# Лабораторная работа №6

## Васильев А.Р. ИУ5-24М

### Цель лабораторной работы: изучение методов классификации текстов.

### Требования к отчету:

Отчет по лабораторной работе должен содержать:

- титульный лист;
- описание задания;
- текст программы;
- экранные формы с примерами выполнения программы.

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

- Способ 1. На основе CountVectorizer или TfidfVectorizer.
- Способ 2. На основе моделей word2vec или Glove или fastText.

In [ ]:
```
import nltk
import spacy
import numpy as np
from sklearn.datasets import fetch_20newsgroups
nltk.download('punkt')
from nltk import tokenize
import re
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

## Будем использовать датасет 20 newsgroups

In [ ]:
```
categories = ["rec.autos", "rec.sport.hockey", "sci.crypt", "sci.med", "talk.
newsgroups_train = fetch_20newsgroups(subset='train', categories=categories)
newsgroups_test = fetch_20newsgroups(subset='test', categories=categories)
```

In [ ]:
```
unique, frequency = np.unique(newsgroups_train.target,
                              return_counts = True)
```

In [ ]:
```
for l, f in zip(unique, frequency):
  print(f'value: {l}, count: {f}')
```

```
value: 0, count: 594
value: 1, count: 600
value: 2, count: 595
value: 3, count: 594
value: 4, count: 377
```

In [ ]:
```
print('Tokenizers NLTK have')
for i in dir(tokenize)[:16]:
  print(i)
```

```
Tokenizers NLTK have
BlanklineTokenizer
LineTokenizer
MWETokenizer
PunktSentenceTokenizer
RegexpTokenizer
ReppTokenizer
SExprTokenizer
SpaceTokenizer
StanfordSegmenter
TabTokenizer
TextTilingTokenizer
ToktokTokenizer
TreebankWordTokenizer
TweetTokenizer
WhitespaceTokenizer
WordPunctTokenizer
```

# Подготовка текстов

In [ ]:
```
from spacy.lang.en import English
import spacy
from nltk.corpus import stopwords

nlp = spacy.load("en_core_web_sm", disable=["parser", "ner"])
nltk.download('stopwords')
stopwords_eng = set(stopwords.words('english'))
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

In [ ]:
```
def prepare(t):
  # t = ' '.join([i.strip().lower() for i in t.split(' ')])
  t = re.sub(r'[^a-zA-Z0-9 \n]', '', t)
  t = re.sub('\s+', ' ', t)
  t = ' '.join([token.lemma_.lower() for token in nlp(t) if token not in stop
  return t

texts = newsgroups_train.data

texts_array = []

for text in texts:
  prepared_text = prepare(text)
  texts_array.append(prepared_text)
```

```
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
<ipython-input-7-fda0f8609e81> in <module>()
     11
     12 for text in texts:
---> 13   prepared_text = prepare(text)
     14   texts_array.append(prepared_text)

<ipython-input-7-fda0f8609e81> in prepare(t)
      3   t = re.sub(r'[^a-zA-Z0-9 \n]', '', t)
      4   t = re.sub('\s+', ' ', t)
----> 5   t = ' '.join([token.lemma_.lower() for token in nlp(t) if token not
in stopwords_eng])
      6   return t
```

```
      7

/usr/local/lib/python3.7/dist-packages/spacy/language.py in __call__(self, te
xt, disable, component_cfg)
    429                 Errors.E088.format(length=len(text), max_length=self.
max_length)
    430             )
--> 431         doc = self.make_doc(text)
    432         if component_cfg is None:
    433             component_cfg = {}

/usr/local/lib/python3.7/dist-packages/spacy/language.py in make_doc(self, te
xt)
    455
    456     def make_doc(self, text):
--> 457         return self.tokenizer(text)
    458
    459     def _format_docs_and_golds(self, docs, golds):

tokenizer.pyx in spacy.tokenizer.Tokenizer.__call__()

doc.pyx in spacy.tokens.doc.Doc.__init__()

doc.pyx in spacy.tokens.doc._get_chunker()

/usr/local/lib/python3.7/dist-packages/spacy/util.py in get_lang_class(lang)
     69     # Check if language is registered / entry point is available
     70     if lang in registry.languages:
---> 71         return registry.languages.get(lang)
     72     else:
     73         try:

/usr/local/lib/python3.7/dist-packages/catalogue.py in get(self, name)
     90         """
     91         if self.entry_points:
---> 92             from_entry_point = self.get_entry_point(name)
     93             if from_entry_point:
     94                 return from_entry_point

/usr/local/lib/python3.7/dist-packages/catalogue.py in get_entry_point(self,
 name, default)
    136         RETURNS (Any): The loaded entry point or the default value.
    137         """
--> 138         for entry_point in AVAILABLE_ENTRY_POINTS.get(self.entry_poin
t_namespace, []):
    139             if entry_point.name == name:
    140                 return entry_point.load()

/usr/local/lib/python3.7/dist-packages/importlib_metadata/__init__.py in get
(self, name, default)
    309
    310     def get(self, name, default=None):
--> 311         flake8_bypass(self._warn)()
    312         return super().get(name, default)
    313

/usr/local/lib/python3.7/dist-packages/importlib_metadata/__init__.py in flak
e8_bypass(func)
    270     import inspect
    271
--> 272     is_flake8 = any('flake8' in str(frame.filename) for frame in insp
ect.stack()[:5])
    273     return func if not is_flake8 else lambda: None
    274

/usr/lib/python3.7/inspect.py in stack(context)
   1511 def stack(context=1):
   1512     """Return a list of records for the stack above the caller's fram
e."""
```

```
-> 1513     return getouterframes(sys._getframe(1), context)
   1514
   1515 def trace(context=1):

/usr/lib/python3.7/inspect.py in getouterframes(frame, context)
   1488     framelist = []
   1489     while frame:
-> 1490         frameinfo = (frame,) + getframeinfo(frame, context)
   1491         framelist.append(FrameInfo(*frameinfo))
   1492         frame = frame.f_back

/usr/lib/python3.7/inspect.py in getframeinfo(frame, context)
   1462         start = lineno - 1 - context//2
   1463         try:
-> 1464             lines, lnum = findsource(frame)
   1465         except OSError:
   1466             lines = index = None

/usr/lib/python3.7/inspect.py in findsource(object)
    778             raise OSError('source code not available')
    779
--> 780     module = getmodule(object, file)
    781     if module:
    782         lines = linecache.getlines(file, module.__dict__)

/usr/lib/python3.7/inspect.py in getmodule(object, _filename)
    731     # Copy sys.modules in order to cope with changes while iterating
    732     for modname, module in sys.modules.copy().items():
--> 733         if ismodule(module) and hasattr(module, '__file__'):
    734             f = module.__file__
    735             if f == _filesbymodname.get(modname, None):

/usr/lib/python3.7/inspect.py in ismodule(object)
     68         __doc__         documentation string
     69         __file__        filename (missing for built-in modules)"""
---> 70     return isinstance(object, types.ModuleType)
     71
     72 def isclass(object):

KeyboardInterrupt:
```

In [ ]:
```
len(texts_array), texts_array[-1]
```

In [ ]:
```
test_texts_arr = []

test_texts = newsgroups_test.data

for text in test_texts:
  prepared_text = prepare(text)
  test_texts_arr.append(prepared_text)
```

# Способ 1 На основе CountVectorizer и TfidfVectorizer

In [ ]:
```
from sklearn.feature_extraction.text import TfidfVectorizer, CountVectorizer
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import classification_report
```

In [ ]:
```
tfidf_vectorizer = TfidfVectorizer()

train_feature_matrix_tfidf = tfidf_vectorizer.fit_transform(texts_array)
test_feature_matrix__tfidf = tfidf_vectorizer.transform(test_texts_arr)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-9-3cb14c8756a3> in <module>()
      2
      3 train_feature_matrix_tfidf = tfidf_vectorizer.fit_transform(texts_arr
ay)
----> 4 test_feature_matrix__tfidf = tfidf_vectorizer.transform(test_texts_ar
r)

NameError: name 'test_texts_arr' is not defined
```

In [ ]:
```
count_vectorizer = CountVectorizer()

train_feature_matrix_count = count_vectorizer.fit_transform(texts_array)
test_feature_matrix_count = count_vectorizer.transform(test_texts_arr)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-10-262f034a8015> in <module>()
      2
      3 train_feature_matrix_count = count_vectorizer.fit_transform(texts_arr
ay)
----> 4 test_feature_matrix_count = count_vectorizer.transform(test_texts_arr
)

NameError: name 'test_texts_arr' is not defined
```

In [ ]:
```
target_values_train = newsgroups_train.target
target_values_test = newsgroups_test.target
```

knn with count vectorizer

In [ ]:
```
knn_count = KNeighborsClassifier()

knn_count.fit(train_feature_matrix_count, target_values_train)
pred_count = knn_count.predict(test_feature_matrix_count)

print(classification_report(target_values_test, pred_count))
```

```
              precision    recall  f1-score   support

           0       0.38      0.71      0.49       396
           1       0.64      0.55      0.59       399
           2       0.63      0.53      0.58       396
           3       0.53      0.36      0.43       396
           4       0.57      0.35      0.44       251

    accuracy                           0.51      1838
   macro avg       0.55      0.50      0.51      1838
weighted avg       0.55      0.51      0.51      1838
```

knn with tfidf vectorizer

In [ ]:
```
knn_tfidf = KNeighborsClassifier()

knn_tfidf.fit(train_feature_matrix_tfidf, target_values_train)
pred_knn = knn_tfidf.predict(test_feature_matrix__tfidf)

print(classification_report(target_values_test, pred_knn))
```

```
              precision    recall  f1-score   support
```

```
              0       0.93      0.91      0.92       396
              1       0.98      0.90      0.94       399
              2       0.84      0.90      0.87       396
              3       0.94      0.65      0.77       396
              4       0.57      0.88      0.69       251

     accuracy                           0.85      1838
    macro avg       0.85      0.85      0.84      1838
 weighted avg       0.88      0.85      0.85      1838
```

# Способ 2 На основе моделей word2vec или Glove или fastText.

In [ ]:

```
import tqdm
from gensim.models import Word2Vec
import gensim.downloader
# gensim.downloader.info()
# glove_vectors = gensim.downloader.load('glove-twitter-25')
glove_vectors = gensim.downloader.load('glove-wiki-gigaword-50')
```

```
[==================================================] 100.0% 66.0/66.0MB downl
oaded
```

In [ ]:

```
class GloveTokenizer:
  def __init__(self, glove_tokenizer):
    self.glove = glove_tokenizer
    self.token_length = 800
    self.embedding_size = 50

  def __getitem__(self, word):
    try:
      vector = glove_vectors.get_vector(word).reshape(1, self.embedding_size)
    except KeyError as e:
      vector = np.zeros((1, self.embedding_size))
    return vector


  def __padd(self, sentence):
    padded_sentence = np.zeros((self.token_length, self.embedding_size))
    for i, token in enumerate(sentence):
        padded_sentence[i] = token
    return padded_sentence

  def tokenize(self, sentence):
    encoded_sentence = []
    sentence = sentence.strip(' ').split(' ')
    for i in sentence:
      token = self.__getitem__(i)
      encoded_sentence.append(token)
    return np.array(self.__padd(encoded_sentence), dtype=np.float16)

tokenizer = GloveTokenizer(glove_vectors)
```

In [ ]:

```
def prepare(t):
  # t = ' '.join([i.strip().lower() for i in t.split(' ')])
  t = re.sub(r'[^a-zA-Z0-9 \n]', '', t)
  t = re.sub('\s+', ' ', t)
  lemmas = [token.lemma_.lower() for token in nlp(t) if token not in stopworc
  t = ' '.join(lemmas)
  vectors = tokenizer.tokenize(t)
  return vectors, len(lemmas)
```

```
vectors_array_train = []
labels_train = []

for enum, text, label in zip(range(len(newsgroups_train.data)), newsgroups_tr
  try:
    vector, length = prepare(text)
    # print(vector, vector.shape)
    vectors_array_train.append(vector)
    labels_train.append(label)
  except IndexError as e:
    print(enum, e)
    continue


vectors_array_train = np.array(vectors_array_train)
print(vectors_array_train.shape)
train_data = vectors_array_train.reshape((-1, vectors_array_train.shape[1]*ve
train_data.shape
```

```
56 index 800 is out of bounds for axis 0 with size 800
58 index 800 is out of bounds for axis 0 with size 800
93 index 800 is out of bounds for axis 0 with size 800
112 index 800 is out of bounds for axis 0 with size 800
145 index 800 is out of bounds for axis 0 with size 800
147 index 800 is out of bounds for axis 0 with size 800
159 index 800 is out of bounds for axis 0 with size 800
214 index 800 is out of bounds for axis 0 with size 800
215 index 800 is out of bounds for axis 0 with size 800
217 index 800 is out of bounds for axis 0 with size 800
222 index 800 is out of bounds for axis 0 with size 800
225 index 800 is out of bounds for axis 0 with size 800
248 index 800 is out of bounds for axis 0 with size 800
265 index 800 is out of bounds for axis 0 with size 800
267 index 800 is out of bounds for axis 0 with size 800
268 index 800 is out of bounds for axis 0 with size 800
281 index 800 is out of bounds for axis 0 with size 800
298 index 800 is out of bounds for axis 0 with size 800
336 index 800 is out of bounds for axis 0 with size 800
361 index 800 is out of bounds for axis 0 with size 800
395 index 800 is out of bounds for axis 0 with size 800
412 index 800 is out of bounds for axis 0 with size 800
416 index 800 is out of bounds for axis 0 with size 800
424 index 800 is out of bounds for axis 0 with size 800
462 index 800 is out of bounds for axis 0 with size 800
468 index 800 is out of bounds for axis 0 with size 800
480 index 800 is out of bounds for axis 0 with size 800
481 index 800 is out of bounds for axis 0 with size 800
523 index 800 is out of bounds for axis 0 with size 800
568 index 800 is out of bounds for axis 0 with size 800
574 index 800 is out of bounds for axis 0 with size 800
585 index 800 is out of bounds for axis 0 with size 800
587 index 800 is out of bounds for axis 0 with size 800
588 index 800 is out of bounds for axis 0 with size 800
591 index 800 is out of bounds for axis 0 with size 800
680 index 800 is out of bounds for axis 0 with size 800
696 index 800 is out of bounds for axis 0 with size 800
708 index 800 is out of bounds for axis 0 with size 800
714 index 800 is out of bounds for axis 0 with size 800
715 index 800 is out of bounds for axis 0 with size 800
733 index 800 is out of bounds for axis 0 with size 800
734 index 800 is out of bounds for axis 0 with size 800
743 index 800 is out of bounds for axis 0 with size 800
753 index 800 is out of bounds for axis 0 with size 800
816 index 800 is out of bounds for axis 0 with size 800
864 index 800 is out of bounds for axis 0 with size 800
901 index 800 is out of bounds for axis 0 with size 800
911 index 800 is out of bounds for axis 0 with size 800
```

```
934 index 800 is out of bounds for axis 0 with size 800
962 index 800 is out of bounds for axis 0 with size 800
965 index 800 is out of bounds for axis 0 with size 800
1015 index 800 is out of bounds for axis 0 with size 800
1022 index 800 is out of bounds for axis 0 with size 800
1112 index 800 is out of bounds for axis 0 with size 800
1116 index 800 is out of bounds for axis 0 with size 800
1155 index 800 is out of bounds for axis 0 with size 800
1167 index 800 is out of bounds for axis 0 with size 800
1170 index 800 is out of bounds for axis 0 with size 800
1173 index 800 is out of bounds for axis 0 with size 800
1181 index 800 is out of bounds for axis 0 with size 800
1209 index 800 is out of bounds for axis 0 with size 800
1265 index 800 is out of bounds for axis 0 with size 800
1269 index 800 is out of bounds for axis 0 with size 800
1292 index 800 is out of bounds for axis 0 with size 800
1294 index 800 is out of bounds for axis 0 with size 800
1309 index 800 is out of bounds for axis 0 with size 800
1339 index 800 is out of bounds for axis 0 with size 800
1357 index 800 is out of bounds for axis 0 with size 800
1372 index 800 is out of bounds for axis 0 with size 800
1398 index 800 is out of bounds for axis 0 with size 800
1412 index 800 is out of bounds for axis 0 with size 800
1413 index 800 is out of bounds for axis 0 with size 800
1422 index 800 is out of bounds for axis 0 with size 800
1426 index 800 is out of bounds for axis 0 with size 800
1429 index 800 is out of bounds for axis 0 with size 800
1442 index 800 is out of bounds for axis 0 with size 800
1457 index 800 is out of bounds for axis 0 with size 800
1462 index 800 is out of bounds for axis 0 with size 800
1481 index 800 is out of bounds for axis 0 with size 800
1490 index 800 is out of bounds for axis 0 with size 800
1501 index 800 is out of bounds for axis 0 with size 800
1527 index 800 is out of bounds for axis 0 with size 800
1544 index 800 is out of bounds for axis 0 with size 800
1551 index 800 is out of bounds for axis 0 with size 800
1584 index 800 is out of bounds for axis 0 with size 800
1586 index 800 is out of bounds for axis 0 with size 800
1592 index 800 is out of bounds for axis 0 with size 800
1605 index 800 is out of bounds for axis 0 with size 800
1620 index 800 is out of bounds for axis 0 with size 800
1625 index 800 is out of bounds for axis 0 with size 800
1672 index 800 is out of bounds for axis 0 with size 800
1728 index 800 is out of bounds for axis 0 with size 800
1766 index 800 is out of bounds for axis 0 with size 800
1781 index 800 is out of bounds for axis 0 with size 800
1797 index 800 is out of bounds for axis 0 with size 800
1804 index 800 is out of bounds for axis 0 with size 800
1811 index 800 is out of bounds for axis 0 with size 800
1812 index 800 is out of bounds for axis 0 with size 800
1819 index 800 is out of bounds for axis 0 with size 800
1825 index 800 is out of bounds for axis 0 with size 800
1832 index 800 is out of bounds for axis 0 with size 800
1854 index 800 is out of bounds for axis 0 with size 800
1883 index 800 is out of bounds for axis 0 with size 800
1891 index 800 is out of bounds for axis 0 with size 800
1908 index 800 is out of bounds for axis 0 with size 800
1924 index 800 is out of bounds for axis 0 with size 800
1950 index 800 is out of bounds for axis 0 with size 800
1958 index 800 is out of bounds for axis 0 with size 800
2005 index 800 is out of bounds for axis 0 with size 800
2061 index 800 is out of bounds for axis 0 with size 800
2086 index 800 is out of bounds for axis 0 with size 800
2104 index 800 is out of bounds for axis 0 with size 800
2105 index 800 is out of bounds for axis 0 with size 800
2138 index 800 is out of bounds for axis 0 with size 800
2149 index 800 is out of bounds for axis 0 with size 800
2167 index 800 is out of bounds for axis 0 with size 800
2175 index 800 is out of bounds for axis 0 with size 800
```

```
2180 index 800 is out of bounds for axis 0 with size 800
2220 index 800 is out of bounds for axis 0 with size 800
2229 index 800 is out of bounds for axis 0 with size 800
2243 index 800 is out of bounds for axis 0 with size 800
2260 index 800 is out of bounds for axis 0 with size 800
2262 index 800 is out of bounds for axis 0 with size 800
2263 index 800 is out of bounds for axis 0 with size 800
2304 index 800 is out of bounds for axis 0 with size 800
2328 index 800 is out of bounds for axis 0 with size 800
2354 index 800 is out of bounds for axis 0 with size 800
2356 index 800 is out of bounds for axis 0 with size 800
2373 index 800 is out of bounds for axis 0 with size 800
2391 index 800 is out of bounds for axis 0 with size 800
2419 index 800 is out of bounds for axis 0 with size 800
2428 index 800 is out of bounds for axis 0 with size 800
2462 index 800 is out of bounds for axis 0 with size 800
2466 index 800 is out of bounds for axis 0 with size 800
2469 index 800 is out of bounds for axis 0 with size 800
2487 index 800 is out of bounds for axis 0 with size 800
2500 index 800 is out of bounds for axis 0 with size 800
2516 index 800 is out of bounds for axis 0 with size 800
2517 index 800 is out of bounds for axis 0 with size 800
2559 index 800 is out of bounds for axis 0 with size 800
2603 index 800 is out of bounds for axis 0 with size 800
2616 index 800 is out of bounds for axis 0 with size 800
2628 index 800 is out of bounds for axis 0 with size 800
2652 index 800 is out of bounds for axis 0 with size 800
2654 index 800 is out of bounds for axis 0 with size 800
2669 index 800 is out of bounds for axis 0 with size 800
2678 index 800 is out of bounds for axis 0 with size 800
2683 index 800 is out of bounds for axis 0 with size 800
2743 index 800 is out of bounds for axis 0 with size 800
2754 index 800 is out of bounds for axis 0 with size 800
(2610, 800, 50)
```

Out[ ]:
```
(2610, 40000)
```

In [ ]:
```
vectors_array_test = []
labels_test= []

for enum, text, label in zip(range(len(newsgroups_test.data)), newsgroups_tes
  try:
    vector, length = prepare(text)
    vectors_array_test.append(vector)
    labels_test.append(label)
  except IndexError as e:
    print(enum, e)
    continue
```

```
67 index 800 is out of bounds for axis 0 with size 800
76 index 800 is out of bounds for axis 0 with size 800
124 index 800 is out of bounds for axis 0 with size 800
137 index 800 is out of bounds for axis 0 with size 800
155 index 800 is out of bounds for axis 0 with size 800
187 index 800 is out of bounds for axis 0 with size 800
292 index 800 is out of bounds for axis 0 with size 800
298 index 800 is out of bounds for axis 0 with size 800
350 index 800 is out of bounds for axis 0 with size 800
432 index 800 is out of bounds for axis 0 with size 800
435 index 800 is out of bounds for axis 0 with size 800
458 index 800 is out of bounds for axis 0 with size 800
476 index 800 is out of bounds for axis 0 with size 800
484 index 800 is out of bounds for axis 0 with size 800
525 index 800 is out of bounds for axis 0 with size 800
556 index 800 is out of bounds for axis 0 with size 800
558 index 800 is out of bounds for axis 0 with size 800
618 index 800 is out of bounds for axis 0 with size 800
680 index 800 is out of bounds for axis 0 with size 800
```

```
683 index 800 is out of bounds for axis 0 with size 800
710 index 800 is out of bounds for axis 0 with size 800
715 index 800 is out of bounds for axis 0 with size 800
720 index 800 is out of bounds for axis 0 with size 800
755 index 800 is out of bounds for axis 0 with size 800
778 index 800 is out of bounds for axis 0 with size 800
780 index 800 is out of bounds for axis 0 with size 800
802 index 800 is out of bounds for axis 0 with size 800
819 index 800 is out of bounds for axis 0 with size 800
825 index 800 is out of bounds for axis 0 with size 800
832 index 800 is out of bounds for axis 0 with size 800
836 index 800 is out of bounds for axis 0 with size 800
862 index 800 is out of bounds for axis 0 with size 800
882 index 800 is out of bounds for axis 0 with size 800
919 index 800 is out of bounds for axis 0 with size 800
956 index 800 is out of bounds for axis 0 with size 800
960 index 800 is out of bounds for axis 0 with size 800
989 index 800 is out of bounds for axis 0 with size 800
1064 index 800 is out of bounds for axis 0 with size 800
1101 index 800 is out of bounds for axis 0 with size 800
1108 index 800 is out of bounds for axis 0 with size 800
1152 index 800 is out of bounds for axis 0 with size 800
1187 index 800 is out of bounds for axis 0 with size 800
1193 index 800 is out of bounds for axis 0 with size 800
1293 index 800 is out of bounds for axis 0 with size 800
1313 index 800 is out of bounds for axis 0 with size 800
1337 index 800 is out of bounds for axis 0 with size 800
1386 index 800 is out of bounds for axis 0 with size 800
1415 index 800 is out of bounds for axis 0 with size 800
1443 index 800 is out of bounds for axis 0 with size 800
1455 index 800 is out of bounds for axis 0 with size 800
1463 index 800 is out of bounds for axis 0 with size 800
1477 index 800 is out of bounds for axis 0 with size 800
1482 index 800 is out of bounds for axis 0 with size 800
1517 index 800 is out of bounds for axis 0 with size 800
1529 index 800 is out of bounds for axis 0 with size 800
1552 index 800 is out of bounds for axis 0 with size 800
1560 index 800 is out of bounds for axis 0 with size 800
1561 index 800 is out of bounds for axis 0 with size 800
1629 index 800 is out of bounds for axis 0 with size 800
1631 index 800 is out of bounds for axis 0 with size 800
1639 index 800 is out of bounds for axis 0 with size 800
1664 index 800 is out of bounds for axis 0 with size 800
1699 index 800 is out of bounds for axis 0 with size 800
1709 index 800 is out of bounds for axis 0 with size 800
1717 index 800 is out of bounds for axis 0 with size 800
1770 index 800 is out of bounds for axis 0 with size 800
1828 index 800 is out of bounds for axis 0 with size 800
1837 index 800 is out of bounds for axis 0 with size 800
```

In [ ]:

```
vectors_array_test = np.array(vectors_array_test)
test_data = vectors_array_test.reshape((-1, vectors_array_test.shape[1]*vecto
test_data.shape
```

Out[ ]:

```
(1770, 40000)
```

In [ ]:

```
from sklearn.neighbors import KNeighborsClassifier

knn_clf = KNeighborsClassifier()

knn_clf.fit(train_data, labels_train)
```

Out[ ]:

```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                     metric_params=None, n_jobs=None, n_neighbors=5, p=2,
                     weights='uniform')
```

In [ ]:
```
pred = knn_clf.predict(test_data[:800])

print(classification_report(labels_test[:800], pred))
```

```
              precision    recall  f1-score   support

           0       0.31      0.70      0.43       172
           1       0.59      0.25      0.36       173
           2       0.46      0.44      0.45       179
           3       0.38      0.17      0.24       162
           4       0.33      0.28      0.30       114

    accuracy                           0.38       800
   macro avg       0.42      0.37      0.36       800
weighted avg       0.42      0.38      0.36       800
```

In [ ]:
```

```