Липецкий государственный технический университет

Факультет автоматизации и информатики

Кафедра автоматизированных систем управления

ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине «Прикладные интеллектуальные системы и экспертные системы»

Предварительная обработка текстовых данных

Студент Сухоруких А.О.

Группа М-ИАП-22

Руководитель Кургасов В.В.

Липецк 2022 г.

Задание кафедры

Вариант 2

1) В среде Jupiter Notebook создать новый ноутбук (Notebook);

2) Импортировать необходимые для работы библиотеки и модули;

3) Загрузить обучающую и экзаменационную выборку в соответствие с вариантом;

4) Вывести на экран по одному-два документа каждого класса;

5) Применить стемминг, записав обработанные выборки (тестовую и обучающую) в новые переменные;

6) Провести векторизацию выборки:

a. Векторизовать обучающую и тестовую выборки простым подсчетом слов (CountVectorizer) и значеним max_features = 10000

b. Вывести и проанализировать первые 20 наиболее частотных слов всей выборки и каждого класса по-отдельности.

c. Применить процедуру отсечения стоп-слов и повторить пункт b.

d. Провести пункты a – c для обучающей и тестовой выборки, для которой проведена процедура стемминга.

e. Векторизовать выборки с помощью TfidfTransformer (с использованием TF и TF-IDF взвешиваний) и повторить пункты b-d.

7) По результатам пункта 6 заполнить таблицы наиболее частотными терминами обучающей выборки и каждого класса по отдельности.

Всего должно получиться по 4 таблицы для выборки, к которой применялась операция стемминга и 4 таблицы для выборки, к которой операция стемминга не применялась

8) Используя конвейер (Pipeline) реализовать модель Наивного Байесовского классификатора и выявить на основе показателей качества (значения полноты, точности, f1-меры и аккуратности), какая предварительная обработка данных обеспечит наилучшие результаты классификации. Должны быть исследованы следующие характеристики:

• Наличие - отсутствие стемминга

• Отсечение – не отсечение стоп-слов

• Количество информативных терминов (max_features)

• Взвешивание: Count, TF, TF-IDF

9) По каждому пункту работы занести в отчет программный код и результат вывода.

10) По результатам классификации занести в отчет выводы о наиболее подходящей предварительной обработке данных (наличие стемминга, взвешивание терминов, стоп-слова, количество информативных терминов).

Классы: 'comp.windows.x', 'rec.sport.baseball', 'rec.sport.hockey'

Ход работы

Импортируем необходимые для работы библиотеки и модули.

- pandas — программная библиотека на языке Python для обработки и анализа данных;

- numPy (сокращенно от Numerical Python)— библиотека с открытым исходным кодом для языка программирования Python. Возможности: поддержка многомерных массивов (включая матрицы); поддержка высокоуровневых математических функций, предназначенных для работы с многомерными массивами;

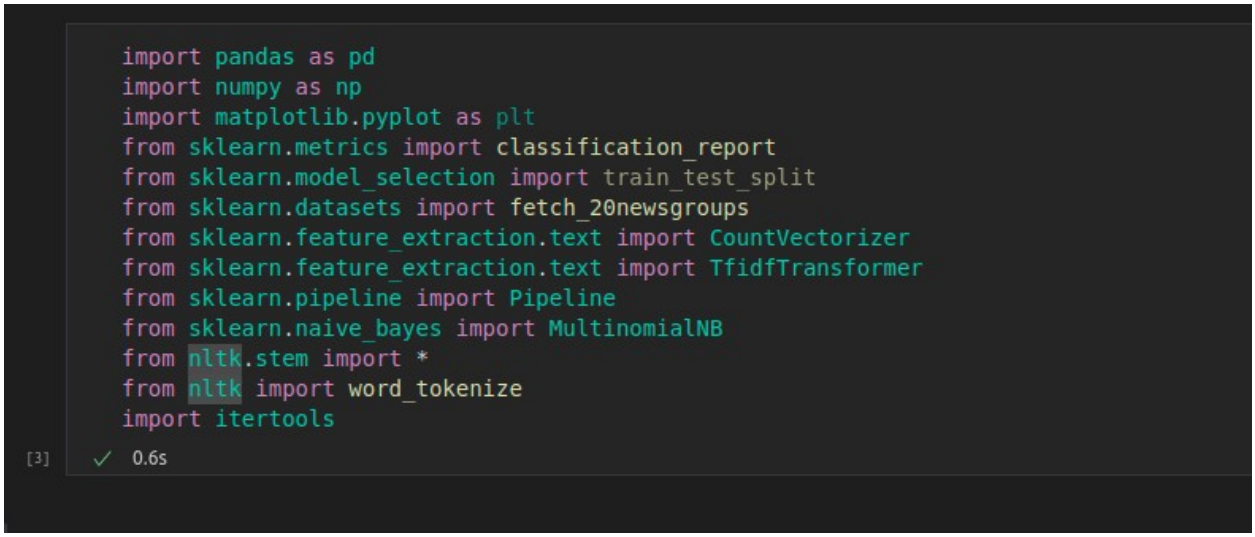
- matplotlib — библиотека на языке программирования Python для визуализации данных двумерной и трёхмерной графикой;

- библиотека NLTK — пакет библиотек и программ для символьной и статистической обработки естественного языка, написанных на языке программирования Python. Содержит графические представления и примеры данных;

- itertools стандартизирует основной набор быстрых эффективных по памяти инструментов, которые полезны сами по себе или в связке с другими инструментами;

scikit-learn – это библиотека Python, которая является одной из самых полезных библиотек Python для машинного обучения. Она включает все алгоритмы и инструменты, которые нужны для задач классификации, регрессии и кластеризации. Она также включает все методы оценки производительности модели машинного обучения.

1) Импортировать необходимые для работы библиотеки и модули;

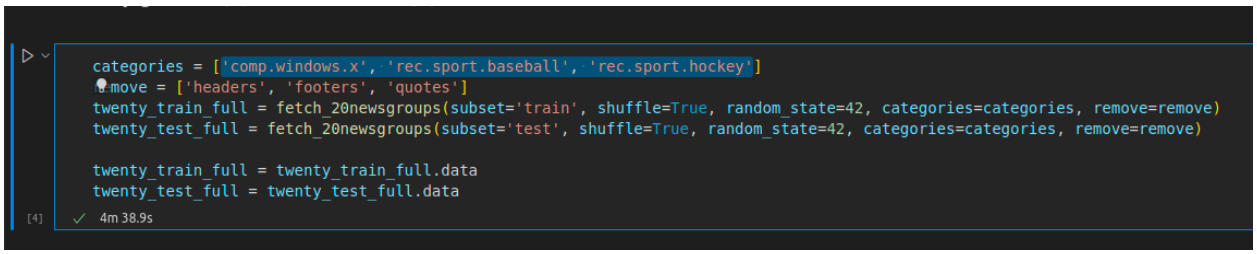


```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import classification_report
from sklearn.model_selection import train_test_split
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.pipeline import Pipeline
from sklearn.naive_bayes import MultinomialNB
from nltk.stem import *
from nltk import word_tokenize
import itertools
```

[3] ✓ 0.6s

Рисунок 1 – Импорт библиотек

2) Загрузить обучающую и экзаменационную выборку в соответствие с вариантом;



```
categories = ['comp.windows.x', 'rec.sport.baseball', 'rec.sport.hockey']
remove = ['headers', 'footers', 'quotes']
twenty_train_full = fetch_20newsgroups(subset='train', shuffle=True, random_state=42, categories=categories, remove=remove)
twenty_test_full = fetch_20newsgroups(subset='test', shuffle=True, random_state=42, categories=categories, remove=remove)

twenty_train_full = twenty_train_full.data
twenty_test_full = twenty_test_full.data
```

[4] ✓ 4m 38.9s

Рисунок 2 – Загрузка выборки

3) Вывести на экран по одному-два документа каждого класса;

```
'\nIf I rember correctly, Lotus Notes gives u this possiblity, among other things...'
```

Рисунок 3 – Документ для класса comp.os.ms-windows.misc

```
"I read in a recent Tidbits(171-2?) about the possibility of putting\na 68030 in a PB100. I am interested in doing so, but woul
d like\nto know more about it. Does it involve just replacing the 68000 that\nis on the daughterboard, or does it involve getti
ng a new daughter-\nboard. Also, would the 68030 be able to run QT with the PB100's\nscreen(not pretty I know, but possible?) A
nd of course, what would\nthe damage be ($). Any info would be appreciated.\nThanks in advance. Jay Fogel\n\n"
```

Рисунок 4 – Документ для класса comp.sys.mac.hardware

```
"\nI'd guess this was a garbled report of the NERVA effort to develop a\nsolid-core fission rocket (the most mundane type of nu
clear rocket).\nThat was the only advanced-propulsion project that was done on a large\nenough scale to be likely to attract ne
ws attention.  It *could* be any\nnumber of things -- the description given is awfully vague -- but I'd\nput a small bet on NER
VA."
```

Рисунок 5 – Документ для класса sci.space

4) Применить стемминг, записав обработанные выборки (тестовую и обучающую) в новые переменные;

Рисунок 6 – Процедура стемминга

5) Провести векторизацию выборки:

a. Векторизовать обучающую и тестовую выборки простым подсчетом слов (CountVectorizer) и значеним max_features = 10000

b. Вывести и проанализировать первые 20 наиболее частотных слов всей выборки и каждого класса по-отдельности.

c. Применить процедуру отсечения стоп-слов и повторить пункт b.

d. Провести пункты a – c для обучающей и тестовой выборки, для которой проведена процедура стемминга.

e. Векторизовать выборки с помощью TfidfTransformer (с использованием TF и TF-IDF взвешиваний) и повторить пункты b-d.

	A	B	C	D	E	F	G
1		Count		TF		TF-IDF	
2		Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами
3							
4	0	('thi', 1125)	('the', 6749)	('thi', 59.88817877128276)	('the', 176.07755353938666)	('window', 23.183318262747868)	('the', 69.14304503181809)
5	1	('use', 988)	('to', 3486)	('use', 46.93261320283235)	('to', 112.74127442605598)	('thi', 23.12751857843546)	('to', 45.04890846112017)
6	2	('file', 763)	('and', 2447)	('window', 42.597525219574734)	('and', 66.43695686807061)	('use', 20.314084934570342)	('is', 29.88739584455872)
7	3	('window', 721)	('is', 2302)	('ani', 30.652341545587962)	('is', 61.64260648039949)	('ani', 15.227798119233983)	('and', 29.339959052194278)
8	4	('program', 591)	('of', 2296)	('thank', 21.222814683958788)	('of', 57.06141787031688)	('server', 13.596676690488847)	('of', 27.909549736261503)
9	5	('entri', 469)	('in', 1849)	('run', 20.298564988539976)	('in', 52.117226198463364)	('widget', 12.505995100435122)	('in', 24.80994001982583)
10	6	('widget', 469)	('for', 1394)	('server', 20.21510861192126)	('it', 48.91942323022249)	('motif', 12.291996104863756)	('it', 24.422443149442163)
11	7	('server', 418)	('it', 1384)	('doe', 20.079841359632432)	('for', 43.426240592814835)	('run', 12.221057425966606)	('for', 21.88712526440163)
12	8	('edu', 377)	('on', 1143)	('ha', 18.972986745642224)	('thi', 39.721026005651055)	('program', 12.082777533961657)	('you', 21.69412035194749)
13	9	('ani', 367)	('that', 1131)	('program', 18.909666267697233)	('that', 38.48440213384221)	('thank', 11.9246033922578)	('that', 20.641194527298392)
14	10	('motif', 357)	('thi', 1125)	('know', 18.3103632139431)	('you', 35.22373608941035)	('know', 11.359528935287118)	('thi', 20.478387820359558)
15	11	('run', 351)	('you', 1066)	('like', 18.067439835022984)	('on', 33.66532615015816)	('doe', 11.268375179632677)	('window', 20.25606083498085)
16	12	('includ', 346)	('be', 1015)	('motif', 17.285279901846774)	('have', 32.325996112983894)	('file', 11.0820663593677)	('on', 18.89697952689806)
17	13	('set', 346)	('use', 988)	('widget', 17.219409743598675)	('use', 31.067854632823902)	('ha', 10.886561179895603)	('have', 18.13678512666158)
18	14	('applic', 332)	('are', 902)	('problem', 16.77160583519499)	('with', 29.390108701958006)	('like', 10.771436349536712)	('use', 17.92430447544591)
19	15	('output', 316)	('if', 836)	('work', 16.690030056650734)	('be', 28.50134264322937)	('applic', 10.707347585888908)	('be', 16.915755211575256)
20	16	('doe', 314)	('with', 821)	('file', 16.267837742673457)	('window', 27.829486978867365)	('display', 10.323932916450032)	('with', 16.715163016432967)
21	17	('avail', 313)	('or', 778)	('applic', 15.762066485831967)	('can', 26.207965525449293)	('problem', 10.170597216005062)	('do', 15.917946622033972)
22	18	('ha', 301)	('file', 763)	('display', 15.127091611959719)	('do', 26.031688837895835)	('work', 9.6935821931121)	('can', 15.775417660556997)
23	19	('work', 297)	('not', 760)	('tri', 15.032835503412675)	('an', 25.560318620220688)	('tri', 9.267086274493444)	('an', 14.847739688603353)
24							
25							
26							
27							
28							
29							

Рисунок 7 – Со стеммингом для comp.windows.x

	Count		TF		TF-IDF	
	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами
0	('thi', 2006)	('the', 15747)	('thi', 141.72279397640855)	('the', 578.9869871527349)	('thi', 55.48698320928098)	('the', 215.27647742894334)
1	('wa', 1448)	('to', 7012)	('wa', 107.29607947837958)	('to', 277.5105984525885)	('wa', 53.07919061045813)	('to', 114.49060106699532)
2	('use', 1130)	('and', 5437)	('game', 77.4041331357221)	('and', 200.61616514687864)	('game', 43.910529528981016)	('and', 87.69641536592607)
3	('game', 1037)	('of', 5008)	('ha', 62.647627160713306)	('in', 178.80949351346268)	('team', 36.14895163007139)	('of', 83.72973399779124)
4	('team', 912)	('in', 4586)	('team', 62.339132757617804)	('of', 177.27317313531762)	('hi', 34.6225688383242)	('in', 82.02034876038448)
5	('ha', 851)	('is', 4061)	('use', 60.14489569791756)	('is', 153.25072611761695)	('year', 33.59812578446207)	('is', 74.800742623246)
6	('year', 817)	('that', 3003)	('hi', 59.43782499655776)	('that', 134.5826880157837)	('ha', 32.618276689614525)	('that', 68.2059599251459)
7	('file', 777)	('for', 2833)	('ani', 57.279631549360936)	('it', 121.64081767176049)	('use', 32.32045014266762)	('it', 63.377601730896856)
8	('hi', 735)	('it', 2787)	('year', 56.538602245273026)	('for', 113.09805914145427)	('ani', 30.890880799498643)	('you', 57.86497701244651)
9	('play', 732)	('on', 2310)	('like', 54.51696717354279)	('you', 95.88651067888384)	('window', 30.626506670619786)	('for', 56.681781334430156)
10	('window', 724)	('be', 2168)	('play', 45.59072193731686)	('thi', 90.29679169755546)	('like', 29.246513479043532)	('he', 52.701288473690965)
11	('like', 657)	('you', 2059)	('know', 45.53483739388707)	('have', 89.47994881568444)	('play', 27.90856004765075)	('have', 50.23419071800302)
12	('ani', 621)	('thi', 2006)	('doe', 45.42365375545463)	('on', 89.07860484463126)	('know', 27.271330191558036)	('on', 49.207504287444486)
13	('run', 615)	('have', 1793)	('think', 44.4952399710279)	('be', 87.08348630671308)	('think', 26.657656715119774)	('thi', 48.988062517955576)
14	('program', 600)	('with', 1708)	('just', 43.87222983981162)	('with', 70.17819208957434)	('player', 26.515653472058656)	('be', 48.987493499956045)
15	('10', 580)	('are', 1674)	('window', 43.401407981085406)	('he', 67.48619409990057)	('doe', 26.350434401319337)	('wa', 46.48640038163983)
16	('doe', 555)	('he', 1524)	('run', 42.12209183127551)	('wa', 67.13462099292734)	('run', 26.161833298548753)	('with', 41.577284328675944)
17	('player', 548)	('if', 1522)	('player', 40.044459890660114)	('but', 64.20796545216623)	('just', 25.410679860785017)	('do', 40.25009158646201)
18	('edu', 544)	('not', 1467)	('time', 36.87544216605833)	('do', 62.54094183567242)	('thank', 23.26725851622436)	('they', 39.95380064551452)
19	('time', 523)	('as', 1453)	('onli', 36.00373220801312)	('if', 61.19748178568055)	('time', 22.445875842617404)	('are', 38.934938887306956)

Рисунок 8 – Со стеммингом для всех категорий

	Count		TF		TF-IDF	
	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами
0	('wa', 488)	('the', 3507)	('wa', 43.650896270696094)	('the', 187.75271864929817)	('wa', 21.303583443331284)	('the', 73.39824242701698)
1	('thi', 416)	('to', 1481)	('thi', 40.46186569117768)	('to', 78.5428617765336)	('thi', 18.344433147219902)	('to', 36.3505793422048)
2	('year', 405)	('and', 1312)	('year', 32.54705898373833)	('and', 67.75412089800828)	('year', 17.26335197345431)	('and', 32.008884822287654)
3	('game', 349)	('of', 1142)	('game', 30.760173912509135)	('in', 60.5450010580712)	('game', 16.807371640244643)	('of', 30.617487467764573)
4	('hi', 337)	('in', 1116)	('hi', 27.60237314507432)	('of', 60.20209451506751)	('hi', 14.915071503151212)	('in', 29.456280732319172)
5	('team', 270)	('that', 883)	('team', 24.02597221907377)	('is', 48.25741727629854)	('team', 14.380650406721234)	('is', 25.989215584632557)
6	('ha', 259)	('is', 875)	('think', 20.478297014660196)	('that', 47.830361674090504)	('run', 12.222773976551435)	('that', 25.510584508260468)
7	('run', 237)	('he', 738)	('ha', 19.984760134307216)	('he', 37.71516633782879)	('ha', 11.741672514911992)	('he', 24.9738551240755)
8	('player', 216)	('for', 580)	('run', 19.66055180606613)	('for', 33.39490247954127)	('think', 11.600344611344717)	('it', 18.784876279173044)
9	('think', 208)	('it', 561)	('like', 17.080484796627637)	('it', 31.704555456764133)	('player', 11.107378991803625)	('wa', 18.556138353876516)
10	('hit', 205)	('have', 529)	('player', 17.027473998426235)	('have', 29.840821960953395)	('hit', 10.617792115696867)	('for', 18.334527654861176)
11	('good', 200)	('be', 510)	('just', 16.671712959598352)	('be', 28.815452466740147)	('pitch', 10.597243380813696)	('have', 18.27192817245304)
12	('pitch', 187)	('wa', 488)	('hit', 15.956690731473008)	('wa', 28.08661464358159)	('just', 9.799114805591033)	('you', 17.539427095231204)
13	('like', 183)	('but', 451)	('pitch', 15.737516827086205)	('you', 26.315555874531896)	('like', 9.798886813200204)	('be', 17.038204506734324)
14	('00', 175)	('you', 443)	('doe', 15.132930689973659)	('thi', 25.51242263647)	('win', 9.725466494383015)	('they', 16.75020581897335)
15	('win', 171)	('on', 442)	('good', 14.536619056720612)	('on', 25.034053958345964)	('doe', 9.622983647004764)	('thi', 15.739368227780147)
16	('play', 163)	('thi', 416)	('did', 14.529212616482056)	('they', 23.683275913791338)	('good', 9.312265022998293)	('on', 15.362454991041353)
17	('just', 161)	('they', 409)	('baseball', 14.484211361683947)	('but', 22.543406577533634)	('baseball', 9.280809207135212)	('year', 14.795378719454591)
18	('did', 150)	('year', 405)	('win', 13.86615081032843)	('at', 21.881943893070986)	('did', 8.939816554390699)	('but', 14.534536397825224)
19	('time', 144)	('at', 404)	('know', 13.595195578337357)	('year', 20.57451779384734)	('know', 8.649335171851126)	('game', 14.164142011694963)

Рисунок 9 – Со стеммингом для rec.sport.baseball

	Count		TF		TF-IDF	
	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами
0	('wa', 762)	('the', 5491)	('wa', 50.15203871391994)	('the', 211.12955241962217)	('wa', 23.36133940232359)	('the', 85.61006180958277)
1	('game', 680)	('to', 2045)	('game', 45.14764928107874)	('to', 84.21875719829619)	('game', 22.232566866103742)	('to', 39.450605569700144)
2	('team', 635)	('and', 1678)	('thi', 39.055502878690326)	('and', 64.77390483752433)	('team', 17.903646722401515)	('in', 32.652018293640175)
3	('play', 564)	('in', 1621)	('team', 37.35448861630797)	('in', 64.72210384814426)	('thi', 17.477917161421182)	('and', 31.69164630209744)
4	('thi', 465)	('of', 1570)	('play', 31.21182425967776)	('of', 58.71305767885902)	('play', 16.12810678887045)	('of', 30.1415800189595)
5	('10', 406)	('that', 989)	('ha', 22.74243751646257)	('that', 47.29987276280994)	('player', 13.380870372760931)	('that', 25.589010815798186)
6	('hockey', 367)	('is', 884)	('player', 22.551885438282852)	('is', 42.35336379019357)	('year', 12.471758807829504)	('is', 23.581772956107983)
7	('25', 352)	('for', 859)	('hockey', 21.58641411333749)	('it', 40.02035067148003)	('ha', 12.462441645111271)	('it', 22.98526691139549)
8	('55', 340)	('it', 842)	('year', 21.00477369512061)	('for', 35.220439435810206)	('hockey', 12.32609891527155)	('he', 21.41784950992936)
9	('player', 332)	('he', 776)	('hi', 19.52311411546069)	('you', 33.498712420706276)	('hi', 11.510625855742392)	('you', 21.331058323921045)
10	('season', 325)	('wa', 762)	('like', 18.528716312290406)	('wa', 30.633344111874845)	('think', 10.881382261023196)	('wa', 20.063201319059996)
11	('hi', 321)	('on', 725)	('think', 17.55562459396268)	('on', 29.64784888901185)	('like', 10.780773726674884)	('for', 19.327227570061634)
12	('11', 311)	('game', 680)	('just', 16.101411881211302)	('be', 29.1901895490001)	('season', 10.385330706405755)	('game', 18.992458722614053)
13	('year', 311)	('be', 643)	('season', 15.886861040064813)	('he', 28.991828609841694)	('just', 9.851146559396701)	('be', 17.927719952417107)
14	('pt', 307)	('team', 635)	('time', 14.504235776761352)	('game', 27.85141869992506)	('win', 9.43124692954132)	('on', 17.779465825043015)
15	('ha', 291)	('play', 564)	('did', 14.399791256589493)	('have', 26.664578514521327)	('time', 9.239888590040328)	('have', 16.935037040848968)
16	('12', 282)	('you', 550)	('win', 13.852075854799367)	('thi', 24.285257405123843)	('did', 9.064435311544207)	('they', 16.660712215581185)
17	('16', 273)	('have', 549)	('playoff', 13.648740059293425)	('they', 23.961866268322044)	('know', 8.896501229794964)	('team', 15.284545147493938)
18	('period', 264)	('at', 547)	('onli', 13.407510165751997)	('team', 22.56275157717991)	('playoff', 8.837366068442966)	('thi', 15.020752206824183)
19	('14', 256)	('but', 532)	('good', 13.142799386221997)	('but', 21.570267570593767)	('nhl', 8.664675767903834)	('but', 14.527253892969995)

Рисунок 10 – Со стеммингом для rec.sport.hockey

	Count		TF		TF-IDF	
	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами
0	('file', 579)	('the', 6750)	('window', 33.493829091087804)	('the', 179.52564999529974)	('window', 18.62263410630872)	('the', 68.4519760979994)
1	('window', 571)	('to', 3486)	('use', 23.36299133935824)	('to', 114.97483605434675)	('server', 12.792738426867421)	('to', 44.42775747206853)
2	('use', 459)	('and', 2447)	('thanks', 20.666595805186912)	('and', 67.83285620602969)	('motif', 12.180890850601092)	('is', 29.161518801317538)
3	('program', 412)	('of', 2296)	('server', 20.216333097277815)	('is', 61.6667177490575)	('use', 12.154245733664446)	('and', 29.052674342859113)
4	('server', 385)	('is', 2269)	('like', 19.565271132371603)	('of', 58.218063099014365)	('thanks', 11.154001474827533)	('of', 27.638370795751744)
5	('edu', 377)	('in', 1848)	('motif', 18.563510497985284)	('in', 53.08672396669423)	('know', 10.821140178418501)	('in', 24.4919449469208)
6	('motif', 356)	('for', 1394)	('know', 18.4154161186558)	('it', 47.15215060133111)	('like', 10.784638083394528)	('it', 23.13880233342308)
7	('widget', 354)	('it', 1261)	('using', 17.427309852987836)	('for', 44.2329558584831)	('does', 9.635880839857878)	('for', 21.5865417689981)
8	('entry', 351)	('on', 1141)	('does', 17.023672845142617)	('this', 40.4925846720753)	('using', 9.398027291603409)	('you', 21.4999693917737)
9	('output', 315)	('that', 1130)	('problem', 14.909743405833549)	('that', 39.221277235426555)	('windows', 9.341850175743255)	('that', 20.28614426057601)
10	('available', 306)	('this', 1125)	('windows', 14.131132198737099)	('you', 35.92030874842972)	('widget', 9.306386140673288)	('this', 20.255725544444637)
11	('com', 274)	('you', 1066)	('program', 14.00077221528705)	('on', 34.242514064021144)	('program', 9.023069498962013)	('on', 18.642354077165816)
12	('set', 274)	('be', 956)	('application', 13.571269773961967)	('with', 30.022306576480474)	('application', 8.976473930392826)	('with', 16.577260844288084)
13	('using', 269)	('are', 893)	('widget', 12.982755334308905)	('have', 29.690554598158364)	('problem', 8.907486584499523)	('have', 16.508083101042423)
14	('mit', 254)	('if', 836)	('file', 12.015114719712582)	('can', 27.315076696894675)	('hi', 8.252838798071473)	('window', 16.12771282102309)
15	('application', 253)	('with', 821)	('code', 11.735283908070118)	('be', 27.210948171156236)	('file', 8.184755851896966)	('be', 15.970070452628836)
16	('like', 247)	('or', 778)	('set', 11.731024419785864)	('an', 26.075261536672475)	('display', 7.692190451120513)	('can', 15.798320612123833)
17	('information', 245)	('can', 756)	('running', 11.513987934502923)	('if', 24.39250127491638)	('x11r5', 7.555969368858565)	('an', 14.645917478059618)
18	('sun', 240)	('an', 730)	('display', 11.262439512269772)	('or', 22.2575884306737)	('xterm', 7.392033646244608)	('if', 14.525084826005761)
19	('does', 232)	('not', 699)	('just', 11.134066103605429)	('any', 21.813290186035825)	('set', 7.355290923099196)	('or', 13.623571462914889)

Рисунок 11 – Без стемминга для comp.windows.x

	Count		TF		TF-IDF	
	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами
0	('team', 679)	('the', 15749)	('like', 54.59034267706766)	('the', 589.4840147265047)	('game', 30.55721189057097)	('the', 214.36387605105372)
1	('game', 633)	('to', 7012)	('game', 52.65391186175265)	('to', 282.9780269588262)	('team', 28.81541385536274)	('to', 113.78471180759847)
2	('year', 629)	('and', 5437)	('team', 50.23374990934391)	('and', 204.73973232096188)	('like', 27.880963215705588)	('and', 87.3636242603679)
3	('file', 586)	('of', 5008)	('just', 47.839597900758605)	('in', 181.95100839538162)	('year', 27.3154013693727)	('of', 83.5045406913417)
4	('like', 582)	('in', 4583)	('don', 46.667780397895164)	('of', 180.65052528156176)	('don', 25.993002073961122)	('in', 81.71464861099017)
5	('10', 580)	('is', 3967)	('year', 46.66093560424913)	('is', 151.84620034325926)	('know', 25.980732209617294)	('is', 72.71056685601074)
6	('window', 573)	('that', 3001)	('know', 45.38244213178166)	('that', 137.00256575679734)	('just', 25.498472063457278)	('that', 67.62424697731576)
7	('edu', 544)	('for', 2833)	('think', 44.39222815673452)	('it', 117.94844823750009)	('think', 25.276905208763342)	('it', 60.63404118880459)
8	('use', 512)	('it', 2597)	('time', 36.49174208101818)	('for', 115.18977217608628)	('window', 24.08872014213688)	('you', 57.61452297032909)
9	('don', 502)	('on', 2307)	('good', 35.62187455720601)	('you', 97.72631057215757)	('games', 21.938746911155285)	('for', 56.101337507476245)
10	('just', 480)	('you', 2059)	('does', 34.76996463457239)	('this', 91.94123274499883)	('thanks', 21.22302451689207)	('he', 52.38043107664733)
11	('time', 466)	('this', 2006)	('window', 34.42979502162383)	('on', 90.5422925115908)	('time', 21.065804609712245)	('on', 48.851522087063)
12	('new', 437)	('be', 1979)	('thanks', 33.545069408891216)	('have', 83.8896451423794)	('good', 21.056475539446414)	('this', 48.544997294911376)
13	('good', 434)	('with', 1708)	('games', 33.05600445722264)	('be', 79.4042202847746)	('does', 20.795041818869304)	('have', 46.82516631808222)
14	('think', 429)	('have', 1652)	('use', 28.892433198026094)	('with', 71.58514903092679)	('players', 19.029886083529593)	('be', 45.129856667459215)
15	('play', 427)	('are', 1634)	('players', 28.10991175375138)	('he', 68.66153507608368)	('use', 18.133972761035878)	('was', 44.487884132057964)
16	('season', 424)	('he', 1524)	('play', 27.022282354210148)	('but', 65.45399825662439)	('play', 17.5914301110468)	('with', 41.34900152421049)
17	('program', 417)	('if', 1522)	('season', 26.63108713464234)	('was', 64.89623089168877)	('season', 17.489935385470744)	('they', 39.69318535367601)
18	('games', 416)	('as', 1453)	('way', 25.574312213576814)	('if', 62.427980984910434)	('hockey', 17.123106074411094)	('but', 38.536211763990764)
19	('11', 415)	('but', 1438)	('did', 24.382018454599265)	('are', 60.70514880130126)	('server', 16.0242298941479)	('are', 38.119468736590726)

Рисунок 12 – Без стемминга для всех категорий

	Count		TF		TF-IDF	
	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами
0	('year', 310)	('the', 3508)	('year', 27.328437909508956)	('the', 189.75201171863503)	('year', 14.150865552494695)	('the', 71.52530093898116)
1	('game', 204)	('to', 1481)	('think', 19.855086670508665)	('to', 79.50866159748803)	('team', 11.453008147196956)	('to', 35.40923223349221)
2	('good', 200)	('and', 1312)	('team', 19.10440941152317)	('and', 68.56931010535685)	('game', 10.78354148594148)	('and', 31.24622062627418)
3	('team', 195)	('of', 1142)	('game', 18.49974721714112)	('in', 61.118250762965616)	('think', 10.571828814475818)	('of', 29.83886400146451)
4	('think', 189)	('in', 1114)	('just', 17.880834591512958)	('of', 60.84448873198823)	('don', 9.631358145255788)	('in', 28.66817537939383)
5	('don', 186)	('that', 882)	('don', 17.251477983402815)	('that', 48.30316258030083)	('just', 9.606608344467812)	('that', 24.78589273581064)
6	('00', 175)	('is', 842)	('like', 16.004033551913114)	('is', 47.18759672480876)	('games', 9.422532464842263)	('is', 24.611093234666686)
7	('just', 161)	('he', 738)	('good', 15.858217089572923)	('he', 38.143154312157066)	('good', 9.239250820662061)	('he', 24.33642098580889)
8	('like', 153)	('for', 580)	('baseball', 15.326411030157013)	('for', 33.80143955713195)	('baseball', 9.118280628917837)	('for', 17.85408755857631)
9	('games', 149)	('it', 543)	('games', 15.186586963622732)	('it', 31.008331018144844)	('like', 8.70513895933545)	('it', 17.83488114065544)
10	('better', 140)	('have', 494)	('time', 12.623170704982257)	('have', 28.198723264493307)	('runs', 8.64254897572228)	('was', 17.454274844220564)
11	('baseball', 137)	('was', 466)	('runs', 12.59916133438563)	('was', 26.7223127997071)	('hit', 8.001550908205388)	('you', 17.021637477648156)
12	('hit', 137)	('but', 451)	('know', 12.58560348462295)	('you', 26.60511283858597)	('know', 7.737753712262402)	('have', 16.81652249766419)
13	('runs', 137)	('be', 448)	('hit', 12.176484861243743)	('this', 25.77982731367013)	('time', 7.485316055886953)	('they', 16.29316671405876)
14	('players', 135)	('you', 443)	('players', 11.29614239488529)	('on', 25.155081362658674)	('players', 7.370947401935824)	('this', 15.289597811048703)
15	('time', 131)	('on', 441)	('better', 10.025123389369568)	('be', 25.070721459170457)	('pitching', 6.9513704933523774)	('be', 15.149747771431635)
16	('02', 125)	('this', 416)	('does', 9.878951573660155)	('they', 23.951408638709445)	('braves', 6.837994737788999)	('on', 14.951226312368018)
17	('won', 124)	('they', 409)	('did', 9.804853955021839)	('but', 22.81733317274584)	('does', 6.6988738870066 8)	('but', 14.156706564766719)
18	('league', 118)	('at', 404)	('pitching', 9.60502110332111)	('at', 22.134317027150583)	('win', 6.6742449360711875)	('at', 13.388275819961727)
19	('03', 116)	('with', 393)	('win', 9.574714345501624)	('are', 20.037180470627636)	('did', 6.406810915458073)	('are', 13.150738124174623)

Рисунок 13 – Без стемминга для rec.sport.baseball

	Count		TF		TF-IDF	
	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами	Без стоп-слов	С стоп-словами
0	('team', 478)	('the', 5491)	('game', 32.48067979748125)	('the', 213.68054815592015)	('game', 16.3621820447421)	('the', 84.0036178948874)
1	('game', 423)	('to', 2045)	('team', 29.681545389112983)	('to', 85.33218873697354)	('team', 14.098565781831738)	('to', 38.665081529313085)
2	('10', 406)	('and', 1678)	('hockey', 23.24432955268019)	('and', 65.66019621011154)	('hockey', 12.178589276699103)	('in', 32.04606452477672)
3	('hockey', 365)	('in', 1621)	('play', 18.308620947881273)	('in', 65.47735012986911)	('games', 10.480501577313447)	('and', 31.125480134214097)
4	('25', 352)	('of', 1570)	('don', 17.77288137024864)	('of', 59.47437076194172)	('don', 10.327349950605772)	('of', 29.625472264539347)
5	('play', 343)	('that', 989)	('think', 17.683650306111264)	('that', 47.90763701907816)	('play', 10.311836562464375)	('that', 25.070798642582655)
6	('55', 340)	('for', 859)	('like', 17.345059884067272)	('is', 41.46126871199282)	('think', 10.260540891834697)	('is', 22.460636534624484)
7	('season', 312)	('is', 856)	('just', 17.13782767553503)	('it', 38.30855000378448)	('players', 9.993766991872079)	('it', 21.49287373245604)
8	('11', 311)	('it', 793)	('games', 16.808370991454076)	('for', 35.66891316799022)	('season', 9.981357980613577)	('he', 21.075672757863803)
9	('12', 282)	('he', 776)	('season', 16.568420610783182)	('you', 33.946611356147756)	('like', 9.681508230349387)	('you', 20.918061780075785)
10	('16', 273)	('was', 731)	('year', 16.17689445964111)	('on', 30.027382738330374)	('just', 9.64422399003983)	('for', 18.897563643644627)
11	('games', 265)	('on', 725)	('players', 16.049272969750437)	('was', 29.633557295937567)	('year', 9.630797590052602)	('was', 18.896296900250352)
12	('period', 257)	('be', 575)	('good', 14.153184016365199)	('he', 29.418526848816647)	('nhl', 8.525334244289347)	('on', 17.425110357801533)
13	('14', 256)	('you', 550)	('nhl', 14.012183267569771)	('be', 26.289954516180654)	('know', 8.441540998481107)	('they', 16.27383213476644)
14	('15', 252)	('at', 547)	('time', 13.41789572558515)	('have', 25.065654965698226)	('good', 8.374197167697206)	('be', 16.07363220286866)
15	('18', 247)	('but', 532)	('know', 13.10218684673364)	('this', 24.561140664414296)	('time', 8.193514753370954)	('have', 15.707471637553976)
16	('20', 242)	('have', 512)	('league', 10.879835024256954)	('they', 24.260640316913864)	('let', 7.39529432050941)	('this', 14.691317767053578)
17	('nhl', 236)	('with', 494)	('teams', 10.630814881299957)	('but', 21.840252374444372)	('detroit', 7.283185011759394)	('but', 14.24362693870247)
18	('13', 233)	('team', 478)	('did', 10.12873791166737)	('with', 21.226302944699118)	('league', 7.083285902127699)	('with', 13.876740518024121)
19	('year', 228)	('as', 466)	('let', 10.088717957524912)	('are', 19.866861444088645)	('teams', 6.928002474068629)	('game', 13.857439899045554)

Рисунок 14 – Без стемминга для rec.sport.hockey

6) Используя конвейер (Pipeline) реализовать модель Наивного Байесовского классификатора и выявить на основе показателей качества (значения полноты, точности, f1-меры и аккуратности), какая предварительная обработка данных обеспечит наилучшие результаты классификации. Должны быть исследованы следующие характеристики:

• Отсечение – не отсечение стоп-слов

• Количество информативных терминов (max_features)

• Взвешивание: Count, TF, TF-IDF

A	B	C	D	E	F	G
	0	1	2	accuracy	macro avg	weighted avg
precision	0,620689655172414	0,851758793969849	0,456043956043956	0,650323774283071	0,64283080172874	0,707280013707122
recall	0,647058823529412	0,606440071556351	0,768518518518519	0,650323774283071	0,67400580453476	0,650323774283071
f1-score	0,6336	0,70846394984326	0,572413793103448	0,650323774283071	0,638159247648903	0,660087259271718
support	306	559	216	0,650323774283071	1081	1081

Рисунок 15 – Пример работы программы со следующими параметрами (max_features = 1000, со стоп словами, без TF, TF-IDF)

	0	1	2	accuracy	macro avg	weighted avg
precision	0,620689655172414	0,851758793969849	0,456043956043956	0,650323774283071	0,64283080172874	0,707280013707122
recall	0,647058823529412	0,606440071556351	0,768518518518519	0,650323774283071	0,67400580453476	0,650323774283071
f1-score	0,6336	0,70846394984326	0,572413793103448	0,650323774283071	0,638159247648903	0,660087259271718
support	306	559	216	0,650323774283071	1081	1081

Рисунок 16 – Пример работы программы со следующими параметрами (max_features=1000, со стоп словами без tf, с idf)

	0	1	2	accuracy	macro avg	weighted avg
precision	0,61128526645768	0,844221105527638	0,516483516483517	0,665124884366327	0,657329962822945	0,703949428789617
recall	0,661016949152542	0,630393996247655	0,743083003952569	0,665124884366327	0,678164649784256	0,665124884366327
f1-score	0,635179153094463	0,721804511278195	0,609400324149109	0,665124884366327	0,655461329507256	0,671857480743635
support	295	533	253	0,665124884366327	1081	1081

Рисунок 17 – Пример работы программы со следующими параметрами

	0	1	2	accuracy	macro avg	weighted avg
precision	0,630094043887147	0,844221105527638	0,527472527472528	0,674375578168363	0,667262558962438	0,710861825529072
recall	0,672240802675585	0,635160680529301	0,758893280632411	0,674375578168363	0,688764921279099	0,674375578168363
f1-score	0,650485436893204	0,724919093851133	0,622366288492707	0,674375578168363	0,665923606412348	0,6803293406725
support	299	529	253	0,674375578168363	1081	1081

(max_features=1000, со стоп словами с tf, без idf)

Рисунок 18 – Пример работы программы со следующими параметрами (max_features=1000, со стоп словами, с tf и idf)

A	B	C	D	E	F	G
	0	1	2	accuracy	macro avg	weighted avg
precision	0,586206896551724	0,743718592964824	0,489010989010989	0,611470860314524	0,606312159509179	0,628435659097758
recall	0,586206896551724	0,629787234042553	0,60958904109589	0,611470860314524	0,608527723896723	0,611470860314524
f1-score	0,586206896551724	0,682027649769585	0,542682926829268	0,611470860314524	0,603639157716859	0,616111387627985
support	319	470	292	0,611470860314524	1081	1081

Рисунок 19 – Пример работы программы со следующими параметрами (max_features=1000, без стоп слов без tf и idf)

A	B	C	D	E	F	G	H
	0	1	2	accuracy	macro avg	weighted avg	
precision	0,586206896551724	0,743718592964824	0,489010989010989	0,611470860314524	0,606312159509179	0,628435659097758	
recall	0,586206896551724	0,629787234042553	0,60958904109589	0,611470860314524	0,608527723896723	0,611470860314524	
f1-score	0,586206896551724	0,682027649769585	0,542682926829268	0,611470860314524	0,603639157716859	0,616111387627985	
support	319	470	292	0,611470860314524	1081	1081	

Рисунок 20 – Пример работы программы со следующими параметрами (max_features=1000, без стоп слов, без tf, с idf)

A	B	C	D	E	F	G
	0	1	2	accuracy	macro avg	weighted avg
precision	0,385579937304075	0,791457286432161	0,598901098901099	0,606845513413506	0,591979440879112	0,662726881292275
recall	0,75	0,583333333333333	0,578249336870027	0,606845513413506	0,63719422340112	0,606845513413506
f1-score	0,509316770186336	0,671641791044776	0,588394062078273	0,606845513413506	0,589784207769795	0,617982496649627
support	164	540	377	0,606845513413506	1081	1081

Рисунок 21 – Пример работы программы со следующими параметрами (max_features=1000, без стоп слов, с tf, без idf)

По результатам классификации наиболее подходящая предварительная обработка данных является со следующими параметрами:

- с tf и tf-idf;

- max_features = 10000;

- со стоп словами.

	precision	recall	f1-score	support
0	0.76	0.89	0.82	270
1	0.94	0.83	0.88	452
2	0.84	0.86	0.85	359
accuracy			0.85	1081
macro avg	0.85	0.86	0.85	1081
weighted avg	0.86	0.85	0.86	1081

Рисунок 22 – Результат работы программы

Код программы

%%

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import classification_report
from sklearn.model_selection import train_test_split
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.pipeline import Pipeline
from sklearn.naive_bayes import MultinomialNB
from nltk.stem import *
from nltk import word_tokenize
import itertools
import nltk

nltk.download('punkt')

# %% [markdown]
# ## Выгрузка данных из датасета

# %%
categories = ['comp.windows.x', 'rec.sport.baseball', 'rec.sport.hockey']
remove = ['headers', 'footers', 'quotes']
twenty_train_full = fetch_20newsgroups(subset='train', shuffle=True,
random_state=42, categories=categories, remove=remove)
twenty_test_full = fetch_20newsgroups(subset='test', shuffle=True, random_state=42,
categories=categories, remove=remove)

twenty_train_full = twenty_train_full.data
twenty_test_full = twenty_test_full.data

# %%
twenty_train = dict()
twenty_test = dict()
for category in categories:
twenty_train[category] = fetch_20newsgroups(subset='train', shuffle=True,
random_state=42, categories=[category], remove=remove)
twenty_test[category] = fetch_20newsgroups(subset='test', shuffle=True,
random_state=42, categories=[category], remove=remove)
twenty_train[category] = twenty_train[category].data
twenty_test[category] = twenty_test[category].data

twenty_train['full'] = twenty_train_full
twenty_test['full'] = twenty_test_full
```

```
# %% [markdown]
# ## Стемминг

# %%
def stemming(data):
porter_stemmer = PorterStemmer()
stem = []
for text in data:
nltk_tokens = word_tokenize(text)
line = ''
for word in nltk_tokens:
line += ' ' + porter_stemmer.stem(word)
stem.append(line)
return stem


stem_train = dict()
stem_test = dict()
for category in categories:
stem_train[category] = stemming(twenty_train[category])
stem_test[category] = stemming(twenty_test[category])

stem_train['full'] = stemming(twenty_train['full'])
stem_test['full'] = stemming(twenty_test['full'])

# %% [markdown]
# ## Векторизация

# %%
def SortbyTF(inputStr):
return inputStr[1]
def top_list(vect, data, count):
x = list(zip(vect.get_feature_names(),np.ravel(data.sum(axis=0))))
x.sort(key=SortbyTF, reverse = True)
return x[:count]

# %% [markdown]
# ## Итоговая таблица

# %%
def process(train, categories):
cats = categories[:]
cats.append('full')
mux = pd.MultiIndex.from_product([['Count','TF','TF-IDF'], ['Без стоп-слов','С стоп-
словами']])
summary = dict()
for category in cats:
summary[category] = pd.DataFrame(columns=mux)
stop_words = [None, 'english']
idf = [False, True]
indx_stop = {
```

```
'english': 'Без стоп-слов',
None: 'С стоп-словами'
}
indx_tf = {
False: 'TF',
True: 'TF-IDF'
}
for category in cats:
for stop in stop_words:
vect = CountVectorizer(max_features=10000, stop_words=stop)
vect.fit(train[category])
train_data = vect.transform(train[category])
summary[category]['Count', indx_stop[stop]] = top_list(vect, train_data, 20)
for tf in idf:
tfidf = TfidfTransformer(use_idf = tf).fit(train_data)
train_fidf = tfidf.transform(train_data)
summary[category][indx_tf[tf], indx_stop[stop]] = top_list(vect, train_fidf, 20)
return summary

summ_without_stem = process(twenty_train, categories)
summ_with_stem = process(stem_train, categories)

# %%
for cat in ['full'] + categories:
summ_without_stem[cat].to_excel('without_stem_' + cat + '.xlsx')
summ_with_stem[cat].to_excel('with_stem_' + cat + '.xlsx')

# %% [markdown]
# ## Pipelines

# %%
import os

# %%
def print_classification_score(clf, data):
print(classification_report(gs_clf.predict(data.data), data.target))

# %%
categories = ['alt.atheism', 'rec.motorcycles', 'talk.politics.guns']
remove = ['headers', 'footers', 'quotes']
twenty_train_full = fetch_20newsgroups(subset='train', shuffle=True,
random_state=42, categories=categories, remove=remove)
twenty_test_full = fetch_20newsgroups(subset='test', shuffle=True, random_state=42,
categories=categories, remove=remove)

# %%
def prespocess(data, max_features, stop_words, use_tf, use_idf):
tf = None
cv = CountVectorizer(max_features=max_features, stop_words=stop_words).fit(data)
if use_tf:
```

```
tf = TfidfTransformer(use_idf=use_idf).fit(cv.transform(data))
return cv, tf

def models_grid_search(data_train, data_test):
max_features = [100,500,1000,5000,10000]
stop_words = ['english', None]
use_tf = [True, False]
use_idf = [True, False]
res = dict()
for param in itertools.product(max_features, stop_words, use_tf, use_idf):
cv, tf = prespocess(data_train.data, param[0], param[1], param[2], param[3])
if tf:
clf = MultinomialNB().fit(tf.transform(cv.transform(data_train.data)), data_train.target)
prep_test = tf.transform(cv.transform(data_test.data))
else:
clf = MultinomialNB().fit(cv.transform(data_train.data), data_train.target)
prep_test = cv.transform(data_test.data)
name =
f'max_features={param[0]}_stop_words={param[1]}_use_tf={param[2]}_use_idf={pa
ram[3]}'
res[name] = pd.DataFrame(classification_report(clf.predict(prep_test), data_test.target,
output_dict=True))
return res

# %%
scores = models_grid_search(twenty_train_full, twenty_test_full)

# %%
if not os.path.exists('scores'):
os.makedirs('scores')
for name, score in scores.items():
score.to_excel('scores/' + name + '.xlsx')

# %%
from sklearn.model_selection import GridSearchCV
parameters = {
'vect__max_features': (100,500,1000,5000,10000),
'vect__stop_words': ('english', None),
'tfidf__use_idf': (True, False),
}

text_clf = Pipeline([
('vect', CountVectorizer()),
('tfidf', TfidfTransformer()),
('clf', MultinomialNB())
])

gs_clf = GridSearchCV(text_clf, parameters, n_jobs=-1, cv=3)
gs_clf.fit(X = twenty_train_full.data, y = twenty_train_full.target)
print_classification_score(gs_clf, twenty_test_full)
```

%%

%%

Вывод

В ходе выполнения данной лабораторной работы мы получили базовые навыки работы с языком python и набором функций для анализа и обработки данных.