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

Отчет
Рубежный контроль №2

По курсу «Технологии машинного обучения»
«Технологии использования и оценки моделей машинного обучения»

ИСПОЛНИТЕЛЬ:

Сафин Рустам
Группа ИУ5-64

_____

"__"_____2020 г.

ПРЕПОДАВАТЕЛЬ:

Гапанюк Ю.Е.

_____

"__"_____2020 г.

Москва 2020

Рубежный контроль №2

Сафин Рустам, ИУ5-64Б, Вариант №10, Задача №1

Задание

Необходимо решить задачу классификации текстов на основе любого выбранного Вами датасета (кроме примера, который рассматривался в лекции). Классификация может быть бинарной или многоклассовой. Целевой признак из выбранного Вами датасета может иметь любой физический смысл, примером является задача анализа тональности текста.

Необходимо сформировать признаки на основе CountVectorizer или TfidfVectorizer.

В качестве классификаторов необходимо использовать два классификатора, не относящихся к наивным Байесовским методам (например, LogisticRegression, LinearSVC), а также Multinomial Naive Bayes (MNB), Complement Naive Bayes (CNB), Bernoulli Naive Bayes.

Для каждого метода необходимо оценить качество классификации с помощью хотя бы одной метрики качества классификации (например, Accuracy).

Сделате выводы о том, какой классификатор осуществляет более качественную классификацию на Вашем наборе данных.

Решение

In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import f1_score, precision_score
from sklearn.linear_model import LogisticRegression
import matplotlib.pyplot as plt
from sklearn.naive_bayes import MultinomialNB, ComplementNB, BernoulliNB
from sklearn.metrics import accuracy_score from sklearn.svm import
LinearSVC
from sklearn.feature_extraction.text import TfidfVectorizer

%matplotlib inline
sns.set(style="ticks")
```

In [2]:

```
data = pd.read_csv('apple-twitter-sentiment-texts.csv')
data
```

Out[2]:

	text	sentiment
0	Wow. Yall needa step it up @Apple RT @heynyla:...	-1
1	What Happened To Apple Inc? http://t.co/FJEX...	0
2	Thank u @apple I can now compile all of the pi...	1
3	The oddly uplifting story of the Apple co-foun...	0
4	@apple can i exchange my iphone for a differen...	0
...	...	...
1625	Those** PICK UP THE SLACK YOU FUCK BOYS @Apple	-1
1626	Finally got my iPhone 6 in the mail and it com...	-1
1627	@umo_games @Apple ended up getting a new compu...	0
1628	The 19-Year-Old #WizKid Who Turned Down @Apple...	0
1629	The iPhone 6 May Have A Longer Upgrade Cycle -...	-1

1630 rows × 2 columns

In [3]:

```
data['sentiment'].value_counts()
```

Out[3]:

```
 0    801
-1    686
 1    143
Name: sentiment, dtype: int64
```

In [5]:

```
data = data[data['sentiment'].isin(['0', '-
1'])] data
```

Out[5]:

	text	sentiment
0	Wow. Yall needa step it up @Apple RT @heynyla:...	-1
1	What Happened To Apple Inc? http://t.co/FJEX...	0
3	The oddly uplifting story of the Apple co-foun...	0
4	@apple can i exchange my iphone for a differen...	0
5	RT @JPDesloges: Apple Acted Unfairly In Suppre...	-1
...	...	...
1625	Those** PICK UP THE SLACK YOU FUCK BOYS @Apple	-1
1626	Finally got my iPhone 6 in the mail and it com...	-1
1627	@umo_games @Apple ended up getting a new compu...	0
1628	The 19-Year-Old #WizKid Who Turned Down @Apple...	0
1629	The iPhone 6 May Have A Longer Upgrade Cycle -...	-1

1487 rows × 2 columns

Разделим выборку на обучающую и тестовую:

In [6]:

```
X = data.drop('sentiment', axis=1)
Y = data['sentiment']
```

In [7]:

```
X
```

Out[7]:

	text
0	Wow. Yall needa step it up @Apple RT @heynyla:...
1	What Happened To Apple Inc? http://t.co/FJEX...
3	The oddly uplifting story of the Apple co-foun...
4	@apple can i exchange my iphone for a differen...
5	RT @JPDesloges: Apple Acted Unfairly In Suppre...
...	...
1625	Those** PICK UP THE SLACK YOU FUCK BOYS @Apple
1626	Finally got my iPhone 6 in the mail and it com...
1627	@umo_games @Apple ended up getting a new compu...
1628	The 19-Year-Old #WizKid Who Turned Down @Apple...
1629	The iPhone 6 May Have A Longer Upgrade Cycle -...

1487 rows × 1 columns

In [8]:

```
Y
```

Out[8]:

```
0      -1
1       0
3       0
4       0
5      -1
       ..
 1625  -1
 1626  -1
 1627   0
 1628   0
 1629  -1
Name: sentiment, Length: 1487, dtype: int64
```

In [9]:

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.25, random_state=1)
print('{}, {}'.format(X_train.shape, X_test.shape))
print('{}, {}'.format(Y_train.shape, Y_test.shape))
```

```
(1115, 1), (372, 1)
(1115,), (372,)
```

In [10]:

```
vectorizer = TfidfVectorizer()
vectorizer.fit(X_train + X_test)
```

Out[10]:

```
TfidfVectorizer(analyzer='word', binary=False, decode_error='strict',
                dtype=<class 'numpy.float64'>, encoding='utf-8',
                input='content', lowercase=True, max_df=1.0, max_features=No
ne,
                min_df=1, ngram_range=(1, 1), norm='l2', preprocessor=None,
                smooth_idf=True, stop_words=None, strip_accents=None,
                sublinear_tf=False, token_pattern='(?u)\\b\\w\\w+\\b',
                tokenizer=None, use_idf=True, vocabulary=None)
```

In [11]:

```
X_train
```

Out[11]:

	text
940	@prettynumbers @TeamCavuto @Apple For so many ...
536	#iPhone6 users complain to @Apple [#Apple] abo...
466	RT @unstanningzarry: my phone keeps fucking fr...
158	Footage of the Apple-1 computer running. #Comp...
368	@FastCoExist If you use @Apple congrats you ju...
...	...
777	#Apple #AAPL Plans to Launch #iPhone6Mini Vers...
983	@Apple honestly sucks
1190	3 More #AppleWatch Details Exposed #aapl http:...
255	BloombergTV: Steve #Wozniak on What Really Ha...
1150	Aw yeah! monarchywpb sticker to ride with @Rhy...

1115 rows × 1 columns

In [13]:

```
X_train_vec = vectorizer.transform(X_train['text'])
X_test_vec = vectorizer.transform(X_test['text'])
```

In [14]:

```
X_train_vec.shape
```

Out[14]:

```
(1115, 1)
```

In [15]:

```
def test(model):
    print(model)
    model.fit(X_train_vec, Y_train)
    print("accuracy:", accuracy_score(Y_test, model.predict(X_test_vec)))
```

In [16]:

```
test(LogisticRegression(solver='lbfgs', multi_class='auto'))
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=100,
                   multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
accuracy: 0.5483870967741935
```

In [17]:

```
test(LinearSVC())
```

```
LinearSVC(C=1.0, class_weight=None, dual=True, fit_intercept=True,
          intercept_scaling=1, loss='squared_hinge', max_iter=1000,
          multi_class='ovr', penalty='l2', random_state=None, tol=0.0001,
          verbose=0)
accuracy: 0.5483870967741935
```

In [18]:

```
test(MultinomialNB())
```

```
MultinomialNB(alpha=1.0, class_prior=None, fit_prior=True)
accuracy: 0.5483870967741935
```

In [19]:

```
test(ComplementNB())
```

```
ComplementNB(alpha=1.0, class_prior=None, fit_prior=True, norm=False)
accuracy: 0.45161290322580644
```

In [20]:

```
test(BernoulliNB())
```

```
BernoulliNB(alpha=1.0, binarize=0.0, class_prior=None, fit_prior=True)
accuracy: 0.5483870967741935
```

Вывод

На данном датасете все предложенные методы показывают одинаковые результаты, можно отметить, что выбивается в худшую сторону метод Complement Naive Bayes.