МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Чорноморський національний університет імені Петра Могили

Факультет комп'ютерних наук

Кафедра інженерії програмного забезпечення

ДОПУЩЕНО ДО ЗАХИСТУ
Завідувач кафедри інженерії
програмного забезпечення
_____ Євген ДАВИДЕНКО
«___»_____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА

АНАЛІЗ НАСТРОЇВ НА ОСНОВІ ДАНИХ СОЦІАЛЬНИХ МЕРЕЖ

Спеціальність 121 Інженерія програмного забезпечення

Освітня програма «Інженерія програмного забезпечення»

Здобувач _____Владислав ШВЕЦЬ
«___»_____ 2024 р.

Керівник керівник PhD, ст. викладачка _____Катерина АНТІПОВА
«___»_____ 2024 р.

Миколаїв – 2024

Чорноморський національний університет імені Петра Могили

(повне найменування закладу вищої освіти)

Факультет	Комп’ютерних наук
Кафедра	Інженерії програмного забезпечення
Рівень вищої освіти	Другий (магістерський)
Освітній ступень	Магістр
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри інженерії
програмного забезпечення
_____Євген ДАВИДЕНКО
«___»_____2024 р.

ЗАВДАННЯ
на кваліфікаційну роботу здобувача

Швеця Владислава Олександровича
(прізвище, ім’я, по батькові здобувача)

1. Тема кваліфікаційної роботи
Аналіз настроїв на основі даних соціальних мереж

1. Затверджена наказом ЧНУ ім. Петра Могили від «04» вересня 2024 р. № 220

2. Строк представлення кваліфікаційної роботи «___»_____2024 р.

3. Очікуваний результат роботи та початкові дані якщо такі потрібні

- розроблений застосунок, початкові дані — датасети повідомлень для аналізу настроїв;

4. Перелік питань, що підлягають розробці:

- дослідження предметної області та аналіз існуючих аналогів ПЗ;
- формування специфікації вимог до ПЗ;
- моделювання та проєктування ПЗ;
- розробка ПЗ;
- здійснення тестування роботи ПЗ;
- проведення аналізу результатів розробки;

5. Перелік графічних матеріалів

- презентація;

Керівник роботи _____ Катерина АНТІПОВА
Особистий підпис *Власне ім'я ПРІЗВИЩЕ*

Здобувач _____ Владислав ШВЕЦЬ
Особистий підпис *Власне ім'я ПРІЗВИЩЕ*

Дата видачі завдання « ____»_____20_____р

КАЛЕНДАРНИЙ ПЛАН

виконання кваліфікаційної роботи

Тема: Аналіз настроїв на основі даних соціальних мереж

№	Найменування роботи	Початок	Закінчення	Примітки
1.	Робота з документацією	10.09.2024 р.	11.09.2024 р.	Виконано
2.	Огляд літератури за темою роботи	13.09.2024 р.	19.09.2024 р.	Виконано
3.	Складання календарного плану КМР	20.03.2024 р.	21.03.2024 р.	Виконано
4.	Аналіз предметної області	24.09.2024 р.	26.09.2024 р.	Виконано
5.	Розробка проєктних рішень	28.09.2024 р.	30.09.2024 р.	Виконано
6.	Моделювання та конструювання ПЗ	01.10.2024 р.	06.10.2024 р.	Виконано
7.	Кодування, тестування та апробація розробленого ПЗ, аналіз результатів тестування, розробка керівництва користувача	11.10.2024 р.	07.11.2024 р.	Виконано
8.	Оформлення КМР та презентації	10.11.2024 р.	20.11.2024 р.	Виконано
9.	Відгук керівника КМР	.2024 р.	.2024 р.	Виконано
10.	Попередній захист	28.11.2024 р.	28.11.2024 р.	Виконано
11.	Завершення оформлення КМР та презентації	12.12.2024 р.	12.12.2024 р.	Виконано
12.	Рецензування	13.12.2024 р.	14.12.2024 р.	
13.	Захист кваліфікаційної роботи	19.12.2024 р.	19.12.2024 р.	

Розробив здобувач Швець Владислав Олександрович _____

(прізвище, ім'я, по батькові) *(підпис)*

«__» _____ 2024 р.

Керівник роботи PhD, ст. викладачка кафедри ІПЗ Антіпова Катерина Олександрівна

«__» _____ 2024 р.

АНОТАЦІЯ

до кваліфікаційної роботи магістра

Аналіз настроїв на основі даних соціальних мереж

Здобувач 608 гр.: Швець Владислав Олександрович

Керівник: PhD, ст. викладачка кафедри ІПЗ Антіпова К. О.

Соціальні мережі стали важливими джерелами інформації для розуміння громадських настроїв, а також для аналізу реакцій на різноманітні події в реальному часі. Одним із основних напрямків сучасного аналізу є обробка та визначення емоційної тональності повідомлень у таких платформах, як Telegram та Reddit. Метою цієї роботи є розробка інструменту для аналізу настроїв на основі даних з цих соціальних мереж.

Об’єктом кваліфікаційної роботи є процеси збору, обробки та аналізу текстових даних з Telegram та Reddit.

Предметом кваліфікаційної роботи є інструментальні засоби та технології, що використовуються для аналізу настроїв, включаючи бібліотеки для обробки природної мови, а також технології для інтеграції з платформами Telegram та Reddit.

Метою роботи є дослідження та впровадження методів обробки текстових даних для автоматичного визначення настроїв у повідомленнях з Telegram та Reddit, з використанням сучасних бібліотек для аналізу настроїв та візуалізації результатів.

Для досягнення поставленої мети були виконані наступні завдання:

1. Проведено огляд сучасних методів аналізу настроїв на основі даних з соціальних мереж. Оцінено використання бібліотек NLP (Natural Language Processing) для аналізу емоційної тональності.

2. Розроблено алгоритми для збору та попередньої обробки текстових даних з Telegram та Reddit.

3. Проведено проєктування програмного забезпечення для аналізу настроїв, включаючи функціональні модулі для збирання та обробки даних, а також для візуалізації результатів.

4. Реалізовано програмні модулі для підключення до API Telegram та Reddit, аналізу текстів за допомогою моделей на основі трансформерів, зокрема DistilBERT, та візуалізації результатів у вигляді графіків.

5. Проведено тестування програмного забезпечення для перевірки точності результатів та ефективності роботи з великими обсягами даних.

У першому розділі було проведено дослідження інструментів для збору та аналізу даних з Telegram і Reddit, що дозволило визначити переваги та обмеження існуючих рішень.

Другий розділ присвячено розробці алгоритмів збору, попередньої обробки даних та аналізу настроїв.

Третій розділ містить опис проєктування програмного забезпечення, включаючи специфікації вимог та проєктування інтерфейсу.

Четвертий розділ присвячено реалізації програмних модулів, їх тестуванню, а також розробці інструкції користувача для роботи з застосунком.

Результатом дипломної роботи є функціональний застосунок для аналізу настроїв на основі даних з Telegram та Reddit, який здатен надавати інформацію про емоційну тональність повідомлень користувачів в реальному часі.

Кваліфікаційна робота викладена на 61 сторінку і містить 4 розділи, 17 ілюстрацій, 10 таблиць та 30 джерела в переліку посилань.

Ключові слова: *Аналіз настроїв, соціальні мережі, Telegram, Reddit, обробка тексту, природна мова, машинне навчання, візуалізація, моделі трансформерів, NLP.*

ABSTRACT

for the Master's Thesis

Sentiment Analysis Based on Social Media Data

Applicant, Group 608: Vladyslav Shvets

Supervisor: PhD, senior lecturer of the department of software engineering

Kateryna Antipova

Social media has become an important source of information for understanding public sentiments and analyzing reactions to various events in real time. One of the key areas in modern analysis is processing and determining the emotional tone of messages on platforms like Telegram and Reddit. The goal of this work is to develop a tool for sentiment analysis based on data from these social networks.

The object of the thesis is processes related to the collection, processing, and analysis of textual data from Telegram and Reddit.

The subject of the thesis is the instrumental tools and technologies used for sentiment analysis, including libraries for natural language processing (NLP), as well as technologies for integrating with Telegram and Reddit platforms.

The aim of this work is to research and implement methods for processing textual data to automatically determine sentiments in messages from Telegram and Reddit, using modern libraries for sentiment analysis and visualizing the results.

To achieve this goal, the following tasks were completed:

A review of modern sentiment analysis methods based on social media data was conducted. The use of NLP libraries for emotional tone analysis was evaluated.

Algorithms for collecting and preprocessing textual data from Telegram and Reddit were developed.

Software design for sentiment analysis was carried out, including functional modules for data collection and processing, as well as for visualizing the results.

Software modules for connecting to Telegram and Reddit APIs, analyzing texts using transformer-based models, particularly DistilBERT, and visualizing results in the form of graphs were implemented.

Software testing was carried out to check the accuracy of results and the efficiency of working with large datasets.

The first chapter reviews the tools for collecting and analyzing data from Telegram and Reddit, determining the advantages and limitations of existing solutions.

The second chapter discusses the development of algorithms for data collection, preprocessing, and sentiment analysis.

The third chapter contains the design of the software, including requirement specifications and user interface design.

The fourth chapter focuses on the implementation of software modules, their testing, and the development of the user manual.

The result of the thesis is a functional application for sentiment analysis based on data from Telegram and Reddit, capable of providing information on the emotional tone of user messages in real time.

The thesis is presented in 61 pages and includes 4 chapters, 17 illustrations, 10 tables, and 30 sources in the reference list.

Keywords: *Sentiment Analysis, Social Media, Telegram, Reddit, Text Processing, Natural Language, Machine Learning, Visualization, Transformer Models, NLP.*

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

API – Application Programming Interface

Telegram API – Telegram Application Programming Interface

Reddit API – Reddit Application Programming Interface

nltk – Natural Language Toolkit

Transformers – бібліотека для трансформерних моделей (не має розшифровки, це назва бібліотеки)

Matplotlib – бібліотека для візуалізації даних (не має розшифровки, це назва бібліотеки)

Seaborn – бібліотека для візуалізації даних (не має розшифровки, це назва бібліотеки)

PyTorch – бібліотека для машинного навчання (не має розшифровки, це назва бібліотеки)

Sentiment Analysis – аналіз настроїв

Tokenization – токенізація

Lemmatization – лематизація

Stopwords – стоп-слова

API Key – ключ API

Client ID – ідентифікатор клієнта

Client Secret – секретний ключ клієнта

Subreddit – підрозділ на Reddit

Hot Posts – популярні пости

Model – модель машинного навчання

Pipeline – конвеєр

DistilBERT – Distilled BERT (скорочена версія BERT)

Sentiment Labels – мітки настроїв

ВСТУП

У цифрову епоху соціальні мережі стали чимось більшим, ніж просто платформами для спілкування; це динамічні екосистеми, які формують думки, тенденції та соціальні рухи. Мільярди користувачів по всьому світу щодня взаємодіють з такими платформами, як Twitter, Facebook, Instagram та LinkedIn, генеруючи величезні обсяги даних у вигляді постів, коментарів, вподобань та поширень. Ці дані відображають колективні настрої, думки та емоції користувачів, що робить їх цінним ресурсом для дослідників та індустрії, які прагнуть зрозуміти суспільні настрої в режимі реального часу.

Аналіз настроїв на основі даних соціальних мереж, який часто називають аналізом настроїв або видобутком думок, став потужним інструментом у таких галузях, як маркетинг, політологія, громадське здоров'я та психологія. Застосовуючи методи обробки природної мови (NLP) до величезних обсягів неструктурованих текстових даних, ми можемо класифікувати і кількісно оцінити емоції та думки, висловлені користувачами на різні теми. Цей процес дозволяє виокремлювати тенденції, прогнози та дієві ідеї, які можуть впливати на прийняття рішень як у державному, так і в приватному секторах.

Унікальність цієї роботи полягає в тому, що воно здатне забезпечити масштабне і точне відображення суспільних настроїв у режимі реального часу. На відміну від традиційних опитувань, які займають багато часу і часто схильні до упередженості, аналіз настроїв на основі даних соціальних мереж забезпечує безперервний зворотний зв'язок з громадською думкою. Ця здатність відстежувати зміну настроїв у міру їхнього розвитку з плином часу має вирішальне значення для галузей та урядів для швидкого реагування на суспільні потреби та занепокоєння [7].

Аналіз настроїв у соціальних мережах є відносно новою галуззю досліджень, але вона набрала значних обертів завдяки вибуховому зростанню соціальних медіа-платформ. На відміну від традиційних ЗМІ, де контент курується і контролюється, соціальні мережі рухаються за рахунок контенту,

створеного користувачами. Ця відмінність створює унікальну можливість для дослідників отримати доступ до величезної кількості нецензурованої інформації в режимі реального часу [9].

Новизна цього дослідження полягає в міждисциплінарному підході, що поєднує НЛП, машинне навчання та методи інтелектуального аналізу даних для аналізу користувацького контенту. Ця тема виділяється тим, що вона стосується не лише вилучення індивідуальних поглядів, але й виявлення тенденцій колективних настроїв. Ці тенденції можуть бути використані для різних цілей прогнозування, таких як прогнозування результатів виборів, поведінки споживачів або навіть коливань на фондовому ринку.

Актуальність цього дослідження підкреслюється його практичним застосуванням у різних галузях. Наприклад, бізнес може використовувати аналіз настроїв для оцінки задоволеності клієнтів і вдосконалення своєї продукції. Політики та урядовці можуть використовувати його для моніторингу громадської думки щодо запропонованих законопроєктів або суспільних проблем. Крім того, організації охорони здоров'я можуть відстежувати настрої громадськості щодо кризових ситуацій у сфері охорони здоров'я, таких як пандемії або програми вакцинації, щоб краще адаптувати свої інформаційно-просвітницькі зусилля.

Поширення фейкових новин, дезінформації та ботів у соціальних мережах також поставило під сумнів надійність даних із соціальних мереж. Це додає унікальності цьому дослідженню, оскільки воно вивчатиме методи фільтрації шуму та виявлення справжніх настроїв користувачів, підвищуючи таким чином точність аналізу настроїв.

У цій кваліфікаційній роботі будуть розглянуті методи, виклики та застосування аналізу настроїв у соціальних мережах, щоб сприяти кращому розумінню того, як аналіз настроїв може бути ефективно застосований. Дослідження також визначить обмеження та етичні міркування, пов'язані з видобутком користувацького контенту для отримання інсайтів.

1 АНАЛІЗ НАСТРОЇВ НА ОСНОВІ ДАНИХ СОЦІАЛЬНИХ МЕРЕЖ

1.1 Сфери застосування аналізу настроїв

Хоча це дослідження зосереджене на аналізі настроїв у соціальних мережах, аналіз настроїв успішно застосовується в різних сферах:

- Аналіз політичних настроїв: Під час виборчих сезонів аналітики використовують платформи соціальних мереж для оцінки громадської думки щодо політичних кандидатів, політики та дебатів. Моделі аналізу настроїв навчені класифікувати пости як позитивні, негативні або нейтральні, що дозволяє прогнозувати поведінку виборців. Дослідження показали, що сплески позитивних настроїв щодо кандидата в Твіттері можуть корелювати зі збільшенням результатів опитувань або явки виборців.

- Моніторинг бренду та управління репутацією: Компанії використовують аналіз настроїв для оцінки громадського сприйняття свого бренду. Наприклад, після запуску продукту компанії можуть відстежувати відгуки клієнтів у режимі реального часу, що допомагає їм на ранніх стадіях виявляти потенційні проблеми. Аналізуючи тональність коментарів клієнтів, компанії можуть вирішити, чи варто коригувати свої маркетингові стратегії або усунути недоліки продукту [5].

- Прогнозування фондового ринку. У фінансовій сфері аналіз настроїв використовується для прогнозування руху цін на акції на основі колективних настроїв, висловлених у соціальних мережах. Наприклад, значне зростання негативних настроїв щодо компанії на таких платформах, як Twitter або Reddit, може свідчити про зниження цін на акції. І навпаки, позитивні настрої можуть сигналізувати про майбутнє зростання вартості акцій.

- Громадське здоров'я та кризовий менеджмент. Аналіз настроїв застосовується для моніторингу кризових ситуацій у сфері охорони здоров'я. Під час пандемії COVID-19 дослідники використовували його для відстеження ставлення громадськості до карантинних заходів та програм вакцинації.

Розуміння того, як люди ставляться до цих питань, дозволило чиновникам у сфері охорони здоров'я розробити більш ефективні комунікаційні стратегії.

- Вивчення думок споживачів та електронна комерція. У секторах роздрібної торгівлі та електронної комерції компанії аналізують відгуки клієнтів та дискусії в соціальних мережах, щоб зрозуміти споживчі вподобання. Це дозволяє компаніям приймати засновані на даних рішення щодо інвентаризації, дизайну продукції та покращення обслуговування клієнтів [2].

1.2 Компоненти процесу аналізу настроїв

Для успішного аналізу настроїв у соціальних мережах необхідно врахувати кілька ключових компонентів і вимог:

- Збір даних. Першочерговою вимогою є доступ до великих обсягів даних з платформ соціальних мереж, таких як Twitter, Facebook і Reddit. Ці дані можна зібрати за допомогою публічних API або методів веб-скрепінгу. Важливо збирати різноманітні набори даних, які включають текстові пости, коментарі, хештеги і навіть мультимедійний контент, наприклад, зображення або відео, якщо це можливо.

- Попередня обробка даних. Дані з соціальних мереж часто зашумлені, містять помилки, сленг і неформальну мову. Тому попередня обробка є важливим кроком, який включає очищення тексту (видалення стоп-слів, розділових знаків), нормалізацію сленгу та обробку абревіатур. Крім того, методи токенізації та лематизації необхідні для підготовки даних до аналізу.

- Інструменти для обробки природної мови (NLP). Такі інструменти, як NLTK, SpaCy або бібліотека Hugging Face Transformers, необхідні для аналізу тексту. Ці інструменти будуть використані для токенізації тексту, виявлення ключових особливостей і класифікації настроїв. Просунуті методи, такі як розпізнавання іменованих об'єктів (NER) і моделювання тем, також можуть бути застосовані для вилучення більш детальної інформації з тексту [3].

Алгоритми класифікації настроїв: Для класифікації настроїв можна використовувати різноманітні моделі машинного навчання. Можна

використовувати традиційні моделі, такі як наївний Байєс або машини опорних векторів (SVM), але моделі глибокого навчання, такі як рекурентні нейронні мережі (RNN) або моделі на основі трансформаторів (наприклад, BERT або GPT), є більш ефективними для врахування контексту і нюансів мови. Ці моделі допоможуть класифікувати настрої як позитивні, негативні або нейтральні [1].

- Візуалізація даних та звітність. Такі інструменти, як Matplotlib, Seaborn або Plotly, будуть використовуватися для візуалізації тенденцій і закономірностей у настроях з плином часу. Інтерактивні інформаційні панелі можуть бути створені за допомогою таких бібліотек, як Dash або Tableau, для відображення інформації в режимі реального часу, що дозволить особам, які приймають рішення, відслідковувати еволюцію суспільних настроїв.

- Етичні та правові міркування. Оскільки це дослідження передбачає збір та аналіз контенту, створеного користувачами, важливо дотримуватися етичних принципів, що стосуються конфіденційності та захисту даних. Особиста інформація користувачів повинна бути анонімізована, а дані повинні збиратися згідно з відповідними правовими нормами, такими як GDPR в Європейському Союзі. За необхідності слід отримувати інформовану згоду [5].

На рисунку 1.1 зображено діаграма IDEF0.

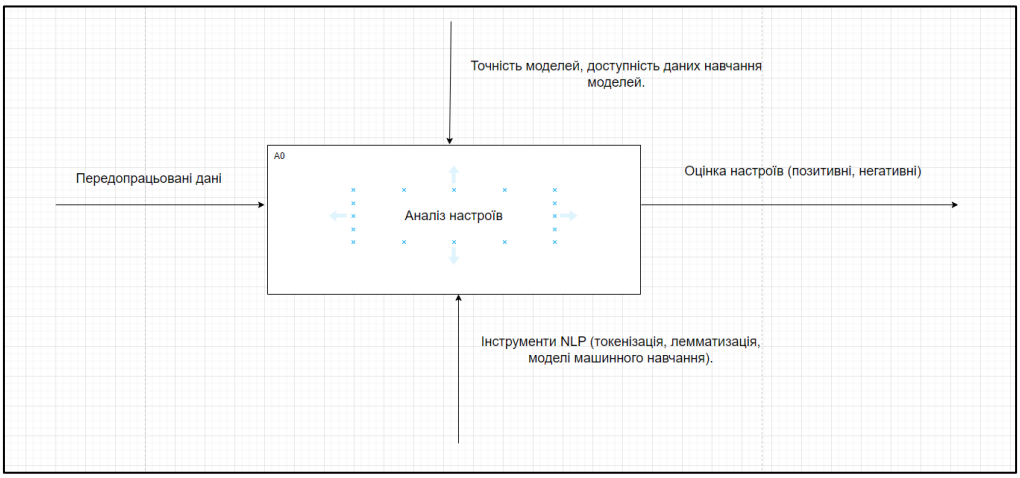

Рисунок 1.1 – Діаграма IDEF0

Щоб ще більше проілюструвати актуальність цього дослідження, нижче наведено реальні приклади застосування аналізу настроїв у різних галузях:

- Автоматизація обслуговування клієнтів. Чат-боти, що працюють на основі аналізу настроїв, можуть виявляти розчарування або задоволеність клієнтів під час розмови і реагувати відповідно. Розуміючи ставлення клієнтів, компанії можуть адаптувати свої стратегії підтримки для надання більш ефективних послуг.

- Моніторинг ЗМІ. ЗМІ та журналісти використовують аналіз настроїв для відстеження громадської думки щодо важливих новин. Аналізуючи дискусії в соціальних мережах, вони можуть оцінити, як різні демографічні групи ставляться до певних подій, і адаптувати своє висвітлення до потреб аудиторії [4].

- Уряд і формування політики. Уряди можуть використовувати аналіз настроїв для моніторингу того, як громадяни ставляться до певної політики чи суспільних змін. Наприклад, аналіз настроїв може допомогти відстежити реакцію громадськості на нові закони, податкову політику або реформу охорони здоров'я.

1.3 Виклики та міркування

Хоча аналіз настроїв має великий потенціал, необхідно вирішити кілька проблем:

- Складність мови. Пости в соціальних мережах часто містять неформальну мову, включаючи сленг, абревіатури та емодзі, які можуть бути складними для інтерпретації моделями НЛП. Крім того, сарказм та іронія створюють значні проблеми для алгоритмів аналізу настроїв, оскільки ці лінгвістичні конструкції можуть передавати настрої, протилежні буквально вираженим.

- Багатомовні дані. Соціальні мережі - це глобальне явище, в якому користувачі публікують дописи різними мовами. Це додає складності процесу аналізу настроїв, оскільки моделі повинні бути навчені ефективно обробляти багатомовні дані або використовувати інструменти перекладу, що може призвести до неточностей [3].

- Упередженість даних. Моделі аналізу настроїв можуть успадковувати упередженість від даних, на яких вони навчаються. Наприклад, певні думки можуть бути надмірно представлені в соціальних мережах, що призводить до викривлення результатів. Крім того, платформи можуть мати демографічні упередження, які впливають на узагальненість результатів.

- Обробка в режимі реального часу. Хоча пакетна обробка історичних даних є поширеним явищем, аналіз настроїв у реальному часі створює технічні проблеми. Він вимагає впровадження масштабованих систем, здатних обробляти та аналізувати потокові дані. Це передбачає обробку великих обсягів дописів і забезпечення того, щоб інсайти надавалися з мінімальною затримкою.

1.4 Використання методів штучного інтелекту для аналізу настроїв

В рамках виконання кваліфікаційної роботи було застосовано сучасні методи штучного інтелекту (ШІ) для аналізу текстової інформації. Основна увага приділялася використанню технологій обробки природної мови (Natural Language Processing, NLP), які забезпечують автоматичне розпізнавання, аналіз і класифікацію настроїв текстових повідомлень.

Найбільш перспективним напрямом в NLP є використання трансформерів — архітектури, що базується на механізмі самоуваги. Для розв’язання задачі аналізу настроїв було обрано модель DistilBERT, яка є компактною версією BERT (Bidirectional Encoder Representations from Transformers). DistilBERT зберігає високу точність аналізу завдяки використанню спеціальних механізмів стиснення моделі, що робить її швидшою та економічнішою у використанні ресурсів [5].

Методи ШІ, використані в роботі, включають:

- Попереднє навчання моделей. Модель DistilBERT навчалася на великих корпусах текстів, що дало змогу розуміти широкий контекст і взаємозв’язки між словами.

- Класифікацію настроїв. Модель здатна ідентифікувати емоційний тон тексту та класифікувати його як позитивний або негативний.

- Оптимізацію обчислень. Для швидкого виконання аналізу була використана інтеграція з бібліотекою Transformers, яка забезпечує роботу моделі з оптимізацією під сучасні апаратні архітектури.

Також у роботі було використано технології лематизації, видалення стоп-слів та токенізації, які дозволяють підготувати текстові дані для аналізу за допомогою ШІ [2].

Таким чином, застосування методів ШІ дало змогу автоматизувати складні процеси аналізу великих обсягів даних, забезпечивши високу точність і ефективність отриманих результатів. Застосовані алгоритми забезпечують не лише розуміння настроїв текстів, але й відкривають нові перспективи для їхнього подальшого використання в практичних і наукових дослідженнях.

Цей підрозділ підкреслює, що методи штучного інтелекту є ключовим інструментом для розв'язання задач аналізу настроїв і демонструє, наскільки ефективними можуть бути сучасні алгоритми обробки природної мови в поєднанні з машинним навчанням.

1.5 Специфікація вимог до програмного забезпечення

Для реалізації системи аналізу настроїв на основі даних соціальних мереж необхідно визначити низку вимог до програмного забезпечення (ПЗ), яке буде забезпечувати ефективну роботу всіх процесів. Ці вимоги охоплюють функціональні та нефункціональні аспекти роботи системи.

Функціональні вимоги

1. Збір даних із соціальних мереж:

- Програмне забезпечення повинно мати можливість інтеграції з публічними API таких платформ, як Twitter, Facebook, Reddit для збору даних у реальному часі.

- Забезпечення можливості веб-скрепінгу для платформ, що не надають відкритих API.

- Можливість збору різних типів контенту, включаючи текстові пости, коментарі, хештеги, а також мультимедійні файли (за необхідності).

2. Попередня обробка даних:

- Очищення зібраних даних від шуму, стоп-слів, розділових знаків та дублікатів.

- Нормалізація сленгу, скорочень та абревіатур для полегшення аналізу.

- Токенізація та лематизація текстових даних для подальшої обробки.

3. Аналіз настроїв:

- Використання методів обробки природної мови (NLP) для класифікації настроїв (позитивний, негативний, нейтральний).

- Реалізація алгоритмів машинного навчання для аналізу настроїв, таких як наївний Байєс, машини опорних векторів (SVM), рекурентні нейронні мережі (RNN) або моделі трансформаторів (BERT, GPT).

- Можливість розпізнавання іменованих сутностей (NER) для виявлення ключових об'єктів у тексті.

4. Візуалізація результатів:

- Побудова графіків та діаграм для візуалізації результатів аналізу настроїв у часі.

- Створення інтерактивних інформаційних панелей для представлення результатів у режимі реального часу.

- Експорт звітів у різних форматах (PDF, Excel) для подальшого використання.

5. Фільтрація фейкових новин та дезінформації:

- Впровадження механізмів виявлення та відсіву фейкових новин, ботів і дезінформації для підвищення точності аналізу.

6. Управління доступом:

- Можливість обмеження доступу до результатів аналізу для різних категорій користувачів (аналітики, адміністратори, користувачі).

- Забезпечення автентифікації та авторизації користувачів системи.

Нефункціональні вимоги

7. Продуктивність:

- Система повинна забезпечувати обробку великих обсягів даних у режимі реального часу з мінімальною затримкою.

- Програмне забезпечення повинно бути масштабованим для роботи з різними соціальними мережами та обробляти тисячі постів одночасно.

8. Надійність:

- Програмне забезпечення має забезпечувати високу надійність під час збору даних і аналізу, мінімізуючи втрату даних або збої в процесі обробки.

- Система повинна мати механізми для відновлення після помилок.

9. Безпека:

- Програмне забезпечення повинно відповідати вимогам конфіденційності та захисту даних користувачів, включаючи анонімізацію особистої інформації.

- Використання протоколів безпеки для захисту даних під час їх збирання та обробки.

10. Масштабованість:

- Програмне забезпечення повинно мати здатність до розширення функціональності для підтримки нових платформ соціальних мереж та методів аналізу.

11. Зручність використання:

- Інтерфейс системи має бути інтуїтивно зрозумілим для користувачів різного рівня технічної підготовки.

- Програмне забезпечення повинно забезпечувати зручний доступ до звітів і результатів аналізу.

Висновки до розділу 1

У першому розділі проведено детальний аналіз предметної сфери аналізу настроїв на основі даних соціальних мереж. Було визначено новизну та актуальність цієї теми, зосереджуючись на її міждисциплінарному підході та широких можливостях практичного застосування. Соціальні мережі стали невичерпним джерелом реального часу для вивчення громадської думки, на

основі якої можуть бути зроблені прогнози в політичній, економічній та інших сферах.

Аналіз настроїв активно використовується для моніторингу громадської думки під час виборчих кампаній, управління репутацією брендів, прогнозування коливань на фондових ринках, а також у кризових ситуаціях, таких як пандемії. Водночас, аналіз на основі соціальних мереж стикається з певними викликами, серед яких обробка неформальної мови, багатомовність, упередженість даних та необхідність масштабованих рішень для обробки даних у реальному часі.

Було також розглянуто ключові вимоги до успішного проведення аналізу настроїв, включаючи необхідність якісного збору та попередньої обробки даних, використання сучасних інструментів для обробки природної мови (NLP), ефективних алгоритмів класифікації настроїв та візуалізації результатів.

Таким чином, аналіз настроїв є потужним інструментом для прийняття рішень у різних галузях, проте він потребує ретельної підготовки та вирішення низки технічних і етичних проблем для забезпечення точності та надійності результатів.

2 МОДЕЛЮВАННЯ ТА ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ АНАЛІЗУ НАСТРОЇВ НА ОСНОВІ ДАНИХ СОЦІАЛЬНИХ МЕРЕЖ

2.1 Діаграма варіантів використання системи

Діаграми використання зазвичай відображають дані за допомогою комбінації лінійних графіків, гістограм та інших візуалізацій. Дані можуть бути представлені у різний спосіб, залежно від конкретних потреб користувача та типу інформації, що відстежується. Загалом, діаграми використання можуть надати цінну інформацію про моделі поведінки та допомогти користувачам приймати більш обґрунтовані рішення щодо споживання ресурсів [7].

Діаграма станів, також відома як діаграма автомата або діаграма станів, - це графічне зображення поведінки системи або об'єкта у відповідь на зовнішні стимули. Вона широко використовується в програмній інженерії, інженерії управління та інших галузях для моделювання поведінки складних систем.

Діаграма станів складається з набору станів, які представляють різні умови або режими, в яких може перебувати система або об'єкт, і переходів, які представляють події або умови, що змушують систему переходити з одного стану в інший. Діаграма може також включати дії, які представляють дії, що виконуються, коли відбувається перехід.

Діаграми станів корисні для моделювання складних систем, оскільки вони дозволяють проєктувальникам розбити систему на менші, більш керовані частини і чітко визначити поведінку кожної частини. Їх також можна використовувати для виявлення потенційних проблем або вузьких місць у системі, а також для тестування і доопрацювання проєкту перед його реалізацією. Діаграми станів можуть бути створені за допомогою різних інструментів, в тому числі спеціально розроблених для цього програм [9].

На таблиці 1 наведено глосарій проєкту

Таблиця 2.1 – Словник проєкту

Аналітик		Спеціаліст, який аналізує соц. мережі
Адміністратор		Спеціаліст, який відповідає за **Управління доступом** до системи, налаштовуючи права доступу для різних користувачів, що гарантує безпеку і правильний розподіл доступу.
Система збору даних		Відповідає за виконання цього процесу автоматично, забезпечуючи отримання необхідних даних.

На рисунку 2.1 – зображено діаграма прецедентів.

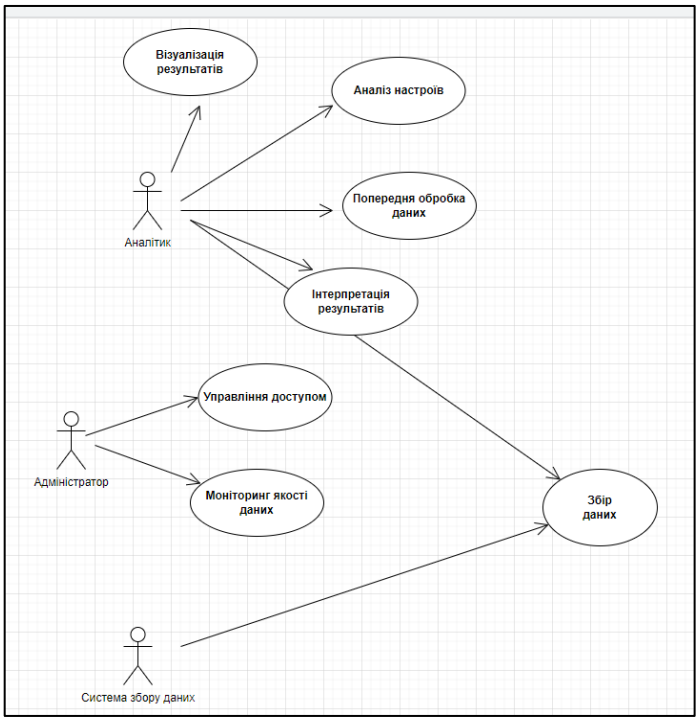

Рисунок 2.1 – Діаграма прецедентів

2.2 Специфікація варіантів використання

Діаграма використання - це тип графіка або діаграми, що відображає дані про використання або споживання певного ресурсу чи продукту протягом певного періоду часу. Це може включати інформацію про те, як часто використовується продукт або ресурс, скільки його використовується кожного разу і як змінюються моделі використання з часом.

Аналітик – починає аналізувати соц. мережі.

Адміністратор – управляє всім доступом до системи.

Система збору даних – починає збір даних.

Варіант використання “ Збір та попередня обробка даних”

Опис сценарію:

Аналітик ініціює процес збору даних, використовуючи спеціальні інструменти або API для отримання текстової інформації з соціальних мереж. Це може бути постійний моніторинг або одноразове завантаження даних.

Постумови:

Після завершення процесу збору даних система автоматично забезпечує доступ до отриманої інформації для подальшої обробки та аналізу. Аналітик отримує повний набір текстових даних із соціальних мереж, готових до подальшої попередньої обробки.

Предумови:

Для реалізації цього варіанту використання необхідно мати налаштовану систему збору даних із соціальних мереж за допомогою API. Також повинна бути доступна платформа для отримання даних у зручному форматі для подальшої обробки.

На таблиці 2.2 наведено варіант використання “Розробка застосунку”.

Таблиця 2.2 – Головний розділ сценарію виконання варіанта використання “Розробка застосунку”.

Варіант використання	**Розробка застосунку**
Актори	Аналітик, система збору даних
Короткий опис	Аналітик ініціює процес Збору даних, використовуючи спеціальні інструменти або API для отримання текстової інформації із соціальних мереж. Це може бути постійний моніторинг або одноразове завантаження даних.
Мета	Збір та обробка даних
Тип	Базовий

На таблиці 2.3 наведено хід подій варіанту використання “ Збір та попередня обробка даних”.

Таблиця 2.3 – Типовий хід подій сценарію виконання варіанта використання “Збір та попередня обробка даних”.

Дії актора	**Відгук системи**
1. Аналітик ініціює процес Збору даних, використовуючи спеціальні інструменти або API для отримання текстової інформації із соціальних мереж. Це може бути постійний моніторинг або одноразове завантаження даних.	4. Інформація: Система діє інформацію по соц. мережам для аналізу.
2.Система збору даних відповідає за виконання цього процесу автоматично, забезпечуючи отримання необхідних даних.	5. Запуск: Включення обробки даних, очистка тексту від зайвих символів.

Продовження таблиці 2.3

3.Після збору даних Аналітик запускає процес Попередньої обробки даних, який включає очищення тексту від зайвих символів, нормалізацію та підготовку до подальшого аналізу.	

На таблиці 2.4 наведено винятки варіанту використання “ Збір та попередня обробка даних ”.

Таблиця 2.4 – Винятки сценарію виконання варіанта використання “ Збір та попередня обробка даних ”.

Дії актори	**Відгук системи**
Виняток №1. Збір даних не працює	
	7.Адміністратор бачить помилку
8. Перезавантаження системи збору даних	

Варіант використання “Аналіз настроїв та візуалізація результатів”

Опис сценарію:

Аналітик ініціює процес збору даних, використовуючи спеціальні інструменти або API для отримання текстової інформації з соціальних мереж. Це може бути постійний моніторинг або одноразове завантаження даних.

Система збору даних відповідає за виконання цього процесу автоматично, забезпечуючи отримання необхідних даних.

Після збору даних Аналітик запускає процес попередньої обробки даних, який включає очищення тексту від зайвих символів, нормалізацію та підготовку до подальшого аналізу.

Постумови:

Після завершення збору та попередньої обробки даних система забезпечує збереження очищених та нормалізованих даних для подальшого аналізу. Аналітик отримує підготовлену для аналізу інформацію, що забезпечує точність подальших кроків.

Предумови:

Перед початком збору даних необхідно налаштувати систему збору даних через API та інструменти для очищення тексту. Також має бути доступна система для попередньої обробки, яка включає нормалізацію та очищення даних від шуму.

На таблиці 2.5 наведено варіант використання “ Аналіз настроїв та візуалізація результатів”.

Таблиця 2.5 – Головний розділ сценарію виконання варіанта використання “Аналіз настроїв та візуалізація результатів”

Варіант використання	Аналіз настроїв та візуалізація результатів
Актори	Аналітик, Адміністратор
Мета	Візуалізація результатів
Короткий опис	Зробити аналіз настроїв
Тип	Базовий

На таблиці 2.6 наведено хід подій варіант використання “ Аналіз настроїв та візуалізація результатів ”.

Таблиця 2.6 – Типовий хід подій сценарію виконання варіанта використання “ Аналіз настроїв та візуалізація результатів ”.

Дії актора	**Відгук системи**

Продовження таблиці 2.6

1. Після того як дані були зібрані та оброблені, Аналітик запускає процес Аналізу настроїв, використовуючи спеціальні алгоритми чи моделі машинного навчання. Цей процес класифікує зібрані дані за настроями: позитивними, негативними або нейтральними.	3. Створюється графік аналізу
2. Після аналізу Аналітик ініціює процес Візуалізації результатів. Це створення графіків і діаграм, які відображають результати класифікації настроїв у зручній для розуміння формі.	

На таблиці 2.7 наведено винятки варіанту використання “ Аналіз настроїв та візуалізація результатів ”.

Таблиця 2.7 – Винятки сценарію виконання варіанта використання “ Аналіз настроїв та візуалізація результатів ”.

Дії актора	Відгук системи
Виняток №2. Аналітик не правильно зробив аналіз	
5. Аналітик перезаписує свій аналіз	4. Адміністратор бачить не той аналіз

Варіант використання ” Інтерпретація результатів та управління доступом”

Опис сценарію:

Після отримання візуалізованих результатів, Аналітик проводить інтерпретацію результатів, роблячи висновки на основі аналізу. Ці висновки можуть використовуватися для прийняття рішень або коригування стратегії.

Адміністратор відповідає за управління доступом до системи, налаштовуючи права доступу для різних користувачів, що гарантує безпеку і правильний розподіл доступу.

Постумови:

Після інтерпретації результатів, Аналітик отримує чітке уявлення про настрої в соціальних мережах і може використати ці дані для коригування стратегії. Адміністратор забезпечує контроль доступу, гарантує безпеку системи і дозволяє коректно розподіляти права доступу для користувачів.

Предумови:

Перед виконанням цього варіанту використання повинні бути зібрані і оброблені дані, а також здійснена візуалізація результатів. Для успішного виконання процесу інтерпретації необхідна наявність доступу до готових результатів аналізу. Адміністратор повинен мати повноваження для налаштування доступу та керування правами користувачів у системі.

На таблиці 2.8 наведено варіант використання “ Інтерпретація результатів та управління доступом”.

Таблиця 2.8 – Головний розділ сценарію виконання варіанта використання ” Інтерпретація результатів та управління доступом”.

Варіант використання	Інтерпретація результатів та управління доступом
Актори	Аналітик, Адміністратор
Короткий опис	Процеси управління доступом і контролю якості даних.
Мета	Розробити процеси управління доступом і контролю якості даних.
Тип	Підлеглий

На таблиці 2.9 наведено хід подій варіанту використання “ Інтерпретація результатів та управління доступом ”.

Таблиця 2.9 – Типовий хід подій сценарію виконання варіанта використання ” Інтерпретація результатів та управління доступом”.

Дії актора	Відгук системи
1. Після отримання візуалізованих результатів Аналітик проводить Інтерпретацію результатів, роблячи висновки на основі аналізу. Ці висновки можуть використовуватися для прийняття рішень або коригування стратегії.	3.Прийшло повідомлення, про результати програми
2. Адміністратор відповідає за Управління доступом до системи, налаштовуючи права доступу для різних користувачів, що гарантує безпеку і правильний розподіл доступу.	

На таблиці 2.10 наведено винятки варіанту використання “ Інтерпретація результатів та управління доступом ”.

Таблиця 2.10 – Винятки сценарію виконання варіанта використання ” Інтерпретація результатів та управління доступом ”.

Дії актора	Відгук системи
Виняток №3. Не прийшов результат	
2. Перезавантаження проєкту	1. Повідомлення про помилку

2.3 Діаграма станів

На рисунках 2.2 – 2.4 наведено діаграми станів.

Рисунок 2.2 – Діаграма стану Збір та обробка даних

На першій діаграмі описується Збір та обробка даних , діаграма виконана послідовно, має старт та фінал. Має 8 Simple State які послідовно з’єднанні друг за другом.

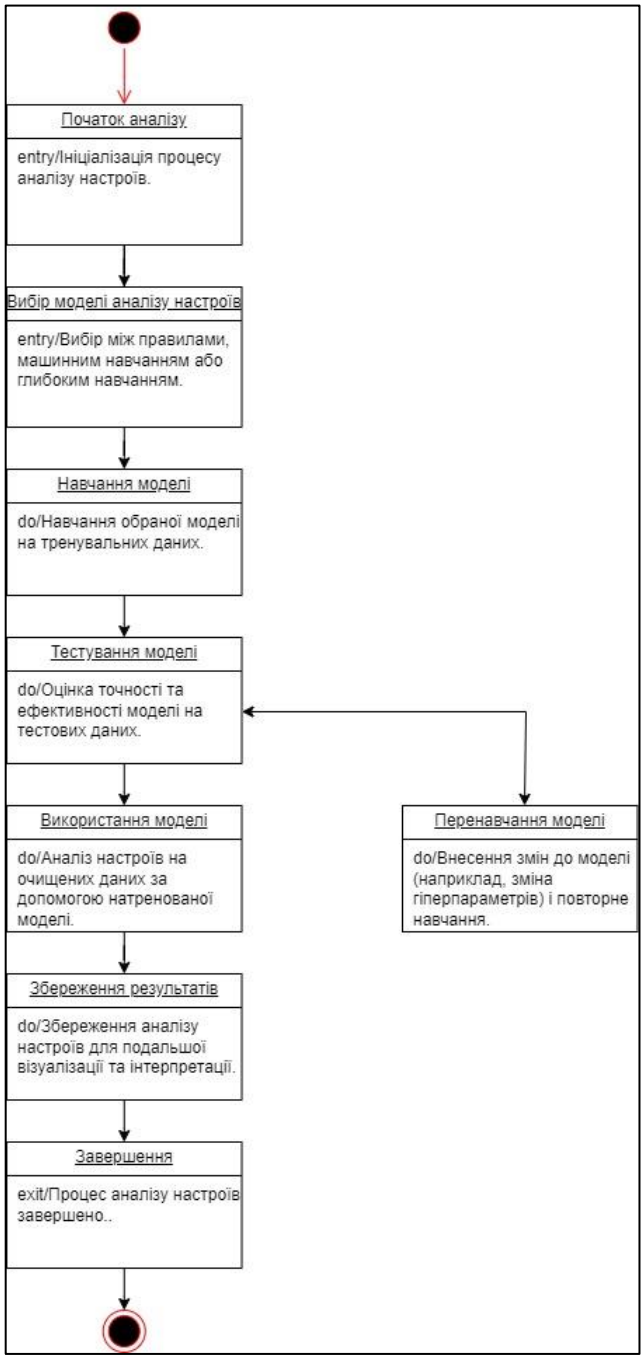

Рисунок 2.3 – Діаграма стану Аналіз настроїв

На другій діаграмі описується Аналіз настроїв, діаграма має 8 State, кожен state має в собі по 1 активності. Перший state має активність entry, другий, третій, четвертий, п’ятий і шостий має активність do, останній state має активність exit.

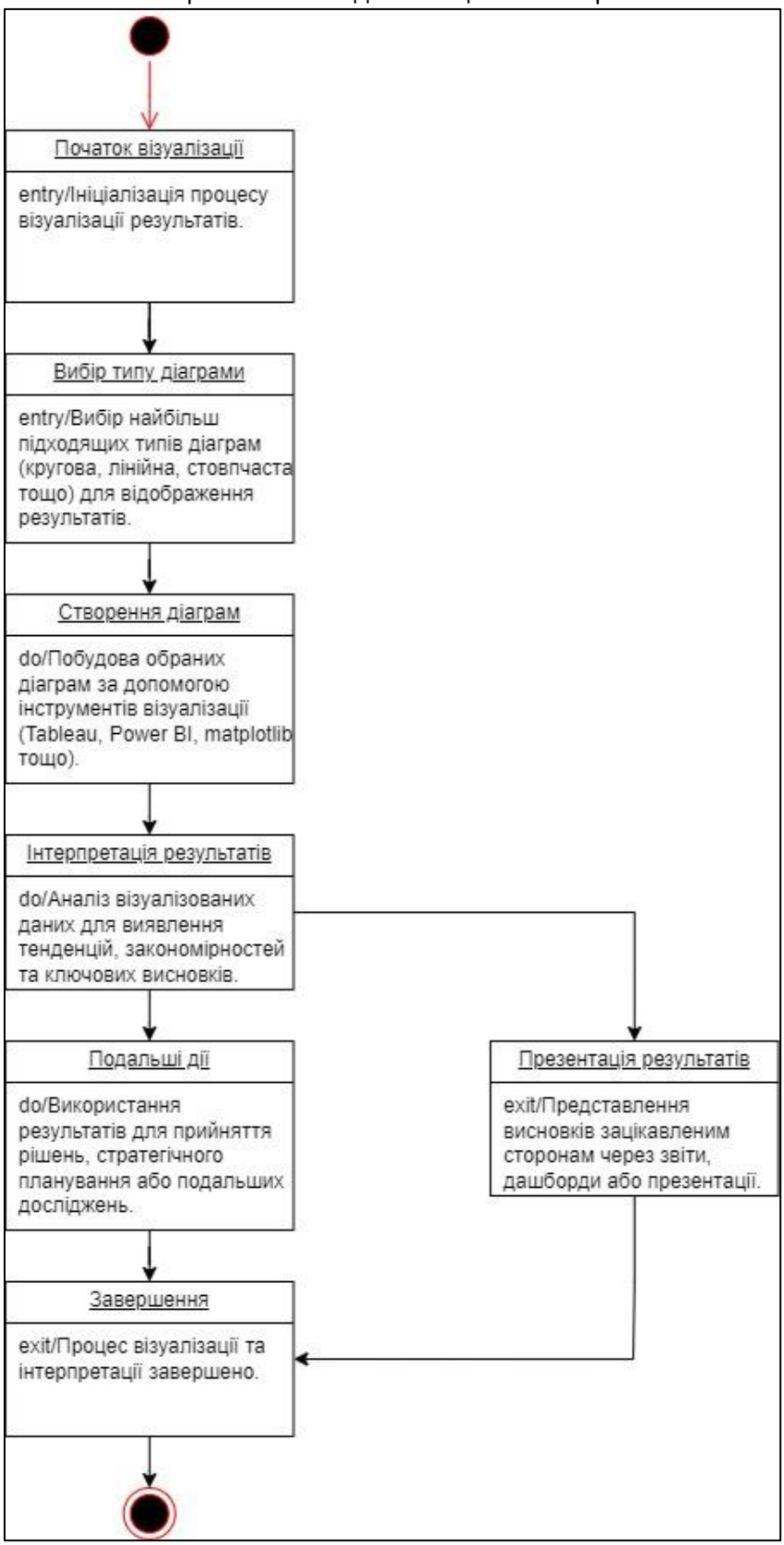

Рисунок 2.4 – Діаграма стану Візуалізація та інтерпретація результатів

На третій діаграмі описується Візуалізація та інтерпретація результатів, має 7 State, кожен з них має по 1 активності entry, do або exit.

Висновки до розділу 2

У другому розділі було розглянуто ключові етапи процесу аналізу настроїв на основі даних соціальних мереж. Було розроблено діаграми варіантів використання, які демонструють основні взаємодії між аналітиками, адміністраторами та системою збору даних. Ці діаграми відображають логічну послідовність дій від збору та обробки даних до аналізу настроїв і візуалізації результатів.

Було окреслено три основні етапи:

1) Збір та попередня обробка даних.

2) Аналіз настроїв і візуалізація результатів.

3) Інтерпретація результатів та управління доступом.

Детально проаналізовано функціональні можливості системи й описано, як різні користувачі (аналітики та адміністратори) взаємодіють із цією системою для досягнення кінцевої мети — отримання зрозумілих і корисних результатів аналізу настроїв, які можуть бути використані для прийняття стратегічних рішень. Система підтримує як автоматизовані процеси збору даних, так і контроль якості даних та управління доступом, що забезпечує її ефективне функціонування.

3 ПРОЄКТУВАННЯ ЗАСТОСУНКУ

3.1 Методи обробки природного тексту

Обробка природного тексту (Natural Language Processing, NLP) є важливою складовою сучасних систем штучного інтелекту. Вона дозволяє комп'ютерам розуміти, аналізувати та генерувати текст, що є важливим для багатьох додатків, зокрема в аналізі настроїв на основі даних з соціальних мереж. У даному розділі будуть розглянуті основні методи обробки тексту, зокрема токенізація, лемматизація, а також трансформери, зокрема модель DistilBERT, та їх застосування в NLP.

3.1 Токенізація

Токенізація є одним з основних етапів обробки природного тексту. Цей процес полягає в поділі тексту на менші одиниці, звані токенами. Токенами можуть бути слова, фрази, символи або навіть рядки тексту, що мають смислове навантаження.

Токенізація є першим кроком для подальшої обробки тексту, адже без цього етапу складно здійснити наступні операції, такі як лемматизація чи аналіз частин мови.

Приклад токенізації:

- Текст: "Ми вивчаємо штучний інтелект"

- Після токенізації: ["Ми", "вивчаємо", "штучний", "інтелект"]

Існують різні типи токенізації:

- Словесна токенізація — коли текст ділиться на окремі слова.

- Токенізація за допомогою пробілів та пунктуації — більш складний підхід, де токенами можуть бути не лише слова, а й розділові знаки.

Для токенізації в Python широко використовуються бібліотеки NLTK або spaCy, які надають вбудовані функції для обробки тексту.

3.2 Лемматизація

Лемматизація — це процес перетворення слова до його базової форми, або леми. Лемма — це форма слова, яка зберігає основне значення. На відміну від стеммінгу, який видаляє суфікси, лемматизація враховує контекст і граматичні правила, щоб отримати правильну лему.

Приклад лемматизації:

- Слова: "йшов", "йшла", "йти" → Лема: "йти"

Лемматизація є більш точним методом обробки тексту, оскільки вона забезпечує коректне розуміння значення слова, навіть якщо воно змінене граматично.

Для лемматизації використовуються різні інструменти, наприклад, бібліотеки spaCy або nltk, які застосовують морфологічні правила для визначення леми слова.

3.3 Трансформери в обробці тексту

Трансформери є однією з найпотужніших архітектур для обробки природного тексту. Модель трансформера була представлена у 2017 році в статті "Attention is All You Need" і швидко стала основою для багатьох досягнень у сфері NLP. Трансформери відрізняються від попередніх моделей, таких як рекурентні нейронні мережі (RNN), тим, що вони не залежать від послідовності вхідних даних і можуть обробляти всю інформацію одночасно за допомогою механізму "уваги" (attention).

Трансформери складаються з двох основних частин: енкодера та декодера. У контексті NLP часто використовується лише енкодер, який відповідає за перетворення тексту в ембеддинги, що представляють зміст тексту у вигляді векторів.

Механізм уваги дозволяє моделі зважати на всі слова в реченні при обробці кожного окремого слова. Це дозволяє трансформеру ефективно працювати з довгими текстами, де важливо враховувати контекст кожного слова.

3.4 DistilBERT

DistilBERT — це легка версія моделі BERT (Bidirectional Encoder Representations from Transformers), яка була розроблена для того, щоб зберегти переваги трансформерів при зменшеному обсязі параметрів і покращеній швидкості роботи. DistilBERT є результатом техніки Distillation, при якій велика модель передає свої знання більш компактній моделі. Завдяки цьому DistilBERT зберігає високу точність, але має значно меншу кількість параметрів, що робить його швидшим і менш вимогливим до ресурсів.

Процес навчання DistilBERT:

- Навчання великої моделі BERT: Спочатку модель BERT навчається на великій кількості текстових даних.

- Distillation: Потім використовуються техніки дистиляції для створення компактної версії моделі, яка зберігає основні властивості і навички.

- Тестування і впровадження: DistilBERT перевіряється на різних задачах NLP, щоб гарантувати, що його продуктивність достатня для практичних застосувань.

На рисунку 3.1 наведено діаграму IDEF0 другого рівня.

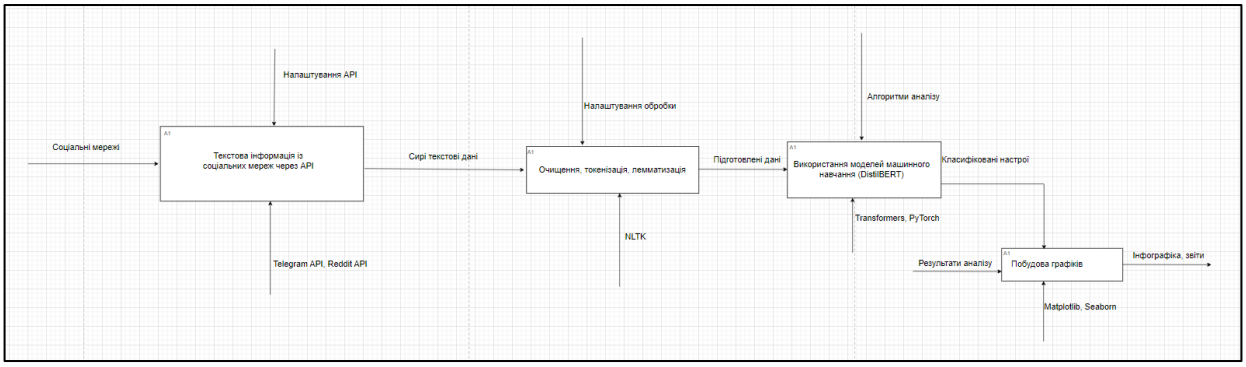

Рисунок 3.1 – Діаграма IDEF0 другого рівня

- DistilBERT є ефективним вибором для задач, де потрібно досягти хороших результатів, але обмеження за ресурсами (наприклад, пам'ять або час на обробку) є важливими.

3.5 Використання трансформерів для аналізу настроїв

Аналіз настроїв (sentiment analysis) є однією з основних задач обробки природного тексту, що дозволяє автоматично визначати емоційне забарвлення тексту — чи є він позитивним, негативним, чи нейтральним. Оскільки велику роль у цій задачі відіграє контекст кожного слова у тексті, то для вирішення цього завдання широко використовуються трансформери, зокрема модель DistilBERT, яка є більш компактною і ефективною версією оригінальної BERT (Bidirectional Encoder Representations from Transformers).

На рисунку 3.2 наведено діаграму IDEF3.

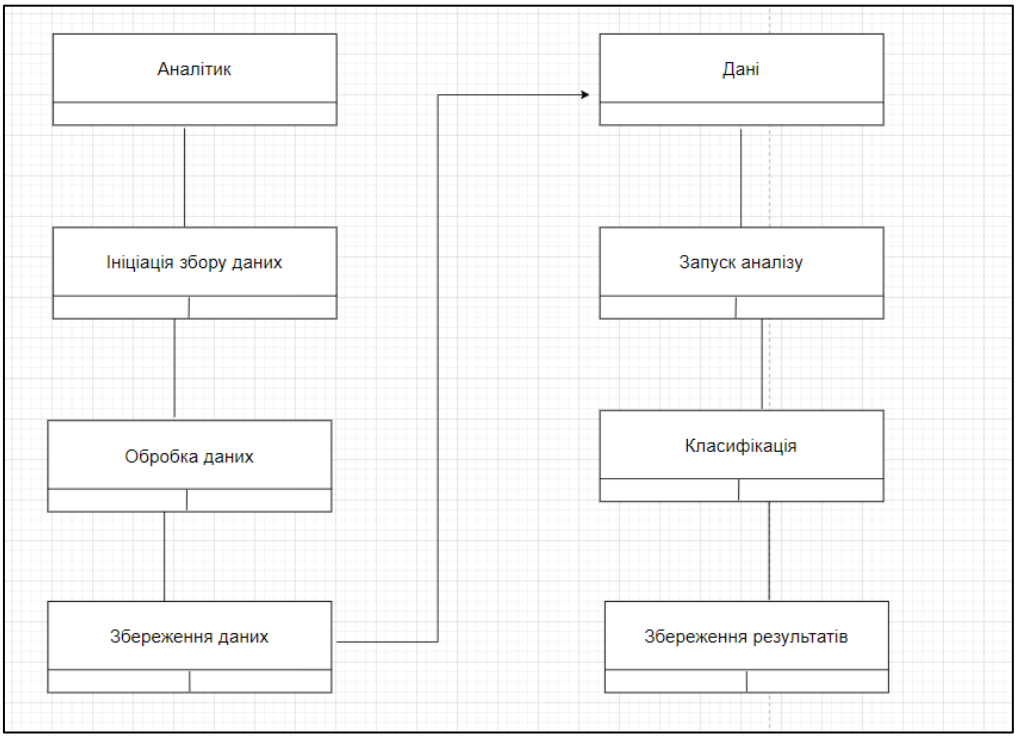

Рисунок 3.2 – Діаграма IDEF3

Моделі на основі трансформерів працюють за допомогою механізму уваги, що дозволяє обробляти контекст кожного слова в реченні незалежно від його позиції. Завдяки цьому моделі можуть точно вловлювати значення слів, яке змінюється в залежності від контексту, що є важливим аспектом для аналізу настроїв у текстах.

У задачі аналізу настроїв трансформери можуть працювати з великими обсягами тексту, точно визначаючи, чи є цей текст позитивним або негативним.

Це дозволяє застосовувати модель до текстів з різних джерел, таких як соціальні мережі, форуми, новини тощо.

3.6 Обробка тексту з різних джерел

Для збору тексту з різних платформ використовуються специфічні бібліотеки, які дозволяють автоматично отримувати дані з Telegram або Reddit. Наприклад, Telethon є популярною бібліотекою для взаємодії з Telegram, а Praw — для Reddit. Ці інструменти дозволяють зібрати великі обсяги текстових даних для подальшої обробки та аналізу.

Після збору даних з платформ, їх необхідно попередньо обробити, щоб привести текст до зручного для аналізу вигляду. Однією з основних процедур є токенізація — процес розбиття тексту на окремі елементи (токени), такі як слова чи фрази. Це дозволяє моделі краще працювати з текстом, оскільки кожен токен є окремим об'єктом для обробки.

Наступним етапом є лемматизація, яка полягає в приведенні слів до їх базової форми (леми). Це необхідно, оскільки різні форми одного й того ж слова можуть призвести до зменшення ефективності моделі. Лемматизація дозволяє зменшити кількість варіантів слів, з якими модель повинна працювати.

3.7 Візуалізація результатів аналізу настроїв

Один з важливих етапів у роботі з текстовими даними — це візуалізація результатів аналізу. Після того, як текст пройшов обробку та був проаналізований моделлю на предмет настроїв, результати можуть бути представлені у вигляді графіків, що дозволяють легко зрозуміти, який настрій переважає серед текстів. Наприклад, можна побудувати стовпчикові графіки, що покажуть кількість позитивних і негативних повідомлень. Це допомагає наочно оцінити загальний емоційний фон зібраних даних.

Візуалізація допомагає краще інтерпретувати результати, зокрема, для аналізу настроїв на різних платформах, таких як Telegram або Reddit. Це важливо для подальших висновків про характер спілкування користувачів на цих платформах і дозволяє виявити тенденції та патерни у настроях.

3.8 Оцінка ефективності моделей

Для оцінки ефективності моделей, які використовуються для аналізу настроїв, важливо застосовувати різні метрики. Однією з основних є точність (accuracy), яка показує, скільки з передбачених моделюваних класів (позитивний або негативний настрій) виявилися правильними порівняно з реальними класами.

Крім того, широко використовуються такі метрики, як F1-метрика, яка є середнім гармонічним точності та відгуку, і відгук (recall), що вимірює здатність моделі знаходити всі позитивні приклади. Ці метрики дозволяють отримати більш точну картину ефективності моделі, особливо в умовах, коли дані можуть бути незбалансованими (наприклад, більшість повідомлень можуть бути нейтральними, а лише мала частина — позитивними чи негативними).

Після виконання аналізу результатів за допомогою цих метрик можна оцінити, наскільки добре модель справляється з завданням класифікації настроїв у конкретних наборах даних.

Висновки до розділу 3

У третьому розділі було застосування трансформерів для аналізу настроїв є потужним інструментом, що дозволяє ефективно обробляти великі обсяги текстових даних і точно визначати емоційне забарвлення. Використання таких моделей, як DistilBERT, забезпечує високу точність при роботі з контекстом кожного слова в тексті. Процес попередньої обробки, який включає токенізацію, лемматизацію та очищення тексту, є необхідним етапом для підготовки даних до аналізу.

Візуалізація результатів аналізу настроїв дає змогу наочно оцінити емоційний фон текстів і робити висновки про загальні тенденції на платформах, таких як Telegram і Reddit. Оцінка ефективності моделей через метрики, такі як точність, F1-метрика та відгук, дозволяє зрозуміти, наскільки добре модель виконує поставлену задачу, і визначити шляхи для її покращення.

4 ПРОЄКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ

4.1 Вибір на налаштування платформ для аналізу

Аналіз настроїв текстового контенту є важливим завданням, яке дозволяє зрозуміти емоційний стан користувачів у соціальних мережах. У цьому проєкті було вирішено працювати з платформами Telegram і Reddit. Обидві платформи відрізняються високою популярністю, активною аудиторією та великою кількістю текстового контенту, що робить їх ідеальними для збору даних і проведення аналізу (рис. 3-3.1) [11].

На рисунку 4.1 – зображено діаграма кількість користувачів використовуючи застосунок Телеграм.

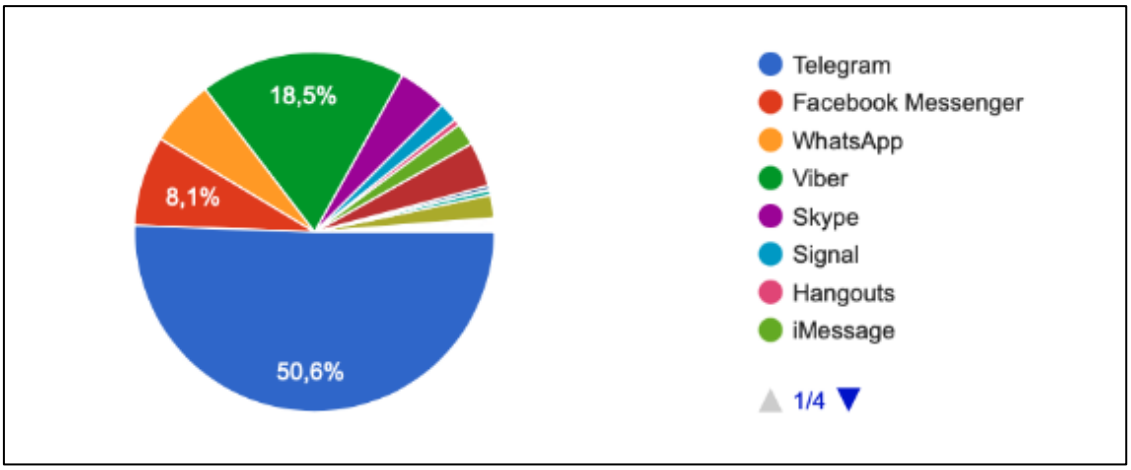

Рисунок 4.1 – Діаграма користувачів використовуючи застосунок Телеграм

На рисунку 4.2 – зображено діаграму популярних для обговорення застосунків.

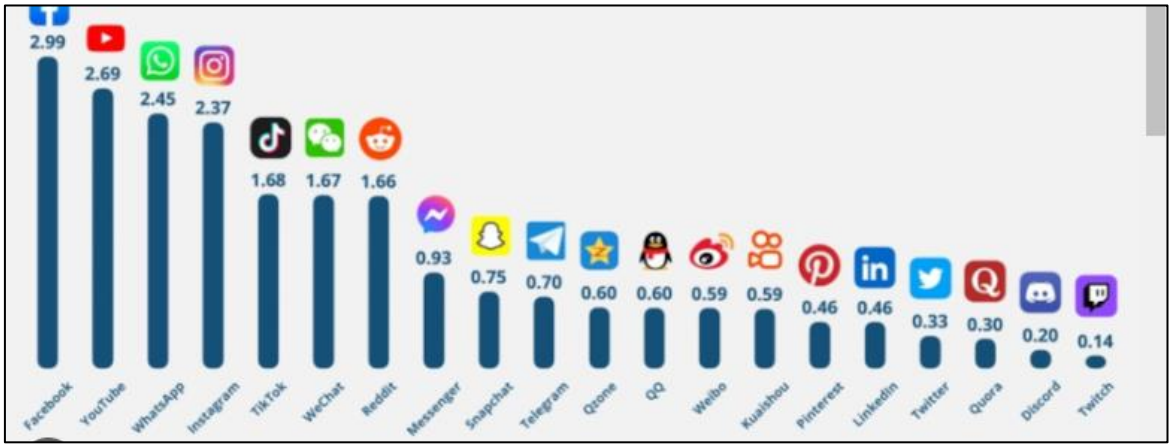

Рисунок 4.2 – Діаграма популярних застосунків для обговорення

Telegram використовується як один із провідних месенджерів з відкритим API, що дозволяє отримувати доступ до даних з каналів, чатів або груп. Його популярність і доступність даних через офіційний API надають можливість зібрати різноманітну інформацію для аналізу.

Reddit, у свою чергу, є платформою для створення і обговорення тем у різних підрозділах (subreddits). Тут можна знайти дискусії на найрізноманітніші теми. Reddit API дозволяє автоматизовано отримувати доступ до цих дискусій, що значно спрощує збір даних для аналізу [15].

4.2 Налаштування доступу до платформ

Для збору даних було проведено ретельну підготовку доступу до API Telegram і Reddit. Це вимагало попередньої реєстрації в обох системах, отримання ключів доступу та налаштування середовища для роботи.

4.2.1 Telegram API

Доступ до даних Telegram забезпечується через Telegram Bot API або Telegram User API. У цьому проєкті використовувався User API, оскільки він дозволяє отримувати дані з відкритих каналів, а не лише через ботів. Для налаштування доступу виконувались наступні кроки:

Реєстрація додатка:

Відкрито сайт Telegram API.

Перейдено до розділу "API Development Tools".

Створено новий додаток, після чого отримано унікальні API_ID та API_HASH [13].

Інтеграція з Python:

Встановлено бібліотеку Telethon, яка забезпечує доступ до Telegram API.

Налаштовано клієнт для роботи з каналом за допомогою сесії.

На рисунку 4.3 – зображено API інформація користувача.

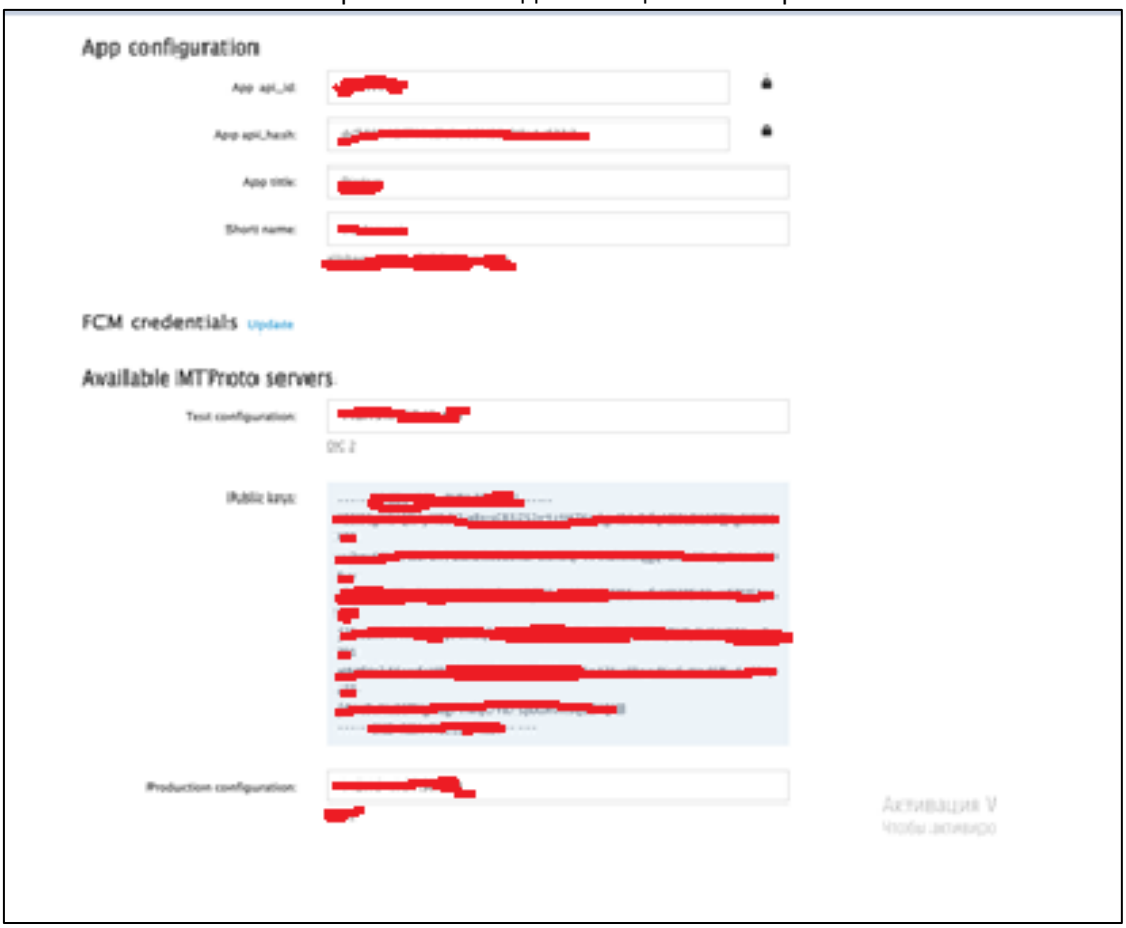

Рисунок 4.3 – API інформація користувача

4.2.2 Reddit API

Reddit API, на відміну від Telegram, є централізованішою платформою для отримання структурованих даних. Процес налаштування виглядає наступним чином:

Реєстрація додатку:

На сайті Reddit Developer Console створено новий додаток.

Отримано значення CLIENT_ID та CLIENT_SECRET.

Інтеграція з Python:

Встановлено бібліотеку praw.

Виконано авторизацію для доступу до API, вказавши облікові дані.

На рисунку 4.4 – зображено API інформація користувача Reddit.

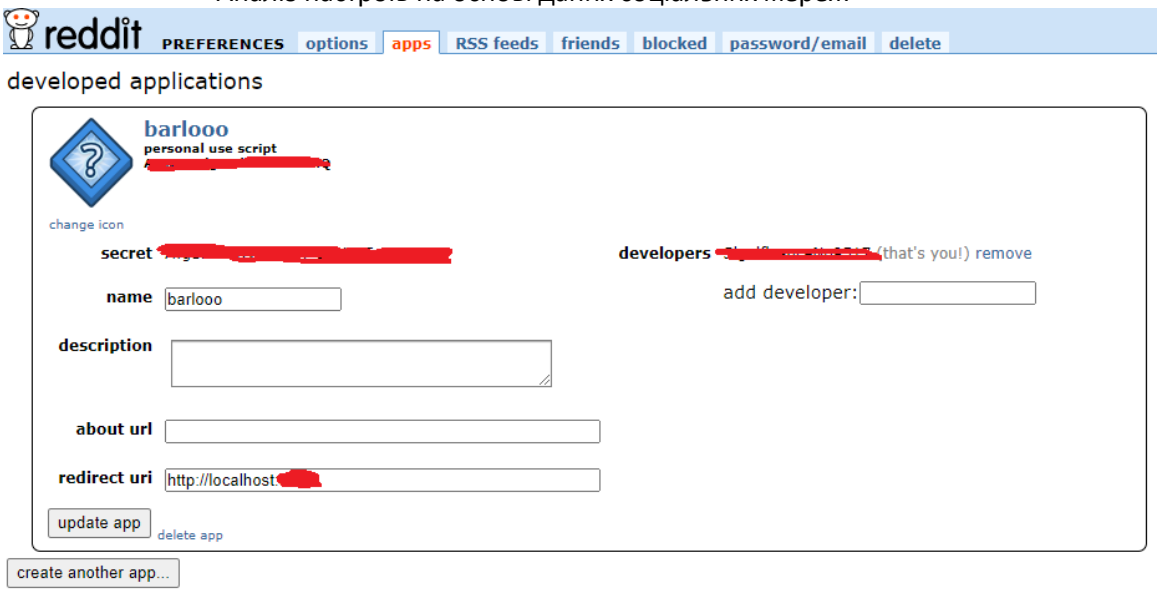

Рисунок 4.4 – API інформація користувача Reddit

4.3 Попередня обробка текстів

Зібрані дані з обох платформ у своєму початковому вигляді не підходять для аналізу, оскільки містять зайві символи, посилання, стоп-слова тощо. Тому важливим етапом проєктування була попередня обробка тексту.

На рисунку 4.5 наведено діаграму IDEF0 першого рівня.

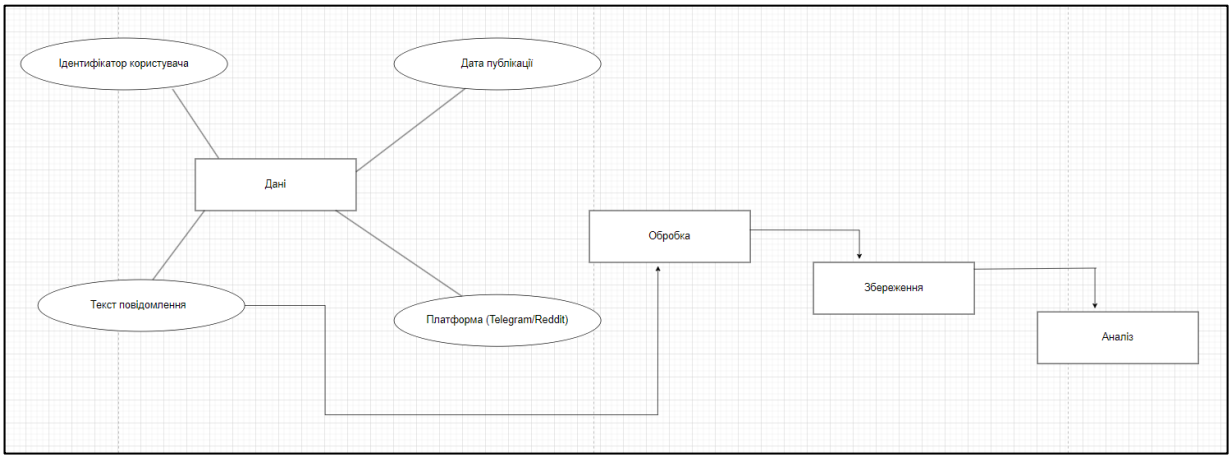

Рисунок 4.5 – Діаграму IDEF0 першого рівня

Обробка тексту складалася з наступних етапів:

Очищення тексту:

- Видалення URL-посилань, хештегів, згадок користувачів та іншого зайвого тексту, який не несе семантичного навантаження.

- Видалення пунктуації та спеціальних символів.

Нормалізація тексту:

- Перетворення тексту до нижнього регістру для забезпечення уніфікації.

Токенізація:

- Розбивання тексту на окремі слова (токени) для подальшої роботи.

Фільтрація стоп-слів:

- Видалення часто вживаних слів (наприклад, "і", "але", "на"), які не мають значення для аналізу настроїв.

Лематизація:

- Зведення слів до їхньої базової форми для зменшення варіативності мовлення.

На рисунку 4.6 наведено діаграма IDEF0 другого рівня деталізація процесу обробки текстів.

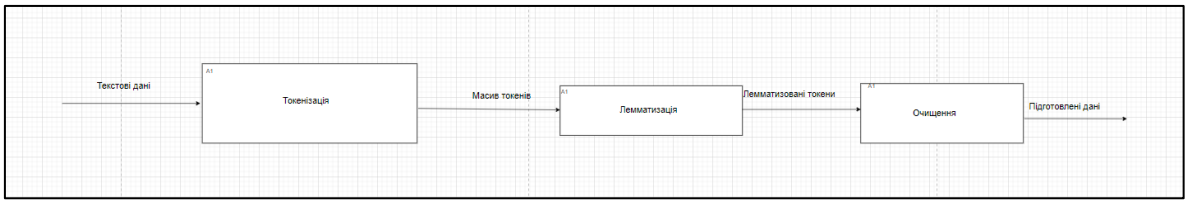

Рисунок 4.6 – Діаграма IDEF0 другого рівня деталізація процесу обробки текстів.

Цей процес забезпечує підготовку тексту до аналізу, роблячи його придатним для роботи з алгоритмами машинного навчання.

4.4 Моделювання аналізу настроїв

Для аналізу настроїв було використано сучасні алгоритми машинного навчання. Зокрема, застосовано модель на основі трансформерів, яка забезпечує високу точність визначення настроїв [13].

Обрано модель distilbert-base-uncased-finetuned-sst-2-english, яка була попередньо навчена на англомовному датасеті для класифікації текстів на позитивні та негативні. Ця модель є спрощеною версією BERT, що робить її швидшою у використанні, не знижуючи якості результатів.

Модель аналізує текст, визначаючи тональність кожного повідомлення. Для цього текст подається на вхід алгоритму, і на виході отримується один із

двох результатів: **позитивний** або **негативний** настрій. У випадку довгих повідомлень текст обмежується до 512 символів, що відповідає максимальній довжині послідовності для моделі [13].

4.5 Загальна структура застосунку

Застосунок побудовано у вигляді багаторівневої системи, що включає кілька основних модулів:

Модуль збору даних:

- Відповідає за отримання текстового контенту з Telegram та Reddit через API.

Модуль обробки текстів:

- Забезпечує очищення та нормалізацію текстових даних перед аналізом.

Модуль аналізу настроїв:

- Використовує алгоритми машинного навчання для класифікації настроїв текстів.

Модуль візуалізації:

- Показує результати аналізу у вигляді графіків для наочного уявлення про розподіл настроїв.

На рисунку 4.7 наведено діаграма класів застосунку.

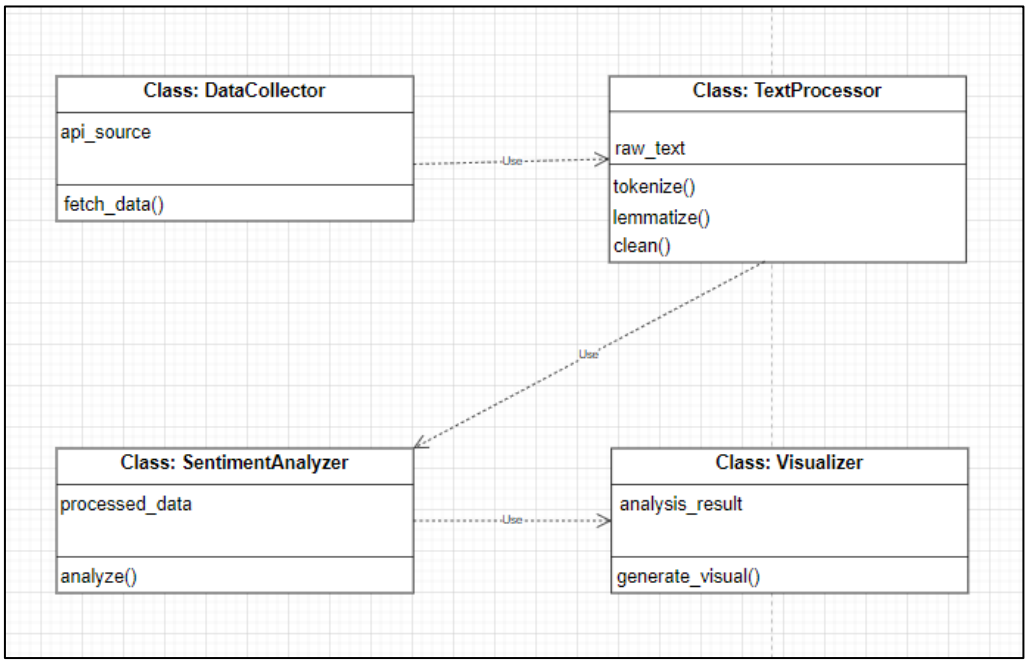

Рисунок 4.7 – Діаграма класів

Клас: DataCollector

Атрибути:

- api_source: джерело даних (наприклад, Reddit або Telegram).

Методи:

- fetch_data(): отримання даних через API.

Class: TextProcessor

Атрибути:

- raw_text: необроблений текст.

Методи:

- tokenize(): токенізація тексту.

- lemmatize(): лематизація тексту.

- clean(): очищення тексту від зайвих символів.

Class: SentimentAnalyzer

Атрибути:

- processed_data: очищені та підготовлені дані.

Методи:

- analyze(): аналіз настроїв з використанням моделі.

Class: Visualizer

Атрибути:

- analysis_result: результати аналізу настроїв.

Методи:

- generate_visual(): генерація візуалізацій для результатів.

Ця структура дозволяє легко масштабувати застосунок, додаючи нові функції або джерела даних у разі потреби.

На рисунку 4.8 наведено діаграма архітектури застосунку.

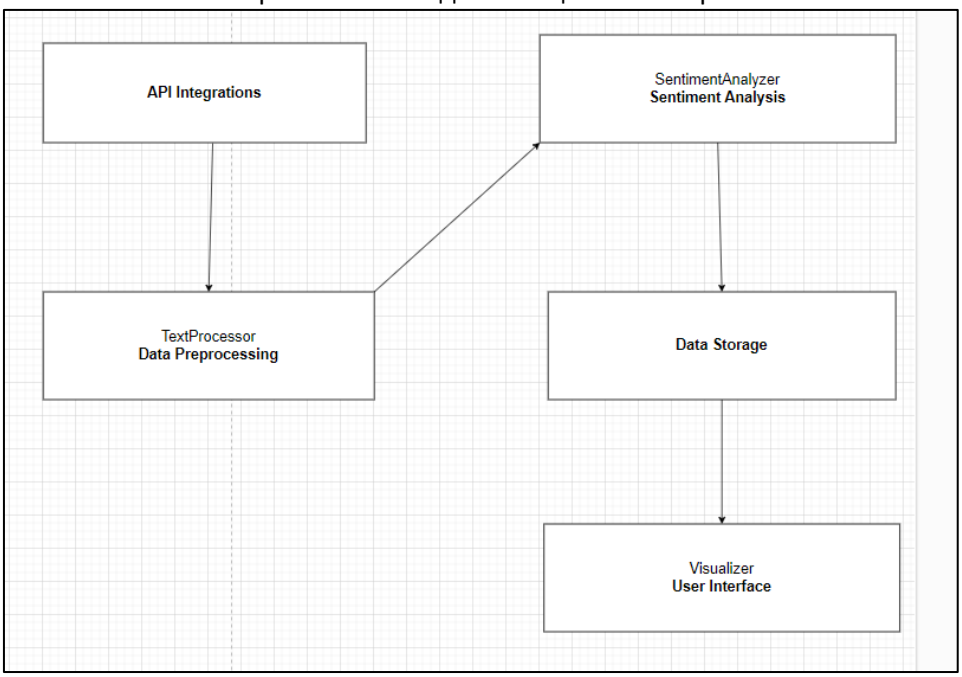

Рисунок 4.8 – Діаграма архітектури застосунку

API Integrations → Data Preprocessing

- Дані, зібрані через Telegram і Reddit API, передаються на етап обробки (наприклад, через API-запити).

- Стрілка вказує, що дані передаються для подальшої обробки.

Data Preprocessing → Sentiment Analysis

- Після того як дані очищено та підготовлено, їх передають у компонент аналізу настроїв для аналізу тексту.

Sentiment Analysis → Data Storage

- Результати аналізу (наприклад, класифікація настроїв) передаються для зберігання в базі даних.

Data Storage → User Interface

- Оброблені та збережені дані передаються в інтерфейс, щоб користувач міг бачити результати у візуалізованому вигляді.

4.3 Огляд використаних бібліотек

Для реалізації функціональності застосунку було використано низку бібліотек Python, які забезпечують роботу з API, обробку тексту, аналіз настроїв

та візуалізацію результатів. Кожна бібліотека має свої унікальні переваги та роль у проєкті [12].

4.3.1 Telethon

Telethon — це асинхронна Python-бібліотека для взаємодії з Telegram API. Вона дозволяє отримувати повідомлення з каналів, надсилати повідомлення, створювати боти та багато іншого [11].

Переваги:

Легкість у використанні.

Підтримка асинхронного програмування, що підвищує ефективність роботи з великим обсягом даних.

Роль у проєкті:

Використовується для отримання текстових повідомлень із Telegram каналів.

4.3.2 PRAW

PRAW (Python Reddit API Wrapper) — це бібліотека, яка надає простий і зручний інтерфейс для взаємодії з Reddit API.

Переваги:

Простота в отриманні публікацій і коментарів із підрозділів Reddit.

Широкі можливості для пошуку та фільтрації контенту.

Роль у проєкті:

Використовується для збору текстового контенту з підрозділу r/Ukraine.

4.3.3 NLTK

NLTK (Natural Language Toolkit) — це набір інструментів для обробки природної мови. Він містить методи для токенізації, лематизації, видалення стоп-слів та інших завдань текстової обробки.

Переваги:

Широкий набір функцій для роботи з текстами.

Підтримка англійської мови та наявність готових словників.

Роль у проєкті:

Застосовується для попередньої обробки текстів із Telegram та Reddit, включаючи очищення тексту, лематизацію та видалення стоп-слів.

4.3.4 Transformers

Transformers — це бібліотека від Hugging Face, яка забезпечує доступ до сучасних моделей глибинного навчання для обробки природної мови, зокрема BERT, DistilBERT та інших.

Переваги:

Можливість використання попередньо навчених моделей для аналізу тексту.

Підтримка обробки великих обсягів текстових даних.

Роль у проєкті:

Використовується для аналізу настроїв текстів за допомогою моделі distilbert-base-uncased-finetuned-sst-2-english.

4.3.5 Matplotlib і Seaborn

Ці бібліотеки використовуються для візуалізації результатів аналізу.

Matplotlib забезпечує створення графіків і діаграм, дозволяючи відобразити розподіл настроїв.

Seaborn надає інструменти для створення більш стильних і зрозумілих графіків, інтегруючись із Matplotlib.

Роль у проєкті:

Візуалізація результатів аналізу настроїв для наочного представлення даних.

4.7 Аналіз результатів роботи застосунку

Застосунок побудовано таким чином, що кожен із етапів роботи виконується незалежно, але при цьому всі частини тісно взаємодіють між собою. Після того, як усі необхідні бібліотеки завантажено, ініціалізуються підключення

до зовнішніх платформ (Telegram та Reddit) за допомогою їхніх відповідних API-ключів.

Завантаження бібліотек: Першим кроком завантажуються усі необхідні бібліотеки та модулі. Вони служать для виконання конкретних завдань: взаємодія з платформами, обробка текстів, аналіз настроїв і побудова графіків.

Підключення до Telegram і Reddit: Далі відбувається підключення до платформ Telegram та Reddit через API. Для цього використовуються відповідні бібліотеки (Telethon і PRAW). Важливою частиною є отримання унікальних ключів доступу до кожної з платформ, що забезпечує безпеку та автентифікацію при взаємодії з серверами.

Збір даних: Після підключення застосунок отримує необхідні дані. У випадку з Telegram — це повідомлення з певного каналу, у випадку з Reddit — пости з субреддіту. Дані, які отримуються із цих платформ, можуть бути у вигляді тексту повідомлень або постів, іноді разом з додатковою інформацією, яку необхідно зберегти для аналізу.

Попередня обробка текстів: Важливим етапом є очищення і підготовка текстів. Це включає видалення непотрібних символів (URL, теги, спеціальні знаки), приведення тексту до нижнього регістру, а також токенізацію та лематизацію. Такі операції допомагають зменшити обсяг даних і підготувати їх для подальшого аналізу.

Аналіз настроїв: Після попередньої обробки тексти передаються в модель DistilBERT для виконання аналізу настроїв. Це дає змогу визначити, чи є текст позитивним, негативним чи нейтральним. Результати зберігаються в списку, який потім використовується для візуалізації.

На рисунках 4.1 – 4.2 зображено результат програми.

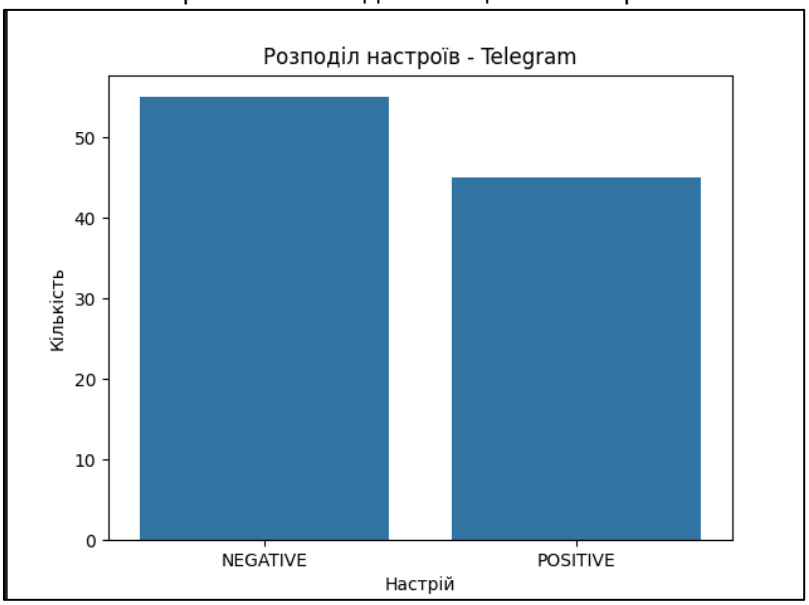

Рисунок 4.9 – Графік Telegram

Рисунок 4.10 – Графік Reddit

Візуалізація результатів: Останнім кроком є візуалізація результатів аналізу настроїв за допомогою графіків, що дозволяє швидко оцінити настрій даних на різних платформах. Графік містить розподіл за категоріями: позитивний, негативний або нейтральний настрій, що дає можливість проаналізувати великі обсяги даних і побудувати корисні висновки.

Висновки до розділу 4

У четвертому розділі було детально описано процес проєктування застосунку для аналізу настроїв на основі текстових даних із платформ Telegram

та Reddit. Ми визначили основні етапи створення застосунку, обґрунтували вибір платформ і описали підхід до інтеграції з їхніми API.

Окрему увагу приділено огляду бібліотек, які забезпечили ефективність роботи із текстовими даними. Telethon і PRAW надали можливість зручно збирати дані з обраних платформ. Бібліотека NLTK забезпечила інструменти для попередньої обробки текстів, включаючи токенізацію, лематизацію та видалення стоп-слів. Завдяки бібліотеці Transformers від Hugging Face було реалізовано аналіз настроїв із використанням сучасних моделей глибинного навчання. Для візуалізації результатів застосовано бібліотеки Matplotlib і Seaborn, що дозволило представити отримані дані у вигляді зрозумілих графіків.

Проєктування системи продемонструвало її модульну структуру, яка складається з чотирьох ключових компонентів: збору даних, їхньої попередньої обробки, аналізу настроїв та візуалізації. Такий підхід дозволяє легко розширювати функціональність застосунку в майбутньому, додаючи нові джерела даних або вдосконалюючи методи аналізу.

Таким чином, описаний процес проєктування закладає основу для успішної реалізації застосунку, результати якої буде розглянуто в наступному розділі.

ВИСНОВКИ

Кваліфікаційна робота присвячена питанням дослідження та вдосконалення напрямку розробки та впровадження технологій для аналізу настроїв на основі даних із соціальних мереж. Аналізу даних із платформ, таких як Telegram та Reddit, та їх подальшому аналізу за допомогою методів обробки природної мови (NLP) та аналізу настроїв, присвячена основна увага роботи. Такий підхід дозволяє отримувати важливу інформацію для визначення емоційної тональності текстів, що є корисним у маркетингових, соціальних та бізнес-дослідженнях.

Результатом виконання кваліфікаційної роботи є функціонуючий застосунок для аналізу настроїв на основі даних із Telegram та Reddit, який дозволяє автоматично збирати повідомлення з цих платформ та визначати настрій користувачів у контексті певних тем чи трендів. Застосунок використовує передові бібліотеки для обробки тексту та аналізу настроїв, зокрема nltk, Transformers, та Matplotlib для візуалізації результатів.

Розробка цього застосунку є актуальним прикладом використання сучасних методів обробки тексту та машинного навчання для аналізу великих обсягів даних з соціальних платформ. Платформи, як Telegram та Reddit, є важливими джерелами інформації, і розробка інструментів для аналізу цих даних дозволяє швидко отримувати уявлення про настрої широкої аудиторії.

В процесі виконання кваліфікаційної роботи було вирішено наступні завдання: – Проаналізовано існуючі інструменти та методи для аналізу настроїв на основі даних з соціальних мереж; визначено їх сильні та слабкі сторони; оцінено можливості збору даних з таких платформ, як Telegram та Reddit. – Спроектовано та змодельовано структуру програмного забезпечення для аналізу настроїв, що включає підключення до API Telegram та Reddit, а також модуль для обробки тексту та визначення емоційної тональності за допомогою моделей DistilBERT. – Розроблено основні функціональні модулі застосунку, що включають збір повідомлень, їх попередню обробку та аналіз настроїв.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. McKinney, W. Python for Data Analysis. New Riders, 2005. p. 27.

2. Matthes, E. Python Crash Course: A Hands-On, Project-Based Introduction to Programming. O'Reilly Media, 2011. p. 168.

3. Berry, P. Head First Python: A Brain-Friendly Guide. New Riders, 2011. p. 231.

4. Sweigart, A. Automate the Boring Stuff with Python: Practical Programming for Total Beginners, 2007. p. 48.

5. Lutz, M. Learning Python: Powerful Object-Oriented Programming, 2009. p. 15.

6. Shaw, Z. E. Learn Python 3 the Hard Way, 2003. p. 292.

7. Downey, A. B. Think Python: How to Think Like a Computer Scientist. O'Reilly Media, 2014 p. 215.

8. Müller, C. S. Introduction to Machine Learning with Python: A Guide for Data Scientists, 2008. p. 79.

9. Luciano, R. Fluent Python: Clear, Concise, and Effective Programming. New Riders, 2014. p. 61.

10. Beazley, D. Python Cookbook: Recipes for Mastering Python 3. Wiley, 2014. p. 12.

11. Rushel, T. Python for Data Analysis: A Practical Guide. Wiley, 2019. p. 85.

12. Vasquez, J. Sentiment Analysis in Python: A Practical Approach. Packt Publishing, 2018. p. 128.

13. Doyle, K. Deep Learning with Python: Concepts and Applications. Springer, 2020. p. 312.

14. Brownlee, J. Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models. CreateSpace, 2017. p. 176.

15. Smith, E. Practical Natural Language Processing with Python. Manning Publications, 2021. p. 203.

16. Bishop, C. M. Pattern Recognition and Machine Learning. Springer, 2006. p. 450.

17. Chollet, F. Deep Learning with Python. Manning Publications, 2018. p. 212.

18. VanderPlas, J. Python Data Science Handbook. O'Reilly Media, 2016. p. 380.

19. Alpaydin, E. Introduction to Machine Learning. MIT Press, 2020. p. 256.

20. Géron, A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, 2019. p. 604.

21. Zhang, L. Sentiment Analysis: Mining Opinions, Sentiments, and Emotions. Springer, 2018. p. 290.

22. Raj, S. Natural Language Processing with Python and SpaCy. Packt Publishing, 2019. p. 150.

23. Mitchell, T. M. Machine Learning. McGraw-Hill, 1997. p. 436.

24. Manning, C. D., & Schütze, H. Foundations of Statistical Natural Language Processing. MIT Press, 1999. p. 657.

25. Ng, A. Machine Learning Yearning. deeplearning.ai, 2018. p. 112.

26. Zhang, Y., & Zhang, X. Applied Machine Learning with Python: Solve Data Analysis Problems Using Python. Packt Publishing, 2019. p. 326.

27. Goodfellow, I., Bengio, Y., & Courville, A. Deep Learning. MIT Press, 2016. p. 775.

28. Brown, M. Introduction to Python Programming. Pearson, 2020. p. 402.

29. Chen, T., & Guestrin, C. XGBoost: A Scalable Tree Boosting System. ACM SIGKDD Explorations Newsletter, 2016. p. 78-85.

30. Sejnowski, T. J. The Deep Learning Revolution. The MIT Press, 2018. p. 336.

ДОДАТОК

```
from telethon.sync import TelegramClient
from telethon.tl.types import PeerChannel
import praw
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
from transformers import pipeline
import matplotlib.pyplot as plt
import seaborn as sns


sentiment_pipeline = pipeline("sentiment-analysis", framework="pt")


nltk.download('stopwords')
nltk.download('punkt')
nltk.download('wordnet')
nltk.download('omw-1.4')


API_ID = '22017729'
API_HASH = 'dc580012f711e5d1e95138a29cdc6332'
CHANNEL_ID = 't.me/RunAsHelicopter'


REDDIT_CLIENT_ID = 'AXwBCJdgFVilAzG3HrD6lQ'
```

```
REDDIT_CLIENT_SECRET = 'ATg1mX9Qv469__i_BEYXzTz6X4y9UQ'
REDDIT_USER_AGENT = 'SignificanceNo9517'


def setup_telegram_client():
    client = TelegramClient('session_name', API_ID, API_HASH)
    client.start()
    return client


def fetch_telegram_messages(client, limit=100):
    messages = []
    channel = client.get_entity(CHANNEL_ID)
    for message in client.iter_messages(channel, limit=limit):
        if message.message:
            messages.append(message.message)
    return messages


def setup_reddit_client():
    reddit = praw.Reddit(
        client_id=REDDIT_CLIENT_ID,
        client_secret=REDDIT_CLIENT_SECRET,
        user_agent=REDDIT_USER_AGENT
    )
    return reddit


def fetch_reddit_posts(reddit, subreddit_name, limit=100):
    subreddit = reddit.subreddit(subreddit_name)
```

```
    posts = []
    for post in subreddit.hot(limit=limit):
        posts.append(post.title + " " + post.selftext)
    return posts


def preprocess_text(text):
    text = re.sub(r"http\S+|www\S+|https\S+", '', text, flags=re.MULTILINE)
    text = re.sub(r'\@\w+|\#', '', text)
    text = re.sub(r'[^\w\s]', '', text)
    text = text.lower()
    tokens = word_tokenize(text)
    lemmatizer = WordNetLemmatizer()
    tokens = [lemmatizer.lemmatize(word) for word in tokens if word not in
stopwords.words('english')]
    return ' '.join(tokens)


def analyze_sentiments(messages):
    max_length = 512
    sentiments = []
    for message in messages:
        truncated_message = message[:max_length]
        sentiment = sentiment_pipeline(truncated_message)
        sentiments.append(sentiment)
    return sentiments


def visualize_sentiments(sentiments, platform_name):
    sentiment_labels = [s[0]['label'] for s in sentiments]
```

```
        sns.countplot(x=sentiment_labels)
        plt.title(f"Розподіл настроїв - {platform_name}")
        plt.xlabel("Настрій")
        plt.ylabel("Кількість")
        plt.show()


    def main():
        # Telegram
        telegram_client = setup_telegram_client()
        telegram_messages = fetch_telegram_messages(telegram_client)
        preprocessed_telegram_messages = [preprocess_text(message) for message in telegram_messages]


        telegram_sentiments = analyze_sentiments(preprocessed_telegram_messages)
        visualize_sentiments(telegram_sentiments, "Telegram")


        reddit_client = setup_reddit_client()
        reddit_posts = fetch_reddit_posts(reddit_client, 'Ukraine', limit=100)
        preprocessed_reddit_posts = [preprocess_text(post) for post in reddit_posts]


        reddit_sentiments = analyze_sentiments(preprocessed_reddit_posts)
        visualize_sentiments(reddit_sentiments, "Reddit")

    if __name__ == "__main__":
        main()
```