Рубежный контроль №2

Студент: Кривцов Н.А.
Группа: ИУ5-22М
Вариант: 7
Классификатор №1: RandomForestClassifier
Классификатор №2: Complement Naive Bayes

Импорт библиотек

In [16]:

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.naive_bayes import ComplementNB
%matplotlib inline
```

Загрузка и просмотр датасета

В рамках настоящего РК будет решена задача классификации текстов обзоров фильмов. Целевой признак - жанр фильма. Исходный датасет также содержит сторонние признаки, которые не будут использоваться в разрабатываемых моделях.

In [2]:

```
data = pd.read_csv('/content/drive/MyDrive/MMO/wiki_movie_plots_deduped.csv')
```

In [3]:

```
data.shape
```

Out[3]:

```
(34886, 8)
```

In [4]:

```
data.head()
```

Out[4]:

	Release Year	Title	Origin/Ethnicity	Director	Cast	Genre	Wiki Page	Plot
0	1901	Kansas Saloon Smashers	American	Unknown	NaN	unknown	https://en.wikipedia.org/wiki/Kansas_Saloon_Sm...	A bartender is working at a saloon, serving dr...
1	1901	Love by the Light of the Moon	American	Unknown	NaN	unknown	https://en.wikipedia.org/wiki/Love_by_the_Ligh...	The moon, painted with a smiling face hangs ov...
2	1901	The Martyred Presidents	American	Unknown	NaN	unknown	https://en.wikipedia.org/wiki/The_Martyred_Pre...	The film, just over a minute long, is composed...
3	1901	Terrible Teddy, the	American	Unknown	NaN	unknown	https://en.wikipedia.org/wiki/Terrible_Teddy	Lasting just 61 seconds and

	Release Year	Title	Origin/Ethnicity	Director	Cast	Genre	Wiki Page	Plot
3	1901	Teddy, the Grizzly King	American	Unknown	NaN	unknown	https://en.wikipedia.org/wiki/Terrible_Teddy,_...	and consisting of two ...
4	1902	Jack and the Beanstalk	American	George S. Fleming, Edwin S. Porter	NaN	unknown	https://en.wikipedia.org/wiki/Jack_and_the_Bea...	The earliest known adaptation of the classic f...

In [5]:

```
# фильтрация датасета и выбор исследуемых признаков
df = data[['Genre', 'Plot']]
df = df[(df.Genre == 'drama') | (df.Genre =='comedy')]
df.Genre.value_counts(normalize=True)
```

Out[5]:

```
drama     0.576622
comedy    0.423378
Name: Genre, dtype: float64
```

In [6]:

```
df.head()
```

Out[6]:

	Genre	Plot
7	comedy	The film is about a family who move to the sub...
14	comedy	Before heading out to a baseball game at a nea...
15	comedy	The plot is that of a black woman going to the...
16	drama	On a beautiful summer day a father and mother ...
17	drama	A thug accosts a girl as she leaves her workpl...

In [26]:

```
VECTORIZERS = [CountVectorizer(analyzer='char_wb', ngram_range=(3, 6)),
               TfidfVectorizer(analyzer='char_wb', ngram_range=(3, 6))]
CLASSIFIERS = [ComplementNB(), RandomForestClassifier(max_depth=3)]
```

In [28]:

```
for v in VECTORIZERS:
  X = v.fit_transform(df.Plot)
  for cl in CLASSIFIERS:
    print("Векторизатор: {}\nКлассификатор: {}".format(type(v), type(cl)))
    print("Точность: {}".format(cross_val_score(cl, X, df.Genre, scoring='accuracy').mean()))
```

```
Векторизатор: <class 'sklearn.feature_extraction.text.CountVectorizer'>
Классификатор: <class 'sklearn.naive_bayes.ComplementNB'>
Точность: 0.7334455950556852
Векторизатор: <class 'sklearn.feature_extraction.text.CountVectorizer'>
Классификатор: <class 'sklearn.ensemble._forest.RandomForestClassifier'>
Точность: 0.5784586504473797
Векторизатор: <class 'sklearn.feature_extraction.text.TfidfVectorizer'>
Классификатор: <class 'sklearn.naive_bayes.ComplementNB'>
Точность: 0.5778785666273711
Векторизатор: <class 'sklearn.feature_extraction.text.TfidfVectorizer'>
Классификатор: <class 'sklearn.ensemble._forest.RandomForestClassifier'>
Точность: 0.5769118693282901
```

Значения показателей качества моделей могут быть обусловлены значениями гиперпараметров моделей, анализ которых в рамках данной работы не производился.

Наилучшую точность показали векторизатор `CountVectorizer` с классификатором `ComplementNB`.