БАКАЛАВРСЬКА РОБОТА

БР. ІП - 09.00.00.000 ПЗ

Група ІП-21-4

Дем’яник Вікторія

2025

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Дем'яник Вікторія Михайлівна

(прізвище, ім'я, по батькові)

УДК 004

(індекс)

БАКАЛАВРСЬКА РОБОТА

Реалізація техніки збору даних з веб-ресурсів соціального нетворкінгу

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121 - Інженерія програмного забезпечення

(шифр і назва спеціальності)

Робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Здобувач освітнього рівня Дем'яник В.М.

(підпис, ініціали та прізвище здобувача)

Науковий керівник Процюк Василь Романович, к.т.н., доцент

(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту
Завідувач кафедри

доц. Бандура В.В.

(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківськ – 2025

Івано-Франківський національний технічний університет нафти і газу

Інститут, факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Ступінь вищої освіти бакалавр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:
Зав. кафедрою **ІПЗ**
доц. В.В. Бандура
“ ____ ” _____2025 р.

ЗАВДАННЯ

НА ДИПЛОМНУ РОБОТУ БАКАЛАВРА СТУДЕНТОВІ

Дем’яник Вікторії Михайлівні

(прізвище, ім’я, по-батькові)

1. Тема проекту (роботи) “ Реалізація техніки збору даних з веб-ресурсів соціального нетворкінгу”

керівник проекту (роботи) Процюк В.Р., доцент

затверджені наказом закладу вищої освіти від **“** 28 ” квітня 2025 р. № 264/7

2. Строк подання студентом проекту (роботи) 10 червня 2025 р.

3. Вихідні дані до проекту (роботи) Результати і матеріали отримані під час проходження переддипломної практики

4. Зміст розрахунково - пояснювальної записки (перелік питань, які потрібно розробити)

1. Аналіз проблематики збору даних з веб-ресурсів соціального нетворкінгу
2. Дослідження структури даних веб-ресурсів соціального нетворкінгу на прикладі tiktok
3. Використання графів для техніки збору даних
4. Теоретично-формальна модель взаємодії користувачів в соціальній мережі
5. Програмна імплементація техніки збору даних з веб-ресурсів соціального нетворкінгу

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

1. Інтерфейс Octoparse (рис. 1.1)
2. Вигляд інструменту ParseHub (рис. 1.2)
3. Хмарна платформа скрапінгу Apify (рис. 1.3)
4. Сторінка TikTok ’For You’ (рис. 2.1)
5. Структура контенту в TikTok (рис. 2.2)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання **28 квітня 2025 р.**

Керівник _____
(підпис)

Завдання прийняв до виконання_____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту	Примітка
1	Аналіз проблематики збору даних з веб-ресурсів соціального нетворкінгу	04.05.2025	виконано
2	Дослідження структури даних веб-ресурсів соціального нетворкінгу на прикладі tiktok	12.05.2025	виконано
3	Використання графів для техніки збору даних	23.05.2025	виконано
4	Теоретично-формальна модель взаємодії користувачів в соціальній мережі	01.06.2025	виконано
5	Програмна імплементація техніки збору даних з веб-ресурсів соціального нетворкінгу	05.06.2025	виконано
6	Оформлення пояснювальної записки дипломної роботи завідувачем кафедри	10.06.2025	виконано

Студент – дипломник _____
(підпис)

Керівник роботи _____
(підпис)

АНОТАЦІЯ

Бакалаврська робота містить 77 сторінок, 37 рисунків, список використаних джерел із 40 найменуваннями.

Мета роботи: розробити, реалізувати та протестувати інструмент збору даних з соціальної мережі TikTok з подальшою обробкою даних у графовій структурі для досліджень соціальних взаємодій

Об'єкт дослідження: веб-ресурси соціального нетворкінгу як джерело великомасштабних даних

Предмет дослідження: технології, інструменти та методи збору і аналізу даних із соціальних мереж на прикладі TikTok

В першому розділі проаналізовано значення даних соціальних мереж у дослідженнях та розглянуто актуальні інструменти для їх збору

В другому розділі подано структуру веб-ресурсу TikTok, визначено технічні особливості збору даних та побудовано модель взаємодій користувачів у вигляді графа

В третьому розділі реалізовано програмний інструмент збору даних із TikTok, здійснено інтеграцію з графовою базою даних та проаналізовано ефективність отриманих результатів.

Висновок: реалізовано програмне рішення для збору, збереження та аналізу даних із використанням Scrapy, Neo4j та Docker. Представлено формальну модель взаємодії користувачів у соціальній мереж

КЛЮЧОВІ СЛОВА: ВЕБ-СКРАПІНГ; ЗБІР ДАНИХ; СОЦІАЛЬНІ МЕРЕЖІ; ДОБУВАННЯ ДАНИХ; АВТОМАТИЗОВАНИЙ ЗБІР ДАНИХ; АНАЛІЗ СОЦІАЛЬНИХ МЕРЕЖ; ВИДОБУВАННЯ ТЕКСТОВИХ ДАНИХ; МОНІТОРИНГ СОЦІАЛЬНИХ МЕРЕЖ.

ANNOTATION

The Bachelor's thesis comprises 77 pages, 37 figures, and a list of 40 references.

The objective of the work is to develop, implement, and test a data collection tool for the social network TikTok, followed by processing the data into a graph structure for social interaction research.

The object of study is social networking web resources as a source of large-scale data.

The subject of study is technologies, tools, and methods for collecting and analyzing data from social networks, using TikTok as an example.

In the first chapter, the significance of social network data in research is analyzed, and current tools for data collection are reviewed.

In the second chapter, the structure of the TikTok web resource is presented, technical features of data collection are defined, and a model of user interactions in the form of a graph is constructed.

In the third chapter, a software tool for collecting data from TikTok is implemented, integration with a graph database is carried out, and the effectiveness of the obtained results is analyzed.

Conclusion: A software solution for data collection, storage, and analysis using Scrapy, Neo4j, and Docker has been implemented. A formal model of user interactions within the social network is presented.

KEYWORDS: WEB SCRAPING; DATA COLLECTION; SOCIAL NETWORKS; DATA EXTRACTION; AUTOMATED DATA COLLECTION; SOCIAL NETWORK ANALYSIS; TEXT DATA EXTRACTION; SOCIAL NETWORK MONITORING

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

API - Application Programming Interface - інтерфейс прикладного програмування

CI - Continuous Integration - безперервна інтеграція

CD - Continuous Delivery - безперервна доставка

DOM - Document Object Model - об'єктна модель документа

OAuth - Open Authorization - протокол авторизації

RPA - Robotic Process Automation - роботизована автоматизація процесів

URL - Uniform Resource Locator - уніфікований локатор ресурсу

XHR - XMLHttpRequest - об’єкт асинхронного запиту в JavaScript

ВСТУП

Соціальні медіа є джерелом цінної інформації для дослідження людської поведінки, як індивідуальної, так і колективної. Дані, що генеруються в соціальних медіа, надають унікальну можливість вивчати соціальні взаємодії, доступ до яких був би значно ускладнений за їх відсутності. Проте, доступ до цих даних пов'язаний із певними викликами. По-перше, він вимагає наявності відповідних технічних знань для первинного збору даних. По-друге, необхідним є врахування правових та етичних аспектів щодо збору та аналізу даних. З одного боку, ці дані являють собою цифрові сліди індивідів, які можуть не усвідомлювати потенційного аналізу їхніх дій. З іншого боку, діяльність користувачів у публічній сфері соціальних медіа, спрямована на поширення контенту широкій аудиторії, ставить питання про публічний характер цих даних. Надання постачальникам платформ монополії на інсайти, отримані з даних, є дискусійним питанням.

У сучасну цифрову епоху соціальні мережі стали не лише засобом комунікації, а й джерелом великого масиву даних, що відображає реальні соціальні процеси, поведінкові патерни користувачів та динаміку інформаційних потоків. Зростання популярності таких платформ, як TikTok, створює нові можливості для міждисциплінарних досліджень, що поєднують соціальні науки, інформатику та аналіз даних. Разом із тим постає низка викликів, пов’язаних із збором, обробкою та інтерпретацією даних у правовому, технічному та етичному контекстах.

Актуальність роботи

З огляду на зростання впливу соціальних медіа на суспільні процеси, актуальним є створення ефективних технічних засобів збору, обробки та аналізу даних з веб-ресурсів соціального нетворкінгу.

Ця робота спрямована на аналіз методологічних та практичних аспектів автоматизованого збору даних із соціальних мереж. У межах дослідження

було здійснено огляд сучасних інструментів веб-скрапінгу, досліджено структуру даних соціальної платформи TikTok, побудовано формальні моделі взаємодії користувачів, а також реалізовано програмну систему збору та обробки інформації з використанням сучасних технологій.

Мета роботи - розробити, реалізувати та протестувати інструмент збору даних з соціальної мережі TikTok з подальшою обробкою даних у графовій структурі для досліджень соціальних взаємодій.

Завдання дослідження

1. Проаналізувати можливості соціальних медіа як джерела даних для досліджень.

2. Дослідити програмні інструменти для веб-скрапінгу.

3. Вивчити технічні аспекти структури веб-ресурсу TikTok.

4. Побудувати формальну модель користувацької взаємодії.

5. Реалізувати програмний інструмент збору та обробки даних.

Об’єкт дослідження - веб-ресурси соціального нетворкінгу як джерело великомасштабних даних.

Предмет дослідження - технології, інструменти та методи збору і аналізу даних із соціальних мереж на прикладі TikTok.

Методи дослідження:

- аналітичний огляд літератури та інструментів;
- технічне моделювання веб-структур;
- графовий аналіз взаємодій;
- програмна реалізація скрапінгових механізмів;
- використання графових БД (Neo4j);
- емпіричне тестування продуктивності.

Наукова новизна

Запропоновано формалізовану модель взаємодій користувачів у TikTok та розроблено комплексний інструмент збору й обробки даних на основі сучасного стеку технологій з використанням графової структури даних.

Практичне застосування

Розроблену систему можна застосовувати для збору та аналізу соціальних даних у дослідженнях, пов’язаних із соціологією, маркетингом, журналістикою даних, безпекою та інформаційним впливом.

Результати роботи мають значення як для науковців, що досліджують цифрові комунікації, так і для розробників, які створюють інструменти аналітики соціальних мереж. Робота закладає основу для подальших досліджень у галузі соціального нетворкінгу, зокрема в напрямку прогнозування соціальних трендів, аналізу впливу та інформаційної безпеки.

Бакалаврська робота містить 77 сторінок, 37 рисунків, 3 розділи список використаних джерел із 40 найменуванням.

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ ЗБОРУ ДАНИХ З ВЕБ-РЕСУРСІВ СОЦІАЛЬНОГО НЕТВОРКІНГУ

1.1. Використання даних соціальних медіа для досліджень у галузі соціальних наук

Поява платформ соціальних медіа трансформувала ландшафт емпіричних досліджень у галузі соціальних наук, надаючи безпрецедентні можливості для макроскопічного аналізу складних соціальних явищ. Великі обсяги даних, що генеруються користувачами, містять потенціал для виявлення закономірностей, тенденцій та динамік, які раніше були недоступні або вимагали значних ресурсів для збору традиційними методами. Однак, реалізація цього потенціалу стикається зі значними технічними та методологічними викликами.

Ключовою перешкодою є складність технічного доступу до цих даних. Хоча деякі платформи соціальних медіа пропонують програмні інтерфейси (API), що дозволяють дослідникам отримувати доступ до даних у структурованому форматі, політика доступу та функціональність цих API можуть бути обмеженими та непостійними. Зокрема, доступ до даних з платформи TikTok, яка стрімко набуває значення як впливовий простір для соціальної взаємодії, формування і поширення контенту, наразі не є тривіальним процесом для наукової спільноти. Незважаючи на експоненційне зростання популярності TikTok та його дедалі більшу роль у публічному та науковому дискурсі, він залишається відносно недостатньо використаним ресурсом для емпіричних досліджень у соціальних науках порівняно з іншими платформами.

Ця робота спрямована на вирішення зазначеної прогалини шляхом представлення та валідації інструментарію, що дозволяє дослідникам здійснювати збір даних з платформи TikTok. Розроблений інструмент

інтегрується з теоретичною моделлю, що розроблена з урахуванням усталених методів і теоретичних концепцій, які застосовуються в соціальних науках. Водночас, модель інкорпорує строгість та формалізований підхід, притаманні комп'ютерним наукам. Такий міждисциплінарний підхід дозволяє не лише подолати технічні бар'єри доступу до даних, але й забезпечує методологічну основу для їх змістовного аналізу в контексті соціальних теорій.

Шляхом сприяння синергії між дослідницькими парадигмами соціальних та комп'ютерних наук, ця робота має на меті стимулювати міждисциплінарну співпрацю. Надання соціологам та іншим фахівцям у галузі соціальних наук надійного та ефективного інструменту для збору, дослідження та аналізу даних з TikTok відкриває нові горизонти для емпіричної перевірки гіпотез, розробки нових теоретичних моделей та отримання глибинних інсайтів щодо сучасних соціальних процесів, що відбуваються на цій динамічній платформі. Це сприятиме більш повному і всебічному розумінню впливу соціальних медіа на індивідуальну та колективну поведінку, формування громадської думки та інші значущі соціальні явища.

1.2. Природа комп'ютерних наук та її застосування у дослідженні соціальних явищ в цифрову епоху

Відомий вислів Алана Перліса, першого лауреата премії Тюрінга, що "Комп'ютерні науки - це не наука про комп'ютери, так само як астрономія - це не наука про телескопи", влучно ілюструє глибоку сутність та амбівалентність термінології в цій галузі. Ця сентенція наголошує на тому, що об'єктом дослідження в комп'ютерних науках є не стільки самі обчислювальні машини, скільки фундаментальні принципи обчислень, алгоритмічні процеси та інформація як така. Комп'ютер виступає радше як

інструмент, що дозволяє формалізувати, моделювати та вирішувати широкий спектр проблем. Ключовим аспектом, на якому також акцентують увагу провідні фахівці [1 - 3], є вивчення та розуміння процесів передачі інформації, яка є невід'ємною складовою будь-якого обчислювального процесу та комунікаційної взаємодії.

Однією з наукових галузей, де дослідження передачі інформації має критичне значення, є соціальні науки. Соціальні науки системно вивчають соціальну поведінку та функціонування різноманітних видів, включаючи Homo sapiens, з метою пізнання закономірностей організації суспільства та його динаміки [4]. В контексті вивчення людського виду, особливого значення набуває його унікальна здатність до створення та використання технологій. Технології використовуються не лише для оптимізації індивідуального існування, але й, що є вкрай важливим, для опосередкування та посилення комунікаційних процесів між індивідами та групами. Таким чином, вивчення інформаційного обміну в людських спільнотах та його впливу на їх структуру та організацію є фундаментальним завданням соціальних наук.

В епоху домінування цифрових технологій та мережевих комунікацій, зокрема соціальних медіа, ландшафт людської взаємодії зазнав кардинальних змін. Соціальні медіа платформи трансформували способи створення, поширення та споживання інформації, що зумовило зростання актуальності досліджень людського спілкування в цьому цифровому середовищі. Ця тенденція логічно призвела до посилення співпраці між соціальними науками та комп'ютерними науками, формуючи нові міждисциплінарні напрями досліджень [5 - 7].

Незважаючи на стрімке зростання популярності та значний соціокультурний вплив платформи TikTok, особливо серед молодіжних когорт, вона досі залишається відносно менш дослідженою академічною спільнотою порівняно з іншими соціальними медіа. Складність сучасних

технологічних систем, що лежать в основі таких платформ, створює значний бар'єр для входу для дослідників-соціологів, які прагнуть глибоко зрозуміти механізми функціонування соціальних медіа та їх вплив на суспільні процеси.

Враховуючи особистий досвід, що поєднує академічну освіту в галузі соціальних наук зі здобуттям кваліфікацій з комп'ютерних наук, автор цієї роботи мав змогу безпосередньо оцінити потенційні переваги інтеграції обчислювальних методів для вирішення дослідницьких завдань у соціальних науках. Однак, суттєві відмінності у методологічних підходах та специфічна термінологія кожної з дисциплін можуть ускладнювати ефективну міждисциплінарну взаємодію.

Метою цієї роботи є сприяння подоланню зазначених бар'єрів та стимулювання співпраці між соціальними та комп'ютерними науками. Це буде досягнуто шляхом розробки та представлення інструментарію, спеціально призначеного для збору емпіричних даних з платформи TikTok, актуальність якої невпинно зростає. Паралельно з розробкою інструменту, буде запропонована чітка та формалізована теоретична рамка, що ґрунтується на синтезі концепцій обох дисциплін і забезпечує методологічну основу для систематизації та змістовної інтерпретації зібраних даних. Таким чином, робота спрямована на забезпечення дослідників необхідними засобами та концептуальною базою для проведення поглибленого аналізу соціальних явищ, опосередкованих платформою TikTok.

1.3. Проблематика та цілі роботи

Сучасний соціальний світ дедалі глибше інтегрований у технологічний контекст, що зумовлює необхідність його вивчення через призму цієї взаємодії. Для дослідження цього складного взаємозв'язку виникли та розвиваються міждисциплінарні наукові напрями, в рамках яких відбувається

конвергенція методів та підходів соціальних та технічних наук. У межах цього широкого контексту, дане дослідження зосереджується на проблемі недостатньої дослідницької уваги до окремих платформ соціальних медіа.

Значна частина емпіричних досліджень у галузі соціальних наук, що використовують дані соціальних медіа, історично зосереджувалася на платформах, які надають відносно легкий доступ до даних через програмні інтерфейси застосунків (API), таких як Twitter (нині X) або Reddit [3]. Однак, такий фокус може призводити до потенційного методологічного ухилу та формування неповного або викривленого розуміння динаміки соціальних медіа в цілому, оскільки висновки, отримані на матеріалах однієї платформи, часто екстраполюються на все онлайн-комунікаційне середовище.

Окрім специфічної проблеми доступу до даних, дане дослідження також спрямоване на вирішення ширшого завдання — стимулювання колаборації між соціальними науками та комп'ютерними науками у вивченні феномену соціальних медіа. Існують фундаментальні відмінності у філософії науки, підходах до побудови теорій та емпіричних методів, які традиційно використовуються в соціальних науках порівняно з дисциплінами, що мають сильніші зв'язки з природничими та точними науками [10]. Це може створювати бар'єри для ефективної комунікації та спільної роботи. Наприклад, галузь складних систем, що вивчає поведінку багатокомпонентних інтерактивних систем з емерджентними властивостями, часто розглядає соціальні системи як один із прикладів таких систем. Проте, ця галузь історично формувалася під значним впливом математики, фізики та комп'ютерних наук, що призвело до певної мовної та концептуальної роз'єднаності із соціальними науками. Таким чином, міждисциплінарні дослідження, що здатні нівелювати цей розрив між природничо-науковими та соціально-науковими парадигмами, подібні до представленого в цій роботі, мають значний потенціал для сприяння взаємному обміну знаннями та поглиблення співпраці між двома областями наукового знання.

Виходячи з окресленої проблематики, головні цілі даної роботи сфокусовані на подоланні обмежень у доступі до даних платформи TikTok та сприянні розвитку міждисциплінарної співпраці між соціальними науками та комп'ютерними науками.

Перша основна ціль полягає у розробці та реалізації програмного інструментарію (веб-скрапера), який забезпечить дослідникам можливість отримувати дані з платформи соціальних медіа TikTok. Цей крок є критично важливим для розширення емпіричної бази досліджень TikTok, враховуючи існуючі складнощі з доступом до даних. При цьому особлива увага приділятиметься не лише технічній реалізації процесу збору даних, але й дотриманню відповідних етичних норм та правових аспектів забезпечення прозорості та легітимності доступу до інформації.

Друга ключова ціль спрямована на проектування як самого інструментарію, так і загальної структури роботи таким чином, щоб максимально сприяти міждисциплінарній колаборації між соціальними науками та комп'ютерними науками. Досягнення цієї цілі передбачає реалізацію двох взаємопов'язаних підцілей:

1. Забезпечення високого рівня ергономічності та доступності розробленого інструментарію для користувача, що дозволить мінімізувати вимоги до специфічних технічних знань, необхідних для отримання даних.

2. Інтеграція функціоналу інструментарію у чітку теоретичну модель, яка буде узгоджуватися з поширеними методами та теоретичними конструкціями, що застосовуються в соціальних науках, одночасно зберігаючи строгість та формалізований підхід, характерний для комп'ютерних наук.

Таким чином, підсумовуючи, основні цілі даної роботи включають:

1. Розробку теоретичної моделі платформи TikTok як соціального середовища, яка буде концептуально доступною та корисною як для дослідників у галузі соціальних наук, так і для фахівців з комп'ютерних наук.

2. Створення та реалізацію інструменту веб-скрапінгу, функціональність якого буде адаптована до розробленої теоретичної моделі.

3. Забезпечення максимальної доступності та зручності використання розробленого інструментарію для дослідників з обох наукових напрямів, сприяючи таким чином їхній ефективній співпраці.

1.4. Значущість отримання даних соціального нетворкінгу

Цей підрозділ має на меті продемонструвати значущість даних TikTok для комп'ютерних наук та соціальних наук, надавши короткий огляд кожної дисципліни та підкресливши їхній зв'язок із соціальними медіа. Під час обговорення соціальних наук я також надам коротке пояснення поширених соціальних теорій та загальноприйнятих методологій, щоб гарантувати, що визначення та, відповідно, зібрані дані узгоджуються з ними.

Соціальні науки охоплюють різноманітні дисципліни, включаючи психологію, науки про комунікацію, антропологію та соціологію, які вивчають соціальну поведінку людини [11]. Однак визначення терміну "соціальний" є складним, оскільки він може мати різні значення в соціальних науках та в повсякденному вжитку. Наприклад, Кембриджський словник визначає соціальний як пов'язаний з "способом життя людей або з рангом особи в суспільстві" [12]. Однак у повсякденній мові цей термін також може описувати поведінку певних тварин, а не лише людей.

На сьогоднішній день соціальні науки характеризуються різноманітністю методологій та перспектив. Незважаючи на це різноманіття, існує постійна проблема щодо спілкування та співпраці між дослідниками в галузі. У [15] автори стверджують, що інтеграція соціальних досліджень у міждисциплінарну співпрацю може призвести до глибшого розуміння колективної людської поведінки за допомогою аналітичних методів. Вони

роблять висновок, що нові інструменти можуть бути важливим фактором у досягненні цієї мети.

У підсумку, ідея про те, що людська поведінка надто складна для вивчення за допомогою статистичних методів, є помилковою. Хоча людська поведінка є складною, це не означає, що не можна знайти та вивчити закономірності. Дослідження [12] є прикладом того, як соціальні дослідження можуть виявляти закономірності в людській поведінці та робити значущі внески в наше розуміння суспільства. Поступальний розвиток у зборі та аналізі даних відкрив нові можливості для соціальних досліджень і дозволив дослідникам краще розуміти людську поведінку та сили, які її формують. Методенстрейт (Methodenstreit) слід розглядати як триваючий діалог про найкращі методи вивчення людської поведінки, а не як бар'єр для наукових досліджень. Методенстрейт – це фундаментальна методологічна суперечка в соціальних науках

"Методенстрейт" допомагає краще зрозуміти триваючу дискусію в соціальних науках щодо визначення терміну "теорія". Деякі дослідники розглядають теорію так само, як її розуміють у природничих науках, як сукупність чітких і несуперечливих тверджень без протиріч. З іншого боку, інші розуміють теорію як парадигму мислення. Розмежування між цими двома формами теорій часто розмите, що призводить до широкого спектру теорій з різними кутами зору, застосуваннями та методологіями.

Ми зосередимося на трьох теоріях, які включають підхід природничих наук і які є актуальними для вивчення соціальних медіа та TikTok. Теорії, які я вирішив представити, це теорія соціальних мереж, структурно-функціональна теорія та теорія соціальної дії.

Стверджується, що суспільство схоже на організм, з різними частинами, які працюють разом для підтримки соціальної стабільності та балансу. Згідно з [11], суспільство можна розглядати як складну систему взаємозв'язаних частин, які працюють разом для підтримки стабільності та

порядку. Теорія, яка виникла з його висновків та результатів, відома як структурно-функціональна теорія [14 - 17]. Структурні функціоналісти вважають, що суспільство складається з кількох різних структур, таких як економіка, сім'я та уряд, кожна з яких має свої власні функції. Ці структури знаходяться в стані рівноваги, підтримуючи стабільність шляхом балансування вимог та потреб різних частин суспільства.

1.5. Аналіз програмних інструментів для збору даних з соціальних мереж

1.5.1. Інструмент веб-скрапінгу Octoparse

Octoparse – це програмний інструмент для веб-скрапінгу, розроблений для вилучення структурованих даних з веб-сайтів. Він позиціонується як рішення, що не вимагає навичок програмування (no-code), що робить його доступним для широкого кола користувачів, включаючи дослідників, маркетологів, бізнес-аналітиків та інших фахівців, які потребують збору великих обсягів веб-даних.

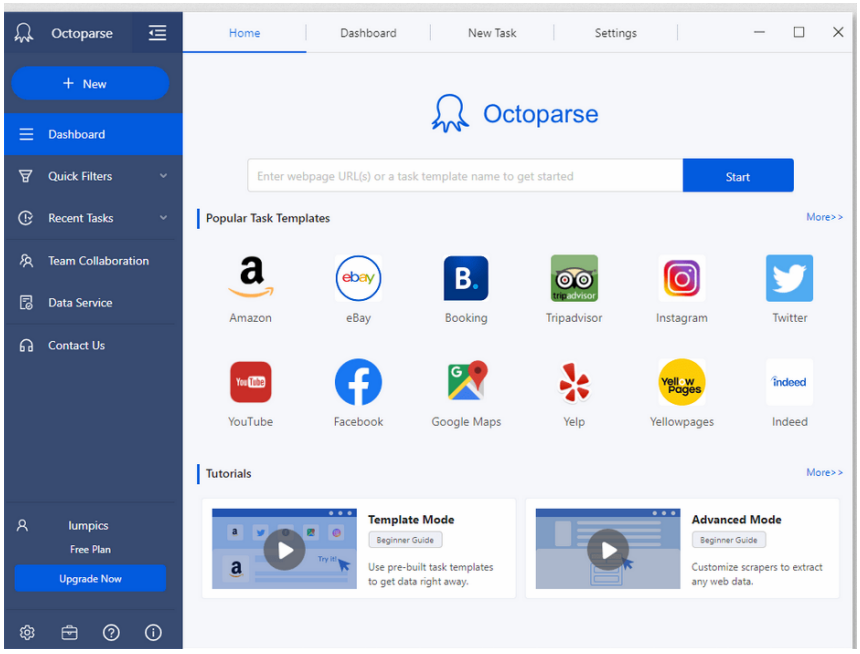


Рисунок 1.1 – Інтерфейс Octoparse

Ключовою перевагою Octoparse є інтуїтивно зрозумілий візуальний конструктор робочих процесів. Користувачі можуть "навчати" програму, які дані потрібно збирати, просто клацаючи на елементи веб-сторінки у вбудованому браузері. Програма автоматично генерує послідовність дій (робочий процес), що імітує взаємодію користувача з сайтом (наприклад, відкриття сторінки, скролінг, кліки по посиланнях, введення тексту).

Інструмент здатен працювати з сучасними веб-сайтами, які активно використовують JavaScript, AJAX, нескінченний скролінг, пагінацію, випадаючі меню та вимагають авторизації (логін/пароль). Це дозволяє збирати дані з ресурсів, які є складними для традиційних методів скрапінгу. Octoparse пропонує набір готових шаблонів для скрапінгу даних з багатьох популярних веб-сайтів, що дозволяє швидко розпочати збір даних без попереднього налаштування.

Платформа надає можливість виконувати завдання збору даних на власних хмарних серверах. Це дозволяє проводити скрапінг у фоновому режимі 24/7, прискорювати процес збору великих обсягів даних та знижувати навантаження на локальний комп'ютер користувача. Для уникнення блокування з боку веб-сайтів, Octoparse включає функції ротації IP-адрес та інші техніки, що допомагають імітувати поведінку реального користувача.

Зібрані дані можуть бути експортовані у різні формати, включаючи CSV, Excel, JSON, а також напряму до баз даних (наприклад, MySQL, SQL Server) або хмарних сервісів (наприклад, Google Sheets).

Octoparse використовується для різноманітних цілей, таких як моніторинг цін конкурентів, збір даних для маркетингового аналізу, генерація лідів (збір контактної інформації, включаючи профілі в соціальних мережах), моніторинг новин та контенту, академічні дослідження.

Щодо збору даних із соціальних мереж, Octoparse має можливості для скрапінгу публічно доступної інформації, такої як пости, коментарі, дані профілів (в межах дозволеного політиками платформ та законодавством).

Однак, варто пам'ятати, що збір даних із соціальних мереж часто обмежений їхніми API та правилами користування, а веб-скрапінг може порушувати ці правила та викликати технічні перешкоди з боку платформи. Незважаючи на це, Octoparse пропонує шаблони та функціонал, який може бути адаптований для роботи з певними аспектами соціальних мереж, де це технічно та юридично можливо.

1.5.2. Інструмент ParseHub

ParseHub – це потужний та гнучкий інструмент для веб-скрапінгу, призначений для вилучення даних з веб-сайтів без необхідності написання коду. Він особливо ефективний для роботи зі складними та динамічними веб-сторінками, які широко використовують JavaScript та AJAX для завантаження контенту.

Як і Octoparse, ParseHub пропонує візуальний підхід до створення завдань скрапінгу. Користувач взаємодіє з веб-сторінкою у вбудованому браузері та клацає на елементи, з яких потрібно зібрати дані. ParseHub використовує власні алгоритми для визначення структури даних та автоматичного вибору схожих елементів на сторінці. ParseHub спеціалізується на скрапінгу даних з веб-сайтів зі складною структурою, включаючи ті, що містять форми для заповнення, випадаючі списки, інтерактивні карти, спливаючі вікна та елементи, що завантажуються за допомогою AJAX або при скролінгу сторінки (нескінченний скролінг). Він може імітувати дії користувача, такі як кліки, введення тексту, навігація по сторінках та автентифікація (вхід на сайт).

Здатність ParseHub ефективно обробляти JavaScript та AJAX робить його придатним для збору даних з більшості сучасних веб-сайтів, контент яких генерується або оновлюється динамічно після завантаження сторінки. ParseHub є десктопним додатком, доступним для операційних систем Windows, Mac та Linux. Це може бути перевагою для користувачів, які

віддають перевагу локальному виконанню завдань або мають обмежений доступ до хмарних ресурсів на певних планах.

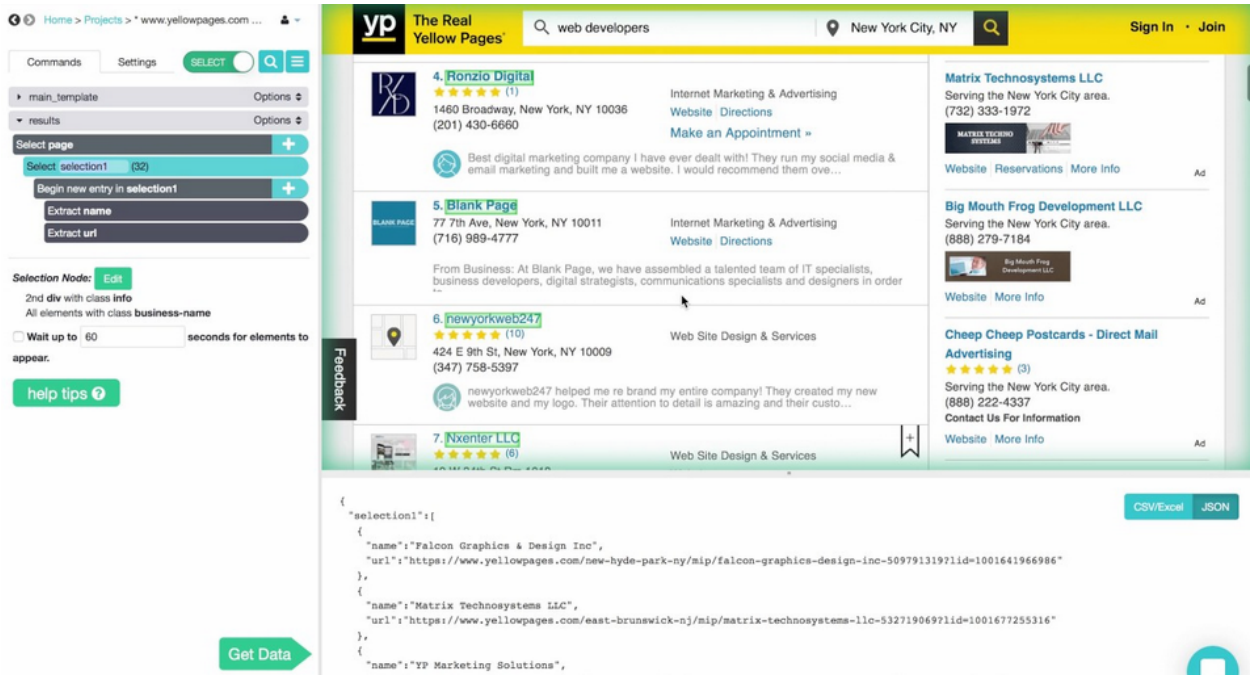


Рисунок 1.2 – Вигляд інструменту ParseHub

Хоча це десктопний додаток, ParseHub також пропонує хмарні сервіси для виконання завдань скрапінгу. Це дозволяє запускати завдання у фоновому режимі, за розкладом (щоденно, щотижнево тощо) та масштабувати збір даних. Зібрані дані можуть бути експортовані у поширених форматах, таких як CSV та JSON. Доступна також інтеграція з хмарними сховищами, такими як Dropbox або Amazon S3, та можливість використання API для доступу до даних з інших програм.

ParseHub орієнтований як на користувачів без досвіду програмування, так і на більш технічно підкованих фахівців (аналітиків, розробників), яким потрібен гнучкий інструмент для складних завдань скрапінгу. Сервіс також пропонує різні тарифні плани, включаючи безкоштовний план з обмеженими можливостями, що робить його доступним для невеликих проектів або ознайомлення.

ParseHub може бути використаний для скрапінгу публічно доступних даних з соціальних мереж, таких як пости, коментарі, інформація профілів, якщо ця інформація відображається у веб-версії платформи та доступна без специфічних обмежень API. Його здатність працювати з динамічним контентом та імітувати дії користувача може бути корисною для навігації по сторінках соціальних мереж. Однак, як і у випадку з будь-яким скрапінгом соціальних мереж, необхідно враховувати політику використання даних конкретної платформи та законодавчі вимоги, оскільки несанкціонований збір даних може бути заборонений. ParseHub також згадує можливість використання API соціальних мереж, якщо вони доступні, для більш надійного збору даних в межах дозволеного.

Загалом, ParseHub є потужним візуальним інструментом, який значно спрощує процес веб-скрапінгу, особливо для сайтів зі складною структурою та динамічним контентом, і може бути ефективним інструментом для дослідників та аналітиків, які працюють з веб-даними.

1.5.3. Хмарна платформа скрапінгу Apify

Apify – це хмарна платформа, яка надає інструменти та інфраструктуру для веб-скрапінгу, автоматизації веб-завдань та обробки даних. На відміну від окремих програм для скрапінгу, Apify є більш комплексною платформою, що базується на концепції "Actors".

Розглянемо ключові концепції та компоненти Apify.

1. Actors (Актори) - це основні блоки платформи Apify. Актори — це, по суті, серверні програми або мікросервіси, які виконують певні завдання. Вони можуть бути написані з використанням Apify SDK (доступний для JavaScript/Node.js та Python) або бути довільним Docker-образом. Актори можуть виконувати різноманітні дії, включаючи:

- Веб-скрапінг та краулінг (збір даних з веб-сайтів).

- Автоматизація дій у браузері (наприклад, заповнення форм, імітація кліків).

- Обробка та трансформація даних.

- Виконання довільних обчислювальних завдань.

- Інтеграція з іншими сервісами через API.

2. Apify Store - це маркетплейс, де користувачі можуть знаходити та використовувати тисячі публічних Акторів, розроблених спільнотою Apify. У магазині є готові Актори для скрапінгу даних з популярних веб-сайтів та соціальних мереж (хоча доступність та функціональність можуть залежати від політики відповідних платформ). Користувачі також можуть ділитися своїми власними Акторами та навіть монетизувати їх.

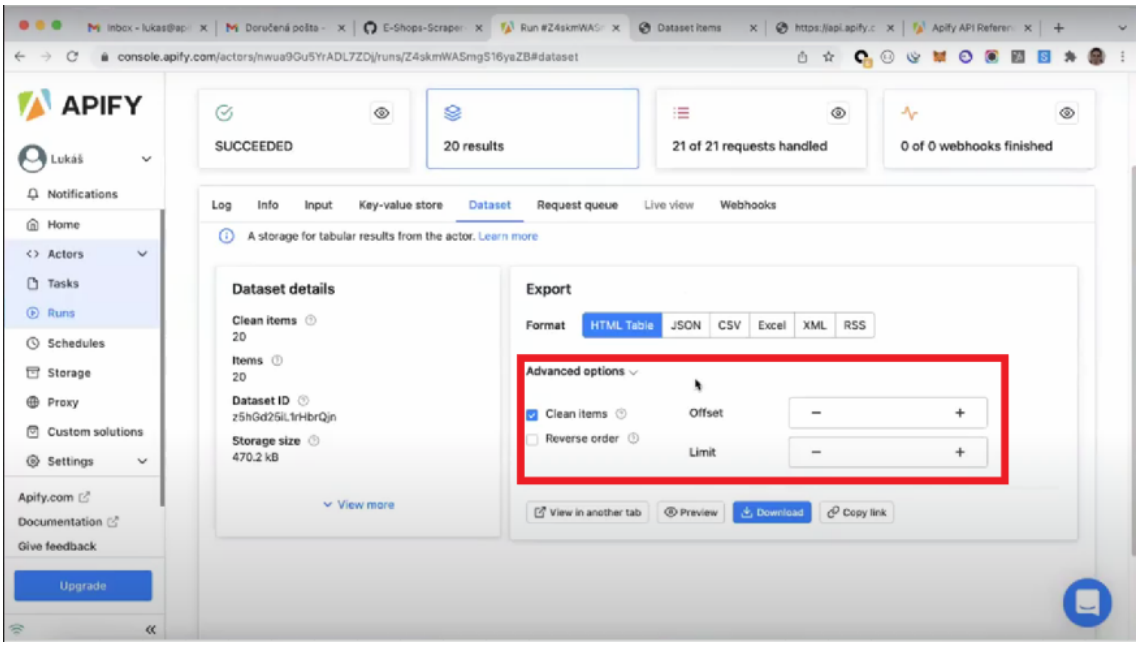


Рисунок 1.3 – Хмарна платформа скрапінгу Apify

3. Платформа Apify (Apify Console) - веб-інтерфейс для керування Акторами, завданнями, даними та іншими ресурсами. Надає можливості для:

- Запуску та моніторингу виконання Акторів.

- Налаштування вхідних параметрів для Акторів.

- Планування регулярних запусків завдань.

- Зберігання та експорту зібраних даних у різних форматах.

- Доступу до логів виконання для налагодження.

- Керування проксі-серверами та обробки блокувань.

Переваги Apify:

1. Завдання виконуються на масштабованій хмарній інфраструктурі Apify, що дозволяє обробляти великі обсяги даних та запускати завдання паралельно без значного навантаження на локальні ресурси користувача.

2. Можливість використання готових Акторів з магазину або розробки власних кастомних рішень робить платформу дуже гнучкою.

3. Вбудовані механізми обробки помилок, повторних спроб, керування проксі та обходу блокувань підвищують надійність збору даних.

4. Можливість планування завдань та інтеграції з іншими сервісами через API або вебхуки дозволяє автоматизувати робочі процеси.

Apify активно використовується для збору даних із соціальних мереж завдяки наявності в Apify Store готових Акторів, розроблених для роботи з такими платформами як Facebook, Instagram, Twitter (X), TikTok, Reddit тощо. Ці Актори дозволяють збирати публічно доступну інформацію (пости, коментарі, дані профілів, хештеги), звісно, в межах обмежень, встановлених самими соціальними мережами та відповідним законодавством.

Важливо підкреслити, що хоча Apify значно спрощує технічний бік скрапінгу, користувачі все одно несуть відповідальність за дотримання правил використання веб-сайтів та соціальних мереж, з яких вони збирають дані, а також відповідних законів про захист даних.

Висновки до розділу

У першому розділі було здійснено комплексний аналіз використання даних із соціальних мереж у дослідженнях соціальних наук, зокрема в контексті цифровізації суспільства. Встановлено, що соціальні медіа

відіграють важливу роль як джерело великого обсягу інформації, що відображає соціальні процеси, поведінку та взаємодію користувачів у режимі реального часу.

Було розглянуто природу комп'ютерних наук та їх інтеграцію в дослідження соціальних явищ, що дозволяє поєднувати традиційні методи аналізу з інструментами машинного навчання, аналізу даних та автоматизації. У межах аналізу проблематики підкреслено виклики, пов'язані з етичністю, правомірністю доступу до даних, технічними обмеженнями та обробкою неструктурованої інформації.

Значущість збору даних соціального нетворкінгу обґрунтована як з наукової, так і з прикладної точок зору — від дослідження громадської думки до прогнозування соціальних тенденцій. Розглянуто та проаналізовано можливості трьох популярних інструментів для веб-скрапінгу: Octoparse, ParseHub і Apify. Кожен із них має свої переваги та обмеження щодо гнучкості налаштування, зручності використання, масштабованості та інтеграції з іншими сервісами.

Таким чином, розділ сформував теоретичне та практичне підґрунтя для подальшої розробки рішень щодо ефективного та етичного збору даних із соціальних платформ

РОЗДІЛ 2. ДОСЛІДЖЕННЯ СТРУКТУРИ ДАНИХ ВЕБ-РЕСУРСІВ СОЦІАЛЬНОГО НЕТВОРКІНГУ НА ПРИКЛАДІ TIKTOK

2.1. Структура та основні компоненти TikTok

Після обговорення значущості даних соціальних медіа для соціальних наук та комп'ютерних наук ми тепер зосередимося на більш вузькій темі. Важливо пам'ятати, що ця робота досліджує конкретну соціальну медіа-платформу: TikTok. Тому, перш ніж переходити до формальних та технічних аспектів, важливо краще зрозуміти загальний контекст TikTok.

Додаток TikTok розвинувся з китайського додатка Douyin, розробленого ByteDance у 2016 році. Douyin дозволяв користувачам створювати та ділитися короткими відео, а також споживати контент від інших користувачів. У вересні 2017 року ByteDance випустила міжнародну версію додатка під назвою TikTok. Додаток мав світовий успіх і до 2021 року був названий найвідвідуванішим веб-сайтом у світі, випередивши Google. Одним із показників його успіху є те, що в багатьох країнах мобільні контракти тепер пропонують опцію включення TikTok, що дозволяє користувачам інтенсивно використовувати платформу. Те, що мобільні оператори визнали це як значущий фактор при виборі контракту користувачами, є значущим [8].

TikTok працює, дозволяючи користувачам отримувати доступ до стрічки контенту, коли вони відвідують додаток або веб-сайт. Після відкриття додатка TikTok або веб-сайту користувачі одразу перенаправляються на сторінку зі стрічкою контенту, як показано на рисунку рис. 2.1.

Рисунок 2.1 демонструє головну сторінку TikTok та три ключові компоненти, позначені 1-3. Сторінка "Для тебе" (1) показує персоналізований контент у вигляді коротких відео. Приклад контенту показано в (2), де відео є

центральним елементом. Відео автоматично починає відтворюватися, якщо виконуються всі технічні вимоги (наприклад, сумісність браузера), і звук можна увімкнути, натиснувши кнопку звуку в нижньому правому куті. Користувачі можуть безпосередньо взаємодіяти з контентом за допомогою вподобань, коментарів або ділитися ним. Для цього їм потрібно увійти в систему, що можна зробити в правому верхньому куті (3).

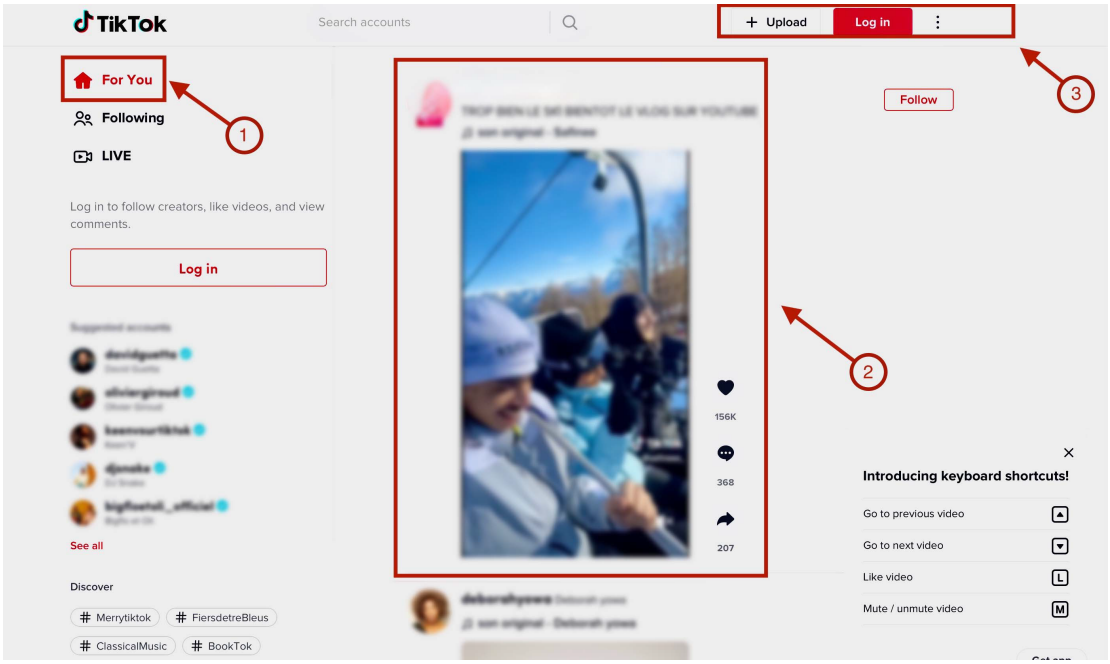


Рисунок 2.1 – Сторінка TikTok ’For You’

Основним елементом TikTok є контент, показаний у (2). Рисунок 2.2 ілюструє анатомію такого контенту.

Як видно на рис. 2.2, кожне відео в соціальних медіа розміщується користувачем, а праворуч від відео інші користувачі можуть бачити коментарі, залишені під відео. Відео супроводжується описом, який користувач пише під час його розміщення. В цьому описі користувач може включати хештеги, які є словами або фразами, що передують символом "#". Хештеги використовуються для групування та категоризації контенту за темою або тематикою, і натиснувши на хештег, користувачі можуть перейти на сторінку, яка показує всі відео, які містять цей конкретний хештег.

Рисунок 2.2 – Структура контенту в TikTok

Крім того, користувач, який розмістив відео, може згадати інших користувачів, написавши "@" перед їхнім ім'ям користувача. Коли користувача згадують, він отримує сповіщення, а потенційні користувачі можуть відвідати сторінку згаданого користувача, натиснувши на його ім'я. Сторінка містить весь контент, який розмістив користувач.

Під описом є посилання на сторінку музики, яка відповідає музиці, присутній у відео. Натиснувши на це посилання, користувач потрапляє на сторінку, де зберігаються всі відео, які містять ту саму музику, подібно до натискання на хештег.

Мета пояснення сторінки TikTok - надати огляд її функцій та можливостей та визначити потенційні області для більш детального вивчення поведінки користувачів на платформі. Сторінка TikTok є складною, а розміщені відео є складними повідомленнями з прихованими намірами. Дослідження сторінки несе в собі небезпеку загубитися в деталях і не побачити загальної картини. Щоб уникнути цього, я постійно задавав питання "Що можливо?" в межах заданого часового обмеження роботи. Це

питання слугувало керівним принципом для фільтрації елементів та зосередження уваги на досяжному. Запропоноване в цій роботі рішення не було досягнуте шляхом зверху вниз, а шляхом дослідження доступних даних та того, які дані будуть корисними. Механізм фільтрації буде додатково посилено в наступних розділах для звуження уваги на найбільш актуальну інформацію.

2.2. Технічний контекст збору даних

У попередньому розділі було проведено огляд TikTok. Як згадувалося раніше, однією з цілей цього дослідження є збір даних для соціальних досліджень. З моєї точки зору, існує два підходи для досягнення цієї мети: зверху вниз та знизу вгору.

Як підхід зверху вниз, планувалось визначити слід соціальної поведінки, присутній на TikTok, а потім отримати доступ до відповідних даних. Однак цей підхід був нездійсненним через суворі обмеження на доступ, встановлені TikTok. Як ми побачимо пізніше в цьому розділі, не всі дані платформи можуть бути отримані за допомогою веб-скрапінгу. Інструмент, який використовується для дослідницьких цілей, повинен гарантувати, що він залишається в межах правових рамок.

Підхід знизу вгору, як випливає з назви, працює в зворотному порядку. Цей підхід передбачає спочатку дослідження доступних даних та способу, яким соціальні науковці та комп'ютерні науковці наразі їх обробляють. На основі цих досліджень буде запропоновано теоретичну модель та відповідним чином налаштування збору даних.

Для реалізації цього підходу спочатку виконаємо огляд веб-скрапінгу та того, як веб-сайти обмежують доступ до своїх даних. Також розглянемо правові аспекти володіння даними та етики.

Щоб ввести читача в концепцію веб-скрапінгу, наведемо відому цитату науково-фантастичного письменника Артура К. Кларка: "Будь-яка досить розвинена технологія не відрізняється від магії." Коли ми вводимо "www.tiktok.com" у поле введення нашого браузера та натискаємо Enter, наш комп'ютер надсилає запит маршрутизатору з проханням про інформацію за цією адресою. Маршрутизатор передає запит наступному серверу, який спрямовує його в правильному напрямку, поки він не досягне сервера, який зберігає інформацію. Сервер перевіряє дозволи запиту, і якщо вони схвалені, відправляє відповідні дані назад.

Google Chrome надає зручний спосіб побачити, що відбувається за лаштунками. Після відкриття браузера, натисніть правою кнопкою миші на сторінці та виберіть "переглянути код". Це відкриє вікно поруч з попереднім виглядом і дозволить вибрати поле Мережа вгорі. Якщо ви відвідаєте "www.tiktok.com", наприклад, ви побачите багато активності в розділі мережі вікна перегляду. Ця опція розкриває частину "магії", яку середній користувач інтернету не бачить і не повинен знати. У вікні можна побачити та переглянути всі файли, які були надіслані на сервер як відповідь на запит, зроблений введенням URL-адреси в браузер.

Якщо хтось хоче досліджувати TikTok, він може відвідати відповідні сторінки та занотувати інформацію. Однак оскільки відповідні файли тимчасово присутні на комп'ютері дослідника, він може написати програму, відому як скрапер, щоб автоматизувати цей процес. Зазвичай дані, які цікавлять, є подібними та можуть бути знайдені за URL-адресами, які можуть бути побудовані з раніше зібраних даних. Ці URL-адреси можуть бути побудовані за допомогою фіксованого набору правил, і їх також можна генерувати програмою, відомою як веб-краулер. На практиці концепції краулінгу та скрапінгу часто використовуються як взаємозамінні, але як згадувалося раніше, вони мають різні значення.

Оскільки процес скрапінгу просто автоматизує завдання, які користувач Інтернету може виконувати вручну, можна стверджувати, що він не є незаконним або неетичним.

Однак легальність та етика скрапінгу в контексті соціальних медіа є ще більш суперечливими. Користувачі соціальних медіа створюють більшість контенту, який підвищує цінність цих веб-сайтів, роблячи їх цінними об'єктами для дослідників та бізнесу. Однак користувачі можуть не усвідомлювати, що їхні дії аналізуються та розуміються цими засобами.

Доступність до веб-скраперів є двоїстою зброєю для соціальних медіа-платформ. З одного боку, це необхідна умова для підвищення популярності веб-сайту, оскільки пошукові системи, такі як Google, використовують скрайпінг для доступу до веб-сайту та появу в результатах пошуку. З іншого боку, дозволяючи скрайперам отримувати доступ до своїх даних, вони можуть потенційно розкрити чутливу інформацію, яку користувачі можуть не бажати робити публічною. Платформи вжили заходів, щоб ускладнити веб-скраперам автоматичний доступ до своїх даних, але цей баланс між доступністю та конфіденційністю залишається викликом.

Щоб краще зрозуміти, як TikTok керує веб-скрапінгом на своєму веб-сайті, я ознайомився з їхніми правилами спільноти. У цих правилах вони згадують веб-скрапінг у наступному реченні: "Не використовуйте автоматичні скрипти, веб-краулінг, програмне забезпечення, обманні техніки або будь-які інші засоби для спроби отримати, отримати або запитати логіни або іншу чутливу інформацію, включаючи непублічні дані, від TikTok або його користувачів" [35]. Наш скрапінговий програмний засіб дотримується цих правил, оскільки він не цілюється на логіни та отримує доступ лише до даних з публічних акаунтів. Крім того, ми дотримуємося файлу robots.txt TikTok.

Файл robots.txt знаходиться на більшості професійних веб-сайтів і вказує, які дані веб-сайт дозволяє веб-скраперам отримувати доступ. Файл

призначений для інструктування веб-скраперів, до яких зазвичай звертаються програми, а не користувачі. Розташування файлу фіксоване, за адресою maindomain/robots.txt, і він має стандартизовану структуру. У випадку з TikTok ми можемо переглянути його файл robots.txt, щоб зрозуміти його позицію щодо веб-скрапінгу.

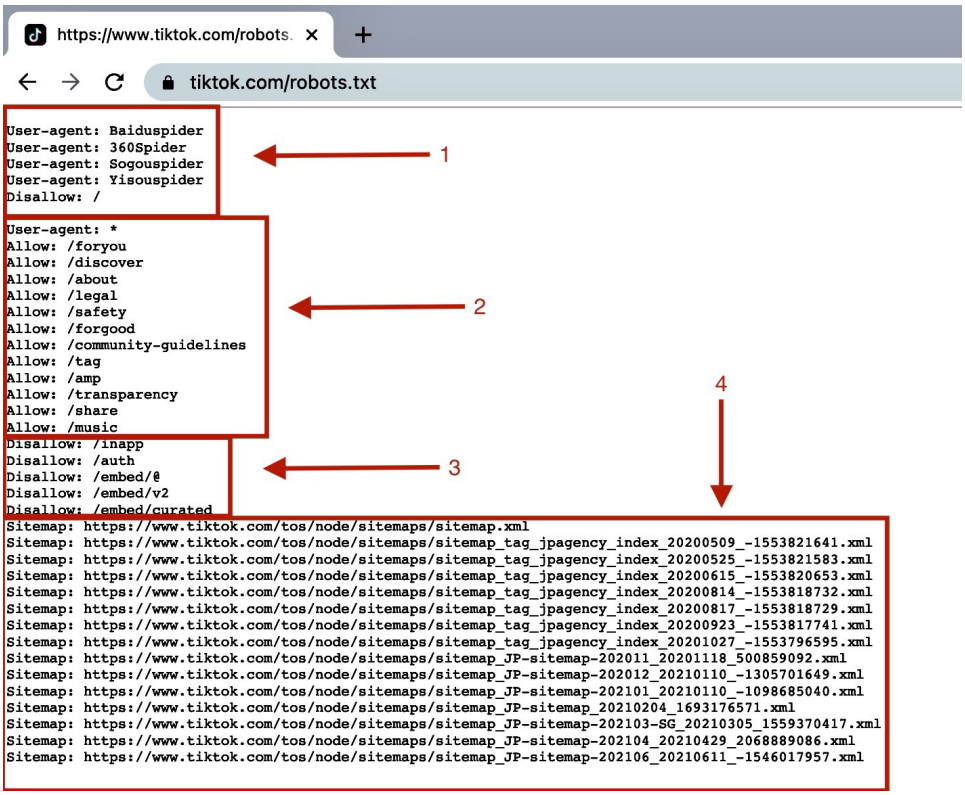


Рисунок 2.3 – Файл robots.txt в TikTok

Рисунок 2.3 демонструє файл robots.txt TikTok. Червоні прямокутники додані для ясності. Перший прямокутник показує агентів користувачів, яким не дозволено отримувати доступ до будь-якого контенту на сторінці. Ці агенти користувачів відповідають певним бізнесам, яким TikTok хоче обмежити автоматичний доступ до свого веб-сайту. У другому прямокутнику ми бачимо всі URL-адреси, до яких TikTok дозволяє доступ усім програмам, крім тих, які вказані в прямокутнику 1. Зокрема, ми можемо побачити, що краулери можуть отримувати доступ до сторінок музики та хештегів. Третій прямокутник містить всі URL-адреси, для яких TikTok забороняє будь-яку

діяльність скрапінгу, включаючи всі URL-адреси автентифікації, які можуть бути переглянуті лише з точки зору авторизованого користувача. Це означає, що контент, який вимагає від користувача увійти в систему, не може бути зібраний. Четвертий прямокутник містить карти сайту, призначені для краулера Google, і не є актуальними для цієї роботи.

Введення концепції веб-скрапінгу та аналіз файлу robots.txt дозволили нам краще зрозуміти інформацію, до якої можна отримати доступ за допомогою веб-скрапінгу на TikTok, включаючи публічні акаунти користувачів, відео та контент, пов'язаний з конкретними хештегами або музичними треками. З цими знаннями ми тепер можемо перейти до останніх підрозділів цього розділу, метою яких є ознайомлення читача з концепцією графів та перекладом отриманих інсайтів у теоретичну модель TikTok.

2.3. Використання графів для техніки збору даних

Ми вже сформували чітке розуміння платформи TikTok та даних, до яких ми можемо отримати доступ за допомогою веб-скрапінгу. Але просто отримання доступу до даних не досягає цілей цієї роботи. Завдання, яке я прагну вирішити в цій роботі, полягає в тому, щоб привести ці дані в таку форму, яка буде зручною для аналізу з точки зору соціальних наук. Я хочу стверджувати, що найкращий спосіб робити це - зберігати дані безпосередньо у формі графу. Мета цього підрозділу - ввести або нагадати читачеві про поняття графу. Тут буде обгрунтовано формальне визначення графу та виконано більш детальне пояснення. Також буде представлено корисність цих математичних об'єктів у аналізі соціальних мереж. Нарешті, буде показано, як алгоритми можуть бути використані для кращого розуміння графів та структур, які вони представляють.

Математичні об'єкти часто використовуються для моделювання певних явищ у реальному світі. Ми спостерігаємо за чимось і хочемо знайти точний

опис, який дозволить описати ці явища максимально повно. Одним можна погодитися, що часто явища можна добре описати за допомогою моделі, яка включає ентітети, які беруть участь у дії, яка нас цікавить, та тим, як вони беруть участь. Давайте розглянемо приклад. Давайте уявимо собі відзначених користувачів TikTok Ешера, Геделя та Кюрі. Давайте уявимо, що Ешер і Гедель розмістили відео vEvE та vGvG відповідно. Ми також знаємо, що Гедель переглянув 66 відео Ешера. Крім того, Ешер також переглянув відео Геделя vGvG. Кюрі переглянула обидва vGvG та vEvE, але не розмістила жодного відео на своєму акаунті. Тепер пропонується невелике завдання – спробувати описати інформацію вище в якомусь вигляді візуалізації та порівняти із запропонованим на рис. 2.4.

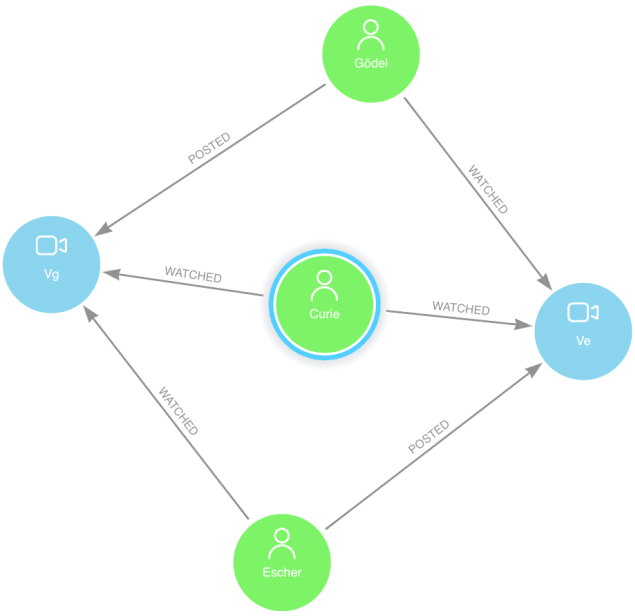


Рисунок 2.4 - Приклад моделі взаємодії в TikTok

Те, що ми бачимо на цьому рисунку, це вже більш складна візуалізація графу. Вона складна, тому що ми маємо різні типи взаємодій між ентітетами, наприклад, "розміщено" та "переглянуто", і зв'язки між ентітетами мають напрямок, вказаний стрілками. Для того щоб ввести об'єкт графу належним

чином, ми спростимо вищезгадану ілюстрацію, не розрізняючи ні різних типів вузлів, ні різних типів зв'язків та їхніх напрямків. Роблячи так, ми отримаємо граф, показаний на рис. 2.5.

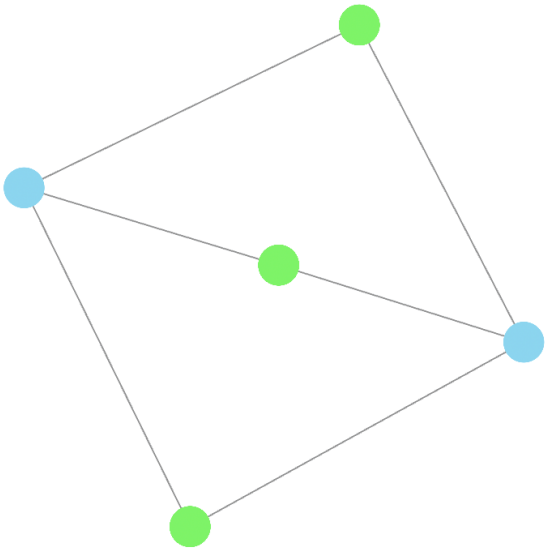


Рисунок 2.5 – Представлення спрощеної взаємодії в TikTok

Ентітети, в нашому випадку користувачі та відео в графі, в комп'ютерних науках часто називаються вузлами. Зв'язки, представлені лініями між вузлами, називаються ребрами. Граф, де ці ребра не мають напрямку, можна описати просто переліком усіх вузлів та усіх ребер між вузлами. Відповідно, ми могли б формально описати граф, візуалізований на рис. 2.5, зазначивши, що граф

$$G_{simple} = (N, E_{simple})$$

де

$$N = \{Goedel, Curie, Escher, V_g, V_e\}$$

$$E_{simple} = \{\{Goedel, V_g\}, \{Goedel, V_e\}, \{Escher, v_g\}, \{Escher, v_e\}, \{Curie, v_g\}, \{Curie, v_e\}\}$$

Ми можемо використовувати схожий спосіб опису більш складного графу, візуалізованого на рис. 2.4. Подібно до простого графу, ми описуємо граф як

$$G_{soph} = (N_{soph}, E_{soph})$$

В цьому випадку ми враховуємо наявність типів вузлів, зазначивши, що

$$N_{soph} = \{V, U\}$$

де V - це набір відео, а U - це набір користувачів. Крім того, напрямок всередині графу враховується шляхом використання кортежів замість множин для представлення одного ребра. Таким чином, ми можемо представити складний граф, встановивши

$$E_{soph} = \{W, P\}$$

де W - це "переглянуті" взаємодії, а P - це "розміщені" взаємодії, тобто

$$W = \{((Goedel, V_e), (Escher, v_g), (Curie, v_g), (Curie, v_a)\}$$
$$P = \{(Goedel, v_g), (Escher, v_e)\}$$

Загалом, багатовимірний граф можна визначити наступним чином.

Багатовимірний орієнтований граф *G=(N,E)* - це кортеж, де *N={N1,⋯,Nk}* - це набір множин, де кожна *Ni* містить скінченну кількість вузлів, а *E={E1,⋯,Ej}* - це набір множин, де кожна множина всередині *E* містить скінченну кількість ребер між вузлами, які містяться в *Ni*, для $1 \le i \le k$.

Графи широко використовуються як у соціальних науках, так і в комп'ютерних науках через їхню корисність у узагальненні інформації. Крім того, графи є добре відомими об'єктами в математиці та комп'ютерних науках із відомими властивостями та застосовними алгоритмами. Хоча неможливо

описати всі ці алгоритми, я згадаю два типи питань, на які можна відповісти, застосовуючи алгоритми до графів.

Перший тип питання: які вузли є важливими в заданому графі? Ці типи питань зазвичай отримують відповіді за допомогою вимірів центральності вузлів. Різні алгоритми можуть бути використані для розрахунку вимірів центральності в залежності від того, що вважається "важливим" в заданій ситуації. Один простий спосіб розрахунку центральності - це сумування вхідних та вихідних ребер вузла, що буде продемонстровано в цій роботі. Цей вимір часто називають ступенем вузла.

Другий тип питання: які вузли, ймовірно, належать один до одного? Ці типи питань зазвичай отримують відповіді з використанням алгоритмів виявлення спільнот. Хоча ці алгоритми трохи складніші, інструменти, представлені в цій роботі, дозволяють легко застосовувати алгоритми виявлення спільнот. Однак оскільки основна мета цієї роботи не є аналізом зібраних даних, то ці алгоритми не будуть застосовані в цій роботі.

2.4. Теоретично-формальна модель взаємодії користувачів в соціальній мережі

На основі аналізу, представленого у попередніх підрозділах, було ідентифіковано питання стосовно даних та потенційні методи їх аналізу. Додатково було розглянуто технічні, етичні та правові обмеження щодо доступу до даних через застосування інструментів веб-скрапінгу. У цьому підрозділі представлено систему визначень та теоретичну модель, що базується на отриманих інсайтах. Метою є забезпечення чіткого та однозначного визначення термінології, що застосовується для представлення даних, зібраних у рамках даного дослідження. Це забезпечить точність та ясність у визначенні концепцій та змінних дослідниками. Такий підхід сприятиме формуванню спільного розуміння термінології в науковій

спільноті, мінімізуючи термінологічну неоднозначність та підвищуючи відтворюваність результатів дослідження. Крім того, точні визначення є необхідною умовою для застосування математичних та статистичних методів.

Метою даного розділу є формулювання визначень ключових концепцій, що стосуються використання соціальних медіа людиною, зокрема платформи TikTok. Ці визначення покликані забезпечити точне окреслення об'єктів вимірювання, що здійснюватиметься за допомогою розробленого інструменту. Не слід вважати, що визначення терміна X як Y (X=Y) імплікує, що X є Y та виключно Y за будь-яких обставин. Натомість, у контексті даної роботи термін X слід інтерпретувати виключно як Y. Усі подальші висновки, що базуються на інтерпретації X як Y, є коректними за умови прийняття даного визначення. Відтак, далі буде визначено основні сутності (ентітети), що формують мікро-макро модель.

Індивід, позначений символом I, є абстрактним об'єктом. Множина кількох індивідів позначається як $I\wedge$, тоді як унікальна множина всіх існуючих індивідів позначається як I.

Дане визначення індивіда є універсальним та допускає адаптацію для різних дослідницьких задач. Наприклад, у дослідженні, зосередженому на віруваннях, абстрактний об'єкт може бути конкретизований для представлення системи вірувань індивіда. Наступним кроком є визначення сутності макро-рівня – суспільства.

Суспільство St в певний момент часу t - це кортеж $(It\wedge, Rt)$, де $It\wedge$ - це множина індивідів, а $Rt=\{(ia,ib,\omega) \mid a,b\in It\wedge,\omega\in R\}n$ - це множина n можливих типів взаємодій між усіма індивідами. Встановлюємо $It\in St\leftrightarrow It \in It\wedge$.

Це визначення узгоджується з розумінням суспільства як мережі, що запропоновано теоретиками соціальних мереж. Таким чином, суспільство визначено як сукупність індивідів та множина взаємодій між ними. Тип

взаємодії не конкретизується на даному етапі, оскільки він визначається специфікою дослідницьких цілей. Визначення індивідів та суспільства забезпечує наявність сутностей як для мікро-, так і для макро-рівнів у структурі моделі. Подальший етап полягає в розгляді переходів між станами моделі. Аналіз розпочинається з мікро-рівня, де індивіди діють відповідно до теорії соціальних дій. Відтак, далі буде формалізовано поняття соціальної дії.

Дія індивіда $It \in It \wedge$ в момент часу t - це відображення $a:(It,Rt,O) \rightarrow Rt+1$, де Rt+1 представляє множину взаємодій між індивідами в суспільстві St. Відображення at визначає, як дія індивіда It в момент часу t впливає на взаємодії між індивідами в суспільстві St+1, а O - це множина інших визначених факторів, від яких може залежати дія.

Інтерпретуючи формальне визначення, дія індивіда детермінується поточними взаємодіями та низкою інших факторів (O). Результат дії визначено як модифікацію множини взаємодій індивіда всередині суспільства, що є ключовим аспектом аналізу в соціальних науках. Базуючись на цьому визначенні, можна перейти до формалізації процесу на макро-рівні наступним чином:

Соціальна взаємодія f - це функція з двовимірного простору дій та взаємодій до простору взаємодій, отже $St+1=f(St,At)$, де St - це суспільство в момент часу t, а At - це множина дій всередині цього суспільства. Множину всіх можливих соціальних взаємодій в момент часу t позначимо як $\otimes \sqcup$.

Представлена формалізація соціальних взаємодій дозволяє враховувати переходи як на мікро-, так і на макро-рівнях моделі. Далі буде представлено визначення поняття соціального середовища в даному контексті.

Соціальне середовище визначається як функція f $SM:\Omega t \rightarrow SMt$. Де SMt - це деякий абстрактний простір в момент часу t.

Згідно з даним визначенням, кожна властивість усіх можливих конфігурацій суспільства відображається унікальним елементом абстрактного простору SMt. Такий підхід дає змогу фіксувати результати

соціальних взаємодій, що проявляються у соціальному середовищі. Для моделювання різних платформ соціальних медіа структура абстрактного простору може бути відповідним чином адаптована. У попередньому розділі було проаналізовано типи даних, доступних для збору за допомогою веб-скрапінгу на платформі TikTok. Відтак, далі буде представлено визначення відповідних сутностей (ентітетів).

Користувач соціальних медіа - це сюр'єктивне підмножина $U \subset FSN:It \rightarrow SMt$.

Отже, користувачі визначені як представники індивідів у межах суспільства. Ця концептуалізація є інтуїтивно зрозумілою. Сюр'єктивність даного відображення відображає той факт, що кожен користувач асоціюється щонайменше з одним індивідом. Визначення соціальних медіа як відображення (а не функції) враховує можливість асоціації одного індивіда або групи з кількома акаунтами в соціальних медіа.

Хештег - це кортеж (s,m), де s - це послідовність символів, така що s(0)=#, а m - це значення слова s(1:довжина(s)). Множину всіх існуючих хештегів позначимо як H. H позначатиме множини хештегів.

Музичний трек визначається як множина M={A,T,MI}, де A - це представлення артистів, які створили музику, T - це відповідна назва, а MI - це будь-яка додаткова інформація, пов'язана з музикою.

Таким чином, наявні всі необхідні компоненти для формалізації поняття посту.

Пост в TikTok P - це множина {UT(It),H,M,Vопис,l,c,C,ME}, де UT(It) - це унікальне представлення користувача певних індивідів або групи індивідів, пов'язаних з постом. Vопис - це рядок, що представляє опис відео. H - це множина хештегів. M - це музичний трек. l - це ціле число, що представляє кількість вподобань. c - це ціле число, що представляє кількість коментарів. C - це множина всіх коментарів, пов'язаних з відео. ME - це список користувачів, згаданих у пості.

Множину всіх постів в момент часу t позначимо як Pt.

Це дозволяє формалізувати визначення платформи TikTok шляхом специфікації структури простору SM.

Веб-застосунок TikTok в момент часу t - це соціальне середовище, де SMt:=Pt.

Наступним етапом є визначення поняття дискурсу, що циркулює на платформі TikTok. Це визначення є центральним, оскільки воно формує основу для моделювання досліджуваних даних. Концепція багатовимірного графа, введена у попередньому підрозділі, та дані про доступні сутності (пости, музичні треки, користувачі) слугують підставою для формулювання наступного визначення, що представляє теоретичну модель дискурсу TikTok в рамках даного дослідження.

Дискурс TikTok в момент часу t - це багатовимірний граф TDt=(N,E), де N={P,H,M,U} є множиною вузлів, що складається з:

P={p1,…,pn} - множина постів.

H={h1,…,hm} - множина хештегів, таких що hi∈Hpj для деякого поста pj∈P, де Hpj - множина хештегів у пості pj.

M={m1,…,ml} - множина музичних треків, таких що mi∈Mpj для деякого поста pj∈P, де Mpj - музичний трек у пості pj.

U={u1,…,uk} - множина користувачів, таких що ui є автором поста pj (ui=UT(It) для pj в момент t) або ui∈MEpj для деякого поста pj∈P, де MEpj - множина згаданих користувачів у пості pj.

Крім того, E є множиною ребер, що складається з підмножин PO, MEN, IN, де:

PO - множина ребер "опублікував": (u,p)∈PO⇔u∈U є автором поста p∈P.

MEN - множина ребер "згадує": (p,u)∈MEN⇔p∈P містить згадку користувача u∈U (тобто u∈MEp).

IN - множина ребер "включення" щодо хештегів та музики: $(p,h) \in IN \Leftrightarrow p \in P$ містить хештег $h \in H$ (тобто $h \in Hp$), та $(p,m) \in IN \Leftrightarrow p \in P$ використовує музичний трек $m \in M$ (тобто $m=Mp$).

Подальша ілюстрація цих визначень буде представлена у третьому розділі.

Висновки до розділу

У другому розділі розглянуто особливості структури даних веб-ресурсу соціального нетворкінгу TikTok, який є однією з найдинамічніших і найпопулярніших платформ сучасності. Проаналізовано основні компоненти платформи, включаючи профілі користувачів, відеоконтент, хештеги, коментарі та механізми взаємодії, які формують основу для збору та аналізу соціальних зв’язків і поведінкових патернів.

Описано технічні аспекти збору даних із TikTok, зокрема особливості структури HTML, використання API (офіційних та неофіційних), обмеження на запити, а також загрози блокування через системи захисту. Це дозволяє оцінити як можливості, так і ризики, пов’язані зі збором великомасштабних даних із подібних платформ.

Окрема увага приділена використанню графових структур для моделювання взаємозв’язків між користувачами, контентом і взаємодіями. Це дає змогу більш глибоко аналізувати соціальну динаміку та виявляти кластери, інфлюенсерів, а також моделі розповсюдження інформації.

На завершення розділу представлено теоретично-формальну модель взаємодії користувачів у соціальній мережі, яка описує процеси створення, обміну та реакції на контент з точки зору інформатики та соціальної поведінки. Такий підхід формує основу для побудови алгоритмів аналізу даних та розвитку систем рекомендацій.

Таким чином, результати розділу створюють практичну та концептуальну базу для ефективного збору, структурування та аналізу даних із TikTok та подібних соціальних платформ.

РОЗДІЛ 3. ПРОГРАМНА ІМПЛЕМЕНТАЦІЯ ТЕХНІКИ ЗБОРУ ДАНИХ З ВЕБ-РЕСУРСІВ СОЦІАЛЬНОГО НЕТВОРКІНГУ

3.1. Опис процесу розробки інтерфейсу застосунку

В попередньому розділі було визначено ціль даного дослідження, що полягає у створенні набору даних з платформи TikTok, та розроблено теоретичну модель структури даних для соціальних досліджень. У цьому розділі детально описано процес практичної реалізації зазначеної теоретичної моделі. Спочатку буде представлена концепція програмного інтерфейсу застосунку (API) та її відношення до розробленого програмного рішення. Далі буде пояснено використання веб-скрапінгової платформи Scrapy у рамках даного проекту. Потім буде розглянуто графову базу даних Neo4j, що була використана для зберігання зібраних даних. Наприкінці, стисло розглянуто використання Docker як інструменту для підвищення доступності та зручності використання розробленого інструменту. Після представлення вказаних компонентів, детальніше буде описано алгоритми, що застосовувалися для реалізації функції скрапінгу.

У попередніх розділах було зазначено, що двома з трьох ключових цілей даної роботи було створення веб-скрапера для збору даних з платформи TikTok та забезпечення легкої доступності розробленого програмного рішення. Хоча це може вказувати на намір створення API, у цьому підрозділі буде пояснено, чому це не є метою даного дослідження. Для цього спочатку буде надано визначення програмного інтерфейсу, а далі буде проведено розмежування між реалізацією веб-скрапера та реалізацією API.

API (програмний інтерфейс застосунку) можна розглядати як програмний компонент, призначений не для безпосереднього використання кінцевим користувачем, а для взаємодії з іншими комп'ютерними

програмами. Початково веб-API представляли собою набір програмних функцій, що спрощували складні взаємодії в мережі Інтернет.

Для ілюстрації функціональності API часто використовується метафора офіціантки в ресторані. Розглянемо сценарій відвідування ресторану. Після того, як відвідувач займає місце, офіціантка приймає замовлення. Офіціантка передає замовлення на кухню шеф-кухарю для приготування страви. Після приготування страви офіціантка доставляє її до столика відвідувача. У даній аналогії відвідувач ресторану (або програма-клієнт) не потребує знань про складні процеси приготування їжі (або взаємодії із сервером), необхідні для отримання замовленої страви (або даних).

Створення API, що забезпечує програмний доступ до даних TikTok у зручному форматі, було б оптимальним рішенням для поставленої задачі. Проте, реалізація повноцінного API виходить за межі обсягу даної бакалаврської роботи. Причиною цього є необхідність наявності вже структурованих даних, до яких можна легко отримати доступ, для розробки API. Хоча платформа TikTok має власний офіційний API, його використання є фінансово витратним та, як наслідок, нереалістичним для більшості академічних дослідницьких проектів. Таким чином, веб-скрапер, розроблений у межах даної роботи, виконує функцію вилучення даних з платформи TikTok та їх збереження у базі даних, тоді як API забезпечував би програмний інтерфейс для доступу до даних, що знаходяться безпосередньо на платформі TikTok.

3.2. Використання фреймворку Scrapy

На даному етапі роботи чітко окреслено вимоги до розробки програмного забезпечення, що підлягають задоволенню. Наступним кроком є визначення методів реалізації цих вимог. Слід зазначити, що в галузі комп'ютерних наук рідко виникає необхідність у розробці рішень "з нуля",

оскільки зазвичай доступні інструменти, придатні для вирішення подібних задач. Однак, необхідним є розуміння цих інструментів та їх обґрунтоване застосування для забезпечення ефективності розробленого рішення.

У галузі комп'ютерних наук існує багато бібліотек для веб-скрапінгу, таких як Selenium і Beautiful Soup, кожна з яких має свої переваги та недоліки для різних цілей. Для реалізації даного дослідження було обрано веб-скрапінгову платформу Scrapy. У цьому підрозділі представлено обґрунтування вибору Scrapy для вирішення поставленої задачі та проведено порівняльний аналіз з іншими популярними інструментами веб-скрапінгу, зокрема Beautiful Soup та Selenium. Вибір було здійснено, виходячи з його відповідності вимогам даного проекту, а не на підставі претензій на його абсолютну перевагу над іншими інструментами.

Scrapy — це веб-скрапінгова платформа, що була розроблена спеціально для Python. Це зумовило вибір Python як основної мови програмування для даного проекту. Сильна інтеграція Python з іншими інструментами та його широке застосування, особливо в наукових контекстах, додатково обґрунтувало його використання. Крім того, відносна простота використання Python зробила його придатним для реалізації у рамках короткострокового проекту, яким є бакалаврською роботою.

Як програмна платформа, Scrapy надає чіткий набір контекстів та методологій для розробників. Це відрізняє її від програмної бібліотеки, яка представляє собою сукупність готового коду, що може бути повторно використаний в інших програмах або проектах для спрощення розробки та зменшення надмірності. На відміну від бібліотеки, програмна платформа надає розробнику керівні принципи дизайну та архітектури, зменшуючи ймовірність помилок та підвищуючи якість коду.

Після ініціалізації проекту Scrapy автоматично готує структуру програми з основними компонентами, необхідними для функцій скрапінгу та краулінгу. Кожен з цих компонентів представлений у вигляді класу Python.

Найважливіші з цих компонентів - це павуки (spiders), елементи (items), конвеєри (pipelines), проміжне програмне забезпечення (middleware) та налаштування (settings). Після розуміння призначення кожного з цих компонентів та їх взаємодії в процесі краулінгу, формується цілісне розуміння принципу роботи Scrapy. Відтак, далі буде розглянуто кожен з цих компонентів окремо, після чого буде представлено теоретичний опис процесу скрапінгу для ілюстрації принципу роботи Scrapy.

Розпочнемо з розгляду павуків (Spiders). Павуки є осердям проекту Scrapy, оскільки вони відповідають за реалізацію основної функціональності скрапінгу. Існує кілька типів павуків для різних цілей, проте всі вони успадковують базовий клас павука. Назва "павук" відображає його функцію "сканування" веб-простору шляхом переходу за виявленими посиланнями. Павук завжди має назву та список URL-адрес для початку процесу скрапінгу. Ці URL-адреси можуть бути налаштовані для передачі як аргумент до павука. Опціонально павук також може мати список дозволених доменів для скрапінгу. Крім того, павук повинен реалізовувати метод з назвою start_requests. Цей метод має повертати об'єкт Request і визначає, як починається процес скрапінгу. Відповідь на ці запити за замовчуванням надсилається назад до методу parse всередині павука. Проте, можна вказати інші методи для цієї мети, які можуть приймати додаткові параметри.

У методі парсингу можлива генерація нових запитів на основі отриманих відповідей, а зібрані дані можуть бути оброблені відповідно до потреб дослідження. Проте, для підвищення підтримуваності та читабельності коду, рекомендовано зберігати дані у структурах, що називаються елементами (Items). За структурою елементи подібні до стандартних словників Python, проте їх структура має бути попередньо визначена. Елементи можуть бути концептуалізовані як контейнери для даних, структура яких відома Scrapy до початку процесу скрапінгу. Методи парсингу повертають елементи, використовуючи оператор yield. Це дозволяє

генерувати послідовність елементів протягом виконання методу, ефективно обробляючи кожний зібраний об'єкт окремо. Використання yield створює генератор, який послідовно надсилає отримані елементи для подальшої обробки конвеєрами.

Конвеєри елементів (Item Pipelines) активуються після збору даних методом парсингу та їх збереження в елементах. Для подальшої обробки даних можливе визначення класу конвеєра та специфікація методів для обробки елементів, які приймають елемент як вхідні дані. Під час виконання процесу скрапінгу Scrapy автоматично передає згенеровані методом парсингу елементи до визначених методів класу конвеєра. Це забезпечує можливість очищення, валідації даних та їх збереження у базі даних або іншому форматі.

Таким чином, Scrapy надає чіткі рекомендації щодо організації процесу збору даних. Проте значна частина цього процесу підлягає налаштуванню через файл налаштувань. Ці налаштування доступні всім компонентам, що беруть участь у процесі краулінгу. Таким чином, можна модифікувати численні параметри та специфікувати хід процесу краулінгу.

Для подальшого налаштування процесу скрапінгу передбачено використання класів проміжного програмного забезпечення (Middleware). Ці класи можуть бути використані для модифікації запитів та відповідей у процесі їх обробки спайдером, дозволяючи реалізувати додаткову логіку на різних етапах виконання. Наприклад, проміжне ПЗ може бути використане для перевірки на дублікати, зміни User-Agent запитів або реалізації ротації проксі. Шляхом специфікації порядку виконання класів проміжного ПЗ можливо побудувати складні ланцюжки обробки, що здійснюють трансформацію даних. У рамках даного проекту проміжне ПЗ було застосовано для перевірки на дублювання запитів шляхом верифікації наявності запитуваного URL у базі даних до початку процесу скрапінгу.

Важливо відзначити, що представлений опис Scrapy не претендує на вичерпне представлення його архітектури. Метою є висвітлення логіки,

необхідної для розуміння процесу реалізації павука, а не детальний опис кожного аспекту фреймворку. Хоча класи, згадані в цьому тексті, є компонентами, з якими розробник найімовірніше взаємодіятиме при побудові Scrapy, існують й інші важливі компоненти, які беруть участь у процесі краулінгу, але можуть бути не повністю доступні користувачеві. У даному проекті ці компоненти не використовувалися безпосередньо розробником. Рисунок 3.1 ілюструє архітектуру Scrapy, що включає Engine (двигун), який є центральним компонентом фреймворку. Хоча Engine відповідає за координацію процесу, для розуміння принципу роботи достатньо концептуалізувати взаємодію компонентів, описаних у даному підрозділі.

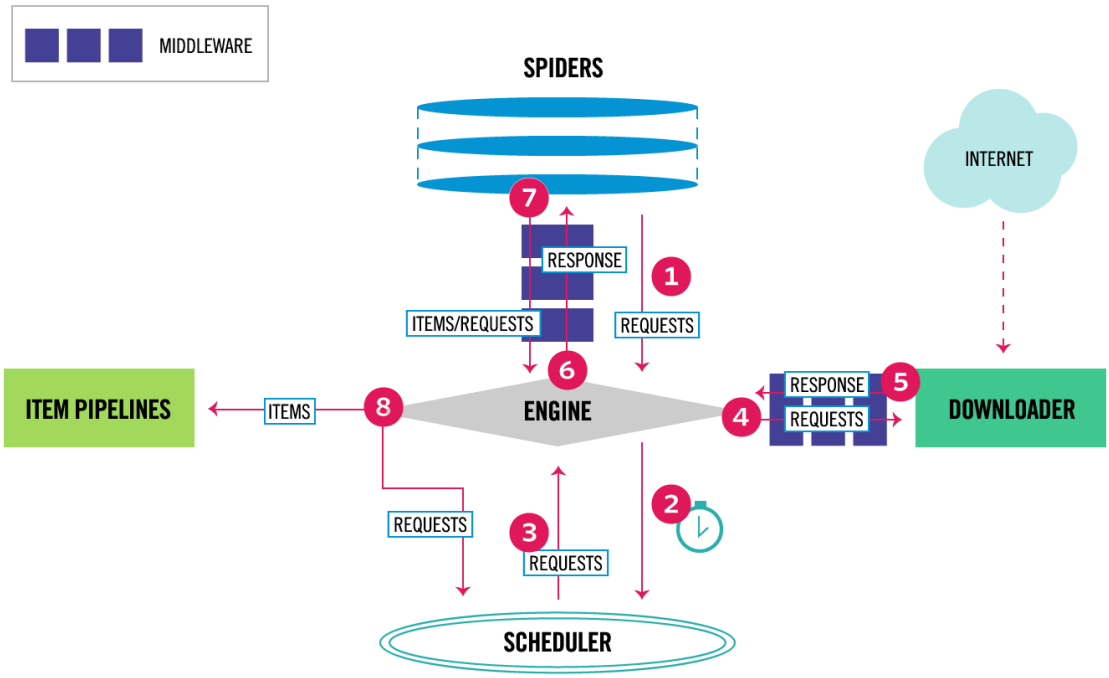


Рисунок 3.1 - Архітектура Scrapy

Після представлення логіки Scrapy, далі буде викладено обґрунтування його вибору порівняно з іншими популярними інструментами веб-скрапінгу, зокрема Beautiful Soup та Selenium.

Beautiful Soup є бібліотекою для парсингу, а не спеціалізованим інструментом веб-скрапінгу. Це робить її придатною для вилучення інформації з файлів після їх отримання. Вона демонструє високу ефективність для цієї задачі, а її простота та легкість використання роблять її оптимальним вибором для невеликих проектів веб-скрапінгу. Наприклад, у випадках, коли місцезнаходження необхідних даних відоме і вони розміщені в межах одного домену, розробка повномасштабного проекту веб-скрапінгу із застосуванням таких інструментів, як Scrapy, може бути надмірною. Натомість, Beautiful Soup може бути ефективно застосована для оперативного отримання необхідних даних.

Selenium є програмним забезпеченням, що дозволяє симулювати взаємодії користувача з веб-браузером. Незважаючи на те, що спочатку Selenium був розроблений для автоматизованого тестування веб-застосунків, він набув популярності як інструмент для веб-скрапінгу, зокрема завдяки здатності виконувати та обробляти вміст JavaScript. Однак, підхід Selenium, що базується на симуляції браузера, може бути менш ефективним для великомасштабних проектів веб-скрапінгу порівняно з оптимізованими методами Scrapy.

Scrapy відрізняється від Selenium та Beautiful Soup тим, що він спеціально призначений для обробки великомасштабних проектів веб-скрапінгу. Платформа є високорозширюваною, що дозволяє розробникам створювати адаптовані веб-скрейпери відповідно до специфічних потреб. Крім того, Scrapy оптимізований для ефективності завдяки використанню асинхронних запитів та паралельної обробки даних, що сприяє прискоренню процесу скрапінгу.

Таблиця 3.1 узагальнює зазначені характеристики та надає додаткову аргументацію на користь вибору Scrapy для реалізації даного проекту збору даних.

Таблиця 3.1 - Порівняння Scrapy, Beautiful Soup та Selenium

	Scrapy	**Beautiful Soup**	**Selenium**
Веб-краулінг	Так	Ні	Так
Парсинг даних	Так	Так	Так
Зберігання даних	Так	Ні	Так
Асинхронність	Так	Ні	Ні
Рендеринг JavaScript	З зовнішніми бібліотеками	Ні	Так
Селектори	CSS, XPath	CSS	CSS, XPath
Проксі	Так	З зовнішніми бібліотеками	Так
Продуктивність	Швидка	Середня	Повільна
Розширюваність	Висока	Обмежена	Обмежена
Крива навчання	Крута	Легка	Крута
Використання	Постійні великомасштабні проекти скрапінгу	Невеликі та середні проекти скрапінгу	Невеликі та середні проекти скрапінгу, які вимагають JavaScript

3.3. Опис графової бази даних Neo4j та Docker

Аналіз, представлений у другому розділі, привів до висновку, що найбільш корисною формою представлення даних для соціальних досліджень є графовий формат. Відповідно до цього висновку було сформульовано теоретичну модель даних. Проте, постає питання практичної реалізації отримання даних у графовій формі. Одним із потенційних варіантів є ручне побудова графа за допомогою традиційних інструментів після збору необхідної інформації за допомогою Scrapy. Однак цей метод є вкрай трудомістким та неефективним для значних обсягів даних. Альтернативним програмним рішенням є використання інструментів, подібних до Gephi, які здатні імпортувати дані у форматі CSV-файлу та візуалізувати їх як граф, дозволяючи подальший аналіз із застосуванням графових алгоритмів. Однак Gephi висуває жорсткі вимоги до коректності формату вхідних даних, і зберігання даних у CSV-файлі може бути неоптимальним підходом для прямого графового аналізу.

Найбільш поширеними системами керування базами даних є реляційні бази даних. Проте, зберігання даних у реляційній формі все одно потребуватиме трансформації даних до графового формату для кожного аналізу.

Ефективним рішенням для зберігання та обробки даних у форматі графа є застосування спеціалізованих систем керування графовими базами даних. Однією з таких є Neo4j. Neo4j — це потужна система управління графовими базами даних, що пропонує широкий спектр функцій та можливостей. Її ключова функціональність як нативної графової бази даних доповнюється ефективною інтеграцією з різними мовами програмування та вичерпною документацією. Програмне забезпечення включає зручний настільний інтерфейс користувача, який полегшує дослідження графів за допомогою інтуїтивно зрозумілої мови запитів Cypher або через графічний інтерфейс. Крім того, Neo4j надає вбудовану підтримку для різних застосунків, призначених для маніпуляції, аналізу, дослідження та візуалізації збереженого графа.

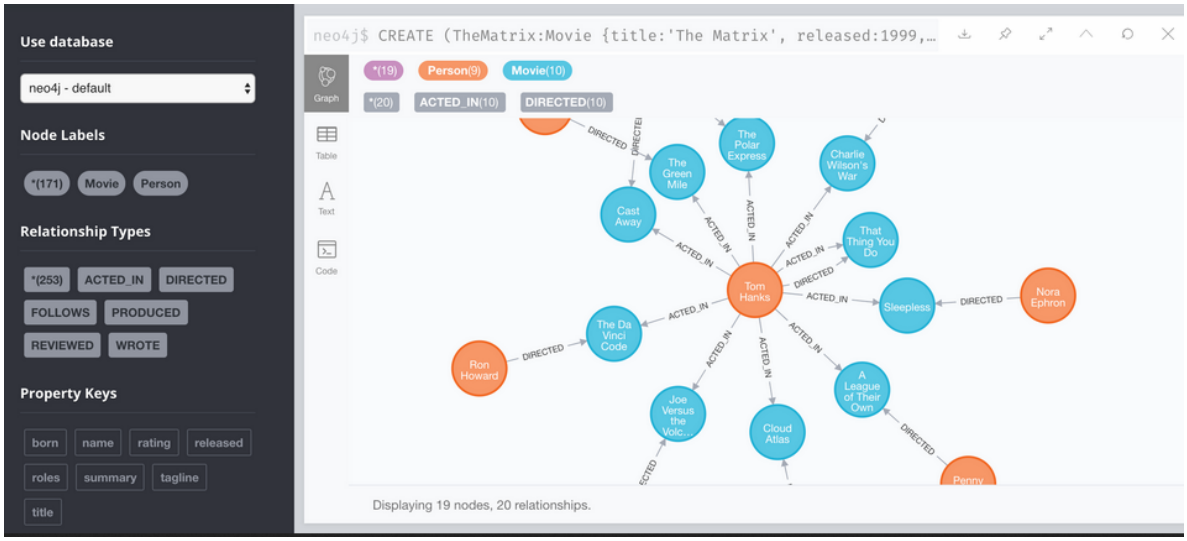


Рисунок 3.2 – Інтерфейс графової БД Neo4j

У рамках даної роботи активно використовувалися два застосунки в середовищі Neo4j: стандартний інтерфейс користувача Neo4j Desktop та

застосунок для візуалізації GraphXR. Інтерфейс користувача переважно використовувався під час розробки для маніпуляцій з графом. GraphXR, у свою чергу, є потужною платформою для дослідження та візуалізації даних, яка використовує графові представлення для сприяння аналізу складних наборів даних. Вона пропонує інтерактивне дослідження даних, аналіз мереж та візуалізацію графів, що допомагає виявляти зв'язки та закономірності в даних. Завдяки наявності плагіну в Neo4j Desktop, забезпечується можливість безпосереднього відкриття даних в інтерфейсі GraphXR та проведення аналізу без необхідності додаткових трансформацій.

На початку цього підрозділу було розглянуто використання Neo4j як графової бази даних для зберігання та візуалізації даних, зібраних для соціальних досліджень. Neo4j надає набір потужних інструментів для дослідження та аналізу даних у форматі графа, що є особливо цінним для аналізу соціальних мереж. Проте, з метою забезпечення легкої доступності інструменту веб-скрапінгу для інших дослідників, необхідно врахувати аспекти його зручності використання та розповсюдження. Для вирішення цього завдання застосовано технологію контейнеризації Docker. Docker дозволяє легко пакувати, розгортати та розповсюджувати інструмент веб-скрапінгу, забезпечуючи послідовність та відтворюваність на різних середовищах, а також ефективне управління та співпрацю під час розробки інструменту.

Docker — це платформа з відкритим вихідним кодом, яка дозволяє розробникам пакувати, розгортати та запускати застосунки в контейнерах. Контейнери є легкими, портативними та самодостатніми середовищами, які інкапсулюють усі компоненти, необхідні застосунку для функціонування, включаючи системні інструменти, бібліотеки та конфігурації. Вони ізольовані від гост-системи, що забезпечує послідовність та відтворюваність виконання застосунку на різних середовищах.

Docker спрощує процес створення, розгортання та запуску застосунків за допомогою контейнеризації. Розробники можуть створити застосунок на своєму локальному комп'ютері, упакувати його в контейнер, а потім запустити на будь-якому іншому комп'ютері з встановленим Docker, гарантуючи ідентичне середовище виконання незалежно від базової інфраструктури.

У випадку даної роботи, Docker дозволив упакувати веб-скрапер та його необхідні залежності (пакети Python, бібліотеки, конфігурації) в окремий контейнер. Одночасно базу даних Neo4j було запущено в іншому контейнері, і ці два контейнери були зв'язані між собою. Таким чином, контейнеризація дозволяє запускати весь процес веб-скрапінгу на будь-якій системі за допомогою однієї команди. Цей контейнеризований варіант веб-скрейпера може бути легко розгорнутий та запущений на будь-якій машині з встановленим Docker. Крім того, Docker спрощує управління контейнеризованим веб-скрапером та надає централізовані репозиторії для обміну програмними рішеннями.

3.4. Алгоритмічна імплементація

У цьому підрозділі детальніше висвітлено процес збору даних на алгоритмічному рівні.

Реалізовано спайдер Scrapy, який приймає URL-адресу платформи TikTok як аргумент при ініціалізації. Починаючи з вказаної URL-адреси, спайдер виконує запит HTML-файлу з відповідного сервера. З отриманого HTML-файлу спайдер ідентифікує 15 відео. Для кожної зі сторінок ідентифікованих відео, спайдер вилучає URL-адреси пов'язаних сторінок: сторінок хештегів, сторінок музичних треків та сторінки користувача, який опублікував відео. Обхід за цими URL-адресами здійснюється рекурсивно. Усі сторінки відео, виявлені в ході цього процесу, передаються функції

парсингу для вилучення відповідних даних. Алгоритм скрапінгу узагальнено у лістингу 3.1, тоді як таблиця 3.2 демонструє всі типи витягнутих даних.

Лістинг 3.1. Псевдокод алгоритму скрапінгу

```
def collectURLs(url) -> void:
    htmlresponse = htmlrequest(url)
    responseUrls = htmlresponse.getAllSignificantURLs(htmlresponse)
    for respUrl in responseUrls:
        if isVideoUrl(respUrl):
            parse(respUrl)
        collectURLs(respUrl)
```

Таблиця 3.2 – Опис усіх змінних, зібраних для кожного відео, витягнутих у функції парсингу

Змінна	Тип	Опис
videoUrl	String	URL-адреса, яка відповідає відео та використовується як ідентифікатор
videoDescription	String	Опис, який відповідає відео
user	String	Унікальний ідентифікатор для кожного користувача
UserScreenname	String	Альтернатива імені користувача
nrComments	Integer	Кількість коментарів під відповідним відео
nrLikes	Integer	Кількість вподобань під відповідним відео
nrForwarded	Integer	Кількість разів, які відео було переслано
hashtags	Список рядків	Хештеги, які присутні в відео
music	String	Назва музики, присутньої в відео
date	String	Вказує дату розміщення відео
mentionedUsers	Список рядків	Вказує інших користувачів, згаданих у пості

Усі дані, отримані після парсингу, інкапсулюються в елементи (items), після чого передаються до конвеєра елементів (Item Pipeline). У рамках цього конвеєра здійснюється завантаження даних до бази даних Neo4j. Змінні, що представляють URL відео (videoURL), музичний трек (music), ім'я користувача (userScreenname) та кожен хештег, розглядаються як ідентифікатори вузлів графа. Вони зв'язуються між собою згідно з логікою,

запропонованою теоретичною моделлю, і далі зберігаються у базі даних. Крім того, до кожного вузла додається атрибут часової мітки, що фіксує момент його додавання до бази даних. Інші зібрані змінні зберігаються як атрибути відповідних вузлів. Результуюча структура даних узагальнена у таблиці 3.3.

Таблиця 3.3 – Опис даних, як вони надсилаються до бази даних у конвеєрі

Змінна	**Роль**
Post	Вузол
VideoURL	Атрибут і ідентифікатор поста
nrComments	Атрибут поста
nrLikes	Атрибут поста
nrForwarded	Атрибут поста
date	Атрибут поста
created 99	Атрибут поста
INCLUDES	Ребро від поста до хештегу або музики
MENTIONS	Ребро від поста до користувача
User	Вузол
UserScreenname	Атрибут і ідентифікатор імені користувача
username	Атрибут користувача
created	Атрибут користувача
POSTED	Ребро від користувача до поста
Hashtag	Вузол
created	Атрибут хештегу
Music	Вузол
created	Атрибут музики

3.5. Оцінка ефективності інструменту на прикладі скрапінгу

В попередньому підрозділі пояснювалися компоненти та алгоритмічна реалізація функції скрапінгу. У цьому розділі буде проведено аналіз часу виконання, щоб дати дослідникам орієнтир для оцінки обчислювальних

витрат на збір бажаних даних. Крім того, буде проведено описовий аналіз, щоб показати, як можна використовувати цей інструмент.

У цьому підрозділі буде оцінено продуктивність інструменту на прикладі скрапінгу. Спочатку буде вказано тривалість часу, протягом якого збиралися дані, та надано метаінформацію про зібрані дані, а потім використано мітки часу даних в базі даних для оцінки продуктивності інструменту. Також буде оцінено проблеми, які виникли під час збору.

Перший скрапінг розпочався з хештегу lgbt. Скрапер був запущений 6 травня 2025 року о 19:00 і зупинений вручну 7 травня 2025 року о 22:00, що призвело до загальної тривалості скрапінгу 27 годин. За цей час було зібрано загалом 4980 елементів. Рисунки 3.3 і 3.4 показують кількість зібраних елементів за час під час збору.

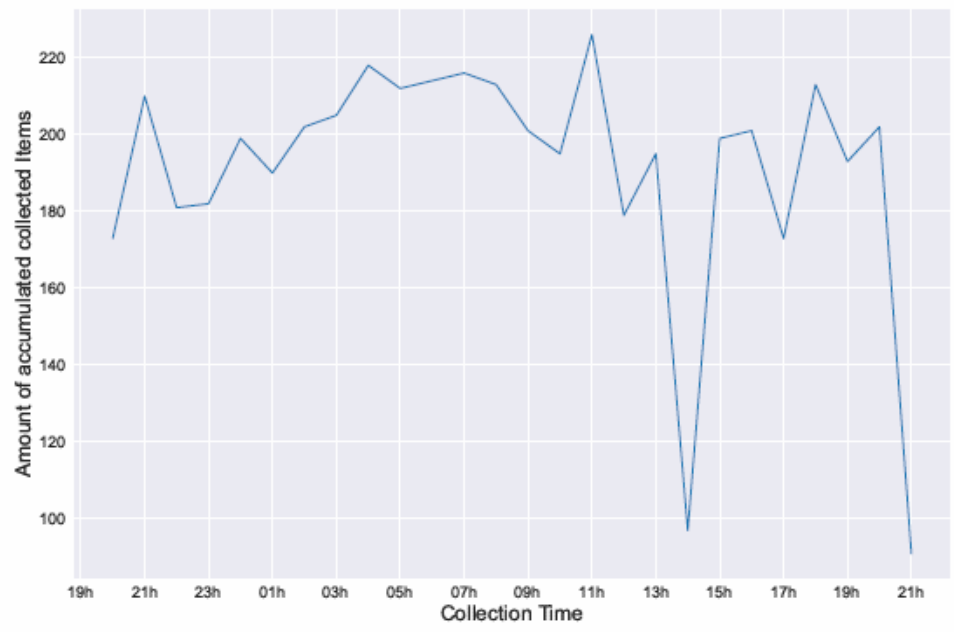


Рисунок 3.3 - Кількість зібраних постів за годину

Рисунки 3.3 і 3.4 демонструють, що збір даних, здається, є досить послідовним. Хоча спостерігаються деякі коливання в кількості зібраних даних, як показано на рис. 3.4, загальна тенденція є лінійною, із середнім

значенням 191 зібраних елементів за годину.

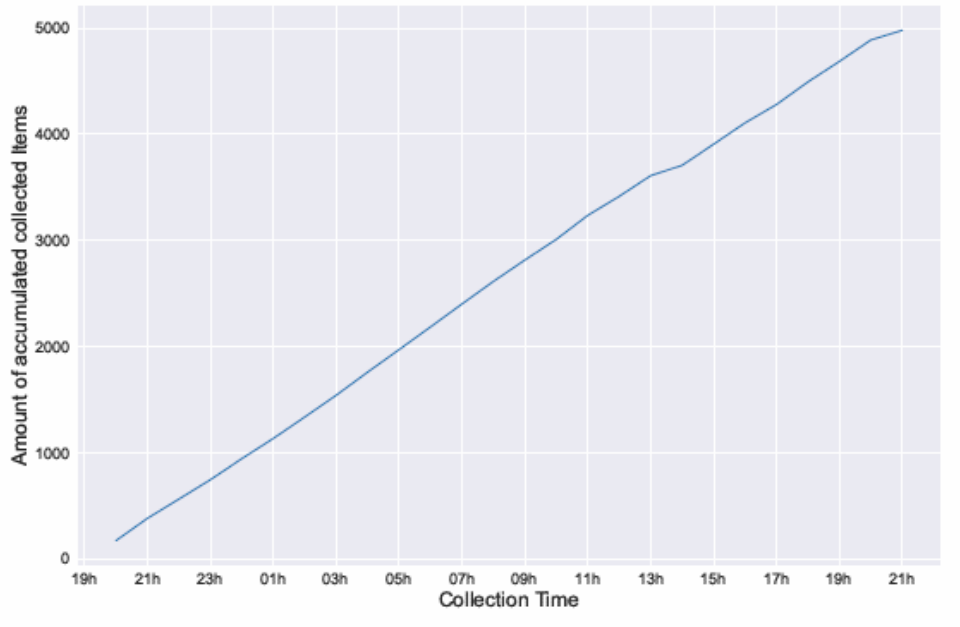


Рисунок 3.4 - Накопичена кількість зібраних постів з часом

Однак були деякі очевидні перешкоди на 14-й годині, які, можливо, були спричинені поганим інтернет-з'єднанням або проблемами з сервером TikTok. Якщо не брати до уваги аномалії на 14-й та 21-й годинах, середня кількість зібраних елементів за годину зростає до 203.

3.6. Аналіз зібраних даних з веб-ресурсу соціального нетворкінгу

У цьому підрозділі представлено результати описового аналізу зібраних даних. Описовий аналіз отриманих даних методом скрапінгу переслідує подвійне призначення. По-перше, отримані загальні метрики можуть розглядатися як наближені оцінки реальних характеристик, притаманних платформі TikTok. По-друге, аналіз спрямований на демонстрацію можливостей дослідження даних та ілюстрацію практичної реалізації теоретичної моделі, сформульованої у другому розділі. Рисунок 3.5

ілюструє загальну структуру даних, що точно відповідає теоретичній моделі, сформульованій у другому розділі.

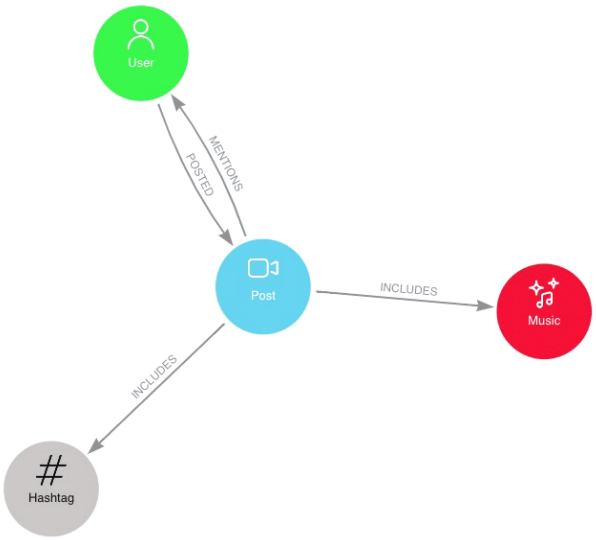


Рисунок 3.5 – Структура зібраних даних

Загалом, зібраний граф налічував 57 997 вузлів та 92 071 ребро. На рисунках 3.6 та 3.7 представлено розподіл цих властивостей (кількості вузлів та ребер) за різними типами вузлів.

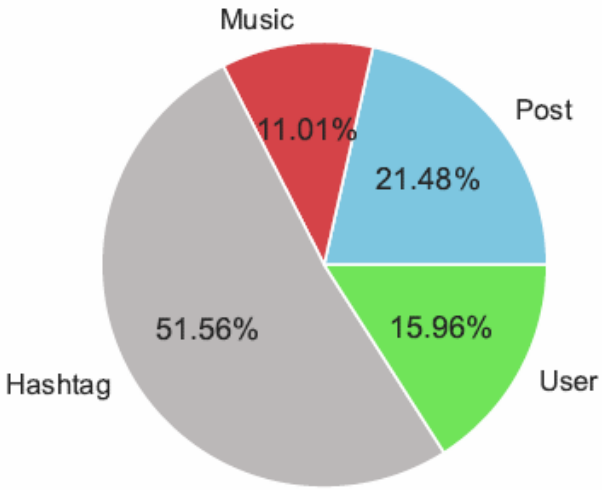


Рисунок 3.6 - Розподіл вузлів за типом

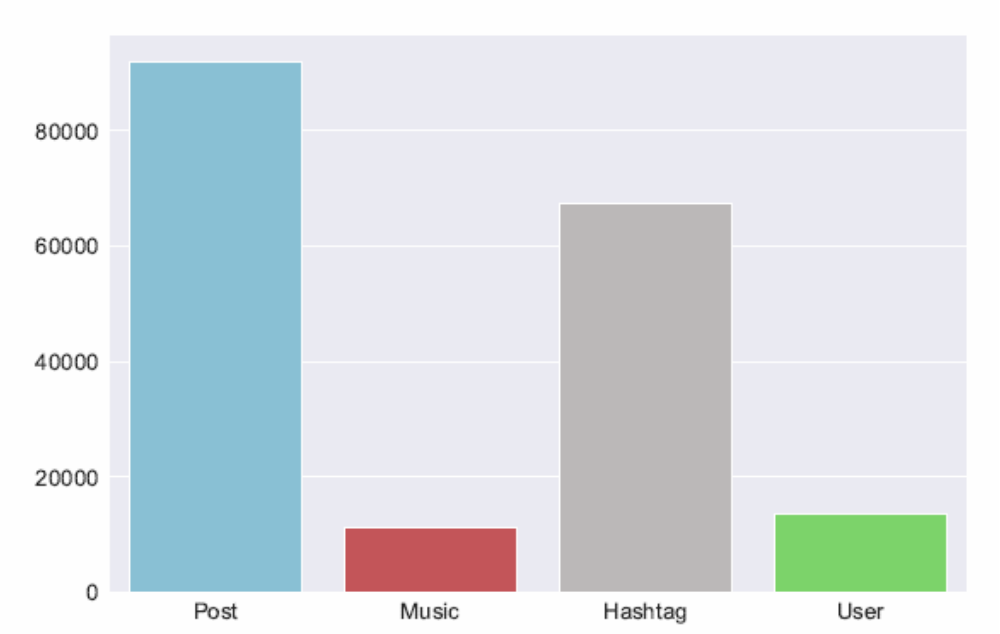


Рисунок 3.7 - Сумарний ступінь за типом вузла

Як видно з рисунків 3.6 та 3.7, більшість вузлів у зібраному наборі даних становлять хештеги. Проте, пости демонструють пропорційно вищу кількість ребер. Це може бути пояснено тим, що кожен пост у використовуваній схемі даних автоматично має три вихідні ребра (зв'язок з автором, музичним треком та хештегами). Висока відносна кількість ребер, пов'язаних із хештегами, також є очікуваною, оскільки кожен пост може містити множину хештегів, тоді як асоціюється лише з одним автором та одним музичним треком. Для детальнішого розуміння подано рисунок 3.8, що ілюструє порівняння середнього ступеня для кожного типу вузла.

Середній ступінь вузла типу 'пост' становить трохи більше семи. Це можна пояснити відносно низькою частотою згадувань інших користувачів у постах та тим фактом, що кожен пост пов'язаний лише з одним музичним треком. На основі цього можна зробити висновок, що кожен пост містить в середньому близько п'яти хештегів. Розподіл ступенів за типом вузла, представлений на рисунку 3.9, надає додаткове розуміння формування цієї кількості ребер.

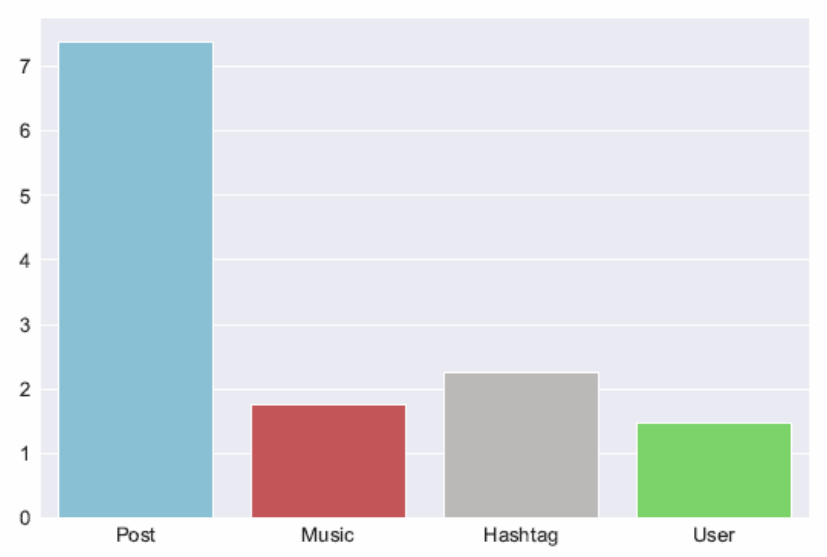


Рисунок 3.8 - Середній ступінь за типом вузла

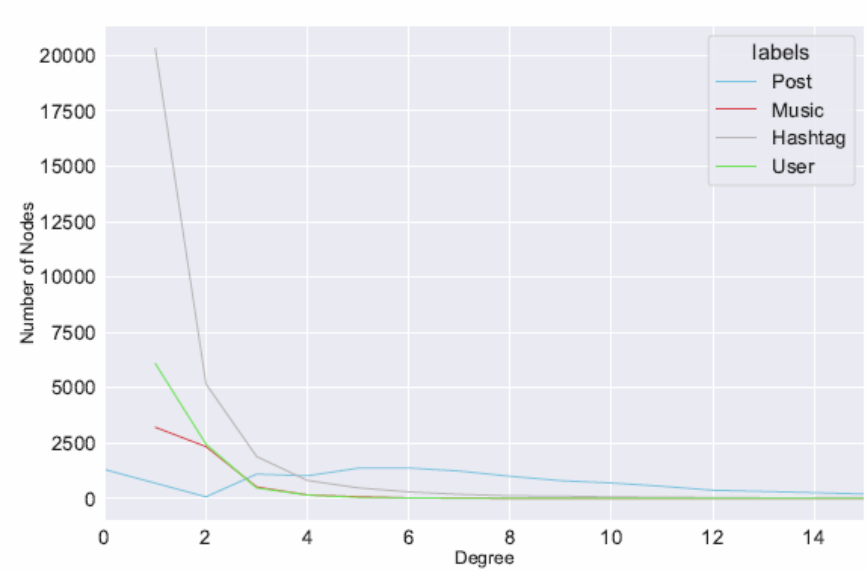


Рисунок 3.9 - Розподіл ступенів за типом вузла

Аналіз розподілу ступенів вузлів надає цінні інсайти щодо структури даних. Спостерігається несподіваний пік на нульовому значенні в розподілі ступенів для вузлів типу 'пост', що суперечить очікуванням, згідно з якими кожен пост повинен мати зв'язки з іншими вузлами (автор, музичний трек, хештеги). Наступний пік спостерігається на значенні шість, після чого крива поступово спадає, прямуючи до нуля для вищих значень ступеня. Особливий інтерес викликає розподіл ступенів для трьох інших типів вузлів

(користувачі, хештеги, музичні треки), який візуально нагадує розподіл Пуассона. Це вказує на наявність невеликої кількості вузлів з високим ступенем та значної кількості вузлів з низьким ступенем. Незважаючи на те, що вісь абсцис на рисунку обмежена значенням ступеня 14, у зібраному наборі даних наявні вузли зі значно вищими значеннями ступеня. Ілюстрацію цього можна побачити на рисунку 3.10, де представлено максимальний ступінь для кожного типу вузла.

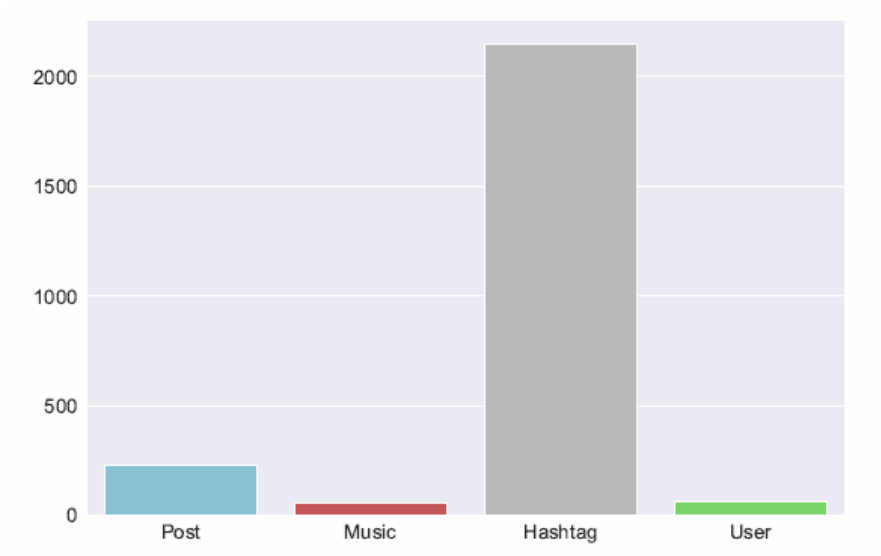


Рисунок 3.10 - Максимальний ступінь за типом вузла

Отримавши початкове уявлення про загальні властивості графа, можна перейти до його детального дослідження. Як зазначалося раніше, для візуалізації та аналізу даних у базі даних Neo4j використовувалося програмне забезпечення GraphXR, що підтримує пряму інтеграцію з Neo4j. GraphXR — це програмне забезпечення або платформа, призначена для інтерактивної візуалізації та аналізу графових даних.

По суті, GraphXR допомагає користувачам працювати зі складними, взаємопов'язаними даними (представленими у вигляді графа з вузлами та ребрами) і отримувати з них цінні інсайти шляхом їх візуального

представлення та дослідження. Рисунок 3.11 демонструє візуалізацію повного графа.

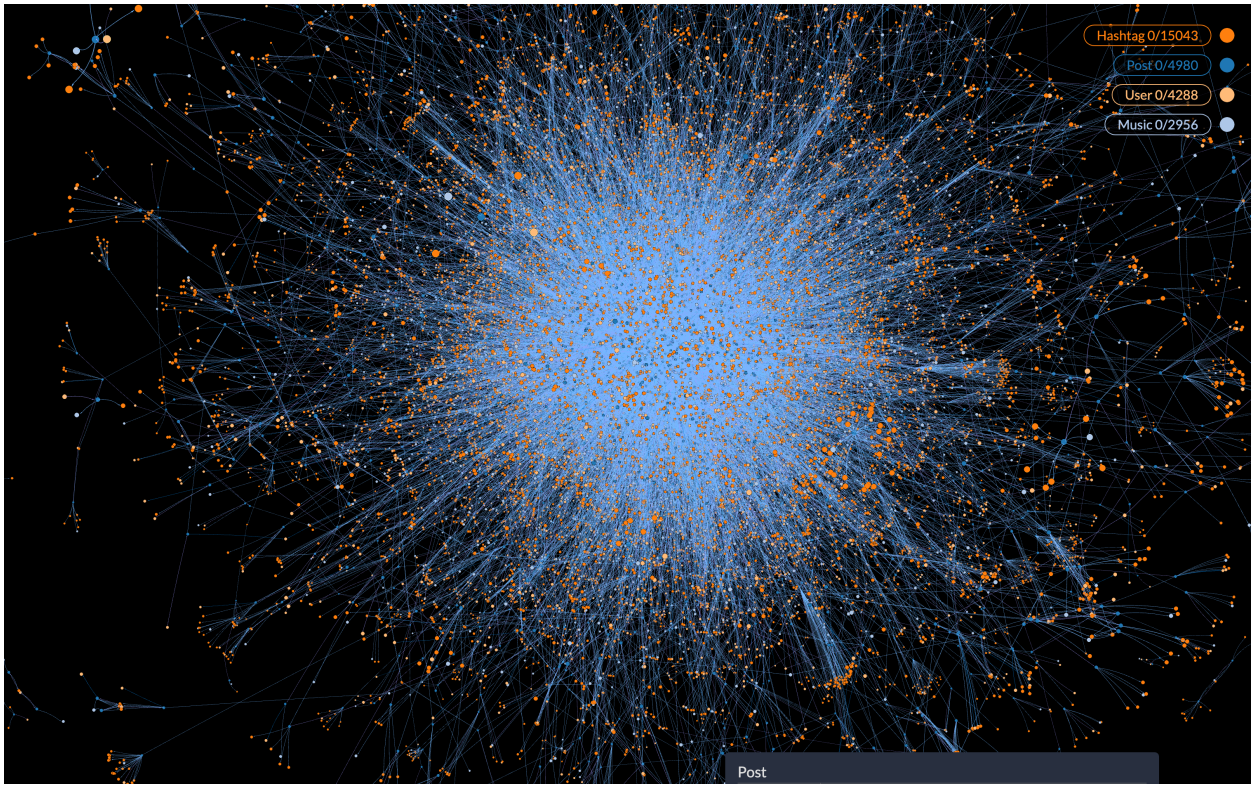


Рисунок 3.11 - Візуалізація графа зібраних даних

Використання GraphXR надає можливість ефективно модифікувати структуру та атрибути графа. Також можливе застосування графових алгоритмів до даних з метою генерації нових метрик, таких як центральність або приналежність до спільноти. Додатково, реалізована функціональність фільтрації вузлів на основі їх атрибутів для контролю візуалізованих елементів. Особливо цінною є можливість швидкого перетворення графа у різні формати візуалізації. Наприклад, рисунок 3.12 ілюструє один з форматів візуалізації, що ефективно демонструє структуру даних.

Як видно з візуалізації та розподілу ступенів, у центрі графа розташована невелика кількість вузлів з високим ступенем зв'язності, що формує високу щільність ребер у центральній частині. Натомість, більшість вузлів розміщені на периферії графа (на зовнішньому колі візуалізації).

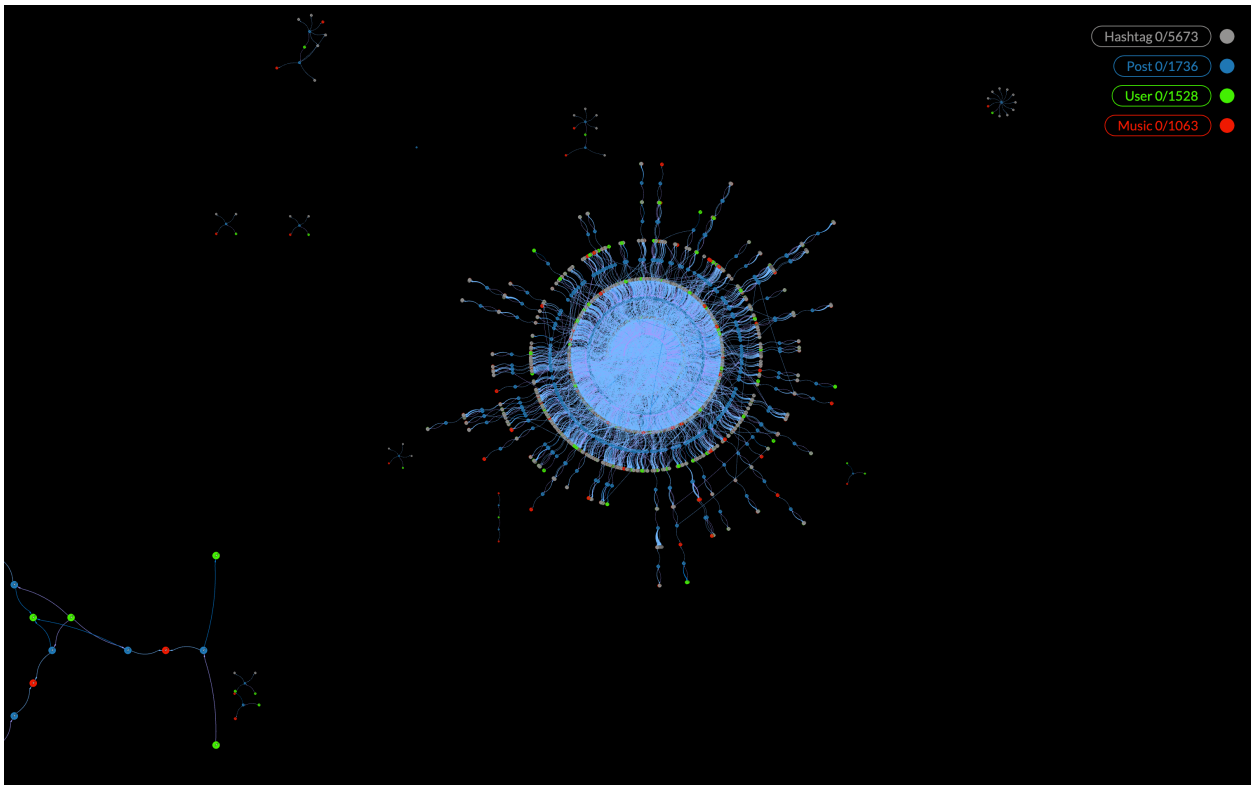
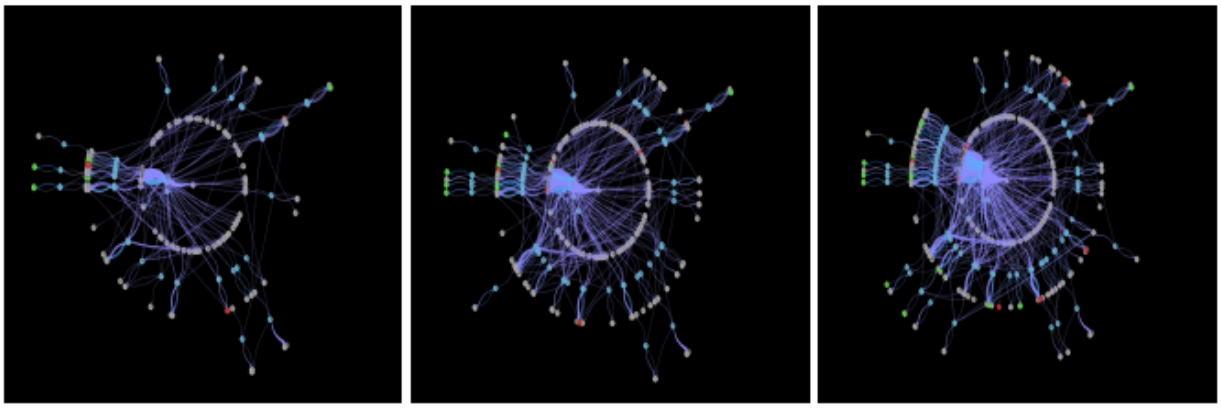


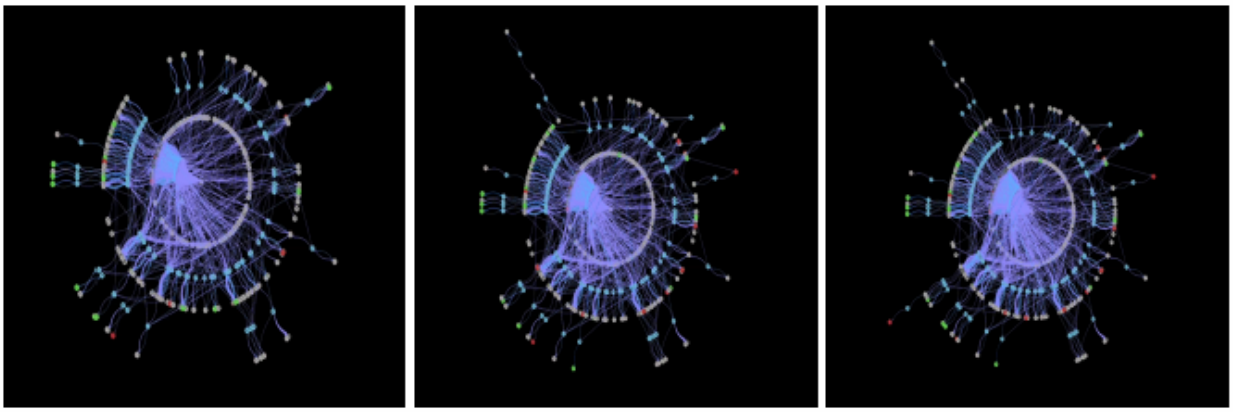
Рисунок 3.12 - Візуалізація графа зібраних даних у вигляді кільця

З метою дослідження впливу алгоритму скрапінгу на формування структури графа, було застосовано фільтрацію за часом збору даних. Це дозволило візуалізувати вузли поступово, відповідно до часу їх додавання до бази даних. Рисунок 3.13 ілюструє цей процес візуалізації.



а) б) в)

Рисунок 3.13 - Візуалізація процесу скрапінгу (початок)



г) д) е)

Рисунок 3.13 - Візуалізація процесу скрапінгу (закінчення)

Отже, перша ціль даної роботи полягала у розробці теоретичної моделі TikTok як соціального середовища, доступної для дослідників як у сфері соціальних наук, так і комп'ютерних наук. У другому розділі було представлено теоретичну модель, що базується на теоретичних та методологічних засадах соціальних наук, а також враховує доступність даних платформи TikTok. Таким чином, першу ціль було досягнуто.

Друга ціль даної роботи полягала в інтеграції розробленого інструменту у теоретичну модель, яка б узгоджувалася з поширеними методами та теоріями в соціальних науках, одночасно відображаючи строгість та формалізм комп'ютерних наук. Як продемонстровано у третьому розділі розроблена програма забезпечує трансформацію зібраних даних безпосередньо у структуру, що відповідає моделі, представленій в другому розділі. Це дозволяє інтерпретувати дані в рамках, визначених моделлю, тим самим сприяючи досягненню другої цілі.

Третя ціль даної роботи полягала в максимальному підвищенні доступності та зручності використання розробленого інструменту веб-скрапінгу. У третьому розділі представлено опис основних компонентів, використаних при розробці інструменту. Крім того, пакування інструменту за допомогою Docker забезпечує його використання без необхідності спеціалізованих знань у програмуванні чи встановлення додаткових

залежностей. Додатково, інтеграція інструменту з базою даних Neo4j дозволяє користувачам отримувати доступ до даних через низку графових застосунків, підтримуваних Neo4j Desktop, що не вимагає наявності знань у програмуванні. Проте, за наявності відповідних знань, доступ до даних також може здійснюватися за допомогою запитів. Також продемонстровано використання GraphXR для дослідження даних та представлено загальні описові статистичні дані, які можуть бути отримані та слугувати відправною точкою для подальшого аналізу.

Аналогічним проектом до даної роботи є неофіційний API TikTok, доступний на платформі GitHub [39]. На початковому етапі даної дипломної роботи було проведено оцінку цього інструменту, проте виявлено низку проблем, про які також повідомлялося спільнотою проекту і які залишалися невирішеними на дискусійному сервері. Відтоді ініціатором проекту було випущено нову версію, спрямовану на усунення виявлених проблем.

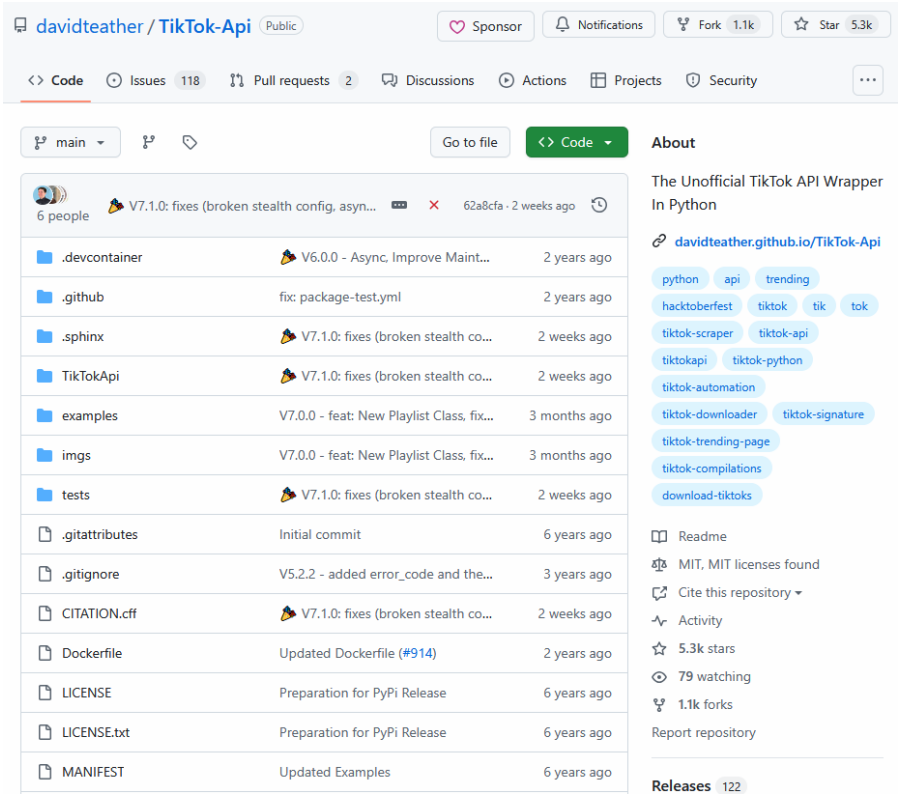


Рисунок 3.14 - Неофіційний API TikTok, доступний на платформі GitHub

Зазначений проект базується на Selenium, що обумовлює як його переваги, так і недоліки. Загалом, програма надає можливість дослідникам здійснювати скрапінг даних за заданим запитом. Однак, вона не пропонує інтегрованої структури даних, що може вимагати додаткової роботи для дослідників залежно від їхніх цілей. Таким чином, незважаючи на вирішення однієї і тієї ж проблеми, підходи обох проектів суттєво відрізняються. Як випливає з назви,

Неофіційний API TikTok функціонує як API, тоді як програма, запропонована в даній роботі, є веб-скрапером з інтегрованою базою даних. Якщо неофіційний API TikTok забезпечує дослідникам можливість оперативного доступу до конкретних даних, то проект, представлений у даній роботі, пропонує рамки для систематичного скрапінгу даних у форматі, що відображає структуру платформи. Потенційно доцільним є інтеграція функціональності неофіційного API TikTok у розроблений інструмент з метою збагачення зібраних даних додатковою інформацією.

Дана робота робить внесок у наукову область зокрема шляхом представлення формалізованої перспективи соціальних медіа у контексті соціальних феноменів. Це може сприяти мотивуванню дослідників використовувати дані з платформи TikTok для вирішення своїх дослідницьких завдань. Строге визначення формальної моделі може сприяти встановленню спільної термінології для досліджень TikTok та полегшити комунікацію результатів між дослідниками.

Незважаючи на успішну розробку інструменту для збору даних з платформи TikTok та забезпечення їх доступності для соціальних досліджень, дана робота має певні обмеження, які необхідно відзначити.

По-перше, дана робота має експериментальний характер. Вона базується на досвіді інтеграції підходів соціальних наук та комп'ютерних наук, а також на розумінні складнощів, пов'язаних з перетином дисциплін, де концепції, очевидні в одній сфері, можуть бути незрозумілими в іншій.

Розроблений інструмент є спробою полегшити доступ до інструментарію і повинен розглядатися як прототип. Запропонований підхід базується на отриманому досвіді. Питання його реальної застосовності та корисності для розуміння даних у представленому форматі потребує подальшої емпіричної перевірки.

Крім того, важливо відзначити, що теоретична модель, розроблена в даній роботі, базується на суб'єктивному судженні щодо її потенційної корисності для наукової спільноти, виходячи з поточних теорій та методів у соціальних науках. Відповідно, необхідно визнати, що розроблена модель може не повною мірою відповідати реальній внутрішній моделі даних платформи TikTok. На жаль, подолання цього обмеження є складним, оскільки лише власник платформи TikTok має доступ до її реальної внутрішньої моделі даних. Ще одним обмеженням даної роботи є обмежений обсяг даних, який можливо зібрати з кожної сторінки. Оскільки Scrapy функціонує шляхом надсилання запитів та парсингу отриманих HTML-відповідей без використання проміжного віртуального браузера, було можливо зібрати дані лише з першої сторінки для кожної сутності, яка зазвичай містила лише 15 елементів. Це може призводити до отримання невеликого обсягу даних для певних контекстів, що ускладнює проведення глибокого аналізу дискурсу навколо конкретної теми. Крім того, коментарі до відео не були включені у дане дослідження, оскільки вони не були безпосередньо доступні для вилучення зі стандартних HTML-відповідей. Проте, зважаючи на те, що коментарі є важливою формою соціальної взаємодії, їх включення могло б бути корисним для аналізу.

Висновки до розділу

У третьому розділі було реалізовано програмну імплементацію техніки збору даних із соціального нетворкінгу на прикладі створення

повнофункціонального програмного інструменту. Детально описано процес розробки інтерфейсу застосунку, орієнтованого на зручність користувача та оптимізацію керування процесом збору даних.

Ключову роль у технічній реалізації відіграв фреймворк Scrapy, який дозволив створити гнучкий та масштабований механізм веб-скрапінгу. Було впроваджено модулі збору, обробки та зберігання даних, що забезпечують ефективне витягування інформації з цільових веб-ресурсів.

Для зберігання та аналізу складних взаємозв'язків між елементами даних застосовано графову базу даних Neo4j, розгорнуту у середовищі Docker для забезпечення портативності та ізольованості розробки. Такий підхід дозволив ефективно представляти структуру соціальних взаємодій.

Алгоритмічна частина включала розробку логіки збирання даних, побудови графа зв'язків та зберігання інформації у базі. Проведено оцінку ефективності інструменту на практичному прикладі збору даних, що підтвердило його працездатність, стабільність і продуктивність при роботі з великими обсягами контенту.

Виконано аналіз зібраних даних, що дозволило зробити первинні висновки про структуру соціального графа, активність користувачів та поширені теми. Це демонструє потенціал запропонованого рішення для подальших досліджень соціальних явищ у цифровому середовищі.

Таким чином, третій розділ підтверджує практичну реалізованість теоретичних засад збору та аналізу даних соціальних мереж, а також демонструє ефективність запропонованого програмного рішення.

ВИСНОВКИ

У межах виконаної дипломної роботи було досліджено теоретичні, технічні та прикладні аспекти збору даних із веб-ресурсів соціального нетворкінгу, зокрема на прикладі платформи TikTok. У першому розділі визначено наукову та практичну цінність використання даних соціальних медіа в дослідженнях соціальних процесів. Проведено огляд актуальних інструментів для збору даних, проаналізовано їх можливості та обмеження. Підкреслено важливість поєднання методів соціальних наук і комп'ютерних технологій у цифрову епоху.

У другому розділі розглянуто структуру даних TikTok, окреслено технічні умови збору інформації, а також запропоновано використання графових моделей для аналізу взаємодій між користувачами. Теоретично-формальна модель, представлена в роботі, дозволяє глибше зрозуміти соціальні механізми функціонування платформи та закладає основу для системного аналізу динаміки онлайн-взаємодій.

У третьому розділі реалізовано повноцінне програмне рішення для збору, зберігання та аналізу даних із TikTok. Було використано фреймворк Scrapy для побудови скрапінг-системи, графову базу даних Neo4j для структурування зв'язків та Docker для зручності розгортання. Проведено експериментальну оцінку ефективності інструменту та виконано первинний аналіз зібраної інформації.

Дана робота спрямована на надання дослідникам можливості доступу до даних для аналізу соціальної поведінки на платформі TikTok з відносно низьким технічним бар'єром. Розроблений інструмент веб-скрпінгу враховує зазначені правові та етичні аспекти. Представлений інструмент забезпечує можливість скрпінгу даних з TikTok. Крім того, у даній роботі запропоновано формальну модель соціальних медіа в цілому та TikTok зокрема, та пов'язано це визначення з рамкою для аналізу соціальних явищ. Незважаючи на певну

обмеженість обсягу зібраних даних, особливо коли йдеться про дані, пов'язані з вузькими сегментами дискурсу, отримані дані можуть бути корисними для розуміння загальних характеристик дискурсу TikTok та для встановлення зв'язків між окремими дискурсами.

Реалізований у даній роботі підхід, що поєднує соціальні науки та комп'ютерні науки, може сприяти розвитку співпраці між цими двома дисциплінами. Встановлення зв'язку між соціальною теорією та строгими формальними визначеннями моделі являє собою експериментальний напрямок, що може стимулювати нові інноваційні дослідження у майбутньому.

Загалом робота демонструє практичну доцільність автоматизованого збору та аналізу даних із соціальних мереж як потужного інструменту для вивчення соціальних взаємодій у цифровому середовищі. Представлений підхід є масштабованим, адаптивним до різних платформ і має потенціал для подальшого вдосконалення в напрямку глибшого соціального, поведінкового та контентного аналізу.

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Mitchell, R. (2018). Web Scraping with Python: Collecting Data from the Modern Web (2nd ed.). O'Reilly Media.
2. Lawson, R. (2015). Web Scraping with Python: Collecting More Data from the Modern Web. Packt Publishing.
3. Grinberg, M. (2018). Flask Web Development: Developing Web Applications with Python. O'Reilly Media.
4. Russell, M. A. (2013). Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Instagram, GitHub, and More (2nd ed.). O'Reilly Media.
5. Singh, S., & Rajput, P. (2020). Web Scraping Techniques: Data Mining and Data Analysis from the Web. Independently Published.
6. J. Adams, S. Leestma, and L. Nyhoff, Turbo C++ an introduction to computing. Prentice-Hall, Inc., 1995.
7. Computer Science Is Not About Computers, Any More Than Astronomy Is About Telescopes – Quote Investigator®, en-US, Apr. 2021. [Online]. Available: https://quoteinvestigator.com/2021/04/02/computer-science/.
8. D. Colander and E. Hunt, Social Science: An Introduction to the Study of Society, 17th ed. New York: Routledge,Mar. 2019, ISBN: 978-0-429-01955-5. DOI: 10.4324/9780429019555.
9. Batrinca, B., & Treleaven, P. C. (2015). Social media analytics: a survey of techniques, tools and platforms. AI & Society, 30(1), 89–116.
10. Glez-Peña, D., Lourenço, A., López-Fernández, H., Reboiro-Jato, M., & Fdez-Riverola, F. (2014). Web scraping technologies in an API world. Briefings in Bioinformatics, 15(5), 788–797.
11. Munzert, S., Rubba, C., Meißner, P., & Nyhuis, D. (2014). Automated Data Collection with R: A Practical Guide to Web Scraping and Text Mining. Wiley.

12. Nguyen, T. T., Nguyen, G. H., & Le, D. H. (2019). A survey on web data extraction. Information Systems Frontiers, 21(5), 1181–1197.
13. Song, M., Kim, H., & Jeong, Y. (2020). Social media mining for product planning: A study on the sports industry. Information Processing & Management, 57(5), 102336.
14. Z. Papacharissi and M. de Fatima Oliveira, “Affective news and networked publics: The rhythms of news storytelling on# egypt,” Journal of communication, vol. 62, no. 2, pp. 266–282, 2012.
15. D. Boyd, “Social network sites as networked publics: Affordances, dynamics, and implications,” in A networked self, Routledge, 2010, pp. 47–66.
16. K. E. Anderson, “Getting acquainted with social networks and apps: It is time to talk about TikTok,” Library Hi Tech News, vol. 37, no. 4, pp. 7–12, Jan. 2020, Publisher: Emerald Publishing Limited, ISSN: 0741-9058. DOI: 10.1108/LHTN-01-2020-0001.https://doi.org/10.1108/LHTN-01-2020-
17. Завантажити програму Octoparse. – https://daad.org.ua/6275-zavantazhiti-programu-octoparse.html
18. Dataset processing on Apify – https://blog.apify.com/dataset-processing/
19. Morstatter, F., Pfeffer, J., Liu, H., & Carley, K. M. (2013). Is the sample good enough? Comparing data from Twitter's Streaming API with Twitter's Firehose. ICWSM.
20. Liu, B. (2011). Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (2nd ed.). Springer.
21. Houssiau, F., et al. (2020). Scraping social media for public health surveillance: Ethical considerations. ACM Computing Surveys (CSUR).
22. Vakulenko, S., et al. (2018). QWant at TREC 2018 Tasks: Web Data Collection and Search. Proceedings of TREC.

23. Olteanu, A., Castillo, C., Diaz, F., & Kıcıman, E. (2019). Social data: Biases, methodological pitfalls, and ethical boundaries. Frontiers in Big Data.
24. Faria, E. R., & Gama, J. (2016). Online learning for content extraction from web pages. Journal of Machine Learning Research.
25. Puthal, D., et al. (2015). Big Data Stream Processing: A Comparative Study on Platforms and Tools. ACM Computing Surveys.
26. Barbosa, L., & Freire, J. (2010). An adaptive crawler for locating hidden-web entry points. Proceedings of the 16th International Conference on World Wide Web (WWW).
27. Hirst, G. (2016). Web scraping ethics: Protecting rights while data mining. Communications of the ACM.
28. Gomes, D., Miranda, J., Costa, M. (2011). A survey on web archiving initiatives. International Conference on Theory and Practice of Digital Libraries (TPDL).
29. Janert, P. K. (2010). Data Analysis with Open Source Tools. O'Reilly Media.
30. Richards, G., & Amadini, R. (2020). Scrapy framework for web scraping: An empirical study. SoftwareX, 12, 100578.
31. Alowibdi, J. S., Buy, U. A., & Yu, P. S. (2013). Empirical Evaluation of Profile Characteristics for Gender Classification on Twitter. IEEE/ACM ASONAM.
32. Miller, T. (2019). Web Scraping with BeautifulSoup and Python: Extracting Information from the Internet. Independently Published.
33. Munzert, S., & Rubba, C. (2017). Collecting digital trace data with web scraping and APIs. Political Analysis.
34. Krotov, V., & Silva, L. (2018). Legality of web scraping: A review of the law and recommendations for practice. Business Horizons.

35. Kallus, N. (2014). Predicting crowd behavior with big public data. Proceedings of the 23rd International Conference on World Wide Web (WWW).
36. Ferrara, E., Varol, O., Davis, C., Menczer, F., & Flammini, A. (2016). The rise of social bots. Communications of the ACM.
37. Albahar, M. A., & Abdelhaq, H. (2021). Scraping Twitter Data Using Python and Tweepy. Journal of Computer Science Applications and Information Technology.
38. Garimella, K., De Francisci Morales, G., Gionis, A., & Mathioudakis, M. (2018). Political discourse on social media: Echo chambers, gatekeepers, and the price of bipartisanship. Proceedings of the Web Science Conference.
39. D. Teather, TikTokAPI. Available: https : / / github. com / davidteather / tiktok - api
40. Zubiaga, A., et al. (2018). Detection and Resolution of Rumours in Social Media: A Survey. ACM Computing Surveys (CSUR).

БІБЛІОГРАФІЧНА ДОВІДКА

Тема дипломної роботи: “ Реалізація техніки збору даних з веб-ресурсів соціального нетворкінгу ”

Обсяг пояснювальної записки: 77 аркушів.

Дата закінчення роботи: 16 червня 2025 р.

Підпис студента _____