Университет ИТМО

Практическая работа №2

По дисциплине “Компьютерная лингвистика”

Автор: Лайок Олег Владимирович

Группа: К3242

Факультет: ИКТ

Преподаватель: Чернышева А.В.

Санкт-Петербург, 2021 г.

2.1: Работа с примерами на регулярные выражения

```
import re
#Напишите регулярное выражение, которое возвращает список первых двух букв каждог
о слова строки. Обратите внимание на работу с дефисом
s = "Привет как дела"
res = re.findall(r'\b\w{2}',s)
print(res)

#Напишите регулярное выражение, которое выбирает из строки все слова, в которых с
трого больше 3 символов.
s = "Привет как у тебя дела"
res = re.findall(r'\w{3,}',s)
print(res)

#Напишите регулярное выражение, которое заменит все подстроки, обозначающие время
 (только время, не даты), в строке на TBD
s = '21.02.2021 в 13:45:49 или в 12:43:34'
res = re.sub(r'\d\d:\d\d:\d\d','TBD',s)
print(res)

#Напишите регулярное выражение, которое заменяет произвольное количество пробельн
ых символов внутри строки на один пробел.
s = "Привет       как  дела     ."
res = re.sub(r'\s{2,}'," ", s)
print(res)

#Напишите регулярное выражение, которое удаляет идущие подряд повторы. Одно слово
 из группы должно остаться.
s = 'Привет привет как дела дела дела'
res = re.sub(r'\b([^\W\d_]+)(\s+\1)+\b', r'\1', s, flags=re.I)
print(res)

#Напишите регулярное выражение, которое определяет, что подстрока является адресо
м электронной почты.
s = 'laolvl@mail.ru и 286507@niuitmo.ru'
res = re.split(' ',s)
for i in res:
    if (re.match(r'\w+@\w+.\w+',i)) != None:
        print(i,' является эмейлом')

#Напишите регулярное выражение, которое возвращает список аббревиатур в строке.
s = 'НИУ ИТМО - вуз в SPB'
res = re.findall(r'[A-ZA-Я]{2,}',s)
print(res)

#Напишите регулярное выражение, которое разделяет текст на предложения
s = 'Greenpeace Франции совместно с другими НКО (Notre Affaire à Tous, Фонд Никол
я Юло и Oxfam France) требуют от властей возместить ущерб, причинённый гражданам
страны из-
за политики в области экологии и начать активные действия в рамках предыдущих сог
```

```
лашений. Соответствующий иск подали ещё два года назад из-
за бездействия государства в решении проблемы климатического кризиса? Сегодня сос
тоялось слушание дела в суде Парижа, решение по которому будет вынесено в течение
 двух недель!'
res = re.split(r'[.!?]',s)
print(res)


#Напишите регулярное выражение, которое определяет, что строка является номеромро
ссийского мобильного телефона любого оператора.
s = "+79106311575"
s.encode('utf-8')
res = re.findall(r'\+79\d{9}|89\d{9}',s)
print(res)

# Напишите регулярное выражение, которое проверяет, что все предложения в строке
начинаются с заглавной буквы.
s = 'Greenpeace Франции совместно с другими НКО (Notre Affaire à Tous, Фонд Никол
я Юло и Oxfam France) требуют от властей возместить ущерб, причинённый гражданам
страны из-
за политики в области экологии и начать активные действия в рамках предыдущих сог
лашений. Соответствующий иск подали ещё два года назад из-
за бездействия государства в решении проблемы климатического кризиса? Сегодня сос
тоялось слушание дела в суде Парижа, решение по которому будет вынесено в течение
 двух недель!'
res = re.split(r'[.!?]',s)
for i in res:
    if (re.match(r'[A-ZA-ЯЁ].*?[.?!]',i)) != None:
        print(i,' является предложением с большой буквой')
```

2.2: Очистка корпуса текстов с помощью модуля re.

Так как мой корпус текстов состоит из записей сообщества Вконтакте, то в нем имеется некоторое количество тегов и символов, которые в дальнейшем будут мешать при работе со словарем.

Для начала я удалил символы новой строки и любые типы ссылок, которые могут встретиться в данном корпусе:

```
import json
import re
with open("VK_data_lentach1.json", "r", encoding ='utf-8') as f:
    text_data = json.load(f)
for i in range(0,len(text_data)):
    text_data[i]= re.sub(r'\n', ' ', text_data[i]) #удаляем символы новой строки
    text_data[i]= re.sub(r'\bhttp.+\b', ' ', text_data[i]) #удаляем ссылки
    text_data[i] = re.sub(r'vk.cc\S+',' ',text_data[i]) #удаляем ссылки вк
```

Далее я перешел к удалению спецсимволов и тэгов уникальных для моего корпуса:

```
      text_data[i]= re.sub(r'l\.tinkoff\.ru/autolentach', ' ', text_data[i])
      text_data[i]= re.sub(r'✹', ' ', text_data[i])
      text_data[i]= re.sub(r'⚡', ' ', text_data[i])
      text_data[i]= re.sub(r'•', ' ', text_data[i])
      text_data[i]= re.sub(r'\\xa0', ' ', text_data[i])
      text_data[i] = re.sub(r'\[.*\]',' ',text_data[i]) #удаляем ссылки на групп
ы
      text_data[i] = re.sub(r'¯.+¯',' ',text_data[i]) #удаляем специальные эмодз
и
      text_data[i] = re.sub(r'#Баян_из_предложки',' ',text_data[i]) #удаляем пер
воапрельский тэг
```

В последнюю очередь, я реши удалить все лишние пробелы, которые получились при удалении спецсимволов, а так же пустые элементы конечноо списка:

```
      text_data[i] = re.sub(r'\s{2,}'," ",text_data[i]) #удаляем образовавшиеся
повторы пробелов
i=0
while i < len(text_data):#удаляем пустые элементы
    if text_data[i] == "" or text_data[i] == " ":
            del text_data[i]
    else:
        i=i+1
with open("VK_data_lentach_cleared.json", "w", encoding='utf-8') as f:
    json.dump(text_data, f, ensure_ascii=False)
f.close()
```

Таким образом, испорльзуя регулярные выражения, было очень удобно очистить текст от ‘мусора’, который в дальнейшем мешал бы правильно токенизировать корпус и проводить морфологический анализ. Данный корпус еще пока не окончательно очищен, однако все, что можно было убрать с помощью модуля re было убрано.

Ссылка на гитхаб со всеми файлами:
https://github.com/OlegLaiok/Comp_Lingv/tree/homework2