МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики
Кафедра програмного забезпечення комп’ютерних систем

КУРСОВА РОБОТА

з дисципліни “Бази даних”

спеціальність 121 – Програмна інженерія

на тему: Система новинних повідомлень в Інтернет

Студент **групи** КП-93
Шевляков А. О. _____
(ПІБ) (підпис)

Викладач
к.т.н, доцент кафедри СПіСКС
Петрашенко А.В. _____
(підпис)

Захищено з оцінкою _____

Київ – 2020

Анотація

У процесі виконання даної курсової роботи було набуто практичні навички написання IAC широкого призначення, з використанням аналізу/фільтрації/пагінації даних та генерації даних у реляційну БД(за допомогою сторонніх засобів збору інформації в Інтернет). Результатом розробки є навички роботи з обраною СУБД та інструментами для управління даною СУБД.

Дана програма здатна аналізувати тренди новин, шукати схожі новини та має великі можливості для розширення функціоналу, завдяки незалежній побудові коду, його модульності(модель MVC)

Користувачу надаються можливості оновлювати та записувати дані у СУБД, а також взаємодіяти з нею за допомогою CRUD операцій.

Нижче будуть описані подробиці виконання та проектування курсової роботи. Початок роботи був запланований на 25 жовтня 2020, а завершення – 5 грудня. План виконаний.

Галузь застосування даного програмного забезпечення - прогноз актуальності новин та прорахунок можливої аудиторії.
Весь код та документацію можна знайти на github репозиторії:
https://github.com/OXOTNIKPROGER/DB

ЗМІСТ

Вступ

Сьогодні новини є дуже важливим інформаційним полем у житті кожної людини. Можливість знати та розуміти, що зараз популярно серед людей, а що ні, на що треба звернути увагу журналістам, а що краще пропустити – все це є дуже важливою задачею аналітиків, у пошуку дійсно цікавих сенсаційних новин. Дане програмне забезпечення базується на таких авторитетних виданнях як ТСН та ЮНИАН, на основі яких і робився аналіз новин в цілому.

На основі всього лише тем, переглядів та часу публікації, можна досягнути великої точності у аналізі та прогнозуванні новин.

У програмі яка базується на цій базі даних є можливість створювати такі сутності як: новини, теги та статистичні дані.

Аналіз інструментарію для виконання курсової роботи

У ролі середовища для розробки було використано PyCharm Community Edition 2020. Дане середовище має широкий функціонал для відладки та компіляції коду, має зручний графічний інтерфейс і є дуже популярною серед програмістів на Python.

Весь код був написаний мовою Python 3.8 , яка не є останньою версією цієї мови, але вона підтримує усі необхідні інструменти для роботи із СУБД та аналізу даних.

У ролі СУБД виступає PostgreSQL та його графічний інтерфейс у вигляді pgAdmin 4. Дана СУБД є досить потужною і популярною для великих обсягів даних. pgAdmin 4 надає зручний користувацький інтерфейс для роботи із даною СУБД.

PostgreSQL – об’єктно-реляційна система управління базою даних. Її перевагами є її популярність, що сприяє тому, що в мережі Інтернет присутня величезна кількість інформації, в якій можна знайти всі відповіді на питання.

Також, дана СУБД має потужні механізми для реплікації та транзакції даних. Ця СУБД є не тільки реляційною, а ще й об’єктно-орієнтовною, із засобами наслідування тощо.

Також використовувалися сторонні бібліотеки та модулі, завантаженні за допомогою утиліти pip.(Про бібліотеки буде розказано більше у відповідних розділах нижче).

Структура бази даних

База даних складається з таких таблиць:

Таблиця News – зберігає дані про новину, його news_id є зовнішнім ключем для інших зв’язаних таблиць.

Таблиця Tags – зберігає усі теги конкретної новини

Таблиця Statistics – зберігає статистичні дані про конкретну новину

Таблиця Content – зберігає посилання на конкретну новину

У БД наявні зв’язки між таблицями. One-to many зв’язок наявний між Tags та News. Це надає можливість мати декілька тегів одній новині.

Також зв’язок one-to-one присутній у таблицях Statistics-News Content-News. Це надає можливість розділити дані про одну новину по темах та по значенню.

Опис програмного забезпечення

Загальна структура програмного забезпечення

Програмне забезпечення побудоване по моделі MVC із рознесенням відповідних модулів.

Уся програма розділена на дві незалежних програми. Перша програма – це генератор новин, друга – аналізатор новин. Доступ до БД реалізований за технологією ORM за допомогою бібліотеки SQLAlchemy (встановлюється pip install sqlalchemy). Також для роботи використовувалися такі бібліотеки як matprolib(дана бібліотека дозволяє візуалізувати аналізовані дані, тобто будувати графіки, діаграми, гістограми тощо), pandas(дана бібліотека дозволяє аналізувати та змінювати DataFrame для виконання необхідних користувачу дій), BeatifulSoup(дана бібліотека дозволяє парсити веб-сторінки у різних форматах, за допомогою тегів)

Опис модулів програмного забезпечення

Генератор новин складається із одного файлу main.py, який має декілька функцій, які здійснюють генерацію даних у введений користувачем файл із вибраного користувачем ресурсу(реалізований парсинг із unian.ua та tsn.ua)

Функції є універсальними, тому можливе розширення функціоналу генератору, наприклад додання нових ресурсів для парсингу. Парсинг реалізований за допомогою бібліотеки requests(встановлюється pip install requests). Дана бібліотека дозволяє виконувати інтернет запити по даному URL та отримувати сторінку у форматі HTML для подальшої роботи з ним. Для знаходження необхідних URL із головної сторінки сайтів, була використана бібліотека BeautifulSoup(була описана раніше). Також для

пришвидчення виконання інтернет запитів, була вивчена багатопоточна архітектура на Python за допомогою бібліотеки concurrent.futures(функція виконання запитів була відправлена на потоки, і по завершенню виконання кожного із потоків, відбувався запис даних у файл директорії користувача).

Аналізатор новин складається з 4 директорій та файлу main.py(директорія webdata наявна тут для прикладу директорії обраною користувачем для генератора).

Оскільки код написаний з використанням моделі MVC то проект відповідно має директорії model, view, controller. Директорія storage та її файл tables.py має описані за технологією ORM SQLAlchemy таблиці бази даних.

Директорія model та її файл DBModel.py містить CRUD операції, описані за допомогою бібліотеки SQLAlchemy, з використанням таблиць із директорії storages. Також даний файл має функції для виконання будь-яких, необхідних користувачу запитів. Даний модуль написаний незалежно від інших, тому він поміщений як бібліотека на ресурс pypi.org(будь-хто може завантажити дану бібліотеку pip install dbcoursework). Оскільки дана бібліотека залежна від sqlalchemy, то на проект користувачу одразу буде завантажена також і дана бібліотека.

Директорія view містить один файл view.py , який містить статичний клас із функціями для реалізації GUI через консоль. Даний модуль не включає жоден файл та бібліотеку, тому він повністю відповідає моделі MVC(у даному файлі містяться функції виводу інформації , та підтвердження дії, які викликаються контроллером як реакція на виконання роботи)

Директорія controller складається із двох файлів controller.py та controller_functions.py. Основна логіка програми розписана у файлі controller_functions.py, натомість у controller.py забезпечений лише необхідний виклик функцій, як реакція на дії користувача. У файлі controller_functions.py було використано такі бібліотеки: concurrent.futures(для пришвидчення виконання коду, шляхом розділення на потоки), requests(для виконання необхідних інтернет-запитів), pandas(для аналізу DataFrame, отриманого із запитів до БД), matprolib(для візуалізації отриманих результатів), BeautifulSoup(для виконання парсингу отриманої сторінки). Також даний файл залежить від таблиць, описаних у директорії storages, та використвує модуль DBmodel. Функції файлу controller_functions.py буде описаний нижче.

Опис основних алгоритмів роботи

Як було сказано вище, у файлі controller_functions.py прописані основні алгоритми для роботи та аналізу даних. Для виконання функціоналу GUI

були створені відповідні функції, які реалізовують основну логіку запиту користувача.

Одним із опцій користувача є запис нових даних до СУБД. Для цього реалізований алгоритм парсингу файлів із заданої користувачем директорії у функції insert_generating_data яка приймає шлях до директорії. У функції відбувається парсинг, формування рядків таблиць, та запис цих рядків до СУБД.

Іншою, не менш важливою опцією користувача є оновлення даних вже існуючих новин, наприклад для побудови більш точної статистики. Для цього реалізована функція update_info, яка використовує багатопоточність, для пришвидчення виконання запитів. Завдяки цьому досягається близько 40 запитів у секунду. Але все ж таки , оскільки даних може бути дуже багато, дана функція рекомендується використовувати не так часто, оскільки динаміка переглядів старих новин не досить активна, щоб повпливати на результати досліджень(наприклад приблизно 500 новин оновлюються за 7 секунд).

Однією із основних опцій користувача є аналізи новин, тому для цього реалізовані дві функції аналізу даних analize_views та analize_views_per_hour для знаходження медіани переглядів по темам та найгарячіших новин відповідно. Дані функції використовують інструменти pandas та matprolib для побудови необхідного DataFrame та його візуалізації шляхом графіків.

Додатково реалізована опція аналізу статистики переглядів по авторам за допомогою функції analize_authors, яка використовує також pandas та matprolib для аналізу та візуалізації результатів.

Також користувачу наявна опція пошуку схожих новин. Для цього реалізована функція analize_similar(приймає номер новини(news_id)) яка працює переважно із таблицею Tags та бібліотекою pandas для знаходження схожих популярних новин.

Згідно до книги «Чистий код» Боба Мартіна, був зроблений рефакторинг коду.

Аналіз функціонування засобів реплікації

Реплікація була реалізована за допомгою другої інстанції власного серверу на одній Windows системі. Реплікація реалізована за принципом master-slave, для цього у PostgreSQL існує відповідна роль. Перверірка трансляції реплікації показане у запиті *select * from pg_stat_replication*

У результаті є можливість використання slave сервера, у разі виходу з ладу основного.

Аналіз функціонування засобів резервування/відновлення

Резервування бази даних реалізовано за допомогою стороннього програмного забезпечення SQLBackupAndFTP. Дана програма дозволяє створити з'єднання із різними СУБД , зокрема із PostgreSQL. Після підключення до необхідного серверу, вибравши необхідну для резервування БД можна вибрати опції та шлях зберігання резервування даних. Наявна можливість циклічного оновлення у обраний час. У даній роботі налаштоване резервування кожен день о 7 годині вечора, та видаленням старих резервних копій , які старіше одного місяця.

Аналіз результатів підвищення швидкодії запитів

Оцінюючи дані та запити, було прийнято рішення використовувати B-tree індекси на поля title , thema таблиці News. Судячи із графіку швидкодії(Дивитися у додатку, перші два та останні 2 стовбці) Це пришвидчило виконання запитів. Була спроба ввести такі самі індекси на news_id(середні графіки), але це дало зворотній ефект. Отже були залишені індекси на поля title та thema. Більше не було створено індексів, оскільки більшість дій виконувалася у коді за допомогою бібліотеки pandas.

Індекс B-tree формує бінарне дерево , що робить пошук по гілках за константний час. Він гарно підходить для розкиданих несортованих даних. Також він використовується у даних, які можна чітко порівняти знаками =, >, <

Опис результатів аналізу предметної галузі

Використовуючи розроблене програмне забезпечення було проаналізовано близько 1000 новин за останні декілька днів(з 9 грудня 2020 до 12 грудня 2020) та отримані такі результати:

Згідно із графіку на рисунку 6 , лідирує тема «Салон краси». Натомість, аналізуючи новини 4 дня тому, на першому місці стояли Фінанси. До того ж, напередодні Нового року, знизилася частка переглядів по темі «Коронавірус».

У першому графіку було використано аналіз переглядів та їх швидкість, для формування найбільш популярних у даний момент часу новин.

У другому графіку (рис 7) використовувалася медіана переглядів по групам(темам). Даний метод аналізу має альтернативу у вигляді середнього значення. Але на відміну від останнього, медіана гарно підходить для датасету, в якому наявні критичні значення(значення які сильно відрізняються від інших).

На рисунках 8 та 9 зображена статистика авторів. При аналізі використовувалася медіана переглядів та підрахунок кількості новин.

На рисунку 10-13 було знайдено схожі новини(лідери за переглядами у своїй темі) для новини із id 738.

Дізнатись id новини можна, попередньо скориставшись списком новин, який наявний у програмі. Аналізуючи початкову новину, видно, що теми перекликаються , отже пошук відбувся коректно.

Висновки

У даній курсовій роботі було розроблено програмне забезпечення для аналізу українських новин, з метою знаходження схожих, трендів тощо. У процесі роботи використовувалися такі сучасні та популярні бібліотеки для роботи із даними мовою Python як pandas та matprolib. Код роботи було логічно розподілено на модулі , згідно схеми MVC(model-view-controller). Для пришвидчення виконання коду було використано інструменти індексації PostgreSQL та використання потоків за допомогою бібліотеки мови Python concurrent.futures. Для належної генерації даних у СУБД було застосовано такі бібліотеки, як BeautifulSoup. Результатом роботи даного програмного забезпечення є графіки найпопулярніших новин та найпопулярніших тем.

Був зроблений рефакторинг коду.

Література

1. Data analysis using mean, median, mode https://www.pbslearningmedia.org/resource/mwnet-math-sp-mmmr/data-analysis-using-mean-median-mode-and-range/
2. Documentation postgreSQL https://www.postgresql.org/docs/
3. Python documentation https://www.python.org/doc/
4. 10 minutes to pandas https://pandas.pydata.org/pandas-docs/stable/user_guide/10min.html
5. How to publish Your own Python Package https://www.codementor.io/@ajayagrawal295/how-to-publish-your-own-python-package-12tbhi20tf
6. Анализ данных с использованием Python https://habr.com/ru/post/353050/
7. Репликация баз данных PostgreSQL по типу master-slave https://www.8host.com/blog/replikaciya-baz-dannyx-postgresql-po-tipu-master-slave/
8. Multithreading API Requests in Python https://creativedata.stream/multi-threading-api-requests-in-python/
9. What is the fastest way to send 10000 HTTP requests in Python https://stackoverflow.com/questions/43466412/what-is-the-fastest-way-to-send-10-000-http-requests-in-python-2-7
10. Web scraping and parsing HTML in Python with Beautiful Soup https://www.twilio.com/blog/web-scraping-and-parsing-html-in-python-with-beautiful-soup
11. “The clean Coder” Bob Martin
12. “Learn python 3 the hard way” Zed A. Shaw

Додатки

А. Графічні матеріали

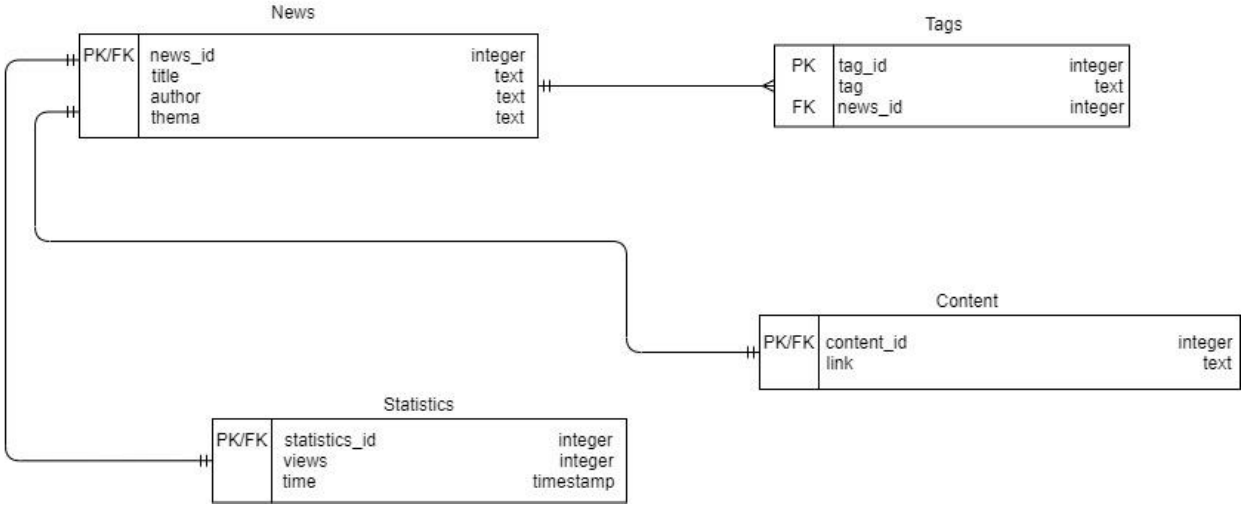

Рис.1 Структура БД

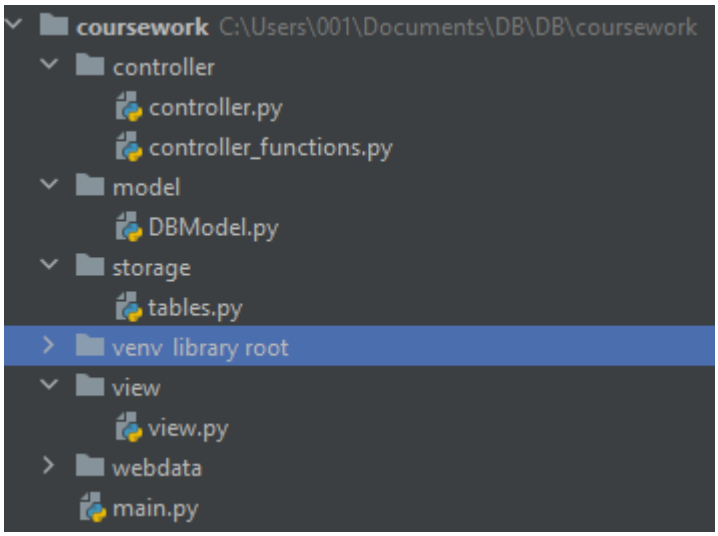

Рис.2 Структура проекту аналізатору новин

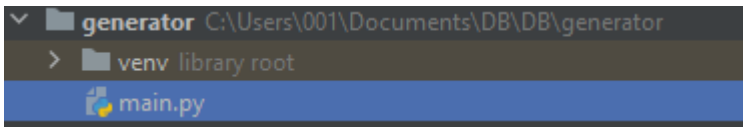

Рис.3 Структура проекту генератора новин

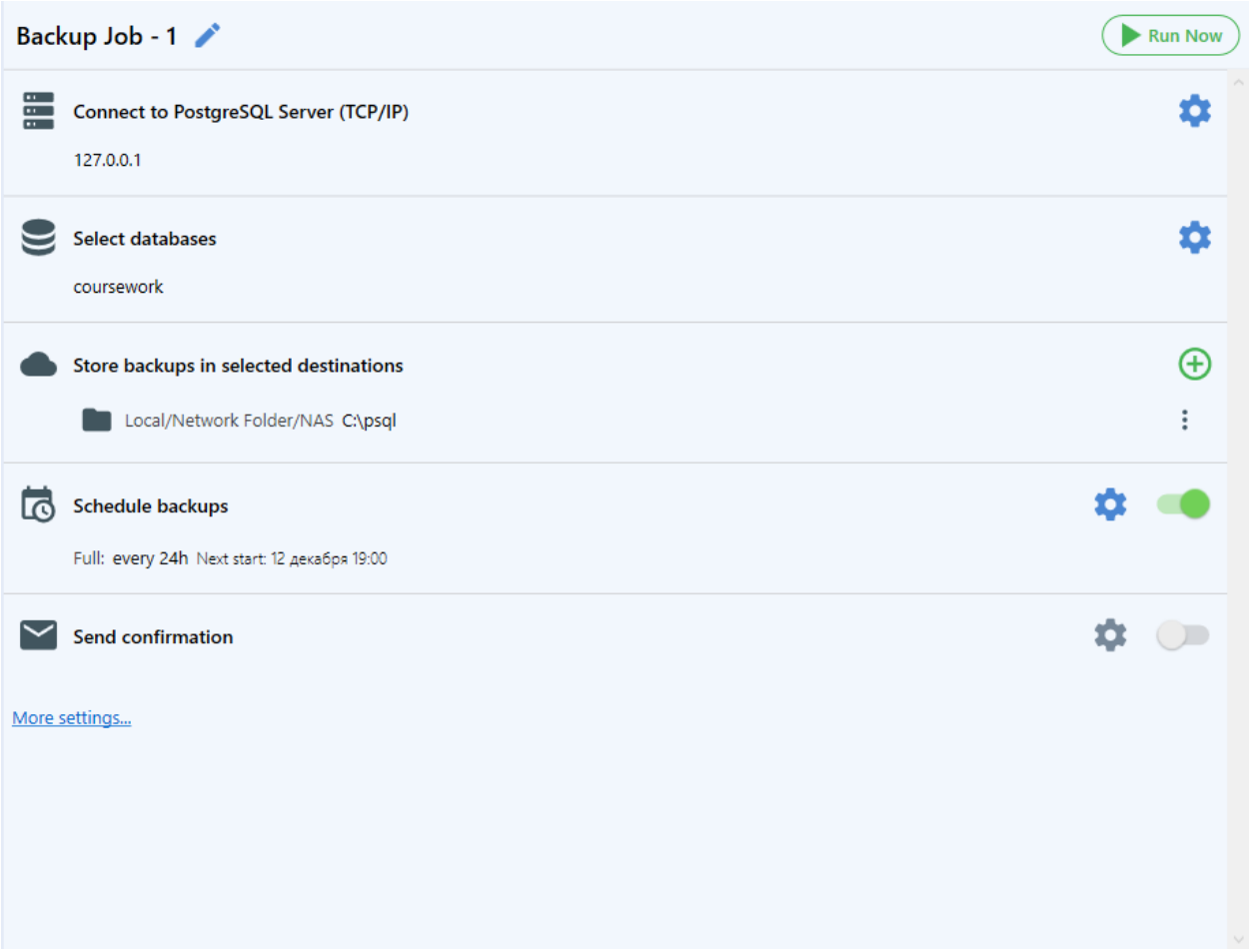

Рис.4 Скріншот SQLBackuoAndFTP із налаштуваннями резервування

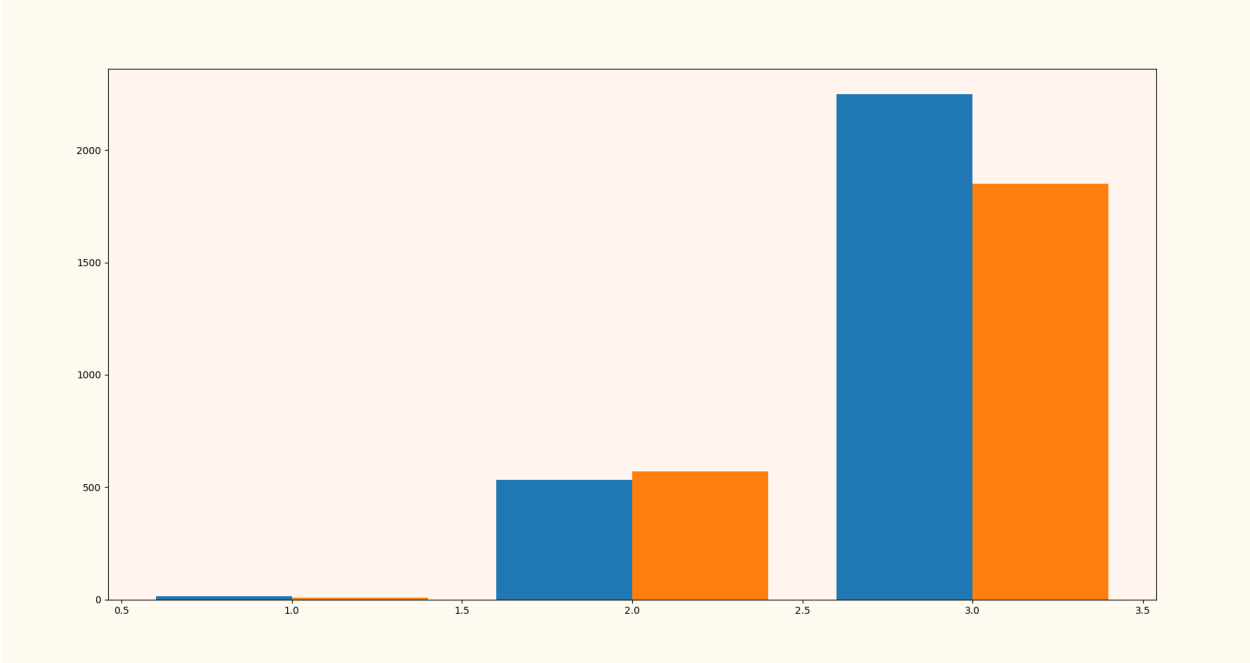

Рис.5 Результати швидкодії до та після індексів(сині – до, помаранчеві - після)

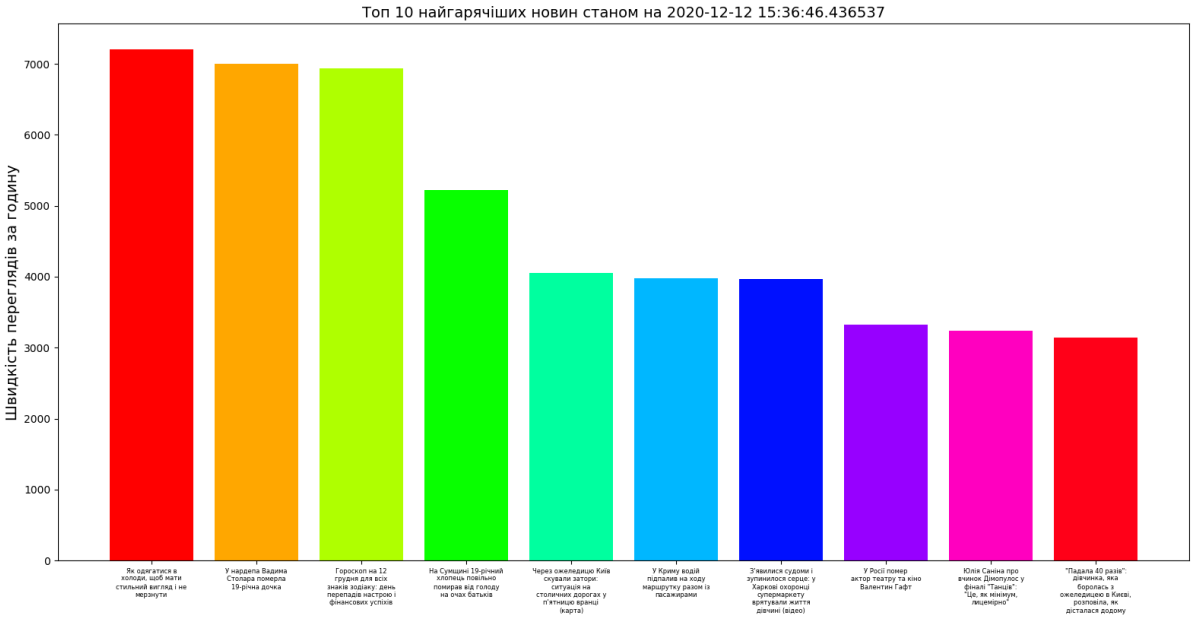

Рис.6 Топ 10 новин за деякий проміжок часу

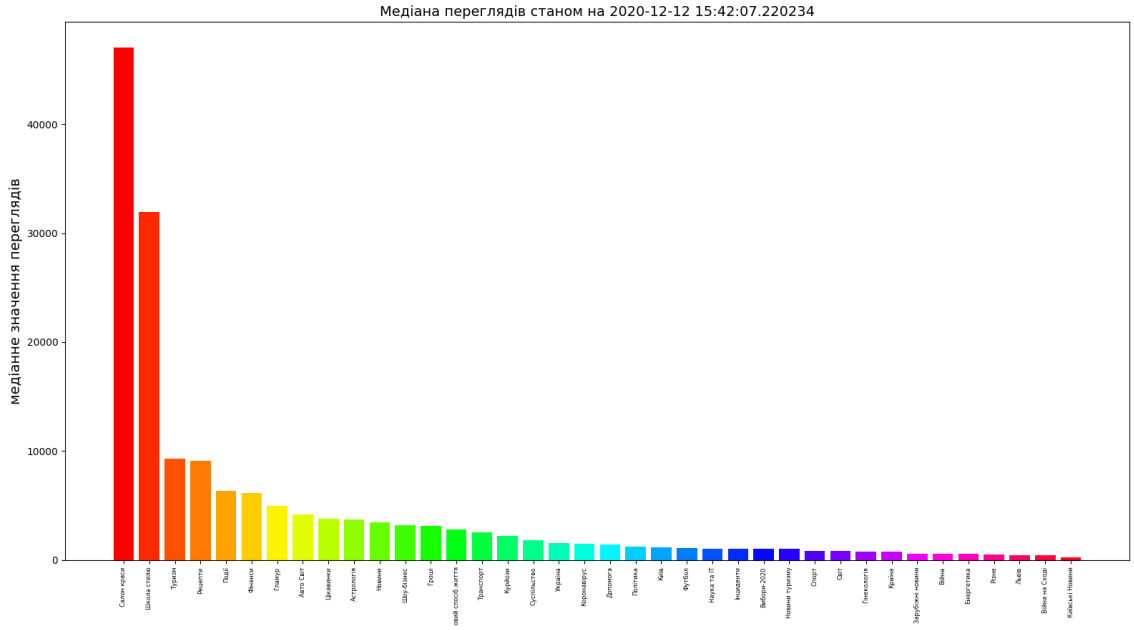

Рис.7 Найпопулярніші теми новин

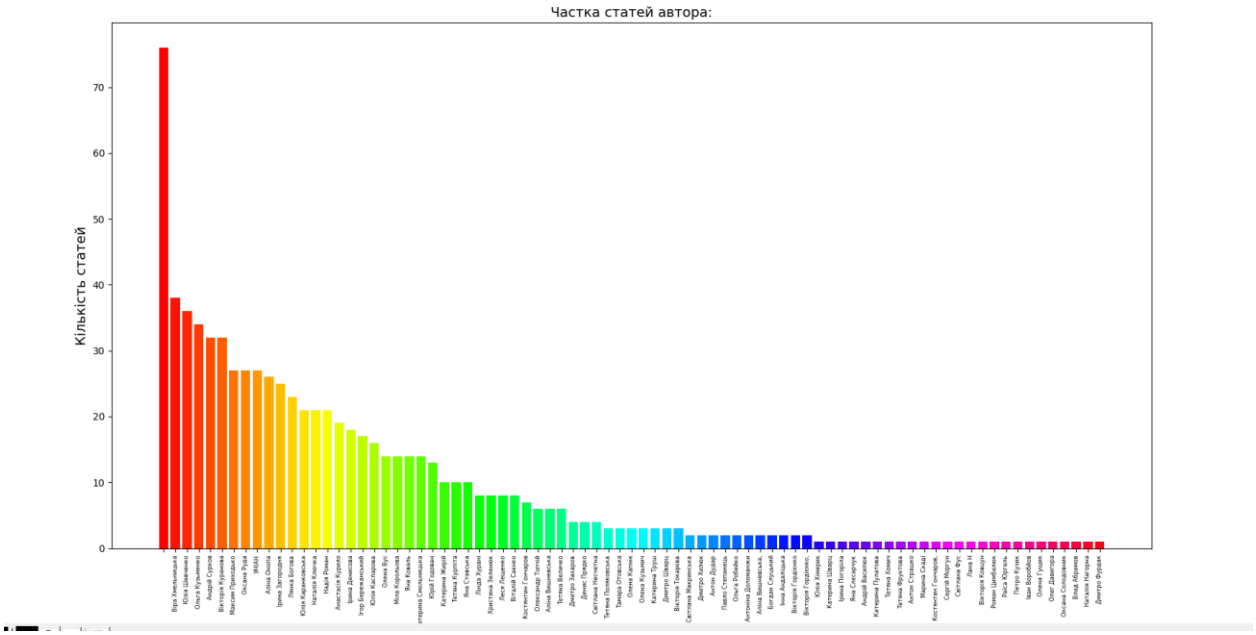

Рис.8 Частка статей автора

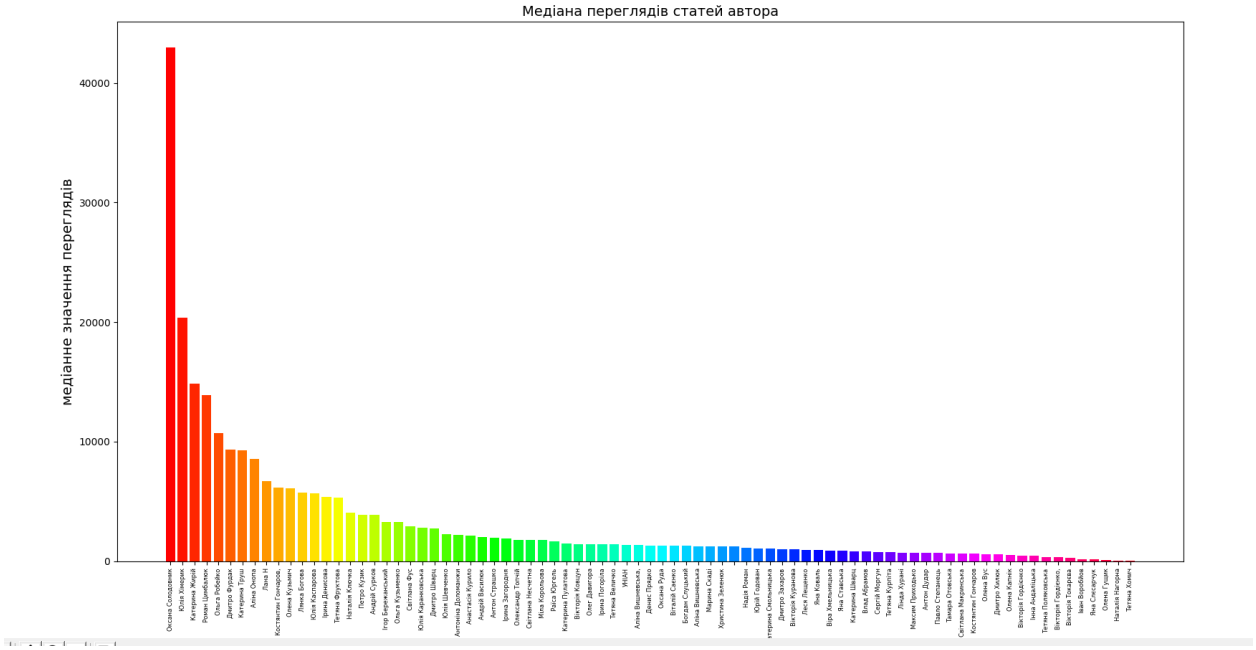

Рис.9 Медіана переглядів статей автора

```
Введіть номер функції, яку хочете виконати:
6
Виберіть до якої новини знайти схожі(введіть id новини, попередньо скориставшись опцією 7)
738
//////////////////////////////
id новини -> 224
Назва новини -> Кім Кардашян замилувала знімками своїх підрослих дітей: "Мої солоденькі"
Тема -> Гламур
Посилання на новину -> https://tsn.ua/glamur/kim-kardashyan-zamiluvala-znimkami-svoyih-pidroslih-ditey-movi-solodenki-1683520.html
//////////////////////////////
id новини -> 767
Назва новини -> Невпізнанна Ніколь Шерзінгер показала, який вигляд мала в дитинстві
Тема -> Гламур
Посилання на новину -> https://tsn.ua/glamur/nevpiznanna-nikol-sherzinger-pokazala-yakiy-viglyad-mala-v-ditinstvi-1686079.html
```

Рис. 10 Приклад виводу ланцюга схожих новин

```
////////////////////////////////
id новини -> 738
Назва новини -> Брітні Спірс сама себе висміяла за подібні фото
тема -> Гламур
////////////////////////////////
```

Рис.11 Новина до якої було знайдено схожі

```
Виберіть до якої новини знайти схожі(введіть id новини, попередньо скориставшись опцією 7)
419
////////////////////////////////
id новини -> 821
Назва новини -> Загиблий киянин врятував життя чотирьох людей: його нирку вперше в Україні пересадили дитині
тема -> Україна
Посилання на новину -> https://tsn.ua/ukrayina/zagibliy-kiyanin-vryatuvav-zhittya-chotiroh-lyudey-yogo-nirku-vpershe-v-ukrayini-peresadili-ditini-1686271.html
////////////////////////////////
id новини -> 861
Назва новини -> "Падала 40 разів": дівчинка, яка боролась з ожеледицею в Києві, розповіла, як дісталася додому
тема -> Україна
Посилання на новину -> https://tsn.ua/ukrayina/padala-40-raziv-divchinka-yaka-borolas-z-ozhelediceyu-v-kiyevi-rozpovila-yak-dobralas-do-domu-1686544.html
////////////////////////////////
id новини -> 592
Назва новини -> Затори в Києві: ситуація на столичних дорогах у четвер вранці (карта)
тема -> Транспорт
Посилання на новину -> https://www.unian.ua/economics/transport/zatori-v-kiyevi-10-grudnya-situaciya-na-dorogah-stolici-10-grudnya-karta-ob-jizdu-novini-kiyeva-11060603.html
////////////////////////////////
id новини -> 263
Назва новини -> Затори в Києві: ситуація на столичних дорогах у вівторок вранці (карта)
тема -> Транспорт
Посилання на новину -> https://www.unian.ua/economics/transport/zatori-v-kiyevi-8-grudnya-situaciya-na-dorogah-stolici-8-grudnya-karta-ob-jizdu-novini-kiyeva-11060603.html
////////////////////////////////
id новини -> 846
Назва новини -> Коронавірус вперше виявили у сніжного барса
тема -> Коронавірус
Посилання на новину -> https://coronavirus.tsn.ua/koronavirus-vpershe-viyavili-u-snizhnogo-barsa-1686499.html
////////////////////////////////
id новини -> 576
Назва новини -> У Києві з-під замерзлого озера дістали 40 ледь живих черепах: чому вони не впали у сплячку
тема -> Цікавинки
```

Рис.12 Ще один приклад виводу ланцюга схожих до новини 419

```
////////////////////////////////
id новини -> 419
Назва новини -> У Києві з озера дістали майже пів сотні черепах, які вмерзли у лід: відео
тема -> Київ
////////////////////////////////
```

Рис.13 новина 419

```
###########Аналізатор новин######################
1)Оновити базу даних новин
2)Записати згенеровані дані
3)Видалити старі новини
4)Знайти найгарячіші новини
5)Знайти найпопулярніші теми новин
6)Знайти схожі новини
7)Вивести усі новини
0) - для виходу
Введіть номер функції, яку хочете виконати:
2
Вкажіть шлях до папки із згенерованими html даними
webdata
Дані вписуються у базу даних
...
Дані записані, користуйтеся
```

Приклад консольного інтерфейсу під час виконання опцій

```
###########Аналізатор новин######################
1)Оновити базу даних новин
2)Записати згенеровані дані
3)Видалити старі новини
4)Знайти найгарячіші новини
5)Знайти найпопулярніші теми новин
6)Знайти схожі новини
7)Вивести усі новини
0) - для виходу
Введіть номер функції, яку хочете виконати:
12
Неправильний номер опції, введіть знову
```

Приклад реакції на некоректні дії користувача

Б. Фрагменти програмного коду

```
DECLARE
BEGIN
    DELETE FROM statistics WHERE statistics_id = OLD.news_id;
    DELETE FROM content WHERE content_id = OLD.news_id;
    DELETE FROM tags WHERE tags.news_id = OLD.news_id;
    RETURN OLD;
END;
```

Код тригеру Before delete який реагує на видалення новини та гарантує видалення усіх зв'язків

```python
def analize_views():
    df = create_News_arguments_table()
    selected_df = df[['views', 'thema']]
    selected_df = selected_df.groupby('thema')['views'].median().reset_index().sort_values(by=['views'], ascending=False)
    plt.title('Медіана переглядів станом на {}'.format(datetime.now()), fontsize=PLOT_LABEL_FONT_SIZE)
    plt.bar(selected_df['thema'], selected_df['views'], color=getColors(len(selected_df['thema'])))
    plt.ylabel('медіанне значення переглядів', fontsize=PLOT_LABEL_FONT_SIZE)
    plt.xticks(rotation=90, fontsize=PLOT_MEANING_FONT_SIZE)
    plt.show()
```

Приклад використання pandas та matprolib для побудови графіків

```python
def update_info():
    contents = dbModel.get_entities(Content)
    amount = 0
    for content in contents:
        amount = amount + 1
    with concurrent.futures.ThreadPoolExecutor(max_workers=amount) as executor:
        future_to_url = (executor.submit(load_url, content.link, content.content_id) for content in contents)
        for future in concurrent.futures.as_completed(future_to_url):
            data = future.result()
            soup = BeautifulSoup(data.text, 'lxml')
            views = get_views(soup)
            if views is None:
                continue
            stat = dbModel.get_entity(Statistics, data.id)
            amount = amount + 1
            stat.views = views
            dbModel.update_entity(stat)
```

Приклад використання багатопоточності бібліотеки concurrent.futures