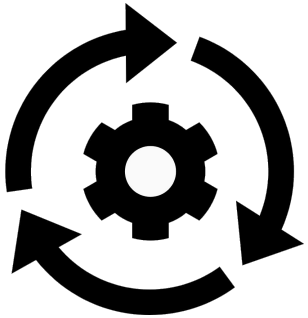
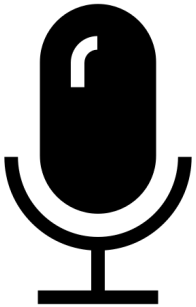
tf-idf

План

Получаем задание



Реализуем



Демонстрируем

Задание #1: **count vectorizer**

Crock Pot Pasta

Never boil pasta again

Pasta Pomodoro

Fresh ingredients Parmesan to taste

Задание #1: **count vectorizer**

```
Crock Pot Pasta

Never boil pasta again
```

```
Pasta Pomodoro

Fresh ingredients Parmesan to taste
```

Список слов

0	crock	7	fresh
1	pot	8	ingredients
2	pasta	9	parmesan
3	never	10	to
4	boil	11	taste
5	again		
6	pomodoro		

Задание #1: **count vectorizer**

```
Crock Pot Pasta

Never boil pasta again
```

```
Pasta Pomodoro

Fresh ingredients Parmesan to taste
```

Список слов

0	crock	7	fresh
1	pot	8	ingredients
2	pasta	9	parmesan
3	never	10	to
4	boil	11	taste
5	again		
6	pomodoro		

Вектора

```
1 [1, 1, 2, 1, 1, 1, 0, 0, 0, 0, 0, 0]
2 [0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1]
```

Задание #1: **count vectorizer**

Реализуйте класс `CountVectorizer`, имеющий метод `fit_transform`

```
corpus = [
    'Crock Pot Pasta Never boil pasta again',
    'Pasta Pomodoro Fresh ingredients Parmesan to taste'
]

vectorizer = CountVectorizer()
count_matrix = vectorizer.fit_transform(corpus)
print(vectorizer.get_feature_names())
Out: ['crock', 'pot', 'pasta', 'never', 'boil', 'again', 'pomodoro',
      'fresh', 'ingredients', 'parmesan', 'to', 'taste']

print(count_matrix)
Out: [[1, 1, 2, 1, 1, 1, 0, 0, 0, 0, 0, 0],
      [0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1]]
```

Задание #2: **term frequency**

Spaghetti and Meatballs

Making your own meatballs and sauce makes it even better

Задание #2: **term frequency**

Spaghetti and Meatballs

Making your own meatballs and sauce makes it even better

$$tf = \frac{\text{повторений}}{\text{всего}}$$

	повторений	tf
spaghetti	1	0,077
and	2	0,154
meatballs	2	0,154
making	1	0,077
your	1	0,077
...		
всего	**13**	

Задание #2: **term frequency**

Penne Alla Vodka

True story: This is the best vodka sauce the Delish team has tasted

$$tf = \frac{\text{повторений}}{\text{всего}}$$

	повторений	**tf**
vodka	?	?
всего	**?**	

Задание #2: **term frequency**

Penne Alla Vodka

True story: This is the best vodka sauce the Delish team has tasted

$$tf = \frac{\text{повторений}}{\text{всего}}$$

	повторений	**tf**
vodka	2	0,125
всего	**16**	

Задание #2: **term frequency**

Реализуйте функцию `tf_transform`

```
count_matrix = [
   [1, 1, 2, 1, 1, 1, 0, 0, 0, 0, 0, 0],
   [0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1]
]
tf_matrix = tf_transform(count_matrix)

print(tf_matrix)
Out: [[0.143, 0.143, 0.286, 0.143, 0.143, 0.143, 0, 0, 0, 0, 0, 0],
      [0, 0, 0.143, 0, 0, 0, 0.143, 0.143, 0.143, 0.143, 0.143, 0.143]]
```

Задание #3: **inverse document-frequency**

Crock Pot Pasta

Never boil pasta again

Pasta Pomodoro

Fresh ingredients Parmesan to taste

$$idf = ln(\frac{\text{всего документов} + 1}{\text{документов со словом} + 1}) + 1$$

	док-ов со словом	**idf**
crock	1	1.405
pot	1	1.405
pasta	2	1
never	1	1.405
boil	1	1.405
...		
всего документов	**2**	

Задание #3: **inverse document-frequency**

Реализуйте функцию `idf_transform`

```
count_matrix = [
   [1, 1, 2, 1, 1, 1, 0, 0, 0, 0, 0, 0],
   [0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1]
]
idf_matrix = idf_transform(count_matrix)

print(idf_matrix)
Out: [1.4, 1.4, 1.0, 1.4, 1.4, 1.4, 1.4, 1.4, 1.4, 1.4, 1.4, 1.4]
```

Задание #4: **tf-idf transformer**

Crock Pot Pasta

Never boil pasta again

Pasta Pomodoro

Fresh ingredients Parmesan to taste

$$tfidf = tf * idf$$

	tf	idf	tf-idf
crock	0.143	1.405	0.201
pot	0.143	1.405	0.201
pasta	0.286	1	0.286
never	0.143	1.405	0.201
boil	0.143	1.405	0.201
...			
pasta	0.143	1	0.143

Для получения значений как в sklearn, нужно использовать

```
from sklearn.preprocessing import normalize
```

Задание #4: **tf-idf transformer**

Реализуйте класс `TfidfTransformer`, имеющий метод `fit_transform`

```
count_matrix = [
   [1, 1, 2, 1, 1, 1, 0, 0, 0, 0, 0, 0],
   [0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1]
]
transformer = TfidfTransformer()
tfidf_matrix = transformer.fit_transform(count_matrix)

print(tfidf_matrix)
Out: [[0.2, 0.2, 0.286, 0.2, 0.2, 0.2, 0, 0, 0, 0, 0, 0],
      [0, 0, 0.143, 0, 0, 0, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2]]
```

Задание #5: **tf-idf vectorizer**

Реализуйте класс `TfidfVectorizer`, имеющий метод `fit_transform`

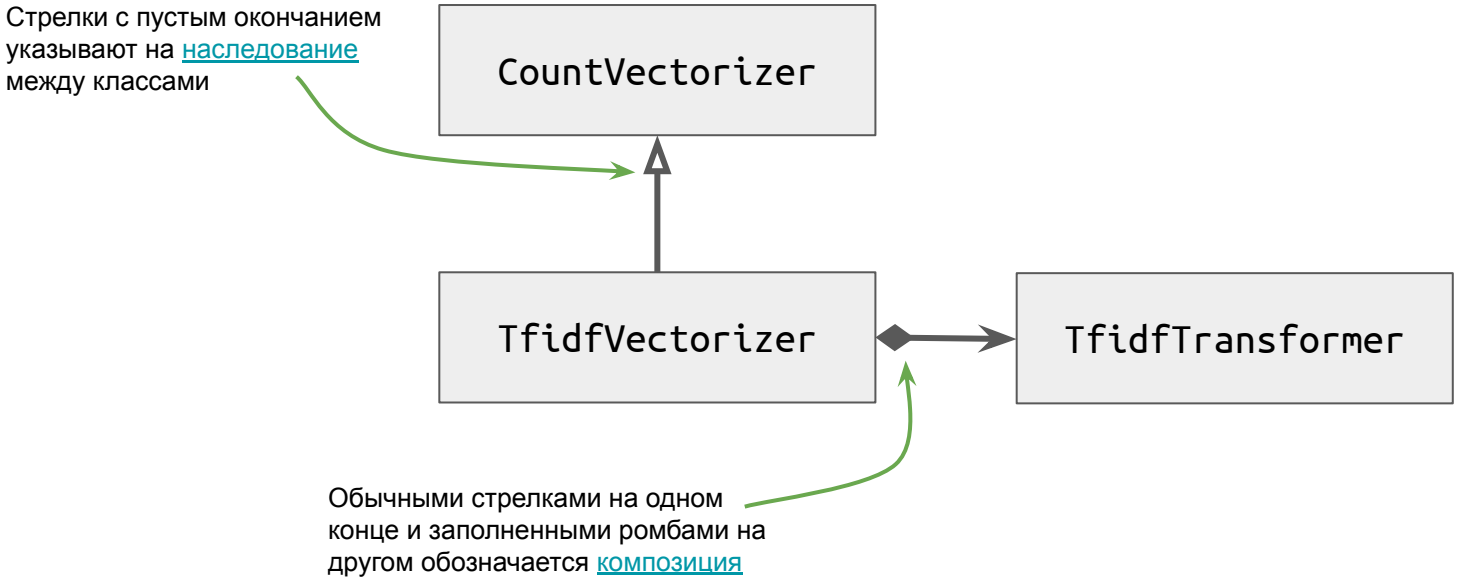
```
corpus = [
    'Crock Pot Pasta Never boil pasta again',
    'Pasta Pomodoro Fresh ingredients Parmesan to taste'
]
vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(corpus)

print(vectorizer.get_feature_names())
Out: ['crock', 'pot', 'pasta', 'never', 'boil', 'again', 'pomodoro',
      'fresh', 'ingredients', 'parmesan', 'to', 'taste']

print(tfidf_matrix)
Out: [[0.2, 0.2, 0.286, 0.2, 0.2, 0.2, 0, 0, 0, 0, 0, 0],
      [0, 0, 0.143, 0, 0, 0, 0.2, 0.2, 0.2, 0.2, 0.2, 0.2]]
```

Задание #5: **tf-idf vectorizer**

Диаграмма классов



Спасибо за проделанную работу!