Московский государственный технический университет им. Н.Э. Баумана
Факультет «Информатика и системы управления»
Кафедра «Системы обработки информации и управления»


# Лабораторная работа №6
# По курсу «Методы машинного обучения»

## «Классификация текста»

**ИСПОЛНИТЕЛЬ:**

Чичикин Тимофей Дмитриевич
Группа ИУ5-25М

_____

**ПРОВЕРИЛ:**

Гапанюк Ю.Е.

_____

Цель работы:

Изучение методов классификации текста.

Задание:

Для произвольного набора данных, предназначенного для классификации текстов, решите задачу классификации текста двумя способами:

1. Способ 1. На основе CountVectorizer или TfidfVectorizer.
2. Способ 2. На основе моделей word2vec или Glove или fastText.
3. Сравните качество полученных моделей.

Описание задания:

Для выполнения лабораторной работы возьмём датасет с обзорами фильмов IMDB для анализа настроений, где выделена целевая переменная: 1 – положительное мнение, а 0 – отрицательное.

Выполнение работы:

1. Классификация текста на основе CountVectorizer
2. Классификация текста на основе модели word2vec
3. Сравнение качества полученных моделей

Вывод:

Была проделана работа по изучению методов классификации текста, в результате чего можно сделать вывод, что для данного датасета наибольшая точность получилась при использовании CountVectorizer и LogisticRegression.

In [ ]:
```
import numpy as np
import pandas as pd
from typing import Dict, Tuple
from scipy import stats
from IPython.display import Image
from sklearn.datasets import load_iris, load_boston
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsRegressor, KNeighborsClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import GridSearchCV, RandomizedSearchCV
from sklearn.metrics import accuracy_score, balanced_accuracy_score
from sklearn.metrics import precision_score, recall_score, f1_score, classific
from sklearn.naive_bayes import ComplementNB
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import cross_val_score
from sklearn.pipeline import Pipeline
from sklearn.metrics import mean_absolute_error, mean_squared_error, mean_squa
from sklearn.metrics import roc_curve, roc_auc_score
from sklearn.svm import SVC, NuSVC, LinearSVC, OneClassSVM, SVR, NuSVR, Linear
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

In [ ]:
```
def accuracy_score_for_classes(
    y_true: np.ndarray,
    y_pred: np.ndarray) -> Dict[int, float]:
    """
    Вычисление метрики accuracy для каждого класса
    y_true - истинные значения классов
    y_pred - предсказанные значения классов
    Возвращает словарь: ключ - метка класса,
    значение - Accuracy для данного класса
    """
    # Для удобства фильтрации сформируем Pandas DataFrame
    d = {'t': y_true, 'p': y_pred}
    df = pd.DataFrame(data=d)
    # Метки классов
    classes = np.unique(y_true)
    # Результирующий словарь
    res = dict()
    # Перебор меток классов
    for c in classes:
        # отфильтруем данные, которые соответствуют
        # текущей метке класса в истинных значениях
        temp_data_flt = df[df['t']==c]
        # расчет accuracy для заданной метки класса
        temp_acc = accuracy_score(
            temp_data_flt['t'].values,
            temp_data_flt['p'].values)
        # сохранение результата в словарь
        res[c] = temp_acc
```

```
    return res

def print_accuracy_score_for_classes(
    y_true: np.ndarray,
    y_pred: np.ndarray):
    """
    Вывод метрики accuracy для каждого класса
    """
    accs = accuracy_score_for_classes(y_true, y_pred)
    if len(accs)>0:
        print('Метка \t Accuracy')
    for i in accs:
        print('{} \t {}'.format(i, accs[i]))
```

In [ ]:
```
# Загрузка данных
df = pd.read_csv('D:\\Ботва\\Магистратура\\2сем\\ММО\\лаб6\\imdb_sup.csv')
text_df=df.head(500).append(df.tail(500))
text_df.drop('Rating', axis=1, inplace=True)
text_df.tail(15)
```

Out[ ]:

| | Review | Sentiment |
|---|---|---|
| **49985** | I had great expectations surrounding this movi... | 0 |
| **49986** | It is playing on SHOWTIME right now but is goi... | 0 |
| **49987** | I love the so-called "blaxploitation" films an... | 0 |
| **49988** | OK, here is the deal. I love action movies and... | 0 |
| **49989** | Grim instead of amusing, mean-spirited instead... | 0 |
| **49990** | This movie did not give Mr. Bachchan justice. ... | 0 |
| **49991** | Oh dear! The BBC is not about to be knocked of... | 0 |
| **49992** | Ridiculous thriller in which a group of studen... | 0 |
| **49993** | If you like poor SE, (some) bad acting and a t... | 0 |
| **49994** | Some Plot Spoilers Ahead.<br /><br />The Nashv... | 0 |
| **49995** | (spoiler) it could be the one the worst movie ... | 0 |
| **49996** | So, you've seen the Romero movies, yes? And yo... | 0 |
| **49997** | Just listen to the Broadway cast album and to ... | 0 |
| **49998** | I have been a fan of the Carpenters for a long... | 0 |
| **49999** | Set in 1945, Skenbart follows a failed Swedish... | 0 |

In [ ]:
```
#Кодирование целевого признаков
text_df.loc[text_df['Sentiment'] == 1, 'Sentiment'] = 'pol'
text_df.loc[text_df['Sentiment'] == 0, 'Sentiment'] = 'otr'
text_df.sort_values(by=['Review'], inplace=True)
text_df
```

Out[ ]:

| | Review | Sentiment |
|---|---|---|
| **49906** | 'Bloody Birthday' is an odd and, at times, hum... | otr |
| **181** | 'Renaissance (2006)' was created over a period... | pol |
| **49867** | (SMALL SPOILERS) I just bought the DVD of this... | otr |
| **49995** | (spoiler) it could be the one the worst movie ... | otr |
| **49834** | **********POSSIBLE SPOILER********** Madonna p... | otr |
| **...** | ... | ... |
| **483** | this one is out there. Not much to say about i... | pol |
| **283** | very few chess movies have been made over the ... | pol |
| **492** | well, i said it all in the summary, i simpley ... | pol |
| **381** | when i first heard about this movie i thought ... | pol |
| **49888** | wow, i just got one watching this.<br /><br />... | otr |

1000 rows × 2 columns

In [ ]:
```
text_df.shape
```

Out[ ]:
```
(1000, 2)
```

In [ ]:
```
# Сформируем общий словарь для обучения моделей из обучающей и тестовой выборк
vocab_list = text_df['Review'].tolist()
vocab_list[1:10]
```

Out[ ]: ["'Renaissance (2006)' was created over a period of six years, co-funded by F
rance, Luxembourg and the United Kingdom at a cost of around \x8014 million.
The final result is a staggering accomplishment of comic-book style animatio
n, aesthetically similar to what Robert Rodriguez and Frank Miller achieved w
ith 'Sin City (2005),' but this film employed motion capture with live-actors
to translate their faces and movements into an entirely animated format. Pres
ented in stark black-and-white, the film looks as though it has been hoisted
from the very pages of the graphic novel on which it was based, and the futur
istic city of Paris looms ominously above us. Directed by French filmmaker Ch
ristian Volckman, in his feature-length debut, 'Renaissance' draws significan
tly from other films in the science-fiction genre, and the tech-noir storylin
e isn't something we haven't seen before, but, from a technical standpoint, i
t is faultless.<br /><br />The year is 2054. The city of Paris is a crumbling
metropolis filled with dark alleys and deserted footpaths, the recent install
ation of modern technology merely offering a thin mask to the pitiable degrad
ation of the darkened buildings. The city's largest corporation, Avalon, achi
eved wealth through offering citizens the promise of beauty and youth, and th
e company's research department is continually striving to invent greater mea
ns of eliminating the aging process. Ilona Tasuiev (voiced by Romola Garai in
the English-language version, which I watched) , a brilliant young scientist,
is mysteriously kidnapped on her return from work, and so it falls to legenda
ry detective Barthélémy Karas (Daniel Craig) to uncover her current whereabou
ts. Possibly holding the key to the woman's disappearance is Bislane (Catheri
ne McCormack), Ilona's hardened elder sister, whose trustworthiness is in que
stion, and Jonas Muller (Ian Holm), the dedicated medical doctor who adored I
lona as his own daughter.<br /><br />The eerie, dimly-lit city of Paris is re
miniscent of Ridley Scott's 'Blade Runner (1982),' and some of the technology
looks as though it might have been borrowed from Tom Cruise in 'Minority Repo
rt (2002)' {which was, coincidentally, also set in the year 2054}. However, d
espite this familiarity, Volckman has created an exciting world for his chara
cters to inhabit. Blending classic film-noir and science-fiction, the result
is an eye-catching collage of harsh lighting and dark shadows, which, I shoul
d warn, occasionally becomes difficult on the viewer's eyes. The dialogue is
a little banal at times, and the story, though engaging, doesn't offer anythi
ng strikingly original {except for the ending, which I thought was a bold twi
st on the usual formula}, but 'Renaissance' is intended to work best as a vis
ual treat, and that it succeeds in this regard cannot be denied.",
'(SMALL SPOILERS) I just bought the DVD of this movie yesterday. I saw it wi
th my friends and I couldn\'t believe what had happened.<br /><br />In the fi
rst 3 movies, the critters at least had a sense of humor (especially the 3rd
movie), but not only did the critters barely ever make an appearance, they we
ren\'t funny! They never made me laugh. I must admit that the story did start
off nicely. After an hour had gone by I remembered that the Critters movies w
ere always very short. So I thought to myself, "Where the $^%#$ are the critt
ers?!?!" They were barely in this movie! If that didn\'t make me mad enough,
the boy named Ethan was sitting on his bed after Charlie had "murdered the sh
ip" and he knew that the critters were still on board! In the first movie the
Brown family was scared out of their minds. But here, Ethan didn\'t even car
e! It was as if the critters weren\'t even a threat!<br /><br />Now what I\'m
about to say next may ruin the ending, but I\'m going to say it anyways. In t
he first movie, at the end, they had to face the giant critter for a final ba
ttle. In the second one, there was the great ball of critter. In the third mo
vie, the critter with his fave burned did a spindash (from Sonic the Hedgeho
g) and was going to attack the little kid. But at the end of the fourth one

(which is what made me the angriest) the bald critter charges toward Ethan, a
nd Ethan kills it as if it were nothing.<br /><br />Now something that I real
ly don\'t understand was what happened to Ug. He was one of my favorite chara
cters in the first two. Then after 50 years, he\'s evil. That was very disapp
ointing. Not only that, but wasn\'t he a faceless bounty hunter? Why was he s
till "Johnny Steele?" Plus he seemed to have a different personality. He seem
ed much smarter and not as monotone like in the first two.<br /><br />Being s
omeone who actually enjoyed the first two critters movies, and loved the thir
d one, I give Critters 4 a 2/10',
 "(spoiler) it could be the one the worst movie you see, you might like it li
ke I did I really like it. Its one of those odds movie. <br /><br />There is
man who seen to have a had day in life. Blacks rats some how feel sorry him
(which I think was a good Idea).<br /><br />The killer rats become friends wi
th man two big man come making feel unwanted so the man set the killer rat on
them and floor to floor both bodies covered in big black rats not that much b
lood.<br /><br />but think about big black rats all over body) but start to l
ittle killing people but rats are going over-bored until the rats kills his f
riends and girlfriends, <br /><br />Ther one scene in were seating on the toi
let while rats are going into the pipes leading to toilet and rat goes up his
you know and come out of his mouth, (which mean the rats must off eaten every
thing inside in body's) I had me laughing for weeks<br /><br />why did i like
this movie, yes it's different of the rest, I for ONE like it when little cre
ature takes on mankind. <br /><br />if you have seen any of these movie slug
s, slither, Them, spiders, snakes, tremors ,Cujo, crocodile, shark, octopu
s.<br /><br />If you liked them and check out Hood rats, <br /><br />it bette
r then Terror toons that all I Can say! 4/10",
 '**********POSSIBLE SPOILER********** Madonna plays an ex-con that needs to
recover some valuable information that might clear her from the murder that s
he was put in prison for four years ago. Griffin Dunne is a tax attorney wh
o\'s marrying his boss\' daughter. Together, the two of them are supposed to
come together in a world where chaos keeps you from getting on the bus...<br
/><br />When you get down to it, this is a stupid movie. Without trying to gi
ve away the plot ****POSSIBLE SPOILER****, the bad guys in the movie are tryi
ng to protect their boss by retrieving the information that would incriminate
them for the murder that Madonna was sent up for. What kind of bad guys do
n\'t commit murder by trying to hide the original murder?!?!?!<br /><br />The
n there\'s the cops who are trailing Madonna who follow the bad guys in a lim
o, where they have the brides-maids all tied up! And let\'s not forget those
same brides-maids who fought from the front gate to the front door, still all
tied together! And I hate to say this, but that patagonious feline sure looks
like a cougar! There might be only four of them in the New York City area, so
they might be endangered there, but I know there\'s plenty of them in the Roc
ky Mountains (see "Charlie the Lonesome Cougar" if you really want to see a l
arge "cat" in the movie). And let\'s look at the old man who falls asleep on
his feet... NAW!<br /><br />The plot is there, but that\'s all there is to th
is movie. I was barely out of my teens when this movie originally came out an
d I was some-what of a fan of Madonna, but that was the only reason I liked t
he movie, but since then, she\'s fell out of popularity with me, and I\'ve fa
ced the fact that she is just a terrible actress (good thing she\'s got that
singing career to fall back on). It rated maybe a "5" back then, but it\'s fa
llen to "barely making a 2" over the years.',
 "-SPOILERS------------ I am a fan of 60's-70's french cinema but not necessa
rily of the more modern,so to be honest i watched this because of Bellucci.Sh
e is very young here,extremely beautiful and on top of this supposedly this m

ovie is where they met with Cassel,so it gives it some extra importance.<br /><br />The movie begins with a very nice style reminiscent of DePalma.Then suddenly we are thrown to flashback,and the back and forth goes on which gets tiring.I don't mind one flash back,but do it and get it over with man!!!Anyway,the movie is still interesting to me until a point when the first and definite hole in the plot,that allows for the rest of the story,never lets me enjoy the rest.I can allow for little holes here and there,but not to base an entire plot on hot air.This is the story of a man who is literally searching for an old flame.This is the main plot.I will go along,when the story at some point will convince me that there are really mysterious things going on,but in this story there's nothing really mysterious.Bellucci-Cassel are a couple ,then Bellucci urgently has to leave for some job in Italy(not the farthest place on earth from Paris)and she leaves him a message,which for reasons later explained he doesn't get.OK,so what?Don't these people have phones?Supposedly she was away for 2months(not a century exactly) and wouldn't she call her boyfriend in Paris to see how he's doing? Of course not.Instead,even after she gets back she forgets all about him.And thats fine,but later in the movie she tells her friend that it was her greatest love and was ready to commit for the first time in her life.Yet she failed to give him a call for 2months and then never tried to get back with him.And what about Cassel's character?He was supposedly unable to locate her in Italy,really hard to find someone in Italy,its probably like Siberia,especially an actress who is probably listed even in the arts papers.And after 2months when she would be back,really hard to find her and ask for an explanation. One thinks she wanted to avoid him,but no,we find out they simply couldn't meet.So hard to meet in Paris. OK,i don't need to go further,because this is the incident where the entire movie is based. What is even worse,Bellucci is not really the star of this movie but this other girl Bohringer is.",
 ". . .but it was on a UHF channel and the reception was very fuzzy. I'd really like to own the movie since the reason I watched it in the first place is because I am a bus driver and at the time I saw this movie, I was driving that model bus. It was only (during his murder trial some 15 years later) that I remember vaguely that OJ was one of the stars in it. I only recall that he was the driver and of the bus' being shot up and driven wildly. I've been looking all over for this movie to no avail, since viewing it in the mid-80s. I liked the movie, I don't usually watch thrillers, but after reading the summary in the TV guide, and viewing its beginning (although fuzzy) I stayed for the whole thing.",
 '... And it\'s a not very good documentary either American MOVIE seems to have confused some people into thinking this is a spoof documentary ( " Mockumentary ) and even some newspaper TV listings described it as such . I\'ll not laugh out loud at that because it\'s easy to mistake this documentary as one big wind up ala THIS IS SPINAL TAP <br /><br />What seems to have caused the confusion is that the documentary centres around budding film maker Mark Borchardt who is .... How can I put it ? Rather self deluded ? Yes but that\'s not necessarily a bad thing since if we had no dreams we\'d all still be living in caves and the fact that Mark is obsessed with horror movies is not to be taken as a criticism since both Sam Raimi ( Yes that one ) and Peter Jackson ( yes that one ) both started out doing low budget horror comedies so again it\'s not a criticism . No it\'s just that Mark Borchardt ( yes that one ) is a parody of American trailer trash <br /><br />Remember in THERE\'S SOMETHING ABOUT MARY Ben Stiller gives a lift to a dodgy hitch hiker ( " Come into my office because you\'re f***in\' fired " ) ? Well that\'s who Mark resembles along with most of Jerry Springer\'s guests so it\'s very easy to see why some

```
people thought this wasn\'t a real documentary . It\'s also not a very good d
ocumentary since Mark and co give me the creeps . Did you know that someone t
hought Mark would grow up to be a serial killer ? Does anyone else think ther
e\'s plenty of time left for this to happen ?',
 "... Oxford, Mississippi, at least. Okay, the Paris we get is Paris, Culver
City apart from the Establishing library footage of the real McCoy but it IS
Paris in spirit than which nothing, nowhere, is better. Okay, Kelly is no Ast
aire but then who is and Caron is no Hepburn, ditto but Alan Lerner is light
years ahead of the vastly overrated Comden and Green who scripted Kelly's oth
er 'big' 50s musical Singin' In The Rain (a curious replication of lyricists
writing screenplays featuring songs by OTHER lyricists and just to balance th
ings the Gershwin numbers are far superior to the Arthur Freed/Nacio Herb Bro
wn numbers so Alan Lerner didn't have to feel too outclassed). The story need
n't detain us any more than the anomalies -Kelly hasn't got change of a match
and is a painter, i.e. bohemian, yet he is able to scare up a perfectly good
suit at a few hours notice when Foch invites him to dinner at her hotel; in t
he well-documented Love Is Here To Stay sequence the lovers are strangely unm
olested by passers-by, other lovers and the bridge in the background is total
ly free of both pedestrian and vehicular traffic - this is, after all, a feel
good musical so it stands or falls by the score and in this case it stands fo
ur square. As feel good musicals go it's definitely in the top 10.",
 '... and I DO mean it. If not literally (after all, I have not seen every mo
vie ever created!), at least, obviously, among the ones, the many I know.<br
/><br />5.3 ??? The rule of thumb with IMDb is this: sometimes movies rated v
ery highly (for example, the piece of Kannes-Kompetition-Krowned-Korean-Kraap
called "Oldboy") can be truly bad. But rarely a movie worth watching is actua
lly rated under 6. This movie, very much worth watching, is. A disgrace.<br
/><br />True, I give it a 10 in protest. The movie is not perfect. Its true r
ating should be an 8 or a 9. It has some acting flaws (Belafonte especially),
the script wanders around, sometimes. However, what we have here is one of th
e greatest directors of all times, the Czech Jan Kadar, directing two of the
greatest actors of all time, the beloved, larger-than-life Zero Mostel and th
e sublime Ida Kaminska in an acting/poetic/moral tour de force. A pair made i
n Heaven! It\'s true that this movie, little flaws apart, does not pander to
the average audiences, but those interested in watching an excellent (while,
again, not beyond criticism) movie of the incomparable director who gave us
"The Shop on the Main Street" (the best movie ever about Holocaust) should no
t miss this just because some silly IMDb rating system decides that "American
Beauty" is better than "The Angel Levine".<br /><br />It isn\'t.']
```

In [ ]:
```
vocabVect = CountVectorizer()
vocabVect.fit(vocab_list)
corpusVocab = vocabVect.vocabulary_
print('Количество сформированных признаков - {}'.format(len(corpusVocab)))
```

```
Количество сформированных признаков - 17896
```

In [ ]:
```
for i in list(corpusVocab)[0:10]:
    print('{}={}'.format(i, corpusVocab[i]))
```

```
bloody=1874
birthday=1772
is=8498
an=781
odd=11070
and=798
at=1158
times=16139
humorous=7838
low=9560
```

# Векторизация признаков на основе CountVectorizer

Подсчитывает количество слов словаря, входящих в данный текст

In [ ]:
```
test_features = vocabVect.transform(vocab_list)
```

In [ ]:
```
test_features
```

Out[ ]:
```
<1000x17896 sparse matrix of type '<class 'numpy.int64'>'
        with 137926 stored elements in Compressed Sparse Row format>
```

In [ ]:
```
test_features.todense()
```

Out[ ]:
```
matrix([[0, 0, 0, ..., 0, 0, 0],
        [0, 0, 0, ..., 0, 0, 0],
        [0, 0, 0, ..., 0, 0, 0],
        ...,
        [0, 0, 0, ..., 0, 0, 0],
        [0, 0, 0, ..., 0, 0, 0],
        [0, 0, 0, ..., 0, 0, 0]], dtype=int64)
```

In [ ]:
```
# Размер нулевой строки
len(test_features.todense()[0].getA1())
```

Out[ ]:
```
17896
```

In [ ]:
```
# Непустые значения нулевой строки
[i for i in test_features.todense()[0].getA1() if i>0]
```

Out[ ]: [1,
1,
2,
1,
1,
1,
1,
2,
7,
1,
3,
1,
5,
3,
1,
2,
1,
2,
1,
2,
1,
1,
2,
3,
6,
1,
2,
1,
1,
1,
1,
1,
2,
1,
1,
1,
1,
1,
1,
1,
1,
2,
1,
1,
1,
1,
1,
1,
2,
2,
1,
2,
1,

1,
1,
1,
1,
1,
1,
1,
2,
1,
1,
5,
1,
1,
1,
1,
1,
6,
5,
1,
2,
1,
1,
2,
1,
1,
1,
4,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
7,
1,
2,
2,
2,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,

1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
1,
2,
1,
1,
17,
3,
2,
1,
4,
2,
3,
1,
1,
5,
1,
1,
2,
1,
1,
1,
1,
1,
1,
2,
1,
1,
2,
1,
1,
1,
2,
2,
3,

```
1,
1,
1,
1,
2,
1]
```

In [ ]:
```
def VectorizeAndClassify(vectorizers_list, classifiers_list):
    for v in vectorizers_list:
        for c in classifiers_list:
            pipeline1 = Pipeline([("vectorizer", v), ("classifier", c)])
            score = cross_val_score(pipeline1, text_df['Review'], text_df['Ser
            print('Векторизация - {}'.format(v))
            print('Модель для классификации - {}'.format(c))
            print('Accuracy = {}'.format(score))
            print('============================')
```

In [ ]:
```
vectorizers_list = [CountVectorizer(vocabulary = corpusVocab)]
classifiers_list = [LogisticRegression(C=3.0), LinearSVC(), KNeighborsClassifi
VectorizeAndClassify(vectorizers_list, classifiers_list)
```

```
c:\users\sveta\documents\virtualenvs\tensorflow\lib\site-packages\sklearn\linea
r_model\_logistic.py:765: ConvergenceWarning: lbfgs failed to converge (statu
s=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regressi
on
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
c:\users\sveta\documents\virtualenvs\tensorflow\lib\site-packages\sklearn\linea
r_model\_logistic.py:765: ConvergenceWarning: lbfgs failed to converge (statu
s=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regressi
on
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
c:\users\sveta\documents\virtualenvs\tensorflow\lib\site-packages\sklearn\linea
r_model\_logistic.py:765: ConvergenceWarning: lbfgs failed to converge (statu
s=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regressi
on
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
```

```
Векторизация - CountVectorizer(vocabulary={'00': 0, '000': 1, '0069': 2, '007':
3, '01': 4,
                               '06th': 5, '08': 6, '0f': 7, '10': 8, '100': 9,
                               '100th': 10, '101': 11, '102': 12, '10th': 13,
                               '11': 14, '112': 15, '11th': 16, '12': 17, '13': 1
8,
                               '13th': 19, '14': 20, '14th': 21, '15': 22,
                               '150': 23, '16': 24, '1600s': 25, '16éme': 26,
                               '17': 27, '1710': 28, '18': 29, ...})
Модель для классификации - LogisticRegression(C=3.0)
Accuracy = 0.7819855783927641
============================
Векторизация - CountVectorizer(vocabulary={'00': 0, '000': 1, '0069': 2, '007':
3, '01': 4,
                               '06th': 5, '08': 6, '0f': 7, '10': 8, '100': 9,
                               '100th': 10, '101': 11, '102': 12, '10th': 13,
                               '11': 14, '112': 15, '11th': 16, '12': 17, '13': 1
8,
                               '13th': 19, '14': 20, '14th': 21, '15': 22,
                               '150': 23, '16': 24, '1600s': 25, '16éme': 26,
                               '17': 27, '1710': 28, '18': 29, ...})
Модель для классификации - LinearSVC()
Accuracy = 0.779977582372792
============================
Векторизация - CountVectorizer(vocabulary={'00': 0, '000': 1, '0069': 2, '007':
3, '01': 4,
                               '06th': 5, '08': 6, '0f': 7, '10': 8, '100': 9,
                               '100th': 10, '101': 11, '102': 12, '10th': 13,
                               '11': 14, '112': 15, '11th': 16, '12': 17, '13': 1
8,
                               '13th': 19, '14': 20, '14th': 21, '15': 22,
                               '150': 23, '16': 24, '1600s': 25, '16éme': 26,
                               '17': 27, '1710': 28, '18': 29, ...})
Модель для классификации - KNeighborsClassifier()
Accuracy = 0.5729981478484473
============================
```

# Разделим на обучающую и тестовую выборки

In [ ]:
```
X_train, X_test, y_train, y_test = train_test_split(text_df['Review'], text_df
```

In [ ]:
```
def sentiment(v, c):
    model = Pipeline(
        [("vectorizer", v),
         ("classifier", c)])
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    print_accuracy_score_for_classes(y_test, y_pred)
```

In [ ]:
```
sentiment(CountVectorizer(), LogisticRegression(C=3.0))
```

```
Метка          Accuracy
otr          0.7765151515151515
pol          0.7669491525423728
```

```
c:\users\sveta\documents\virtualenvs\tensorflow\lib\site-packages\sklearn\linea
r_model\_logistic.py:765: ConvergenceWarning: lbfgs failed to converge (statu
s=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regressi
on
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
```

# Классификация текста на основе моделей word2vec

In [ ]:
```
import gensim
from gensim.models import word2vec
```

In [ ]:
```
import re
from typing import Dict, Tuple
from sklearn.metrics import accuracy_score, balanced_accuracy_score
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import Pipeline
from nltk import WordPunctTokenizer
from nltk.corpus import stopwords
import nltk
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\sveta\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Out[ ]:
```
True
```

In [ ]:
```
# Подготовим корпус
corpus = []
stop_words = stopwords.words('english')
tok = WordPunctTokenizer()
for line in text_df['Review'].values:
    line1 = line.strip().lower()
    line1 = re.sub("[^a-zA-Z]"," ", line1)
    text_tok = tok.tokenize(line1)
    text_tok1 = [w for w in text_tok if not w in stop_words]
    corpus.append(text_tok1)
```

In [ ]:
```
corpus[:10]
```

Out[ ]:
```
[['bloody',
  'birthday',
  'odd',
  'times',
  'humorous',
  'low',
  'budget',
  'horror',
  'flick',
  'along',
  'lines',
  'mikey',
  'less',
  'intelligent',
  'version',
  'good',
  'son',
  'br',
  'br',
  'set',
  'small',
  'californian',
  'town',
  'three',
  'babies',
  'born',
  'height',
  'eclipse',
  'planetary',
  'alignment',
  'means',
  'somehow',
  'born',
  'without',
  'emotions',
  'ten',
  'years',
  'later',
  'three',
  'little',
  'psychopaths',
  'take',
  'killing',
  'spree',
  'away',
  'parents',
  'siblings',
  'teachers',
  'anyone',
  'else',
  'irritates',
  'one',
  'teenage',
  'girl',
```

```
'knows',
'truth',
'able',
'stop',
'explanation',
'babies',
'across',
'world',
'born',
'time',
'equally',
'twisted',
'go',
'br',
'br',
'slasher',
'film',
'tame',
'terms',
'violence',
'gore',
'suppose',
'highlights',
'problem',
'casting',
'child',
'characters',
'killers',
'much',
'expose',
'young',
'actors',
'instead',
'amusing',
'little',
'disturbing',
'seeing',
'three',
'ten',
'year',
'olds',
'plotting',
'murders',
'carrying',
'plans',
'using',
'guns',
'knives',
'crossbows',
'main',
'reason',
'descend',
'totally',
'ridiculous',
```

```
 'child',
 'actors',
 'convincing',
 'roles',
 'way',
 'slyly',
 'play',
 'little',
 'innocents',
 'front',
 'undiscerning',
 'adults',
 'showing',
 'dark',
 'side',
 'girl',
 'knows',
 'truth',
 'br',
 'br',
 'bloody',
 'birthday',
 'rather',
 'mediocre',
 'horror',
 'flick',
 'scares',
 'little',
 'blood',
 'shock',
 'factor',
 'kids',
 'killers',
 'bit',
 'unique',
 'way',
 'one',
 'watch',
 'nothing',
 'else'],
['renaissance',
 'created',
 'period',
 'six',
 'years',
 'co',
 'funded',
 'france',
 'luxembourg',
 'united',
 'kingdom',
 'cost',
 'around',
 'million',
```

```
'final',
'result',
'staggering',
'accomplishment',
'comic',
'book',
'style',
'animation',
'aesthetically',
'similar',
'robert',
'rodriguez',
'frank',
'miller',
'achieved',
'sin',
'city',
'film',
'employed',
'motion',
'capture',
'live',
'actors',
'translate',
'faces',
'movements',
'entirely',
'animated',
'format',
'presented',
'stark',
'black',
'white',
'film',
'looks',
'though',
'hoisted',
'pages',
'graphic',
'novel',
'based',
'futuristic',
'city',
'paris',
'looms',
'ominously',
'us',
'directed',
'french',
'filmmaker',
'christian',
'volckman',
'feature',
'length',
```

```
'debut',
'renaissance',
'draws',
'significantly',
'films',
'science',
'fiction',
'genre',
'tech',
'noir',
'storyline',
'something',
'seen',
'technical',
'standpoint',
'faultless',
'br',
'br',
'year',
'city',
'paris',
'crumbling',
'metropolis',
'filled',
'dark',
'alleys',
'deserted',
'footpaths',
'recent',
'installation',
'modern',
'technology',
'merely',
'offering',
'thin',
'mask',
'pitiable',
'degradation',
'darkened',
'buildings',
'city',
'largest',
'corporation',
'avalon',
'achieved',
'wealth',
'offering',
'citizens',
'promise',
'beauty',
'youth',
'company',
'research',
'department',
```

```
'continually',
'striving',
'invent',
'greater',
'means',
'eliminating',
'aging',
'process',
'ilona',
'tasuiev',
'voiced',
'romola',
'garai',
'english',
'language',
'version',
'watched',
'brilliant',
'young',
'scientist',
'mysteriously',
'kidnapped',
'return',
'work',
'falls',
'legendary',
'detective',
'barth',
'l',
'karas',
'daniel',
'craig',
'uncover',
'current',
'whereabouts',
'possibly',
'holding',
'key',
'woman',
'disappearance',
'bislane',
'catherine',
'mccormack',
'ilona',
'hardened',
'elder',
'sister',
'whose',
'trustworthiness',
'question',
'jonas',
'muller',
'ian',
'holm',
```

```
'dedicated',
'medical',
'doctor',
'adored',
'ilona',
'daughter',
'br',
'br',
'eerie',
'dimly',
'lit',
'city',
'paris',
'reminiscent',
'ridley',
'scott',
'blade',
'runner',
'technology',
'looks',
'though',
'might',
'borrowed',
'tom',
'cruise',
'minority',
'report',
'coincidentally',
'also',
'set',
'year',
'however',
'despite',
'familiarity',
'volckman',
'created',
'exciting',
'world',
'characters',
'inhabit',
'blending',
'classic',
'film',
'noir',
'science',
'fiction',
'result',
'eye',
'catching',
'collage',
'harsh',
'lighting',
'dark',
'shadows',
```

```
 'warn',
 'occasionally',
 'becomes',
 'difficult',
 'viewer',
 'eyes',
 'dialogue',
 'little',
 'banal',
 'times',
 'story',
 'though',
 'engaging',
 'offer',
 'anything',
 'strikingly',
 'original',
 'except',
 'ending',
 'thought',
 'bold',
 'twist',
 'usual',
 'formula',
 'renaissance',
 'intended',
 'work',
 'best',
 'visual',
 'treat',
 'succeeds',
 'regard',
 'cannot',
 'denied'],
['small',
 'spoilers',
 'bought',
 'dvd',
 'movie',
 'yesterday',
 'saw',
 'friends',
 'believe',
 'happened',
 'br',
 'br',
 'first',
 'movies',
 'critters',
 'least',
 'sense',
 'humor',
 'especially',
 'rd',
```

```
'movie',
'critters',
'barely',
'ever',
'make',
'appearance',
'funny',
'never',
'made',
'laugh',
'must',
'admit',
'story',
'start',
'nicely',
'hour',
'gone',
'remembered',
'critters',
'movies',
'always',
'short',
'thought',
'critters',
'barely',
'movie',
'make',
'mad',
'enough',
'boy',
'named',
'ethan',
'sitting',
'bed',
'charlie',
'murdered',
'ship',
'knew',
'critters',
'still',
'board',
'first',
'movie',
'brown',
'family',
'scared',
'minds',
'ethan',
'even',
'care',
'critters',
'even',
'threat',
'br',
```

```
'br',
'say',
'next',
'may',
'ruin',
'ending',
'going',
'say',
'anyways',
'first',
'movie',
'end',
'face',
'giant',
'critter',
'final',
'battle',
'second',
'one',
'great',
'ball',
'critter',
'third',
'movie',
'critter',
'fave',
'burned',
'spindash',
'sonic',
'hedgehog',
'going',
'attack',
'little',
'kid',
'end',
'fourth',
'one',
'made',
'angriest',
'bald',
'critter',
'charges',
'toward',
'ethan',
'ethan',
'kills',
'nothing',
'br',
'br',
'something',
'really',
'understand',
'happened',
'ug',
```

```
 'one',
 'favorite',
 'characters',
 'first',
 'two',
 'years',
 'evil',
 'disappointing',
 'faceless',
 'bounty',
 'hunter',
 'still',
 'johnny',
 'steele',
 'plus',
 'seemed',
 'different',
 'personality',
 'seemed',
 'much',
 'smarter',
 'monotone',
 'like',
 'first',
 'two',
 'br',
 'br',
 'someone',
 'actually',
 'enjoyed',
 'first',
 'two',
 'critters',
 'movies',
 'loved',
 'third',
 'one',
 'give',
 'critters'],
['spoiler',
 'could',
 'one',
 'worst',
 'movie',
 'see',
 'might',
 'like',
 'like',
 'really',
 'like',
 'one',
 'odds',
 'movie',
 'br',
```

```
'br',
'man',
'seen',
'day',
'life',
'blacks',
'rats',
'feel',
'sorry',
'think',
'good',
'idea',
'br',
'br',
'killer',
'rats',
'become',
'friends',
'man',
'two',
'big',
'man',
'come',
'making',
'feel',
'unwanted',
'man',
'set',
'killer',
'rat',
'floor',
'floor',
'bodies',
'covered',
'big',
'black',
'rats',
'much',
'blood',
'br',
'br',
'think',
'big',
'black',
'rats',
'body',
'start',
'little',
'killing',
'people',
'rats',
'going',
'bored',
'rats',
```

```
'kills',
'friends',
'girlfriends',
'br',
'br',
'ther',
'one',
'scene',
'seating',
'toilet',
'rats',
'going',
'pipes',
'leading',
'toilet',
'rat',
'goes',
'know',
'come',
'mouth',
'mean',
'rats',
'must',
'eaten',
'everything',
'inside',
'body',
'laughing',
'weeks',
'br',
'br',
'like',
'movie',
'yes',
'different',
'rest',
'one',
'like',
'little',
'creature',
'takes',
'mankind',
'br',
'br',
'seen',
'movie',
'slugs',
'slither',
'spiders',
'snakes',
'tremors',
'cujo',
'crocodile',
'shark',
```

```
 'octopus',
 'br',
 'br',
 'liked',
 'check',
 'hood',
 'rats',
 'br',
 'br',
 'better',
 'terror',
 'toons',
 'say'],
['possible',
 'spoiler',
 'madonna',
 'plays',
 'ex',
 'con',
 'needs',
 'recover',
 'valuable',
 'information',
 'might',
 'clear',
 'murder',
 'put',
 'prison',
 'four',
 'years',
 'ago',
 'griffin',
 'dunne',
 'tax',
 'attorney',
 'marrying',
 'boss',
 'daughter',
 'together',
 'two',
 'supposed',
 'come',
 'together',
 'world',
 'chaos',
 'keeps',
 'getting',
 'bus',
 'br',
 'br',
 'get',
 'stupid',
 'movie',
 'without',
```

```
'trying',
'give',
'away',
'plot',
'possible',
'spoiler',
'bad',
'guys',
'movie',
'trying',
'protect',
'boss',
'retrieving',
'information',
'would',
'incriminate',
'murder',
'madonna',
'sent',
'kind',
'bad',
'guys',
'commit',
'murder',
'trying',
'hide',
'original',
'murder',
'br',
'br',
'cops',
'trailing',
'madonna',
'follow',
'bad',
'guys',
'limo',
'brides',
'maids',
'tied',
'let',
'forget',
'brides',
'maids',
'fought',
'front',
'gate',
'front',
'door',
'still',
'tied',
'together',
'hate',
'say',
```

```
'patagonious',
'feline',
'sure',
'looks',
'like',
'cougar',
'might',
'four',
'new',
'york',
'city',
'area',
'might',
'endangered',
'know',
'plenty',
'rocky',
'mountains',
'see',
'charlie',
'lonesome',
'cougar',
'really',
'want',
'see',
'large',
'cat',
'movie',
'let',
'look',
'old',
'man',
'falls',
'asleep',
'feet',
'naw',
'br',
'br',
'plot',
'movie',
'barely',
'teens',
'movie',
'originally',
'came',
'fan',
'madonna',
'reason',
'liked',
'movie',
'since',
'fell',
'popularity',
'faced',
```

```
 'fact',
 'terrible',
 'actress',
 'good',
 'thing',
 'got',
 'singing',
 'career',
 'fall',
 'back',
 'rated',
 'maybe',
 'back',
 'fallen',
 'barely',
 'making',
 'years'],
['spoilers',
 'fan',
 'french',
 'cinema',
 'necessarily',
 'modern',
 'honest',
 'watched',
 'bellucci',
 'young',
 'extremely',
 'beautiful',
 'top',
 'supposedly',
 'movie',
 'met',
 'cassel',
 'gives',
 'extra',
 'importance',
 'br',
 'br',
 'movie',
 'begins',
 'nice',
 'style',
 'reminiscent',
 'depalma',
 'suddenly',
 'thrown',
 'flashback',
 'back',
 'forth',
 'goes',
 'gets',
 'tiring',
 'mind',
```

```
'one',
'flash',
'back',
'get',
'man',
'anyway',
'movie',
'still',
'interesting',
'point',
'first',
'definite',
'hole',
'plot',
'allows',
'rest',
'story',
'never',
'lets',
'enjoy',
'rest',
'allow',
'little',
'holes',
'base',
'entire',
'plot',
'hot',
'air',
'story',
'man',
'literally',
'searching',
'old',
'flame',
'main',
'plot',
'go',
'along',
'story',
'point',
'convince',
'really',
'mysterious',
'things',
'going',
'story',
'nothing',
'really',
'mysterious',
'bellucci',
'cassel',
'couple',
'bellucci',
```

```
'urgently',
'leave',
'job',
'italy',
'farthest',
'place',
'earth',
'paris',
'leaves',
'message',
'reasons',
'later',
'explained',
'get',
'ok',
'people',
'phones',
'supposedly',
'away',
'months',
'century',
'exactly',
'call',
'boyfriend',
'paris',
'see',
'course',
'instead',
'even',
'gets',
'back',
'forgets',
'thats',
'fine',
'later',
'movie',
'tells',
'friend',
'greatest',
'love',
'ready',
'commit',
'first',
'time',
'life',
'yet',
'failed',
'give',
'call',
'months',
'never',
'tried',
'get',
'back',
```

```
'cassel',
'character',
'supposedly',
'unable',
'locate',
'italy',
'really',
'hard',
'find',
'someone',
'italy',
'probably',
'like',
'siberia',
'especially',
'actress',
'probably',
'listed',
'even',
'arts',
'papers',
'months',
'would',
'back',
'really',
'hard',
'find',
'ask',
'explanation',
'one',
'thinks',
'wanted',
'avoid',
'find',
'simply',
'meet',
'hard',
'meet',
'paris',
'ok',
'need',
'go',
'incident',
'entire',
'movie',
'based',
'even',
'worse',
'bellucci',
'really',
'star',
'movie',
'girl',
'bohringer'],
```

```
['uhf',
 'channel',
 'reception',
 'fuzzy',
 'really',
 'like',
 'movie',
 'since',
 'reason',
 'watched',
 'first',
 'place',
 'bus',
 'driver',
 'time',
 'saw',
 'movie',
 'driving',
 'model',
 'bus',
 'murder',
 'trial',
 'years',
 'later',
 'remember',
 'vaguely',
 'oj',
 'one',
 'stars',
 'recall',
 'driver',
 'bus',
 'shot',
 'driven',
 'wildly',
 'looking',
 'movie',
 'avail',
 'since',
 'viewing',
 'mid',
 'liked',
 'movie',
 'usually',
 'watch',
 'thrillers',
 'reading',
 'summary',
 'tv',
 'guide',
 'viewing',
 'beginning',
 'although',
 'fuzzy',
```

```
 'stayed',
 'whole',
 'thing'],
['good',
 'documentary',
 'either',
 'american',
 'movie',
 'seems',
 'confused',
 'people',
 'thinking',
 'spoof',
 'documentary',
 'mockumentary',
 'even',
 'newspaper',
 'tv',
 'listings',
 'described',
 'laugh',
 'loud',
 'easy',
 'mistake',
 'documentary',
 'one',
 'big',
 'wind',
 'ala',
 'spinal',
 'tap',
 'br',
 'br',
 'seems',
 'caused',
 'confusion',
 'documentary',
 'centres',
 'around',
 'budding',
 'film',
 'maker',
 'mark',
 'borchardt',
 'put',
 'rather',
 'self',
 'deluded',
 'yes',
 'necessarily',
 'bad',
 'thing',
 'since',
 'dreams',
```

```
'still',
'living',
'caves',
'fact',
'mark',
'obsessed',
'horror',
'movies',
'taken',
'criticism',
'since',
'sam',
'raimi',
'yes',
'one',
'peter',
'jackson',
'yes',
'one',
'started',
'low',
'budget',
'horror',
'comedies',
'criticism',
'mark',
'borchardt',
'yes',
'one',
'parody',
'american',
'trailer',
'trash',
'br',
'br',
'remember',
'something',
'mary',
'ben',
'stiller',
'gives',
'lift',
'dodgy',
'hitch',
'hiker',
'come',
'office',
'f',
'fired',
'well',
'mark',
'resembles',
'along',
'jerry',
```

```
 'springer',
 'guests',
 'easy',
 'see',
 'people',
 'thought',
 'real',
 'documentary',
 'also',
 'good',
 'documentary',
 'since',
 'mark',
 'co',
 'give',
 'creeps',
 'know',
 'someone',
 'thought',
 'mark',
 'would',
 'grow',
 'serial',
 'killer',
 'anyone',
 'else',
 'think',
 'plenty',
 'time',
 'left',
 'happen'],
['oxford',
 'mississippi',
 'least',
 'okay',
 'paris',
 'get',
 'paris',
 'culver',
 'city',
 'apart',
 'establishing',
 'library',
 'footage',
 'real',
 'mccoy',
 'paris',
 'spirit',
 'nothing',
 'nowhere',
 'better',
 'okay',
 'kelly',
 'astaire',
```

```
'caron',
'hepburn',
'ditto',
'alan',
'lerner',
'light',
'years',
'ahead',
'vastly',
'overrated',
'comden',
'green',
'scripted',
'kelly',
'big',
'musical',
'singin',
'rain',
'curious',
'replication',
'lyricists',
'writing',
'screenplays',
'featuring',
'songs',
'lyricists',
'balance',
'things',
'gershwin',
'numbers',
'far',
'superior',
'arthur',
'freed',
'nacio',
'herb',
'brown',
'numbers',
'alan',
'lerner',
'feel',
'outclassed',
'story',
'detain',
'us',
'anomalies',
'kelly',
'got',
'change',
'match',
'painter',
'e',
'bohemian',
'yet',
```

```
 'able',
 'scare',
 'perfectly',
 'good',
 'suit',
 'hours',
 'notice',
 'foch',
 'invites',
 'dinner',
 'hotel',
 'well',
 'documented',
 'love',
 'stay',
 'sequence',
 'lovers',
 'strangely',
 'unmolested',
 'passers',
 'lovers',
 'bridge',
 'background',
 'totally',
 'free',
 'pedestrian',
 'vehicular',
 'traffic',
 'feelgood',
 'musical',
 'stands',
 'falls',
 'score',
 'case',
 'stands',
 'four',
 'square',
 'feel',
 'good',
 'musicals',
 'go',
 'definitely',
 'top'],
['mean',
 'literally',
 'seen',
 'every',
 'movie',
 'ever',
 'created',
 'least',
 'obviously',
 'among',
 'ones',
```

```
'many',
'know',
'br',
'br',
'rule',
'thumb',
'imdb',
'sometimes',
'movies',
'rated',
'highly',
'example',
'piece',
'kannes',
'kompetition',
'krowned',
'korean',
'kraap',
'called',
'oldboy',
'truly',
'bad',
'rarely',
'movie',
'worth',
'watching',
'actually',
'rated',
'movie',
'much',
'worth',
'watching',
'disgrace',
'br',
'br',
'true',
'give',
'protest',
'movie',
'perfect',
'true',
'rating',
'acting',
'flaws',
'belafonte',
'especially',
'script',
'wanders',
'around',
'sometimes',
'however',
'one',
'greatest',
'directors',
```

```
'times',
'czech',
'jan',
'kadar',
'directing',
'two',
'greatest',
'actors',
'time',
'beloved',
'larger',
'life',
'zero',
'mostel',
'sublime',
'ida',
'kaminska',
'acting',
'poetic',
'moral',
'tour',
'de',
'force',
'pair',
'made',
'heaven',
'true',
'movie',
'little',
'flaws',
'apart',
'pander',
'average',
'audiences',
'interested',
'watching',
'excellent',
'beyond',
'criticism',
'movie',
'incomparable',
'director',
'gave',
'us',
'shop',
'main',
'street',
'best',
'movie',
'ever',
'holocaust',
'miss',
'silly',
'imdb',
```

```
 'rating',
 'system',
 'decides',
 'american',
 'beauty',
 'better',
 'angel',
 'levine',
 'br',
 'br']]
```

In [ ]:
```
%time
model_imdb = word2vec.Word2Vec(corpus, workers=4, min_count=10, window=10, sam
```

```
Wall time: 0 ns
```

In [ ]:
```
# Проверим, что модель обучилась
print(model_imdb.wv.most_similar(positive=['find'], topn=5))
```

```
[('someone', 0.9996957778930664), ('audience', 0.9996758103370667), ('far', 0.9
996746182441711), ('everything', 0.9996718168258667), ('nothing', 0.99967104196
54846)]
```

In [ ]:
```
def sentiment(v, c):
    model = Pipeline(
        [("vectorizer", v),
         ("classifier", c)])
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    print_accuracy_score_for_classes(y_test, y_pred)
```

In [ ]:
```
class EmbeddingVectorizer(object):
    '''
    Для текста усредним вектора входящих в него слов
    '''
    def __init__(self, model):
        self.model = model
        self.size = model.vector_size

    def fit(self, X, y):
        return self

    def transform(self, X):
        return np.array([np.mean(
            [self.model[w] for w in words if w in self.model]
            or [np.zeros(self.size)], axis=0)
            for words in X])
```

In [ ]:
```
def accuracy_score_for_classes(
    y_true: np.ndarray,
    y_pred: np.ndarray) -> Dict[int, float]:
    """
    Вычисление метрики accuracy для каждого класса
```

```
    y_true - истинные значения классов
    y_pred - предсказанные значения классов
    Возвращает словарь: ключ - метка класса,
    значение - Accuracy для данного класса
    """
    # Для удобства фильтрации сформируем Pandas DataFrame
    d = {'t': y_true, 'p': y_pred}
    df = pd.DataFrame(data=d)
    # Метки классов
    classes = np.unique(y_true)
    # Результирующий словарь
    res = dict()
    # Перебор меток классов
    for c in classes:
        # отфильтруем данные, которые соответствуют
        # текущей метке класса в истинных значениях
        temp_data_flt = df[df['t']==c]
        # расчет accuracy для заданной метки класса
        temp_acc = accuracy_score(
            temp_data_flt['t'].values,
            temp_data_flt['p'].values)
        # сохранение результата в словарь
        res[c] = temp_acc
    return res

def print_accuracy_score_for_classes(
    y_true: np.ndarray,
    y_pred: np.ndarray):
    """
    Вывод метрики accuracy для каждого класса
    """
    accs = accuracy_score_for_classes(y_true, y_pred)
    if len(accs)>0:
        print('Метка \t Accuracy')
    for i in accs:
        print('{} \t {}'.format(i, accs[i]))
```

In [ ]:
```
# Обучающая и тестовая выборки
boundary = 900
X_train = corpus[:boundary]
X_test = corpus[boundary:]
y_train = text_df['Sentiment'][:boundary]
y_test = text_df['Sentiment'][boundary:]
```

In [ ]:
```
sentiment(EmbeddingVectorizer(model_imdb.wv), LogisticRegression(C=3.0))
```

```
Метка          Accuracy
otr          0.5319148936170213
pol          0.5094339622641509
```

Наибольшая точность получилась при использовании CountVectorizer и LogisticRegression