1. **Instruction**

We will discuss two different types of text classification techniques and one clustering techniques to solve book classification.

We will use Python, Beautiful-soup, Sci-kit-learn, Gensim and the XGBoost library for solving this problem.

1. **Getting the data**

We used Beautiful-soup package to scrab the book content from Gutenberg.



Figure 1

Then we used regular expression to divide text into different chapter and label them.



Figure 2

Finally, we excluded stop words, punctuation and stemming the words. And then created segments on the text, the distance between two segments was 100 words (some segments will be less than 100 words)

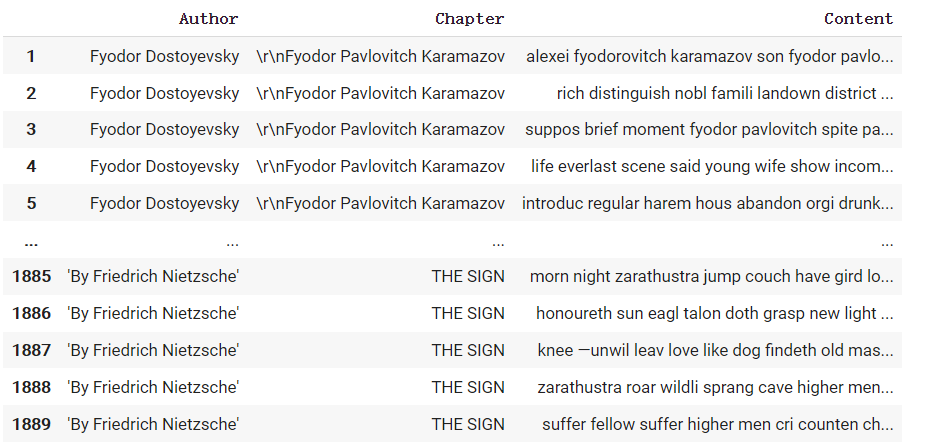


Figure 3

1. **Transforamation**

We need to transform text into vector, the most popular models for transformation are ‘Doc2Vec’ & ‘Tf-Idf’.

* **Doc2Vec**

We imported Doc2Vec method from ‘gensim’ package and made the transformation.

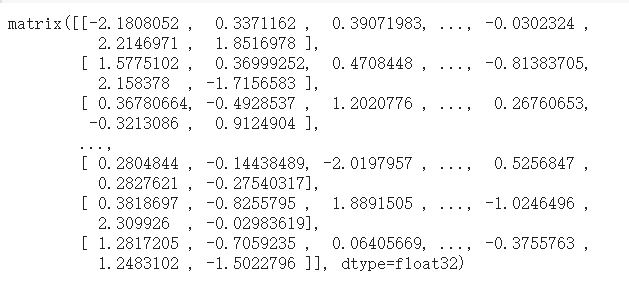


Figure 4

* **Tf-Idf**

We also tried Tf-Idf transformer. We can see the result after transformation.

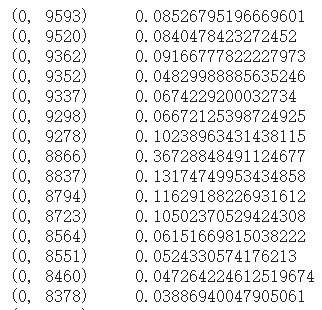


Figure 5

1. **Binary Classification**

* **Doc2Vec+XGBoost**

After we used doc2vec to transform the text, we can use these numeric features into any classification method we want. We tried XGBoost method and did a 5-fold cross validation model.

The train data(X\_train) was the book content in the dataframe and the label of the train data was the authors name, so this is a binary classification.



Figure 6

We can see that both Cross Validation and Test Data achieved very high accuracy.

* **Doc2Vec+SVM**

We also tried SVM method to do classification, SVM also achieved very high accuracy.



* **Tf-idf+XGBoost**

Then we used XGBoost classification method for tfidf.



Figure 7

We can see that Tf-Idf+XGBoost method also achieved very high accuracy.

* **Tf-idf+SVM**

For comparison, we also used SVM to classify our data. The accuracy of this method achieved one! How amazing!



1. **Multi-class classification**

Since we achieved very high accuracy in binary classification, we also want to see how it performs under multiple label classifications. We used chapter content as train data and the chapter name replaces the authors name as the data label.

* **Doc2vec+XGBoost**

The accuracy of the classification is much lower than binary classification.





* **Doc2vec+SVM**

The SVM method achieved better than XGBoost.



* **Tf-idf+XGBoost**

By using Tf-idf transformation the XGBoost method achieved much lower accuracy.



* **Tf-idf+SVM**

For the same time, we used SVM method to do comparison.



The accuracy is better than XGBoost method but also very low.

1. **Clustering between two authors**

In order to get the insight of semantic comparison, we also do clustering between two authors. We used three different clustering method and compared the accuracy between them.

* **Elbow method**

First of all, we need to use elbow method to determine the best number of k.

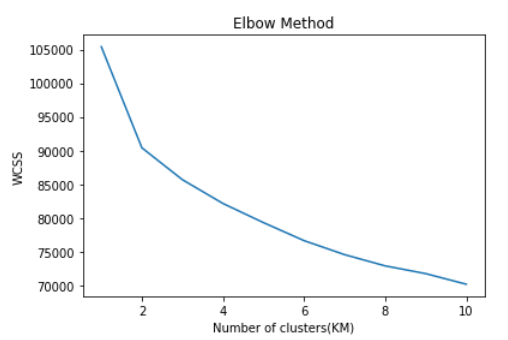


Figure 8

The turning point shows on the picture is 2, means that 2 is the best number of k.

* **Doc2vec**

We used three different methods (k-means, Hierarchical clustering, Gaussian Mixture Model) to cluster the text. The number of centers is 2, since we have two different authors. The kappa value of different methods is 0.950, 0.905.956 and the silhouette value of different methods is 0.1539,0.1466,0.1521, respectively.

From the result, we can see that the clustering method also achieved very high accuracy, that means computer can easily tell the two authors apart.

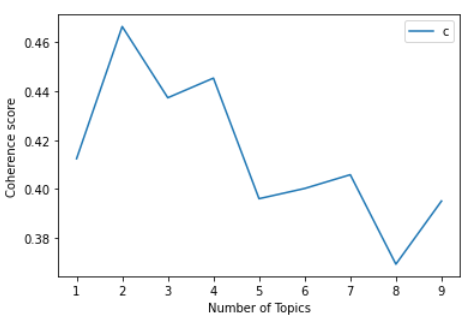
* **Tf-idf**

For tfidf transformation, we also tried three clustering method, the kappa value is 0.944,0 (because all texts are clustered into one author), 0.945 and the silhouette value is 0.0145, 0.0071, 0.0145.

That means tfidf can also give the right answer.

1. **LSI model**

LSI model uses singular value decomposition to reduce the dimension of the text. LSI model can also give us the number of topics by calculating the coherence scores.



Figure

We can see that the highest coherence score is occurring when the number of topics is 2. We can also see that the details of different topics.

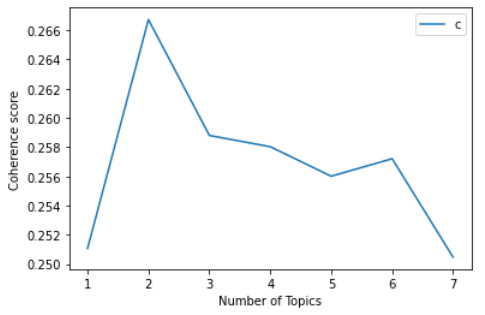




Figure

1. **LDA model**

We also used coherence score to determine the best number of topics in LDA model.



Figure

This result shows the same answer with LSI model, the best number of topics is 2.

1. **Error Analysis**

We created word cloud for different author and see the difference.

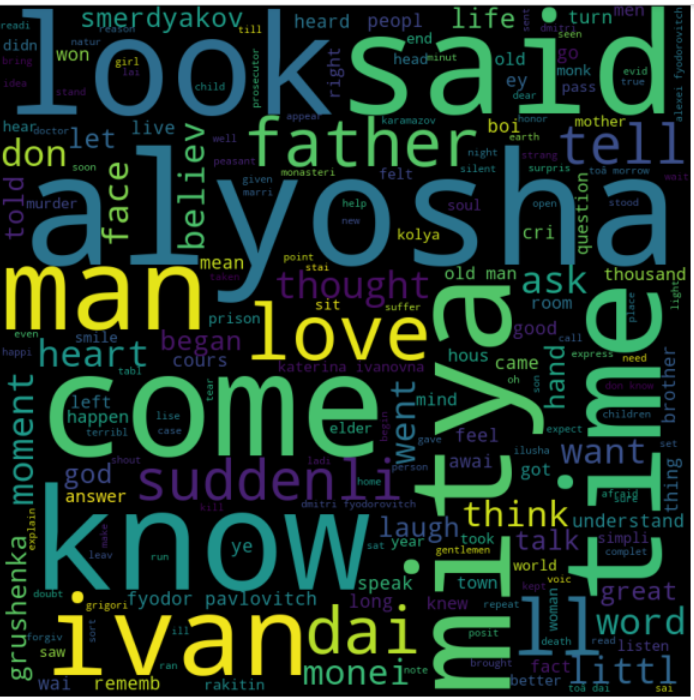


Figure 12 Word cloud for ‘Fyodor Dostoyevsky’

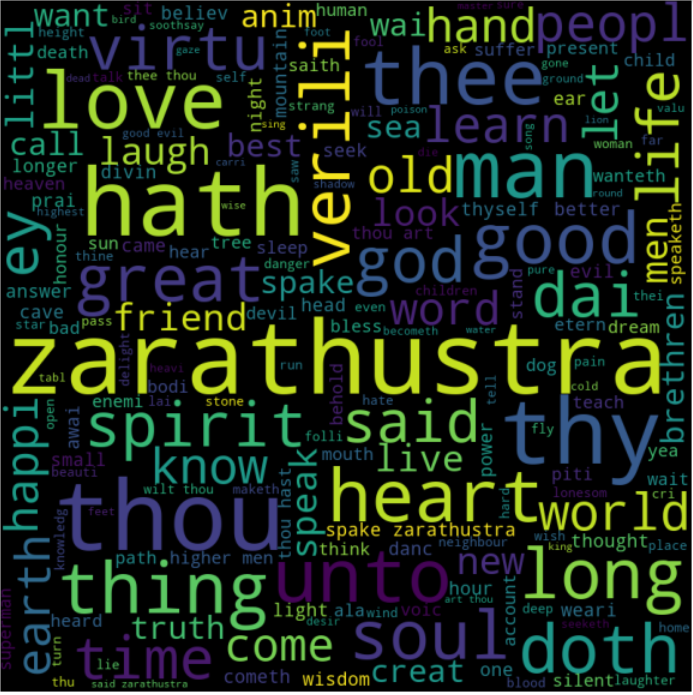


Figure 13 Word cloud for ‘Friedrich Nietzsche’

Bigger words mean higher frequency, so for author Fyodor Dostoyevsky the most frequent words are ‘Alyosha’, ’said’, ‘know’, ‘come’ and ‘man’.

For author Friedrich Nietzsche, the most frequent words are ‘thou’, ‘zarathustra’, ‘hath’, ‘thy’ and ‘unto’.

The two authors have completely different vocabulary habits, which is why we can achieve such a high accuracy.

1. **Conclusion**

Dov2Vec is a very popular model in NLP, it achieved only a little higher accuracy in our cases. This reason can be explained. The two authors have totally different vocabulary habits, machine can easily tell them apart. Doc2Vec is a more advanced model, so it can perform better.