1. **Постановка задачи**

Классификация письменных документов.

Применить 3-4 классификатора библиотеки scikit-learn для классификации документов, имеющихся в полученном файле.

2. **Классы документов**

№ класса	Класс	Число документов в классе
1	‘автомобиль’	249
2	‘здоровье’	157
3	‘культура’	358
4	‘наука’	227
5	‘недвижимость’	98
6	‘политика’	600
7	‘происшествие’	436
8	‘реклама’	94
9	‘семья’	101
10	‘спорт’	373
11	‘страна’	146
12	‘техника’	289
13	‘экономика’	272
Общее	**13**	**3400**

3. **Порядок предварительной обработки корпуса**

- С помощью команды `re.sub(r'[^А-я #\n]', '', data)` удаляем всё кроме русских слов, решётки и знака переноса строки
- С помощью команды `data.lower()` приводим к ниженему регистру
- Далее `data.split('\n')` делит большую строку на мелкие строки – документы, `nltk.word_tokenize(txt[i])` – токенезация,

```
morph = pymorphy2.MorphAnalyzer()
txt[i] = [morph.parse(word)[0].normal_form for word in txt[i] if word not in stop_words]
```

– отбор и лемматизация (нормализация) только тех слов, которые не являются стоп-словами.

Стоп-слова – это часто используемые слова, которые не вносят никакой дополнительной информации в текст.

Собрали стоп-слова из двух библиотек:

```
from nltk.corpus import stopwords
from stop_words import get_stop_words
stop_words = list(stopwords.words('russian'))
print(len(stop_words)) # 151
stop_words_ = list(get_stop_words('russian'))
print(len(stop_words_)) # 421
stop_words.extend(stop_words_)
```

```
print(len(stop_words)) # 572
```
Примеры стоп-слов:
[…,
'но',
'да',
'ты',
'к',
'у',
'же',
'вы',
'за',
'бы',
'по',
'только',
'ее',
'мне',
'было',
'вот',
'от',
'меня',
'еще',
'нет',
'о',
'из',
'ему',
'теперь',
'когда',
'даже',
'ну',
'вдруг',
…]

- Разделим метки и данные.

4. Размер словаря корпуса до и после предварительной обработки

Размер словаря до удаления иностранных слов, цифр и посторонних символов	Размер словаря после удаления иностранных слов, цифр и посторонних символов	Размер словаря после удаления верхних регистров и стоп-слов
84687	**81286**	**34989**

5. Примененные классификаторы и их параметры

Классификатор	Сокращённо	Время обучения, с	Параметры
KNeighborsClassifier	(KNN) - Метод K-ближайших соседей	0.0923	n_neighbors=12
PassiveAggressiveClassifier		40.737	по умолчанию
LinearDiscriminantAnalysis	(LDA) - Линейный дискриминантный анализ	93.7723	по умолчанию

В случае использования метода для классификации объект присваивается тому классу, который является наиболее распространённым среди k соседей данного элемента, классы которых уже известны.

6. **Точность классификации документов обучающего и проверочного множеств (общая и по классам)**

KNeighborsClassifier		
Категория	Оценочное %	Обучающее %
Общее	52.94	57.02
‘автомобиль’	36.84	38.54
‘здоровье’	8.33	15.04
‘культура’	40.74	49.46
‘наука’	62.5	68.45
‘недвижимость’	25.0	22.09
‘политика’	65.81	70.60
‘происшествие’	55.56	67.63
‘реклама’	0	9.64
‘семья’	8.33	19.48
‘спорт’	100	98.98
‘страна’	20	15.08
‘техника’	50.77	68.3
‘экономика’	51.67	52.83

PassiveAggressiveClassifier		
Категория	Оценочное %	Обучающее %
Общее	87.06	99.93
‘автомобиль’	82.46	100
‘здоровье’	87.50	100
‘культура’	82.72	100
‘наука’	100	100
‘недвижимость’	58.33	100
‘политика’	91.45	100
‘происшествие’	96.67	100
‘реклама’	45.45	100
‘семья’	87.50	100
‘спорт’	96.20	100

‘страна’	40.00	98.41
‘техника’	91.38	100
‘экономика’	89.83	100

LinearDiscriminantAnalysis		
Категория	Оценочное %	Обучающее %
Общее	32.97	98.2
‘автомобиль’	49.12	99.48
‘здоровье’	45.83	96.99
‘культура’	30.86	98.92
‘наука’	32.5	97.86
‘недвижимость’	33.33	94.19
‘политика’	29.06	98.34
‘происшествие’	43.33	99.42
‘реклама’	18.18	98.80
‘семья’	62.50	94.81
‘спорт’	15.19	97.62
‘страна’	20.00	98.41
‘техника’	21.54	98.21
‘экономика’	36.67	98.11