Discrimination Analysis
And
Uncovering the Unnoticed

By

Umar Mohamed Ibrahim

Mahmoud Yahia Ahmed

Abdulrahman Mohamed Abdelsalam

Mahmoud Maged Mohamed

Group number (DSA_202101_ 12)

Discrimination Analysis

- Goal:

  Implementing a classification strategy of five different books by five different authors which have the same genres and are semantically the same.

- Dataset:

  five books were selected from Gutenberg's digital library, and all of them have the same fiction genre. Books Names e.g

  - Carroll-Alice.
  - Chesterton-Ball.
  - Edgeworth-Parents.
  - Melville Moby Dick.
  - Burgess Buster brown.

- Project Phases:

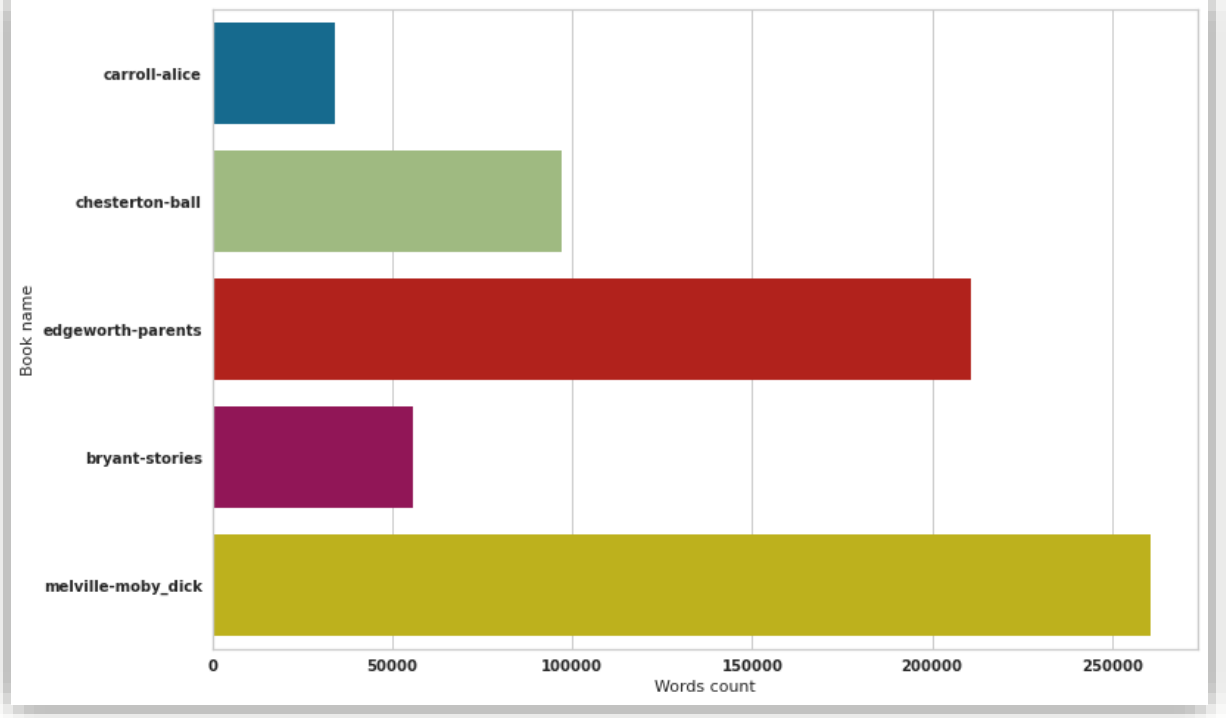
1) Data Pre-processing:
   - 1.1 Read Books from Gutenberg using NLTK.
   - 1.2 Clean Special Characters.
   - 1.3 Remove Stop Words and Stemming.

2) Feature Engineering:
   - 2.1 Split the data into train, and test data.
   - 2.2 Transformation (Embedding).
   - 2.3 Select Dimensionality Reduction.

3) Modelling and Evaluation:
   - 3.1 Modeling Multi-class classification solutions.
   - 3.2 Choosing and Tunning the Champion model.
   - 3.3 Evaluation and Error Analysis.

1.1) Read Books from Gutenberg using NLTK

The illustration shows a huge variation in the word counts per book, and because that data is not imbalanced, 200 random paragraphs were chosen from each book, and each paragraph has 100 words, so our corpus will contain 1K paragraph of 5 books.



1.2) Clean Special Characters

The illustration shows the function which removes the special characters.

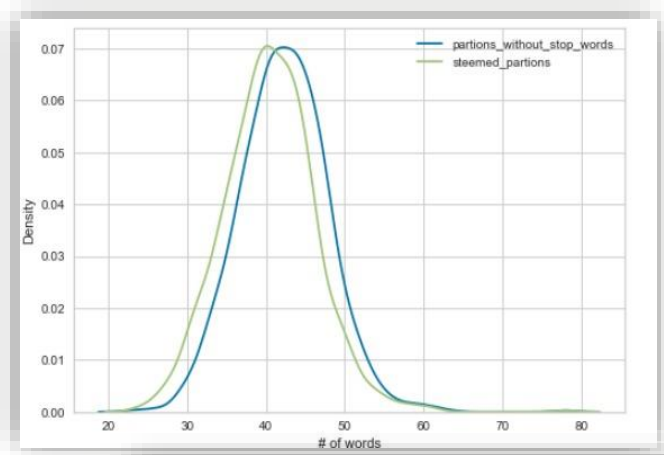
```
def clean(txt):
    '''
    Elminate any non word character from the input text, and remove any additionl spaces.

    Args:
      - txt (string) -> the unclean text.
    '''
    txt = re.sub(r'\W', ' ', txt)
    txt = re.sub(r' \w ', ' ', txt)
    txt = re.sub(r' +', ' ', txt)
    return txt.strip().lower()
```

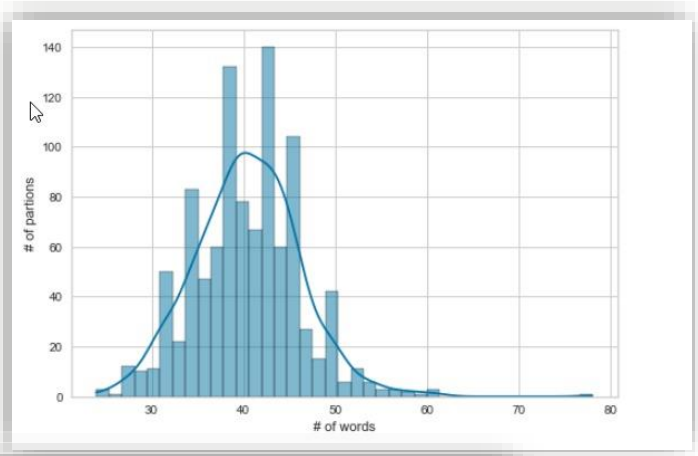
```
{'\n', '\r', '!', '"', '$', '%', '&', "'", '(', ')', '*', '+', ',',
 '-', '.', '/', ':', ';', '<', '=', '>', '?', '@', '[', ']', '~' }
```

1.3) Remove Stop Words and Stemming.

the curve shows how removing stop words and stemming the data affect the words count distribution.



the histogram shows the frequencies after removing stop words and stemming.



carroll-alice	chesterton-ball	edgeworth-parents	bryant-stories	melville-moby_dick
alice	turnbull	come	little	whale
little	man	know	come	man
think	macian	good	man	like
know	like	mr	look	ship
look	come	look	king	sea
like	know	think	day	ahab
begin	thing	little	great	boat
come	look	time	run	ye
thing	think	tell	time	head
time	face	ll	know	old

Here are some of the most common words that had used by our different authors.

2.1) Split the data into train, and test data

We split the data into 70% for training and 30% for testing, and instead of splitting a Validation portion from the data, the Cross-Validation technique would be used.

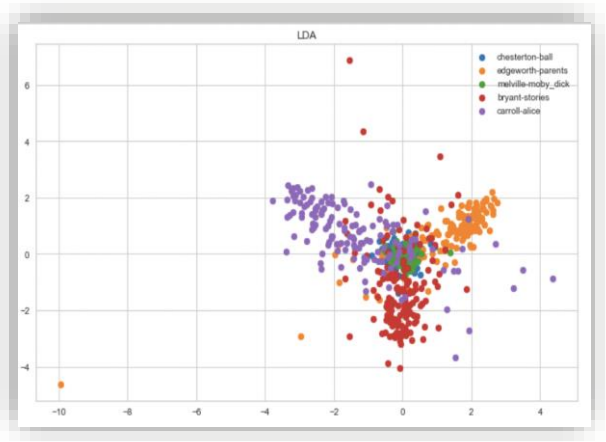
2.2) Transformation (Embedding).

Multiple embedding techniques were used at that phase, some of them depend on word occurrence e.g. "BOW", "Tf-IDF", "N-gram", in addition to some techniques that maintain the meaning e.g. "Glove", "Word2Vec" and "FastText", but the last three techniques we used pre-trained models to apply transfer-learning and get the contextual embedding for each word instead of naive occurrence-based one.

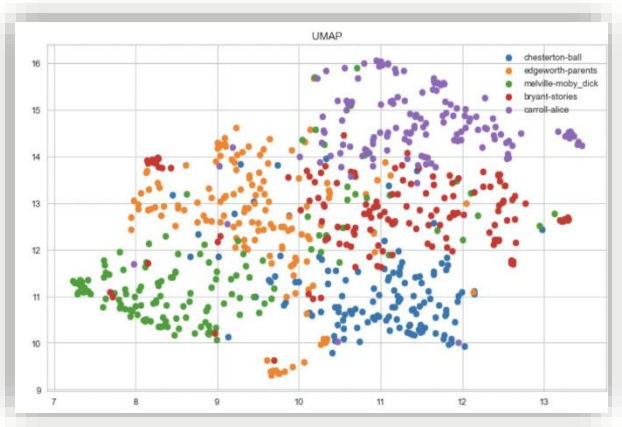
2.3) Select Dimensionality Reduction.

Many embedding techniques need to be tested, but to make the learning harder, and for reducing the number of dimensions, a dimensionality reduction algorithm will be used here, as example BOW produces an embedding vector for each word with a length of more than 6000, and we aim to reduce it to just 2 features, but without losing massive information, and many algorithms used here e.g LDA, PCA, UMAP, t-SNE.

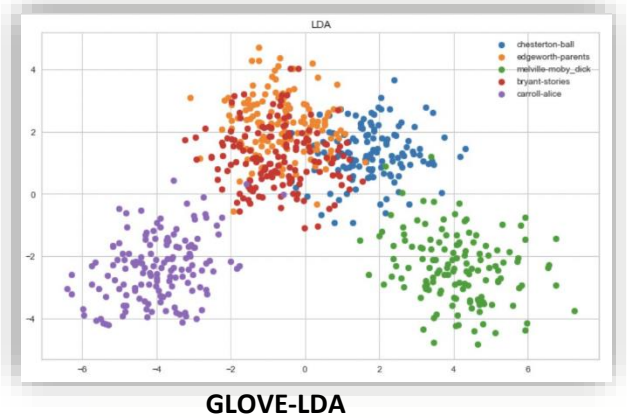
Discrimination Analysis



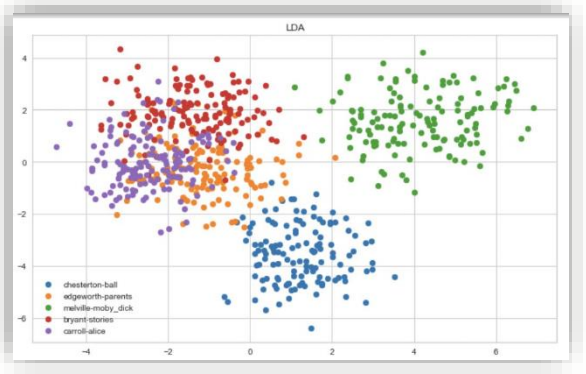
LDA-BOW



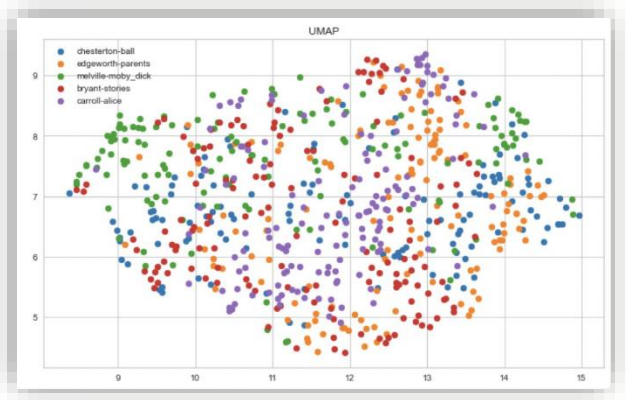
UMAP TF-IDF



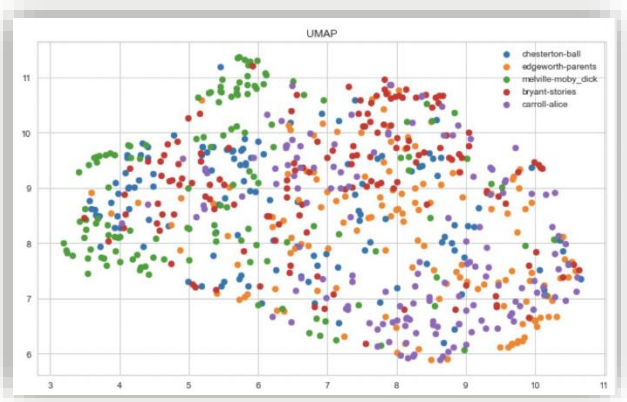
GLOVE-LDA



WORD2VEC-LDA

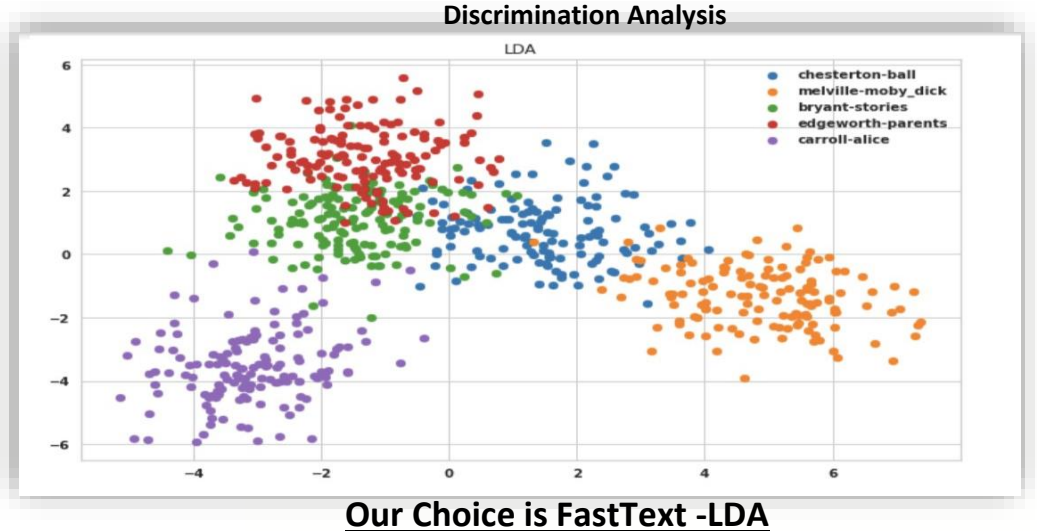


GLOVE-UMAP



FASTTEXT-UMAP

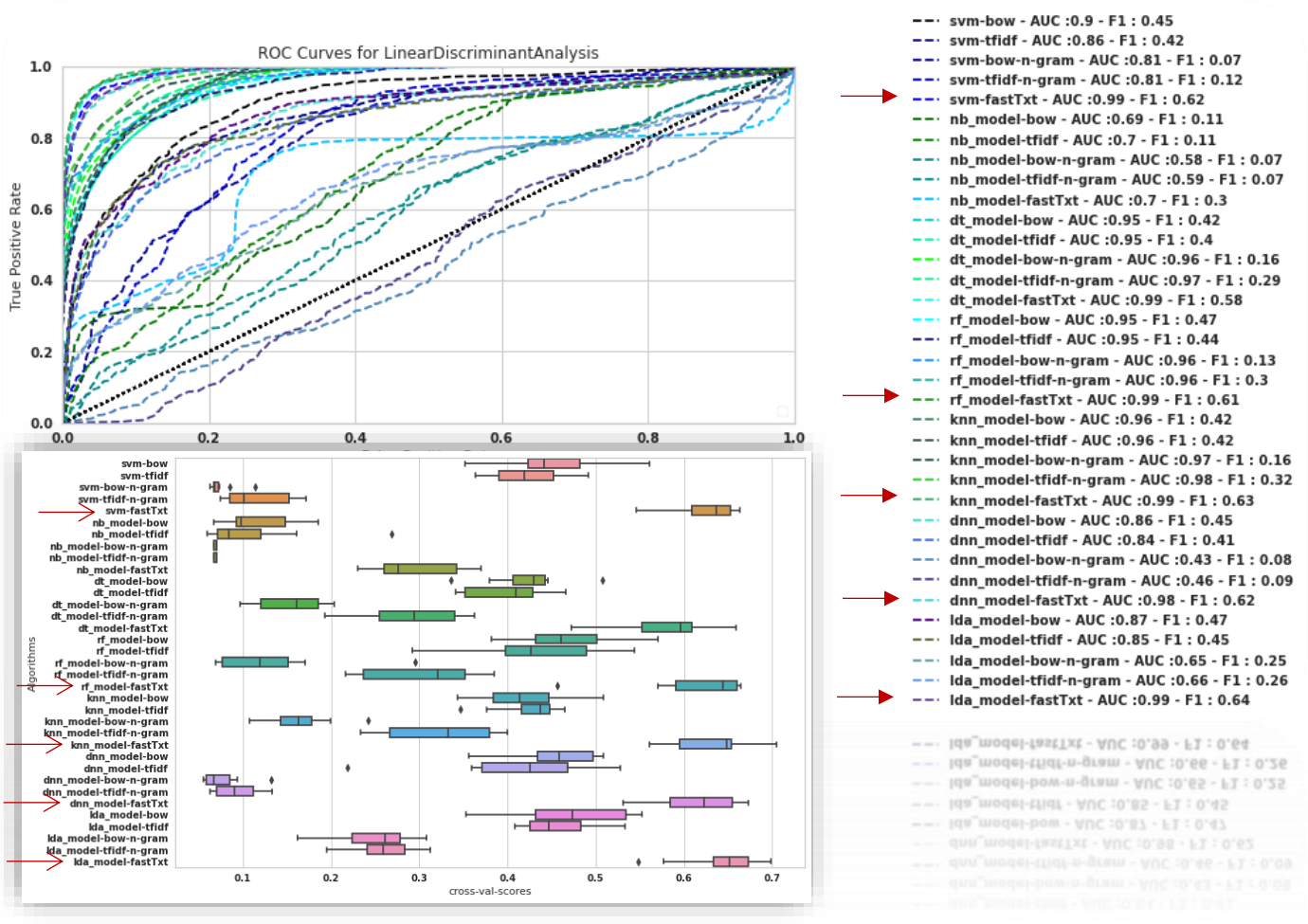
Discrimination Analysis



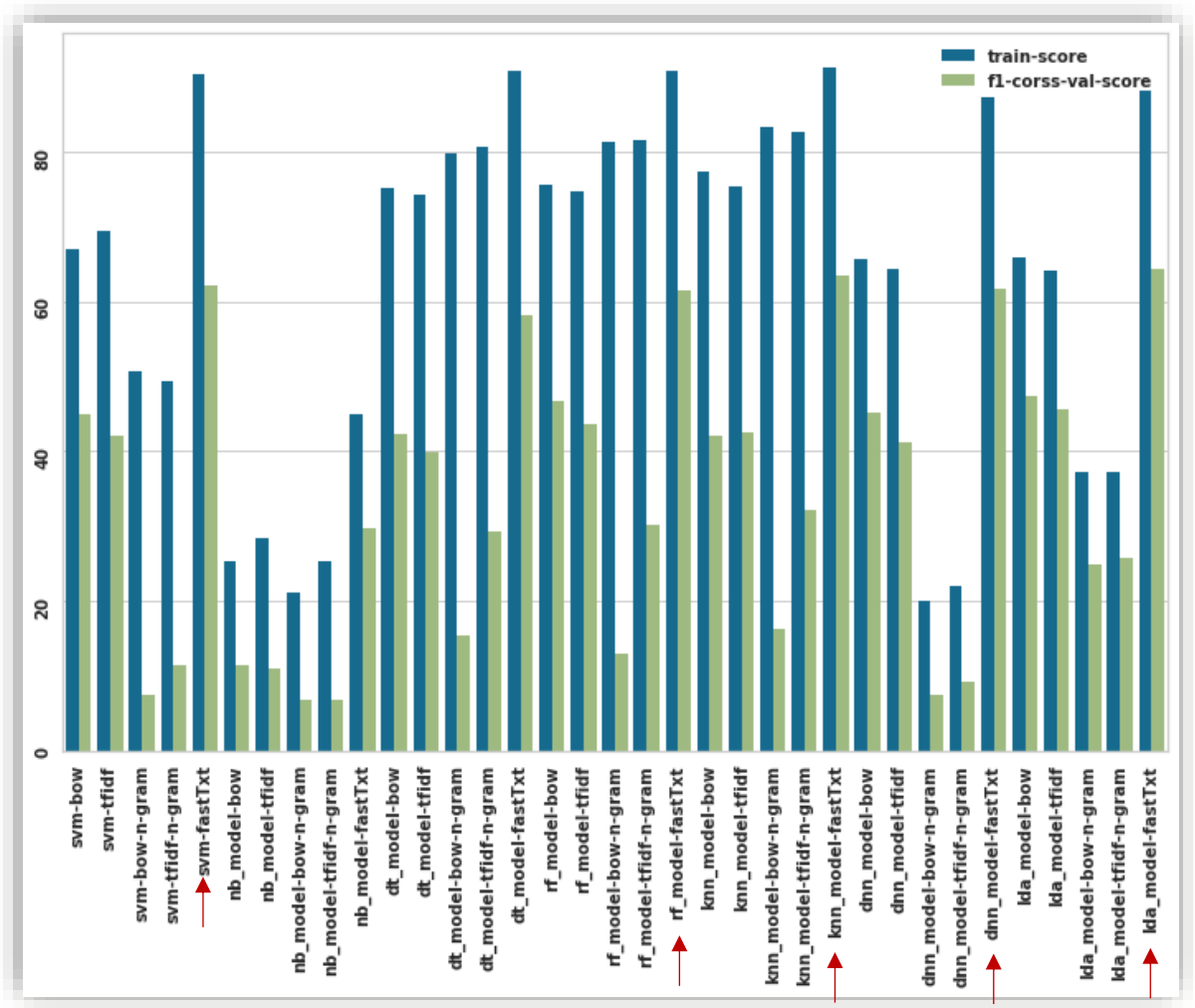
Our Choice is FastText -LDA

There is an interesting notice here, algorithms like t-SNE & UMAP, have the same behavior which successes in distinguishing between the points of each book when occurrence-based embedding algorithms have been used especially with TF-IDF, but they failed with the contextual embedding base e.g glove and word2vec, but in contrast, the LDA achieve a greater separation when the embedding produced from contextual-embedding based e.g glove and word2vec. So when having a contextual-embedding base algorithm, LDA will be the best suitable dimensionality reduction algorithm.

3.1) Modeling Multi-class classification solutions.

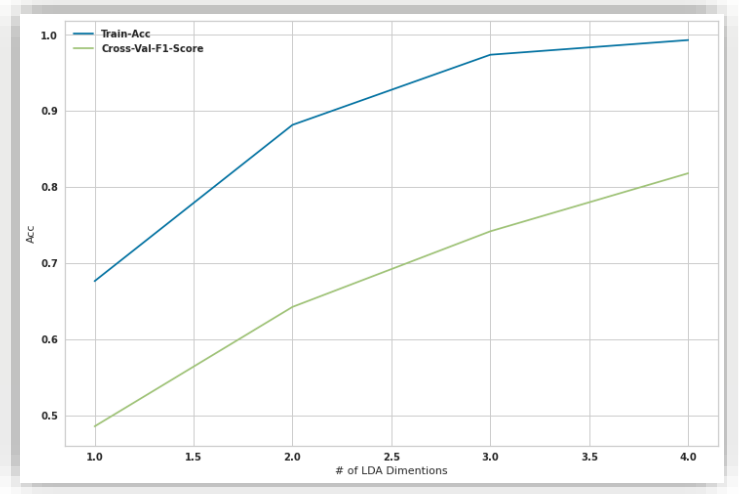


Discrimination Analysis



The champion model is 'lda_model-fastTxt' which used fastTxt Embedding with LDA dimensionality reduction then applied LDA Classifier. And this model was selected based on the maximum cross validation metric.

3.2 Tunning the Champion model.



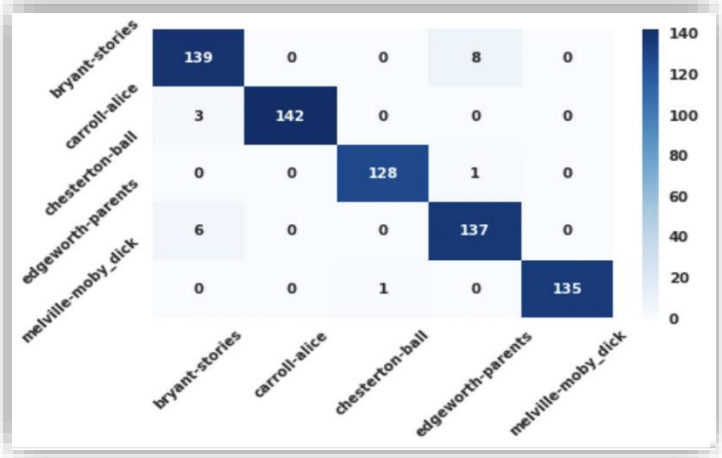
This figure illustrates how the number of features produced by LDA dimensionality reduction affect the accuracy, and we will get along with 3 dimensions.

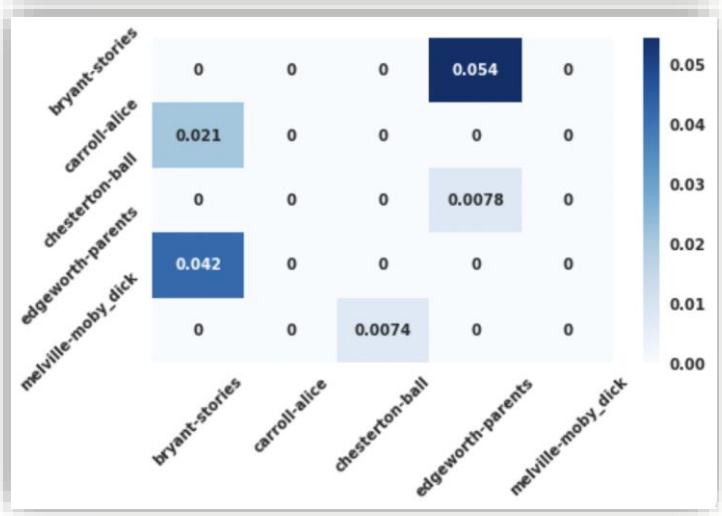
3.3 Evaluation and Error Analysis.

Training Set Evaluation

acc :: 0.9728571428571429

	precision	recall	f1-score	support
bryant-stories	0.94	0.95	0.94	147
carroll-alice	1.00	0.98	0.99	145
chesterton-ball	0.99	0.99	0.99	129
edgeworth-parents	0.94	0.96	0.95	143
melville-moby_dick	1.00	0.99	1.00	136
accuracy			0.97	700
macro avg	0.97	0.97	0.97	700
weighted avg	0.97	0.97	0.97	700

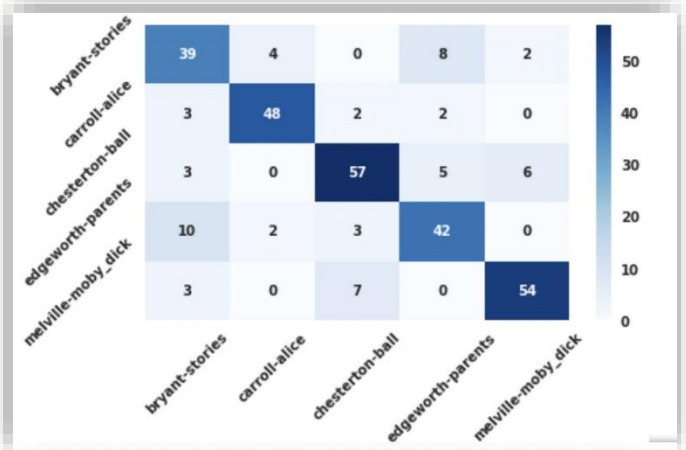


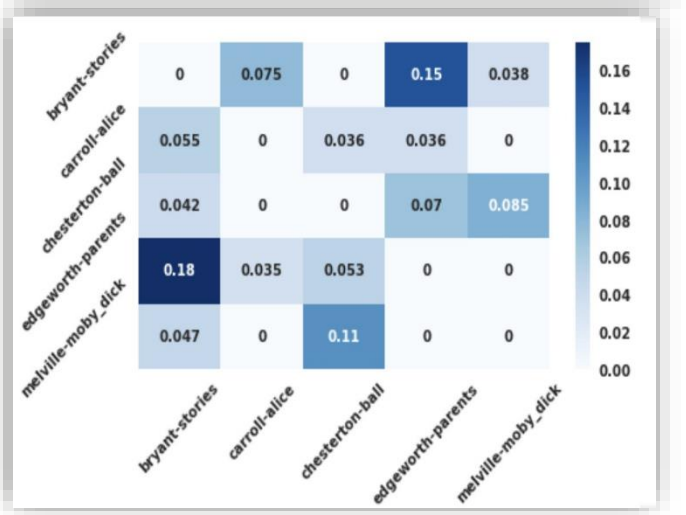


Testing Set Evaluation

acc :: 0.8

	precision	recall	f1-score	support
bryant-stories	0.67	0.74	0.70	53
carroll-alice	0.89	0.87	0.88	55
chesterton-ball	0.83	0.80	0.81	71
edgeworth-parents	0.74	0.74	0.74	57
melville-moby_dick	0.87	0.84	0.86	64
accuracy			0.80	300
macro avg	0.80	0.80	0.80	300
weighted avg	0.80	0.80	0.80	300





Conclusion:

The model here is overfitting on the training-set and it needs some regularization, but as is so far, it achieve a 74% f1-score on cross-validation 10 Folds and 80% on the test-set, and according to the confusion matrix of Error analysis, the higher error exists with those two books "byrant-stories", "Edgeworth-parents", which very make sense, because that very clear from the Embedding Space figure FastText-LDA, the red points of "Edgeworth-parents" overlapped with blue ones of "byrant-stories", which make them writing style same a little bit.