Міністерство освіти і науки України
Національний технічний університет України «КПІ» імені Ігоря Сікорського
Кафедра обчислювальної техніки ФІОТ

ЗВІТ
з лабораторної роботи №5
з навчальної дисципліни «Вступ до технології Data Science»

Тема:

ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

Виконав:
Студентка 2 курсу кафедри ІПІ ФІОТ,
Навчальної групи ІТ-03
Цуканова М.С.

Перевірив:
Професор кафедри ОТ ФІОТ
Писарчук О.О.

Київ 2022

І. Мета:

виявити дослідити та узагальнити особливості інтелектуального аналізу даних з використанням спеціалізованих пакетів мови програмування Python.

ІІ. Завдання:

Результат представити у формі:

1. Результати архітектурного проектування скрипта, що реалізує інтелектуальний аналіз даних.
2. Програмний скрипт та результати його функцілнування.
3. Аналітичний звіт за результатами інтелектуального аналізу даних.

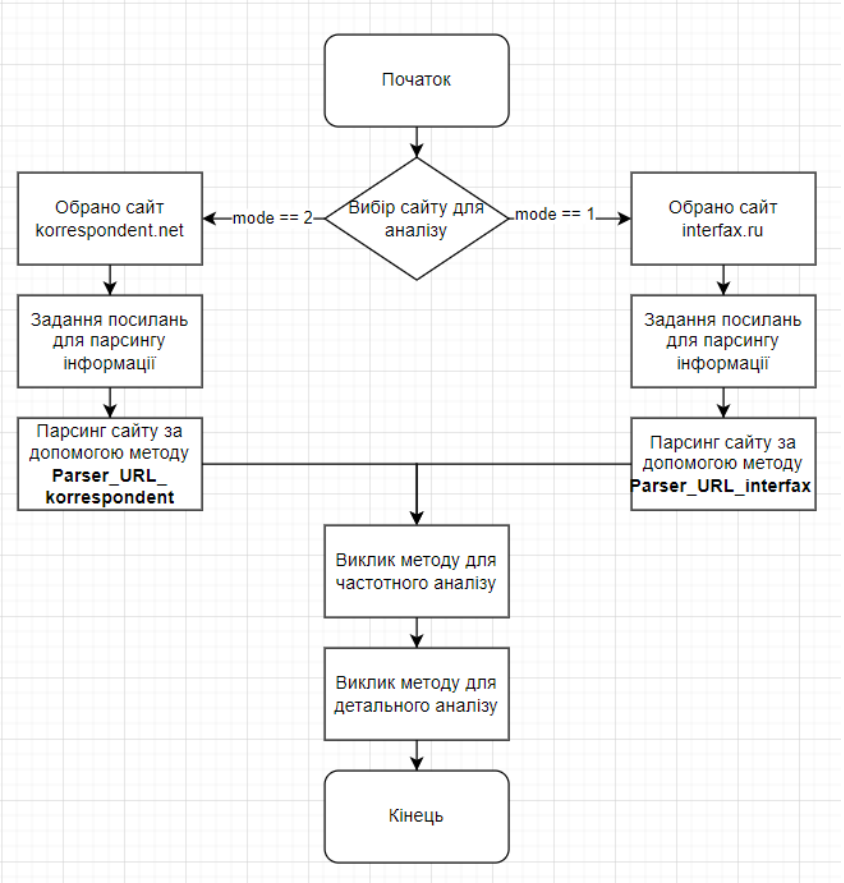
Завдання ІІ рівня складності – максимально 10 балів.

Реалізувати завдання відповідно до варіанту таблиці Д2 додатку 1.

Варіант (порядковий номер в списку групи)	**Технічні умови завдання**
1,16	Розробити програмний скрипт, що реалізує аналіз зміни активного контенту сайтів новин за даними 2 діб з моніторингом не менше 2 інформаційних джерел.

ІІІ. Результати виконання лабораторної роботи.

3.1. Результати архітектурного проектування на їх опис



мал.1, загальний алгоритм роботи програми

Перед початком роботи виводиться строка для вибору сайту для аналізу. В залежності від сайту в метод парсингу передаються посилання на сторінки, з яких потрібно брати інформацію. Перед парсингом попередній вміст файлу видаляється.

Парсинг:

Оскільки на різних сайтах різний код html, то і парсери відрізняються. Для сайту interfax спочатку вся інформація з класу ‘an’ записується в документ text_1.txt. Для обрання тільки тексту заголовків в цього документів проведено додатковий парсинг документу text-1.txt, його обробка та запис результуючих строк в файл text_1_clear.txt.

Для сайту korrespondent данные з сайту переносятся одразу в документ text_2.txt, оскільки для нього не потрібно додатково обробляти дані.

Частотний аналіз контенту сайту:

Для частотного аналізу контенту сайту використовується метод text_mining_wordcloud.

1. Весь текст з документу заноситься з один масив
2. За допомогою бібліотеки NLTK викликаємо масив stopwords з стоп-словами. З масиву з усіма слова прибираємо слова, які знаходяться в масиві з стоп-словами.
3. Підрахунок кількості кожного слова в масиві
4. Токенізація слів
5. Виведення 10ти найпопулярніших слів в консоль
6. Створення wordcloud з цих слів

Докладний аналіз контенту сайту:

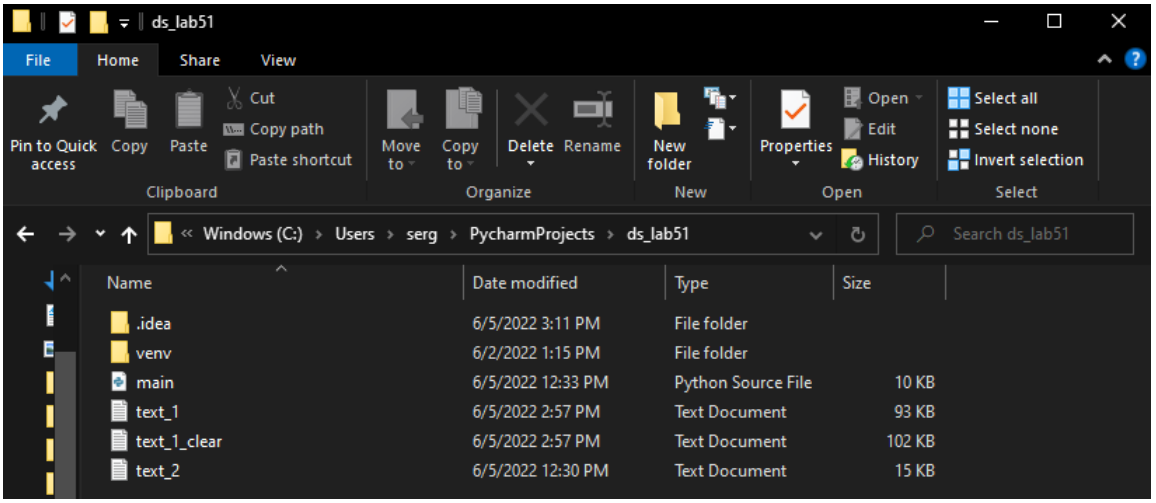
Для докладного аналізу контенту сайту використовується метод text_mining_ru.

1. Токенізація всіх слів з документу
2. Видалення стоп-слів (за допомогою stopwords NLTK)
3. Видалення лишніх символів
4. Стеммінг слів (прибирання закінчень для більш об’єктивного оцінювання)
5. Додавання слів в словник та підрахунок їх кількості
6. Виведення кількості слів, кількостів унікальних слів та всі використані слова.

3.2. Опис структури проекту програми в середовищі PyCharm.

Для реалізації розробленого алгоритму мовою програмування Python з використанням можливостей інтегрованого середовища PyCharm сформовано проект.

Проект базується на лінійній бізнес-логіці функціонального програмування та має таку структуру.

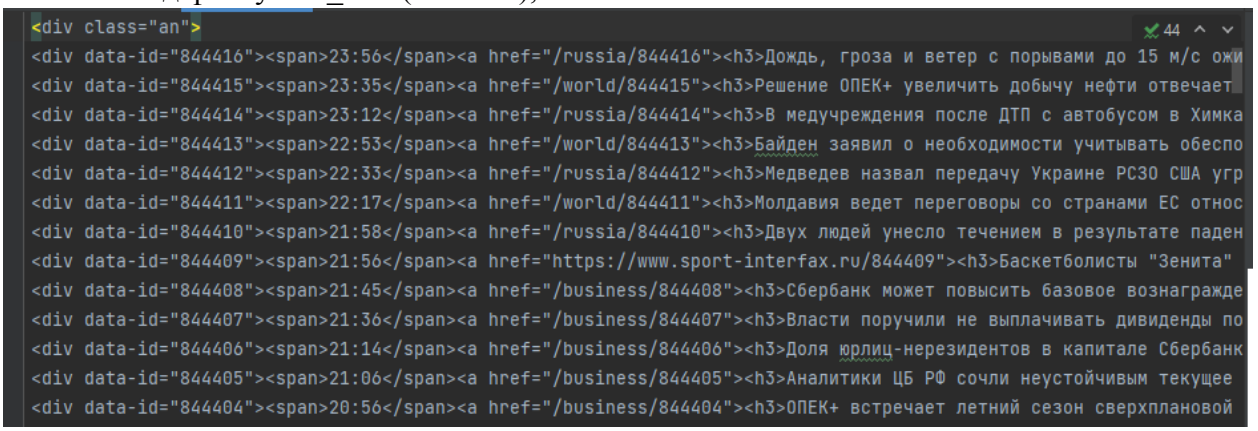


мал. 2, структура проекту

ds_lab51 - головний каталог проекту
main.py - файл програмного коду лабораторної роботи
text_1.txt - файл, куди додається інформація з сайту інтерфакс
text_1_clear.txt - файл тільки з текстом з сайту інтерфакс
text_2.txt - файл з текстом з сайту корерспондент

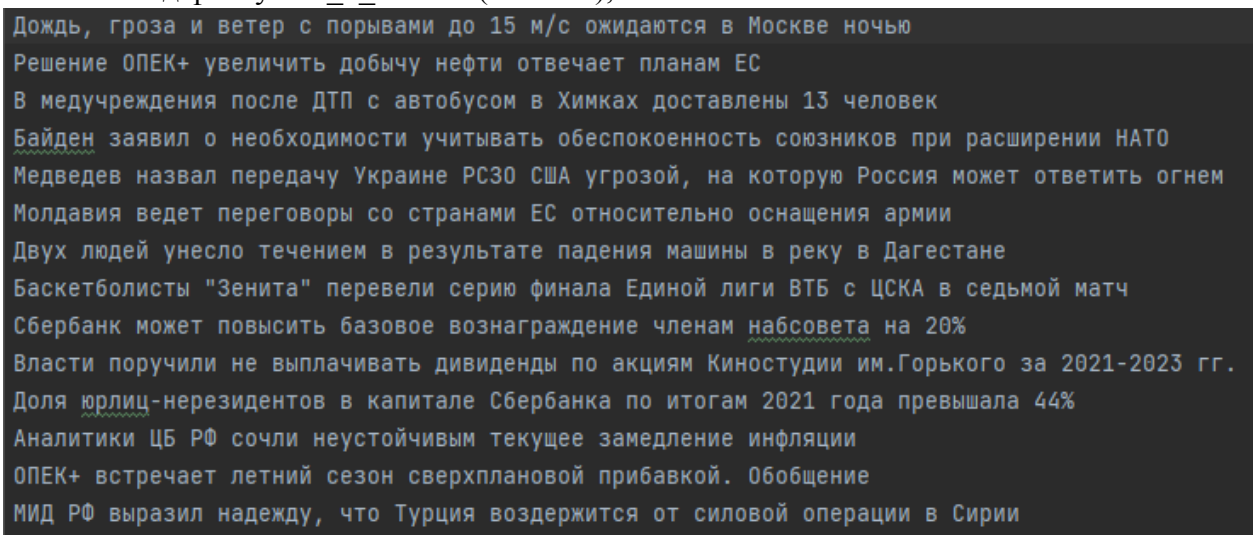
3.3. Результати роботи розробленого скріпта відповідно до завдання

1. Склад файлу text_1.txt(частина), мал.3



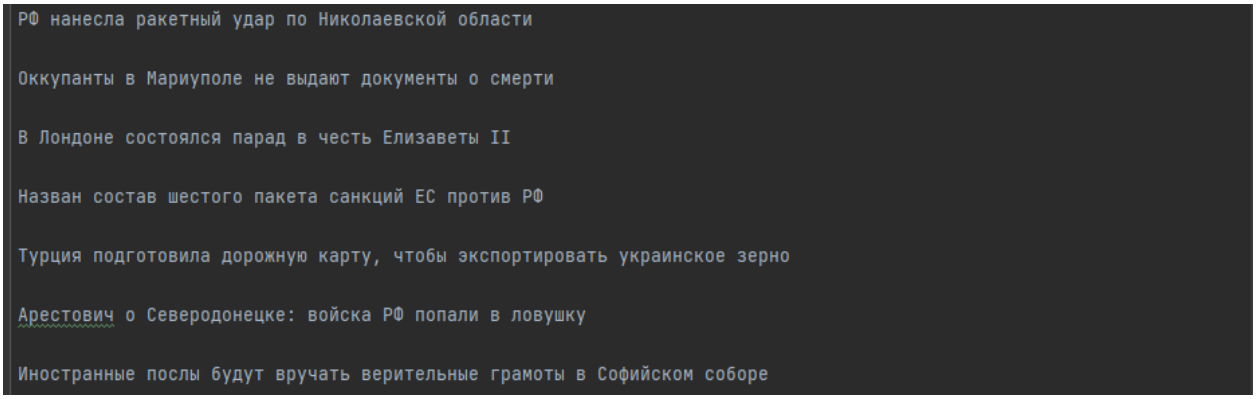
мал.3, частина файлу text_1.txt

2. Склад файлу text_1_clear.txt(частина), мал.4



мал.4, частина файлу text_1_clear.txt

3. Склад файлу text_2.txt(частина), мал.5



мал.5, частина файлу

Результат аналізу сайту interfax.ru за 2 та 3 червня

1. Wordcloud, мал.5

мал.5, workcloud часткового аналізу контенту

2. Частотний аналіз контенту сайту, мал.6

```
places 1 place,РФ - 186 times
places 2 place,ЕС - 83 times
places 3 place,рублей - 42 times
places 4 place,России - 41 times
places 5 place,ОПЕК - 40 times
places 6 place,США - 37 times
places 7 place,человек - 36 times
places 8 place,Украины - 35 times
places 9 place,Москве - 33 times
places 10 place,санкций - 31 times
```

мал.6, частотний аналіз контенту сайту

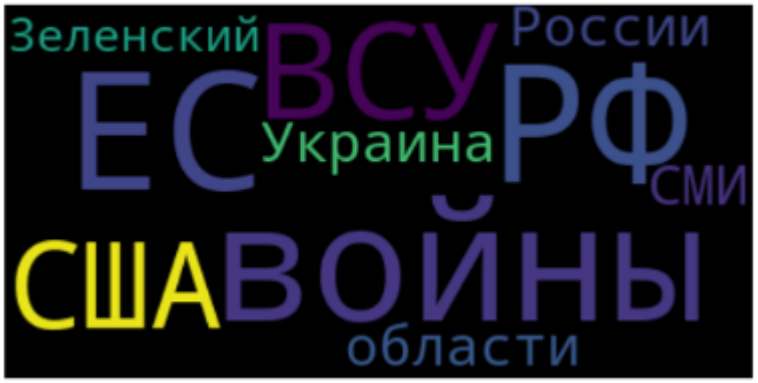
3. Докладний аналіз контенту сайту(частина), мал. 7-10

```
Оберіть інформаційне джерело:
1 - https://www.interfax.ru/
2 - https://korrespondent.net/
mode:1
Обрано інформаційне джерело: https://www.interfax.ru/
Докладний частотний аналіз інформаційного джерела: 1 : https://www.interfax.ru/
Кількість слів: 11921
Кількість унікальних слів: 1770
Усі використані слова:
```

доллар	13	квадрокоптер	1	удушен	5
долож	6	квартальн	5	ужесточен	4
домашн	1	квот	16	узбекиста	1
доначислен	5	киев	5	уйдут	1
донбасс	4	кинорынк	2	указа	5
дополнительн	7	киностуд	6	украин	90
доппроцент	4	киргиз	11	украинск	29
допуст	13	киргизск	2	укреплен	9
дорог	12	кисточк	5	уменьшат	5

Результат аналізу сайту korrespondent.net за 2 та 3 червня

1. Wordcloud, мал.11



мал.11, workcloud часткового аналізу контенту

2. Частотний аналіз контенту сайту, мал.12

```
places 1 place,РФ - 32 times
places 2 place,войны - 14 times
places 3 place,ЕС - 11 times
places 4 place,ВСУ - 10 times
places 5 place,США - 10 times
places 6 place,Украина - 9 times
places 7 place,области - 8 times
places 8 place,России - 8 times
places 9 place,СМИ - 7 times
places 10 place,Зеленский - 6 times
```

мал.6, частотний аналіз контенту сайту

3. Докладний аналіз контенту сайту(частина), мал. 13-16

```
Докладний частотний аналіз інформаційного джерела: 2 : https://korrespondent.net/
Кількість слів: 1606
Кількість унікальних слів: 948
Усі використані слова:
```

полномасштабн	1	украден	2	роспропаганд	1
получ	1	украин	56	росс	14
получа	1	украинк	1	российск	5
польш	3	украиноязычн	1	россия	1
поляков	1	украинск	7	россиян	2
помощ	2	украинц	6	рсзо	2
пообеща	1	украст	1	русск	1
				рф	32

3.4. Програмний код, що забезпечує отримання результату

```
import operator
import nltk
import pandas as pd
import re
import requests
from wordcloud import WordCloud
import matplotlib.pyplot as plt
from bs4 import BeautifulSoup
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords
from nltk.stem import SnowballStemmer
```

мал.17, імпорт необхідних бібліотек

```
# ---------- Парсер САЙТУ Interfax.ru для отримання html структури і вилучення з неї стрічки новин
def Parser_URL_interfax(url):
    response = requests.get(url)
    soup = BeautifulSoup(response.content, "html.parser")

    quotes_1 = soup.find_all('div', class_="an")
    output_file = open('C:/Users/serg/PycharmProjects/ds_lab51/text_1.txt', 'a')
    output_file_final = open('C:/Users/serg/PycharmProjects/ds_lab51/text_1_clear.txt', 'a')

    for quote in quotes_1:
        quote.encoding = 'cp1251'
        output_file.write(str(quote))

    with open(r'C:/Users/serg/PycharmProjects/ds_lab51/text_1.txt') as f:
        f = f.read()

    soup2 = BeautifulSoup(f, "html.parser")
    for link in soup2.find_all('h3'):
        link.encoding = 'cp1251'
        output_file_final.write(link.text + '\n')
    return
```

мал.18, метод для парсингу інформації в сайту interfax.ru

```
# ---------- Парсер САЙТУ Korrespondent для отримання html структури і вилучення з неї стрічкі новин  --------
def Parser_URL_korrespondent (url):
    response = requests.get(url)
    soup = BeautifulSoup(response.content, "html.parser")
    quotes_2 = soup.find_all('div', class_='article__title')
    output_file_2 = open('C:/Users/serg/PycharmProjects/ds_lab51/text_2.txt', 'a')

    for quote in quotes_2:
        quote.encoding = 'cp1251'
        output_file_2.write(quote.text)
    return
```

мал.19, метод для парсингу інформації в сайту korrespondent.net

```python
# -------------- Частотний text mining -------------------------
def text_mining_wordcloud(f):
    text = str(f.readlines())
    # -------------- Аналіз тексту на частоту слів БЕЗ СОЮЗНИХ СЛІВ (наприклад НА, ЗА і т.д.)
    # words = re.findall('[a-zA-Z]{2,}', text)  # regex для англійських слів
    words = re.findall('[а-яА-Я]{2,}', text)  # regex для  російських слів
    stats = {}

    stop_words = stopwords.words("russian")
    stop_words.append('млн')
    stop_words.append('млрд')
    stop_words.append('июня')
    stop_words.append('против')
    stop_words.append('заявил')
    stop_words.append('Украине')
    stop_words.append('Украины')
    stop_words_title = [None] * len(stop_words)
    for i in range(len(stop_words)):
        stop_words_title[i] = stop_words[i].title()

    for word in words:
        if word in stop_words:
            for i in range(words.count(word)):
                words.remove(word)
        if word in stop_words_title:
            for i in range(words.count(word)):
                words.remove(word)
```

```python
    for w in words:
        stats[w] = stats.get(w, 0) + 1
    #print(stats)
    w_ranks = sorted(stats.items(), key=lambda x: x[1], reverse=True)[0:10]
    _wrex = re.findall('[а-яА-Я]+', str(w_ranks))
    _drex = re.findall('[0-9]+', str(w_ranks))

    pl = [p for p in range(1, 11)]
    for j in range(len(_wrex)):
        places = '{} place,{} - {} times'.format(pl[j], _wrex[j], _drex[j])
        print('places', places)
    text_raw = " ".join(_wrex)

    wordcloud = WordCloud().generate(text_raw)
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.axis("off")
    plt.show()
    return
```

мал.20-21. метод для частотного аналізу контенту сайтів

```python
# ---------- Докладний частотний text mining --------------
def text_mining_ru(filename):
    #nltk.download('punkt')
    #nltk.download('stopwords')
    with open(filename) as file:
        text = file.read()

    tokens = word_tokenize(text)
    #видалення союзних слів
    stop_words = stopwords.words("russian")
    filtered_tokens = []
    for token in tokens:
        if token not in stop_words:
            filtered_tokens.append(token)
    #видалення символів та номерів
    regex_numbers = re.compile('^[0-9]{1,4}([,:-][0-9]{1,4})*(\.[0-9]+)?$')
    digits = ['.', '-', ',', '/', '' '', '!', '@', '" "',
              '#', '№', '$', ':', ';', '%', '^', '&', '?',
              '*', '(', ')', '_', '+', '=', '[', ']', '{', '}',
              '"', ' ', '<', '>', '|', '`', '~', ',']

    for token in filtered_tokens:
        if token in digits:
            filtered_tokens.remove(token)
    for token in filtered_tokens:
        if (regex_numbers.search(token) != None):
            filtered_tokens.remove(token)
```

```python
    words = []
    snowball = SnowballStemmer(language="russian")
    for i in filtered_tokens:
        word = snowball.stem(i)
        words.append(word)

    words.sort()
    words_dict = dict()

    for word in words:
        if word in words_dict:
            words_dict[word] = words_dict[word] + 1
        else:
            words_dict[word] = 1

    print("Кількість слів: %d" % len(words))
    print("Кількість унікальних слів: %d" % len(words_dict))
    print("Усі використані слова:")
    for word in words_dict:
        print(word.ljust(20), words_dict[word])

    return
```

мал.22-23, метод для докладного аналізу контенту сайтів

```
# -------------- Головні виклики парсера для отримання даних text mining --------------------
print('Оберіть інформаційне джерело:')
print('1 - https://www.interfax.ru/')
print('2 - https://korrespondent.net/')
mode = int(input('mode:'))
if (mode == 1):
    with open('C:/Users/serg/PycharmProjects/ds_lab51/text_1.txt', 'w'):
        pass
    with open('C:/Users/serg/PycharmProjects/ds_lab51/text_1_clear.txt', 'w'):
        pass
    print('Обрано інформаційне джерело: https://www.interfax.ru/')
    url = 'https://www.interfax.ru/'
    url1 = 'https://www.interfax.ru/news/2022/06/02/all'
    url2 = 'https://www.interfax.ru/news/2022/06/02/all/page_2'
    url3 = 'https://www.interfax.ru/news/2022/06/02/all/page_3'
    url4 = 'https://www.interfax.ru/news/2022/06/03/all'
    url5 = 'https://www.interfax.ru/news/2022/06/03/all/page_2'
    url6 = 'https://www.interfax.ru/news/2022/06/03/all/page_3'
```

```
    #print (' ------------------ Новини Інтерфакс за 02.06 ----------')
    Parser_URL_interfax(url1)
    Parser_URL_interfax(url2)
    Parser_URL_interfax(url3)
    #print(' ------------------ Новини Інтерфакс за 03.06 ----------')
    Parser_URL_interfax(url4)
    Parser_URL_interfax(url5)
    Parser_URL_interfax(url6)

    print('Докладний частотний аналіз інформаційного джерела:', mode, ':', url)
    filename = 'C:/Users/serg/PycharmProjects/ds_lab51/text_1_clear.txt'
    words_dict = text_mining_ru(filename)

    f = open('C:/Users/serg/PycharmProjects/ds_lab51/text_1_clear.txt', 'r')
    print('Домінуючий контент сайту:', mode, ':', url)
    text_mining_wordcloud(f)
```

мал.24-25, метод main для сайту interfax.ru

```python
if (mode == 2):
    with open('C:/Users/serg/PycharmProjects/ds_lab51/text_2.txt', 'w'):
        pass
    print('Обрано інформаційне джерело: https://korrespondent.net/')
    url = 'https://korrespondent.net/'
    with open('C:/Users/serg/PycharmProjects/ds_lab51/text_2.txt', 'w'):
        pass
    #------------------- Новини за 02.06 -------------------------
    url11 = 'https://korrespondent.net/all/2022/june/2/print/'
    url21 = 'https://korrespondent.net/all/2022/june/2/p2/print/'
    url31 = 'https://korrespondent.net/all/2022/june/2/p3/print/'
    url41 = 'https://korrespondent.net/all/2022/june/2/p4/print/'
    url51 = 'https://korrespondent.net/all/2022/june/2/p5/print/'
    #------------------- Новини за 03.06 -------------------------
    url61 = 'https://korrespondent.net/all/2022/june/3/print/'
    url71 = 'https://korrespondent.net/all/2022/june/3/p2/print/'
    url81 = 'https://korrespondent.net/all/2022/june/3/p3/print/'
    url91 = 'https://korrespondent.net/all/2022/june/3/p4/print/'
    url101 = 'https://korrespondent.net/all/2022/june/3/p5/print/'
```

```python
    Parser_URL_korrespondent(url11)
    Parser_URL_korrespondent(url21)
    Parser_URL_korrespondent(url31)
    Parser_URL_korrespondent(url41)
    Parser_URL_korrespondent(url51)
    Parser_URL_korrespondent(url61)
    Parser_URL_korrespondent(url71)
    Parser_URL_korrespondent(url81)
    Parser_URL_korrespondent(url91)
    Parser_URL_korrespondent(url101)

    # ------------- Частотний text mining аналіз даних від новосних сайтів ---------
    f = open('C:/Users/serg/PycharmProjects/ds_lab51/text_2.txt', 'r')
    print('Домінуючий контент сайту:', mode, ':', url)
    text_mining_wordcloud(f)
    print('Докладний частотний аналіз інформаційного джерела:', mode, ':', url)
    filename = 'C:/Users/serg/PycharmProjects/ds_lab51/text_2.txt'
    text_mining_ru(filename)
```

мал. 26-27, метод main для сайту korrespondent.net

3.5. Аналітичний звіт

Для аналізу було обрано контент сайтів korrespondent.net та interfax.ru.

«Кореспондент» (рос. «Корреспондент») — тижневий суспільно-політичний журнал в Україні. Видається російською мовою. Кореспондент є членом Української асоціації видавців періодичної преси (УАВПП). Входить в UMH group (головний офіс в м. Київ). Має інформаційно-новинний інтернет-ресурс Кореспондент.net.

«Інтерфакс» — незалежне інформаційне агентство, одне з трьох провідних агентств Росії, і найбільша в країнах СНД інформаційна група, що об'єднує понад три десятки компаній — мережу національних, регіональних та галузевих агенцій. «Інтерфакс» став одним із перших у СРСР незалежних новинних агентств.

Оскільки Інтерфакс має головний офіс в Москві та відноситься до російських медіа, було цікави дослідити інформацію цього сайту в порівнянні з українським видавництвом. Далі надано таблицю з порівнянням кількості згадувань найпопулярніших слів кожного видавництва. Дані надані за 2 та 3 червня.

Слово та його повідні	interfax.ru	korrespondent.net
РФ, Россия	347	52
Война	0	17
ЕС	83	11
ВСУ	0	10
США	37	10
Украина, украинский	119	72
СМИ	5	7
Зеленский	2	6
Рубль	53	0
ОПЕК	40	1
Москва	51	1
Санкции	80	7
Путин	28	4
Донбасс	4	4
Лукашенко	14	1
Пшеница	10	1
Кількість слів	11921	1606
Кількість унікальних слів	1770	948
% унікальності слів	14.8%	59%

IV. Висновки.

У цій лабораторній роботі було реалізовано скрипт для дослідження та узагальнення особливостей інтелектуального аналізу даних з використанням спеціалізованих пакетів мови програмування Python. Було досліджено та проаналізовано контент двох сайтів новим за період з 2 по 3 червня. В ході виконання лабораторної роботи було використано бібліотеки WordCloud, Matplotlib та NLTK мови Python. Середа розробки - PyCharm.

Виконав: студент Цуканова М.С.