Liliia Ukhobotova

**Homework 2**

**Script**

import os

import re

import string

import nltk

from nltk.probability import FreqDist

from nltk.corpus import stopwords

from nltk import word\_tokenize,pos\_tag

nltk.download('punkt')

nltk.download('stopwords')

nltk.download('averaged\_perceptron\_tagger')

nltk.download('wordnet')

nltk.download('omw-1.4')

from nltk.stem import SnowballStemmer

import pymorphy2

from wordcloud import WordCloud

import matplotlib.pyplot as plt

txt\_data=''

s= []

single\_root\_words =[]

**#Reading a file**

dir\_name = "C:\Лилия\Загрузки\DATASETS2"

snowball = SnowballStemmer(language="russian")

def remove\_chars\_from\_text(text, chars):

return "".join([ch for ch in text if ch not in chars])

def stemming(text):

result=[]

for word in text:

result.append(snowball.stem(word))

return result

def remove\_urls(text):

url\_pattern = re.compile(r'https?://\S+|www\.\S+')

return url\_pattern.sub(r'', text)

all\_files = os.listdir(dir\_name)

**#reading csv-files into a string**

**#lowercasing**

for item in all\_files:

if item.endswith(".csv"):

file = open(os.path.join(dir\_name, item), "r")

txt\_data = txt\_data+file.read().lower()

**#removing URLs**

txt\_data = remove\_urls(txt\_data)

**#removing stop-words and punctuations**

txt\_data=re.sub(';', ' ', txt\_data)

txt\_data=re.sub('\n', ' ', txt\_data)

txt\_data=re.sub('\t', ' ', txt\_data)

txt\_data=re.sub('-', ' ', txt\_data)

spec\_chars = string.punctuation + '«»—…'

txt\_data = remove\_chars\_from\_text(txt\_data, spec\_chars)

txt\_data = remove\_chars\_from\_text(txt\_data, string.digits)

**#tokenization**

list\_of\_words = word\_tokenize(txt\_data)

**#the number of words = list length**

print('Количество слов в исходном тексте ',len(list\_of\_words))

**#removing small words**

rus\_smallwords=stopwords.words("russian")

#there are words in the texts that can also be removed

rus\_smallwords.extend(['также', 'об', 'по', 'рф','это','авг','например','очень', 'риа', 'лет'])

list\_of\_words = [word.strip() for word in list\_of\_words if word not in rus\_smallwords]

**#the number of words in the cleaned-up text = list length**

print('Количество слов в очищенном тексте - ',len(list\_of\_words))

**#transforming the list of words into a text to use the tools of nltk-library**

txt=nltk.Text(list\_of\_words)

**#transforming the list into the list of tuples containing words and their frequency in the text**

fdist = FreqDist(txt)

**#10 most common words**

print('10 самых популярных слов в очищенном тексте- ',fdist.most\_common(10))

**#visualizing a graph**

fdist.plot(10,cumulative=False)

**#Removing 2 words taking into account single root words**

txt2=nltk.Text(stemming(list\_of\_words))

fdist2 = FreqDist(txt2)

single\_root=fdist2.most\_common(10)

print('Самые популярные корни слов - ',single\_root)

for i in range(2):

s.append( single\_root[i][0])

single\_root\_words = [word.strip() for word in list\_of\_words if snowball.stem(word) in s]

**#removing duplicates and sorting**

single\_root\_words=sorted(list(set(single\_root\_words)))

print('Наиболее повторяющиеся слова с учетом словоформ - ', single\_root\_words)

**#removing single root words from the text**

clear\_text= [word.strip() for word in list\_of\_words if word not in single\_root\_words]

**#text lemmatization**

morph = pymorphy2.MorphAnalyzer()

clear\_text = [morph.parse(word)[0].normal\_form for word in clear\_text]

**#getting a list of 10 most common words**

txt=nltk.Text(clear\_text)

fdist = FreqDist(txt)

print('10 самых популярных слов после удаления этих слов- ',fdist.most\_common(10))

**#visualizing a new graph**

fdist.plot(10,cumulative=False)

**#Vizualizing a tag-cloud of most common words**

txt\_raw=" ".join(clear\_text)

w\_cloud=WordCloud(max\_words=50,background\_color="salmon").generate(txt\_raw)

plt.figure(figsize=(10,15))

plt.imshow(w\_cloud)

plt.axis("off")

plt.show()

**Results and Visualisation**

The input documents are publications of 10 news outlets. They contain information on publication date, url, edition, topic, authors, title, text, reposts, likes, views.

The results show that the number of words in initial text = 22665, the number of words in the cleaned text – 16237. The most popular word is ‘товар’.

Extended results and visualizations are below:



