

After extracting all emails from the top 20 authors with the highest email frequency, we generated a basic dataset and saved it as a CSV file named "enron.csv". Upon analyzing the distribution of sent emails amongst these authors, we observed that it was non-uniform. As a result, we recognized the need for random stratified sampling in all subsequent procedures. Later we applied various different models like multinomialNB ,random forest classifier, and support vector classifier using Count Vectorizer and the results obtained are as follows:-

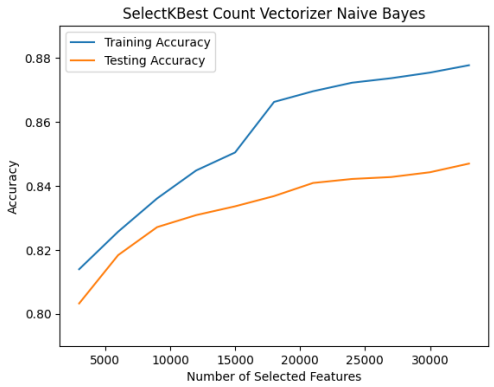

Figure 3: Model training using Count Vectorizer

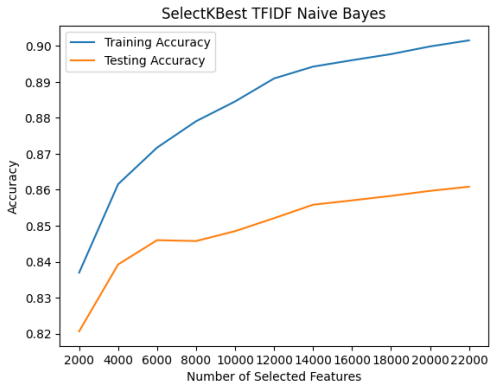

Figure 4: Model training using TF-IDF Vectorizer

Models using Count Vectorizer	
Models Used	Accuracy
MultinomialNB	80.32%
Random Forest Classifier	86.14%
Support Vector Classifier	83.22%

The best accuracy which was obtained using the count vectorizer was of support vector classifier which was 86.14% . Than we tried using this model along with the TF-IDF vectorizer. Below table suggests the results obtained from it.

Models using TF-IDF Vectorizer	
Models Used	Accuracy
MultinomialNB	81.92%
Random Forest Classifier	86.66%
Support Vector Classifier	84.83%

Logistic regression using word2vec	
Models Used	Accuracy
Training accuracy	74.47%
Testing accuracy	73.78%

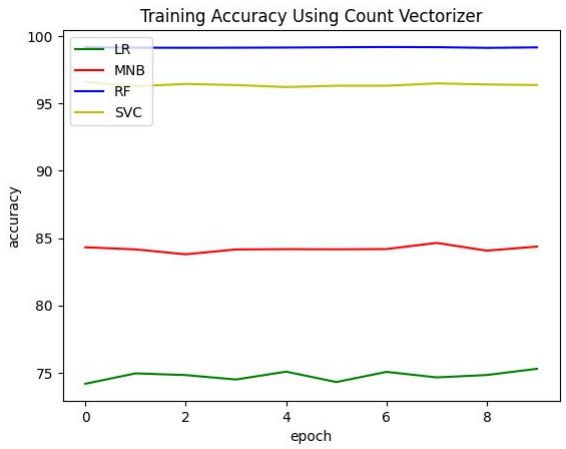

Figure 5: Model training accuracy using Count Vectorizer

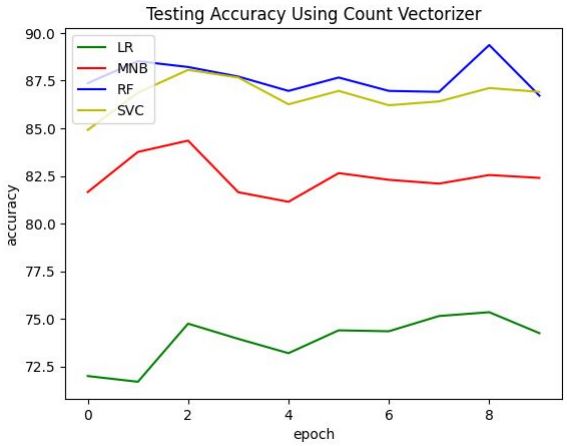

Figure 6: Model testing accuracy using Count Vectorizer

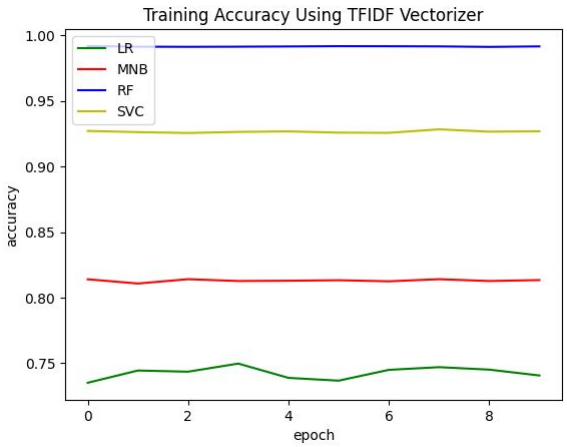

Figure 7: Model training accuracy using TFIDF Vectorizer

, ,

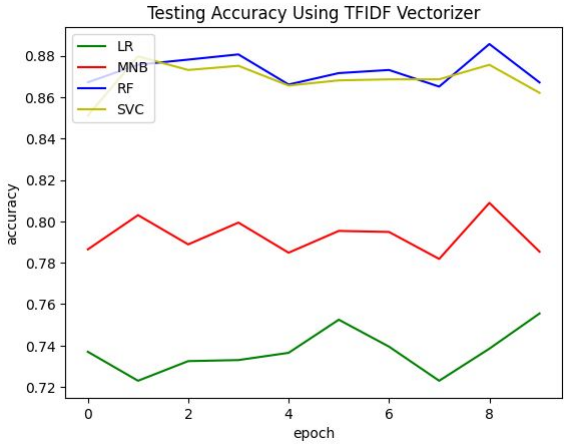

Figure 8: Model testing accuracy using TFIDF Vectorizer

Our experiment has revealed that the implementation of a random forest classifier with 10 k-fold has yielded the most impressive results to date. To identify the most appropriate hyperparameters for this classifier, we intend to identify the configuration that produces the highest Area Under Curve (AUC) value for the Receiver Operating Characteristic (ROC). This necessitates a thorough analysis of the AUC value for varying hyperparameter values, with the objective of selecting the most optimal one.

We then used word2vec to convert the text content of the emails into numerical vectors that were used as input features for the logistic regression model. The training accuracy for 10 rounds came around 74.47% and testing accuracy came around 73.78%.The training accuracy and testing accuracy results suggest that the logistic regression model with word2vec achieved an accuracy that was lower than the random forest classifier for the enron dataset. After

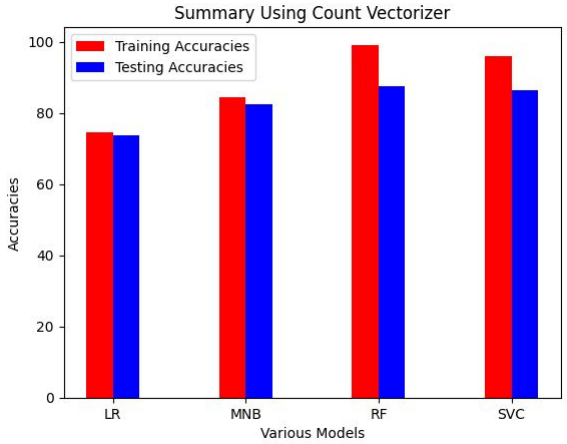

Figure 9: Summary using Count Vectorizer

comparing the performance of Count Vectorizer and TF-IDF Vectorizer for various models, it was observed that TF-IDF Vectorizer outperformed Count Vectorizer. Specifically, the highest accuracies were achieved using TF-IDF Vectorizer in conjunction with the Random Forest Classifier. This result aligns with the hypothesis that TF-IDF Vectorizer, which considers both term frequency and inverse document frequency, would perform better than Count Vectorizer. Although the accuracy results of Random Forests and Support Vector Classifier with TF-IDF were comparable, the F1 score of Random Forest Classifier was found to be better than that of SVC. Consequently, it was decided to take into account F1 score metrics in addition to accuracy when selecting the best performing model. Based on this evaluation, Random Forest Classifier was determined to be the most effective model.

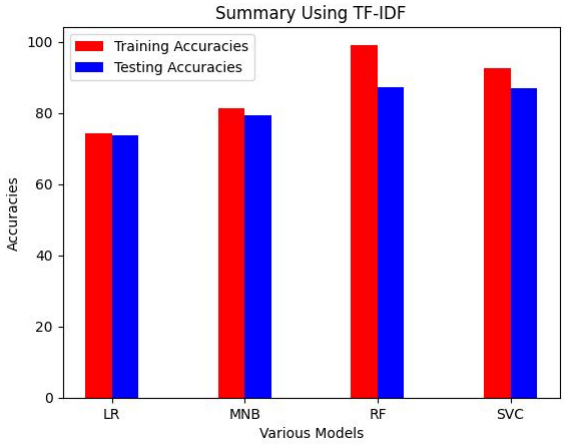

Figure 10: Summary using TFIDF Vectorizer

The summary of all the models using TFIDF Vectorizer and Count Vectorizer can be shown as above.

4.2 Model Evaluation

Model evaluation is a crucial step in building machine learning models as it helps in determining the effectiveness of the models.We evaluated multiple machine learning models such as MultinomialNB, random forest classifier, support vector classifier, and logistic regression using word2vec to classify emails from the Enron dataset. The results obtained from evaluating the different machine learning models using various metrics such as accuracy, precision, recall, F1 score, and ROC curve suggest that the random forest classifier with Tfidf vectorizer achieved the best performance for our dataset. The random forest classifier with TF-IDF vectorizer achieved an accuracy of 88.67%, which is the highest among all the models evaluated. The model also achieved high precision and recall scores, indicating that it was able to correctly identify positive cases and minimize false positives and false negatives. The F1 score of the model was also high, indicating a good balance between precision and recall. In addition, the confusion matrix for the random forest classifier with Tfidf vectorizer showed that the model correctly classified a large number of emails, with only a few misclassifications. This indicates that the model was able to effectively learn the patterns in the data and make accurate predictions. The confusion matrix for it is shown as belows. Overall, these results suggest that the random forest classifier with Tfidf vectorizer is a highly effective model for

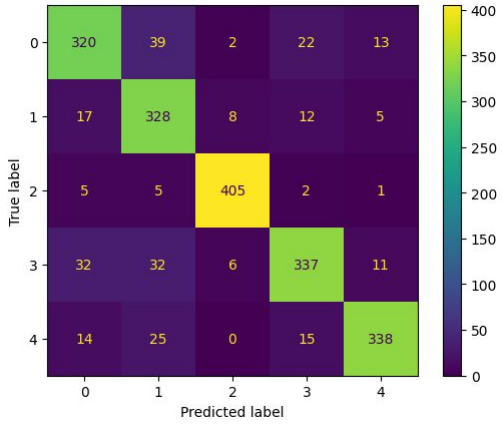

Figure 11: Confusion Matrix for RFC using TF-IDF Vectorizer

our dataset. The model achieved high accuracy, precision, recall, and F1 score, and was able to correctly classify a large number of emails.

5 DISCUSSION

This project aimed to identify authors based on their content using various machine learning models. We used a combination of data preprocessing, feature engineering, and hyperparameter tuning techniques to optimize the performance of the models. The models were then evaluated using various metrics such as accuracy, precision, recall, F1 score, and ROC curve, and the results were compared to determine the most effective model. Additionally, we conducted a comparison of our results with prior work, which also achieved similar accuracy levels in identifying authors based on their writing style. Our study's results are consistent with other studies that have shown that random forest classifier can be used to accurately identify authors based on their writing style. Our study builds on the prior work by comparing the performance of two different text vectorization techniques, which can provide valuable insights for future research in this area. [2] [5]

The prior work used the random forest classifier for email classification. In that work, the author used the random forest algorithm to classify emails into four categories: personal, promotional, social, and spam. They used various text preprocessing techniques such as tokenization, stemming, stop-word removal, and feature selection using mutual information. Their results showed that the random forest classifier achieved an accuracy of 96.25%, which was better than other classifiers they tested. In our hypothesis, we planned to build on this work by comparing the performance of two vectorizers - TF-IDF and count vectorizer - with various different models. Our hypothesis was that the TF-IDF vectorizer would outperform the count vectorizer due to its emphasis on the significance of a word's frequency in the corpus. Our results showed that our hypothesis was supported as the random forest classifier along with TF-IDF vectorizer performed better than the count vectorizer in terms of accuracy, precision, and recall.

6 CONCLUSION

In conclusion, we have explored the problem of multi-class authorship attribution and evaluated different feature extraction methods and classification algorithms. Our findings suggest that TFIDF vectorizer with custom features using Random Forests provides the best F1 score and scalability of the model, outperforming other combinations of methods. Additionally, our results show that increasing the SelectKBest features can improve the performance of the model.

The significance of our work lies in its potential applications, such as in forensic linguistics and cybersecurity, where authorship attribution is a critical task. The accurate identification of the author of an unknown text can provide valuable information for solving crimes and preventing security breaches. Therefore, our work can contribute to the development of more efficient and accurate authorship attribution techniques. Further research can be done by testing the scalability of other feature extraction methods, such as Doc2Vec, and by scaling the models to include more authors. Overall, our work can pave the way for future research in the field of authorship attribution.

7 CODE

The code and the link to the dataset used can be found at our GitHub

8 TEAM MEETINGS

We met at the following dates for completing the project:

- 9 April, 2023: 1:00 - 5:00 PM (Everyone was present)
- 14 April, 2023: 2:00 - 5:00 PM (Everyone was present)
- 17 April, 2023: 2:00 - 6:00 PM (Everyone was present)
- 19 April, 2023: 1:00 - 3:00 PM (Everyone was present)
- 22 April, 2023: 1:00 - 8:00 PM (Everyone was present)
- 23 April, 2023: 2:00 - 7:00 PM (Everyone was present)

REFERENCES

[1] Rathee, Dhruv & Mann, Suman. (2022). Detection of E-Mail Phishing Attacks - using Machine Learning and Deep Learning. International Journal of Computer Applications. 183. 1-7. 10.5120/ijca2022921868.

[2] Samar Al-Saqqa and Arafat Awajan. 2019. The Use of Word2vec Model in Sentiment Analysis: A Survey. In Proceedings of the 2019 International Conference on Artificial Intelligence, Robotics and Control (AIRC '19). Association for Computing Machinery, New York, NY, USA.

[3] Abbas, Muhammad & Ali, Kamran & Memon, Saleem & Jamali, Abdul & Memon, Saleemullah & Ahmed, Anees. (2019). Multinomial Naive Bayes Classification Model for Sentiment Analysis.

[4] H. Zhang and D. Li, "Naïve Bayes Text Classifier," 2007 IEEE International Conference on Granular Computing (GRC 2007), Fremont, CA, USA, 2007

[5] Sarwat Nizamani, Nasrullah Memon, CEAI: CCM-based email authorship identification model, Egyptian Informatics Journal,Volume 14, Issue 3,2013.

[6] de Vel, O.: Mining e-mail authorship. In: ACM International Conference on Knowledge Discovery and Data Mining (KDD) (2000).

[7] Bhargava, M., Mehndiratta, P., Asawa, K. (2013). Stylometric Analysis for Authorship Attribution on Twitter. In: Bhatnagar, V., Srinivasa, S. (eds) Big Data Analytics. BDA 2013. Lecture Notes in Computer Science, vol 8302. Springer, Cham. https://doi.org/10.1007/978-3-319-03689-2_3