## Data:

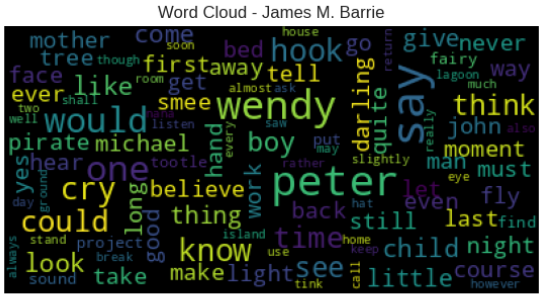
Our data consists of five books from the Gutenberg library. those books are:

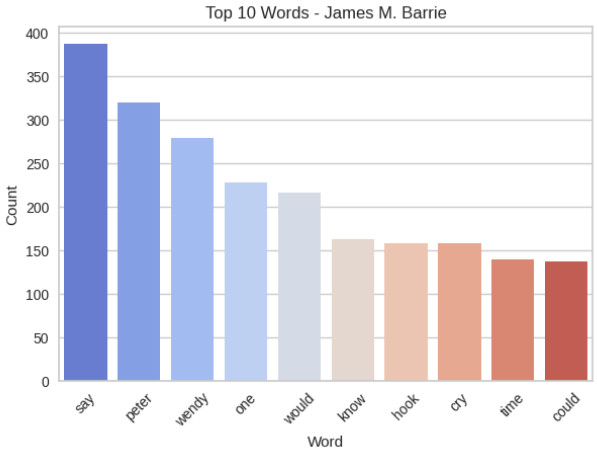
1. The Adventures of Pinocchio by Carlo Collodi
2. Peter Pan by J. M. Barrie
3. The Wonderful Wizard of Oz by L. Frank Baum
4. Alice's Adventures in Wonderland by Lewis Carroll
5. Gulliver's Travels by Jonathan Swift

All books are from the children's literature genre with authors with different nationalities. After reading the books We start the data preparation by:

- tokenizing the book to get a list of separated words
- converting all words into lowercase words
- removing the stop words
- splitting each book to 200 samples, each sample containing 100 words
- lemmatizing every word then we combine all the samples into the complete data frame containing 1000 rows

then we explore our data by plotting the most frequent 200 words of every book in a word cloud and then we plot the top 10 words to see how much they differ from other books.




## Feature Engineering:

We implemented three functions to transform our data (Bag of Words – TFIDF – N-Gram) then we explore their outputs on the TSNE figures. We see that there is a clear difference between different authors in each of BOW and TFIDF while there isn't a clear difference in Bi-Gram or Tri-Gram. Then we label encode the target column (Authors' names)

**Modeling:**

For each of the previous techniques we split the data into training and testing datasets and pass them to our models. Our main evaluation metrics are Accuracy and F1 Score so we compare our models using these metrics across different transformation functions

| Model | Bag of Words | TFIDF | Bi-Gram | Tri-Gram |
|---|---|---|---|---|
| SVM | Accuracy: 0.96<br>F1 score: 0.97 | Accuracy: 0.96<br>F1 score: 0.97 | Accuracy: 0.82<br>F1 score: 0.83 | Accuracy: 0.71<br>F1 score: 0.73 |
| Naïve Bayes | Accuracy: 0.96<br>F1 score: 0.97 | Accuracy: 0.96<br>F1 score: 0.97 | Accuracy: 0.95<br>F1 score: 0.95 | Accuracy: 0.88<br>F1 score: 0.88 |
| KNN | Accuracy: 0.95<br>F1 score: 0.96 | Accuracy: 0.94<br>F1 score: 0.94 | Accuracy: 0.86<br>F1 score: 0.86 | Accuracy: 0.62<br>F1 score: 0.60 |
| Random Forrest | Accuracy: 0.95<br>F1 score: 0.96 | Accuracy: 0.95<br>F1 score: 0.96 | Accuracy: 0.73<br>F1 score: 0.73 | Accuracy: 0.48<br>F1 score: 0.49 |
| SGD | Accuracy: 0.97<br>F1 score: 0.97 | Accuracy: 0.96<br>F1 score: 0.96 | Accuracy: 0.92<br>F1 score: 0.92 | Accuracy: 0.79<br>F1 score: 0.80 |
| XGB | Accuracy: 0.95<br>F1 score: 0.95 | Accuracy: 0.95<br>F1 score: 0.95 | Accuracy: 0.83<br>F1 score: 0.83 | Accuracy: 0.75<br>F1 score: 0.75 |
| BERT | Accuracy: 72.5 | | | |

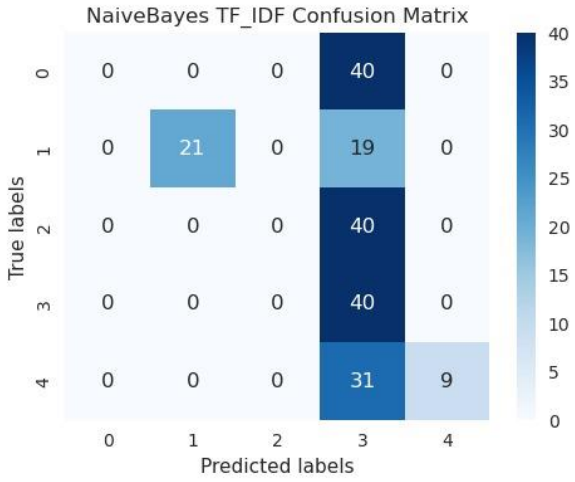
Based on the results, we choose Naïve Bayes as our champion model since it has the highest f1 score and accuracy across all our models. And it has similar performance in both the Bag of words and the TFIDF functions with a better performance of the Bag of words function in the cross validation with accuracies of [0.925 0.95 0.9375 0.9375 0.95 0.975 0.975 0.975 0.95 0.975].

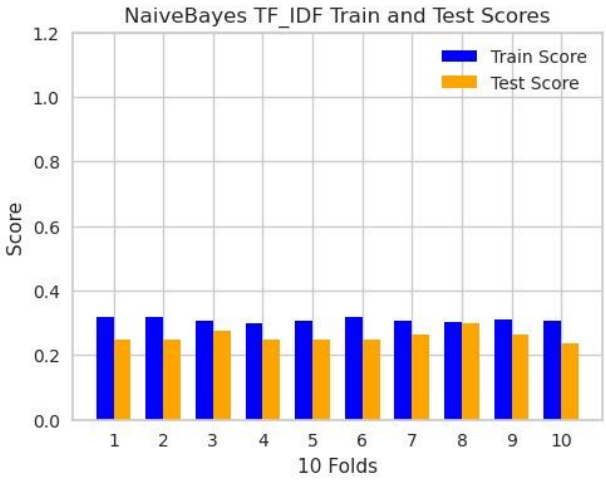
Our champion model has its default parameters with testing accuracy of 96% and F1 score of 97%. Having such little differences in accuracies indicate that there is no overfitting.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Carlo Collodi | 1.00 | 0.97 | 0.99 | 40 |
| James M. Barrie | 1.00 | 0.95 | 0.97 | 40 |
| Jonathan Swift | 1.00 | 0.97 | 0.99 | 40 |
| L. Frank Baum | 1.00 | 0.93 | 0.96 | 40 |
| Lewis Carroll | 0.85 | 1.00 | 0.92 | 40 |
| accuracy | | | 0.96 | 200 |
| macro avg | 0.97 | 0.97 | 0.97 | 200 |
| weighted avg | 0.97 | 0.96 | 0.97 | 200 |

## Error Analysis:

After choosing our champion model as the Naïve Bayes we explore its parameters and change them to reduce its accuracy to see which parameters affect our model the most. When we use "alpha = 0.5", "fit_prior = False" and "class_prior = [0.1, 0.2, 0.1, 0.5, 0.1]" the accuracy of the BOW drops to 94% , TFIDF drops to 35% , Bi-Gram drops to 89% and the Tri-Gram drops to 75%. We also tried changing the number of words per partition from 100 words to 30 words which decreased our accuracy with BOW to 92% , TFIDF to 90% , Bi-Gram to 75% ,and Tri-Gram to 55%

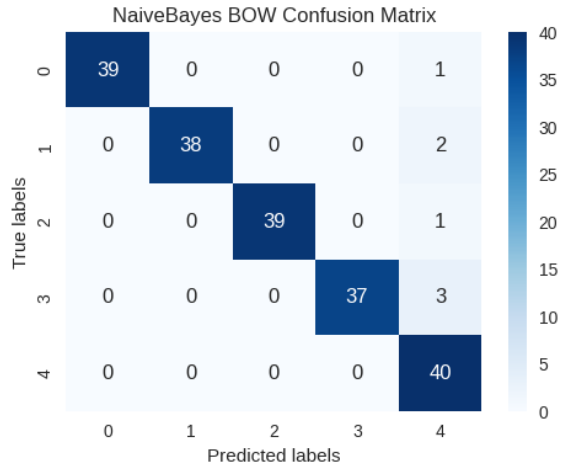


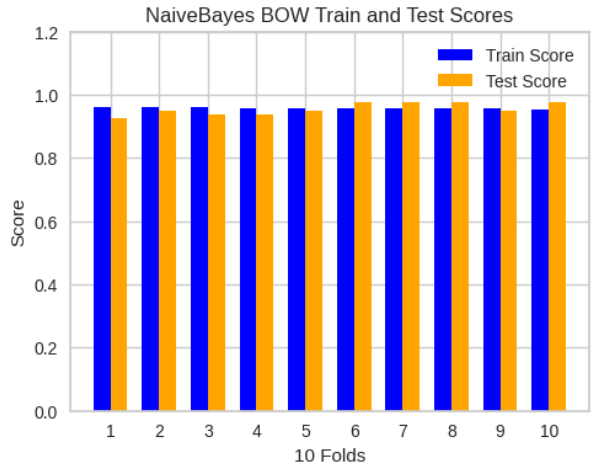



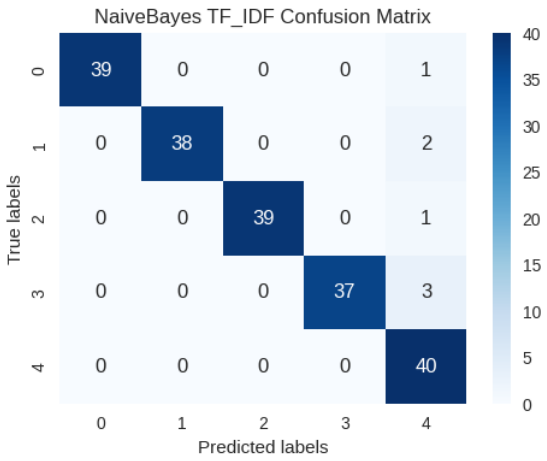

As we can see from the confusion matrix the model predicts most cases as class 3 of which we set the class prior to 0.5 in the class_prior parameter and predicts some cases of class 1 correctly because we set a higher class prior than other classes

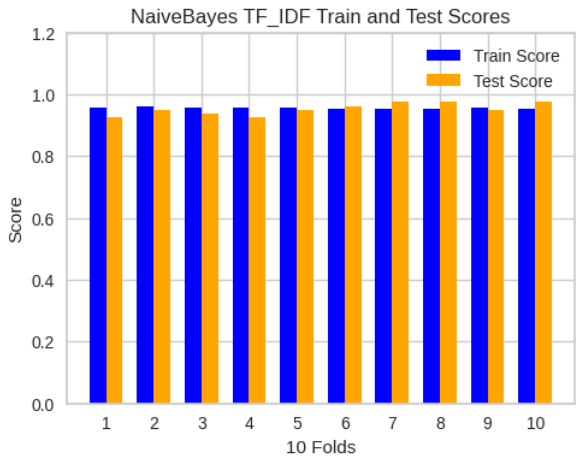
## Testing Results:

For every model of the models we use, we output the classification report and plot the confusion matrix and plot the validation results against the test results for every transformation method we use.






**README:**

There are two notebooks (BERT.ipynp – book_classification.ipynp) the first notebook contains the implementation of the BERT model with its own data importing, preprocessing and feature engineering. The second notebook contains all other models with its own version of data preprocessing as mentioned in the data section.

Both notebooks have commands to install and import the necessary libraries in order for the notebooks to run correctly.