



```
In [102]: y_pred = final_model.predict(x_test)
y_pred = np.argmax(y_pred,axis=1)
y_ground_truth = y_test

print(classification_report(y_ground_truth, y_pred))

63/63 [=====] - 0s 2ms/step
precision    recall  f1-score   support

0           0.95         0.92         0.93         206
1           0.94         0.89         0.96         206
2           0.81         0.87         0.84         203
3           0.93         0.87         0.90         219
4           0.94         0.91         0.92         194
5           0.88         0.93         0.91         187
6           0.97         0.93         0.95         183
7           0.92         0.96         0.94         180
8           0.94         0.93         0.93         202
9           0.86         0.83         0.84         218

accuracy          0.91         0.91         0.91         1998
macro avg         0.91         0.91         0.91         1998
weighted avg      0.91         0.91         0.91         1998
```

I decided to use the SVM as my final model and use it to make some predictions because it performed better than the other 3 models. I selected a random sample from the test set and made predictions using it to demonstrate the model's accuracy and efficiency. The ground truth represents the genre index as it should actually be, while the predicted index is what my SVM model predicted as the outcome.

```
In [54]: sample = x_test.sample()
prediction = best_svm_model.predict(sample)
prediction_index = prediction
ground_truth = y_test[sample.index].iloc[0]
print("Random row to be predicted: ", sample.index)
print("Ground_truth: {}, Predicted Index: {}".format(ground_truth, prediction_index))

Random row to be predicted:  Int64Index([2256], dtype='int64')
Ground_truth: 2, Predicted Index: [2]
```