

**Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення**

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «МАГІСТР»**

**РОЗРОБКА ТА ДОСЛІДЖЕННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ
АНАЛІЗУ ВІДГУКІВ НА ТОВАРИ З ВИКОРИСТАННЯМ МЕТОДІВ
АНАЛІЗУ ТОНАЛЬНОСТІ ТЕКСТУ**

**DEVELOPMENT AND RESEARCH OF AN INTELLIGENT SYSTEM FOR
ANALYZING PRODUCT REVIEWS USING TEXT TONE ANALYSIS
METHODS**

спеціальність 121 «Інженерія програмного забезпечення»
освітня програма «Інженерія програмного забезпечення»

Виконав: здобувач вищої освіти
групи ПЗм-21
Калюжний Д. В.
Керівник:
к.т.н., доцент
Ящук А. А.

Кваліфікаційну роботу
допущено до захисту
«__» _____ 20__ р.
Гарант освітньої програми:
к.т.н., доцент Суринович О.М.

Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення
Ступінь вищої освіти магістр
Галузь знань: 12 «Інформаційні технології»
Спеціальність: 121 «Інженерія програмного забезпечення»
Освітня програма: «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ
Завідувач кафедри

«__» _____ 202__ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА ДРУГОГО (МАГІСТЕРСЬКОГО) РІВНЯ ВИЩОЇ ОСВІТИ

Калюжному Дмитру Валерійовичу

1. Тема кваліфікаційної роботи: Розробка та дослідження інтелектуальної системи аналізу відгуків на товари з використанням методів аналізу тональності тексту.

Керівник роботи: Ящук Андрій Анатолійович, доцент, к.т.н.

затверджені наказом закладу вищої освіти від «29» березня__2025 року № 190/01-02_____

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи: 4 грудня 2025 р.

3. Вихідні дані до роботи технічне та програмне забезпечення EOM

4. Зміст розрахунково-пояснювальної записки: аналіз проблематики визначення тональності тексту та вибір методів дослідження, обґрунтування технологій і реалізація системи аналізу відгуків на товари, експериментальне дослідження результативності програмного забезпечення.

5. Перелік графічного матеріалу 16 рисунків, 4 таблиці, 3 листинги коду.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
<i>Аналіз проблеми за темою роботи та постановка завдань дослідження</i>	<i>Яцук А. А.</i>		
<i>Теоретичне дослідження та практична реалізація</i>	<i>Яцук А. А.</i>		
<i>Експериментальне дослідження системи</i>	<i>Яцук А. А.</i>		
<i>Нормоконтроль</i>	<i>Повстяна Ю. С.</i>		
<i>Гарант ОП</i>	<i>Андрущак І. Є.</i>		
<i>Показник запозичень тексту</i>		___%	
<i>Академічна добросовісність</i>	<i>Яцук А. А.</i>		

7. Дата видачі завдання «02 квітня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи магістра	Строк виконання етапів роботи	Примітка
1	Провести огляд літературних джерел по темі кваліфікаційної роботи	02.05.2025	
2	Провести аналіз загальної проблеми і вибір напрямків дослідження	24.09.2025	
3	Розробити функціональну схему роботи програмного продукту	01.11.2025	
4	Описати засоби розробки об'єкта проектування	19.11.2025	
5	Практична реалізація об'єкта проектування	26.11.2025	
6	Розробити методика для проведення експерименту	05.11.2025	
7	Провести аналіз результатів експерименту	15.11.2025	
8	Здача чистового варіанту кваліфікаційної роботи на кафедрі	04.12.2025	

Здобувач вищої освіти

_____ (підпис)

Каложний Д. В.

(прізвище, ініціали)

Керівник кваліфікаційної роботи

_____ (підпис)

Яцук А. А.

(прізвище, ініціали)

АНОТАЦІЯ

Калюжний Д. В. Розробка та дослідження інтелектуальної системи аналізу відгуків на товари з використанням методів аналізу тональності тексту. Рукопис.

Кваліфікаційна робота магістра ОП «Інженерія програмного забезпечення» спеціальності 121 «Інженерія програмного забезпечення». Луцький національний технічний університет. Луцьк, 2025.

Кваліфікаційна робота магістра складається з вступу, 3 розділів, висновків, списку використаних джерел, додатків.

У першому розділі наведено огляд предметної області та існуючих підходів до визначення тональності тексту. У другому розділі описано процес розроблення інтелектуальної системи оцінки відгуків покупців на товари, обґрунтування вибору технологій і реалізацію основних функціональних модулів. У третьому розділі подано експериментальні результати, аналіз точності визначення тональності текстів і оцінку ефективності розробленої системи.

Ключові слова: аналіз тональності, відгуки користувачів, трансформери, BERT, scraping, NLP, e-commerce.

ABSTRACT

Kaluzhnyi D. V. Development and Research of an Intelligent System for Analyzing Product Reviews Using Text Tone Analysis Methods.

Qualification master's thesis EP «Software Engineering» specialty 121 «Software Engineering». Lutsk National Technical University. Lutsk, 2025.

Qualification master's thesis consists of an introduction, 3 chapters, conclusions, a list of sources used, and an appendix.

The first section provides an overview of the subject area and existing approaches to determining the tone of the text. The second section describes the process of developing an intelligent system for evaluating customer reviews of products, the justification of the choice of technologies and the implementation of the main functional modules. The third section presents experimental results, an analysis of the accuracy of determining the tone of texts and an assessment of the effectiveness of the developed system.

Key words: sentiment analysis, user reviews, transformers, BERT, scraping, NLP, e-commerce.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 АНАЛІЗ ПРОБЛЕМАТИКИ ЗА ТЕМОЮ РОБОТИ ТА ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ.....	9
1.1 Огляд і аналіз предметної області проблеми, результатів існуючих теоретичних та експериментальних досліджень	9
1.2 Огляд і аналіз методів та засобів розробки інтелектуальної системи аналізу відгуків на товари для вирішення проблеми дослідження	17
1.3 Постановка завдання на кваліфікаційну роботу магістра	21
РОЗДІЛ 2 ТЕОРЕТИЧНЕ ДОСЛІДЖЕННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ АНАЛІЗУ ВІДГУКІВ НА ТОВАРИ.....	23
2.1 Обґрунтування вибору шляхів, технологій, алгоритмів і засобів вирішення поставленого завдання.....	23
2.2 Практична реалізація об'єкта проектування.....	28
РОЗДІЛ 3 ЕКСПЕРЕМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ РЕЗУЛЬТАТИВНОСТІ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ АНАЛІЗУ ВІДГУКІВ НА ТОВАРИ.....	40
3.1 Методика проведення дослідження.....	40
3.2 Обробка та аналіз отриманих результатів.....	44
ВИСНОВКИ	50
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	53
ДОДАТКИ.....	57

ВСТУП

Актуальність теми. Розвиток електронної комерції супроводжується зростанням обсягів користувацьких відгуків, у яких міститься важлива інформація про задоволеність клієнтів і якість товарів. Ручний або поверхневий аналіз таких даних не дозволяє оперативно виявляти тенденції та повністю використовувати їх аналітичний потенціал. У світі активно застосовують методи аналізу тональності тексту на основі обробки природної мови й машинного навчання, однак більшість рішень орієнтовані на англійськомовні дані та погано працюють з українськомовними текстами. Відсутність комплексних систем для повного циклу обробки українськомовних відгуків і формування зрозумілих аналітичних показників для бізнесу зумовлює актуальність розроблення інтелектуальної системи аналізу відгуків на товари.

Мета кваліфікаційної роботи магістра полягає у створенні інтелектуальної системи автоматичного аналізу тональності українськомовних користувацьких відгуків на товари та оцінюванні ефективності її роботи на основі сучасних методів обробки природної мови і машинного навчання.

Для досягнення поставленої мети необхідно розв'язати такі завдання:

- проаналізувати сучасний стан проблеми аналізу тональності тексту та використання користувацьких відгуків у системах електронної комерції;
- сформулювати вимоги до інтелектуальної системи аналізу тональності;
- обґрунтувати вибір методів обробки природної мови та моделей машинного навчання, придатних для класифікації тональності українськомовних текстів;
- спроектувати архітектуру інтелектуальної системи аналізу відгуків на товари та структуру її основних програмних модулів;
- реалізувати програмний прототип системи, що забезпечує збір, попередню обробку, аналіз тональності та збереження результатів обробки відгуків;

- сформувати та розмітити корпус українськомовних відгуків, провести експериментальні дослідження роботи системи, порівнявши різні моделі аналізу тональності;

- оцінити отримані результати, визначити переваги, недоліки та перспективи подальшого розвитку розробленої системи.

Об'єктом дослідження є процес аналізу тональності текстових відгуків користувачів в інформаційних системах електронної комерції.

Предметом дослідження є методи, моделі та програмні засоби побудови інтелектуальної системи автоматичного аналізу тональності українськомовних відгуків на товари.

Наукова новизна одержаних результатів: запропоновано комплексний підхід до побудови інтелектуальної системи аналізу тональності українськомовних користувацьких відгуків, який поєднує спеціалізовані процедури попередньої обробки текстів із застосуванням сучасних моделей машинного навчання та глибокого навчання.

Практичне значення одержаних результатів полягає в розробленні програмного прототипу інтелектуальної системи аналізу тональності відгуків на товари, який може бути інтегрований у інформаційну інфраструктуру підприємств електронної комерції. Система забезпечує автоматизований збір та обробку великих обсягів відгуків, формування узагальнених показників тональності, виявлення проблемних товарів і тенденцій у настроях споживачів. Отримані результати можуть бути використані для побудови інформаційних панелей моніторингу задоволеності клієнтів, підтримки рішень у маркетинговій та сервісній політиці.

Апробація результатів дослідження. Основні положення та результати кваліфікаційної роботи магістра були висвітлені у статті [1] у науковому збірнику «Студентський науковий вісник» (додаток А).

РОЗДІЛ 1

АНАЛІЗ ПРОБЛЕМАТИКИ ЗА ТЕМОЮ РОБОТИ ТА ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ

1.1 Огляд і аналіз предметної області проблеми, результатів існуючих теоретичних та експериментальних досліджень

Сентимент-аналіз (аналіз тональності тексту) розглядається як один з ключових інструментів обробки природної мови для інтерпретації емоційного забарвлення повідомлень користувачів у соціальних мережах, відгуках та коментарях. Сучасні оглядові роботи показують перехід від простих лексичних правил і класичних моделей машинного навчання до глибоких нейронних мереж і трансформерних архітектур, які краще враховують контекст і структуру тексту [2]. Узагальнену схему основних рівнів (документний, реченнєвий, аспектний) подано на рисунку 1.1 [3].

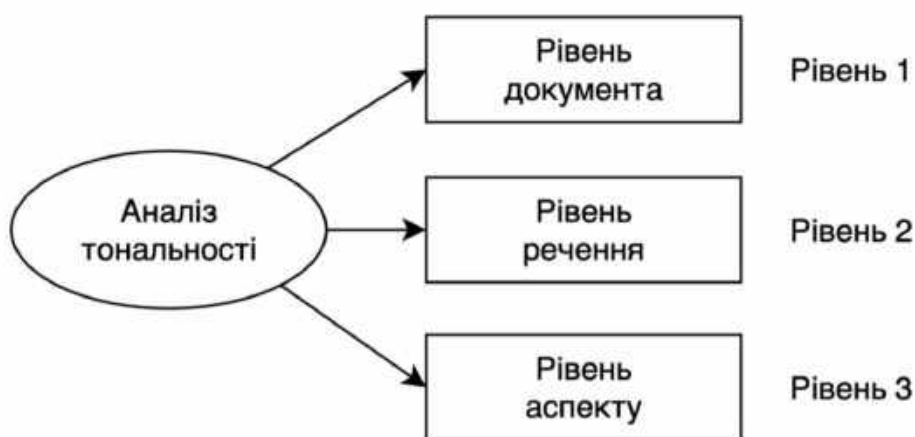


Рисунок 1.1 – Таксономія рівнів і методів аналізу тональності тексту

У систематичному огляді застосувань глибокого навчання для аналізу тональності коментарів запропоновано таксономію задач (класифікація відгуків, аналіз твітів, змішані сценарії «відгуки + твіти»), а також проаналізовано типові набори даних, архітектури (CNN, LSTM, BiLSTM, attention-моделі, трансформери) та відкриті дослідницькі проблеми: дисбаланс класів, доменна адаптація, пояснюваність моделей та масштабування до потоків «великих

даних» [2]. Більш детальну узагальнену структуру компонентів систем аналізу тональності (рівні аналізу, джерела даних, моделі, вихідні метрики) наведено на рисунку 1.2, де продемонстровано місце окремих класів методів у загальній таксономії систем sentiment-аналізу [4].

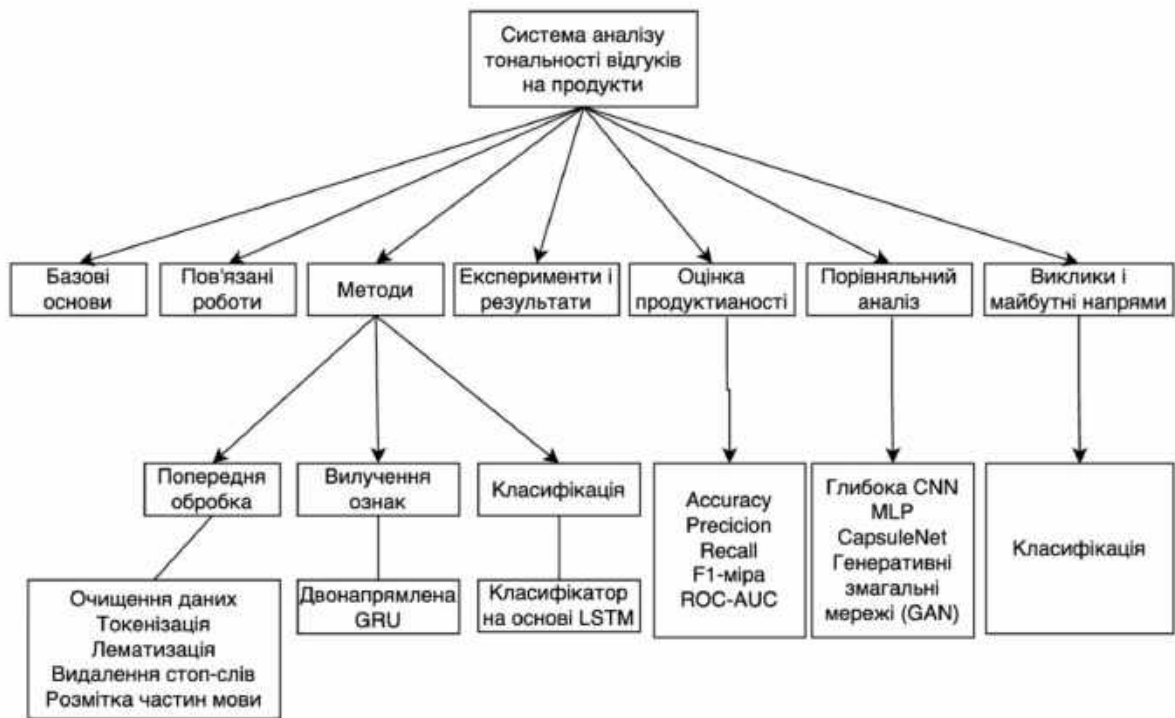


Рисунок 1.2 – Узагальнена таксономія систем аналізу тональності

Це формує методологічне підґрунтя для побудови інтелектуальних систем аналізу користувацьких відгуків. Подібні підходи успішно застосовуються й у суміжних доменах, наприклад для аналізу в реальному часі текстових відгуків студентів, що підтверджує універсальність методів аналізу тональності.

Окремий напрямок становлять роботи, орієнтовані безпосередньо на товарні відгуки в e-commerce. У праці P. Ghatoga та співавторів проведено комплексне порівняння традиційних алгоритмів машинного навчання (SVM, Random Forest тощо) з попередньо навченими великими мовними моделями (зокрема GPT-4) у задачі автоматичної класифікації тональності відгуків клієнтів. Показано, що LLM-моделі здатні забезпечувати високу точність у режимі zero-shot, однак потребують ретельного налаштування під конкретний

домен і мають значні обчислювальні витрати [5].

Типовий конвеєр такої системи – від збору відгуків, попередньої обробки, векторизації тексту до порівняння різних моделей та оцінювання якості – схематично представлений на рис. 1.3, що ілюструє послідовність етапів аналізу тональності товарних відгуків [5].

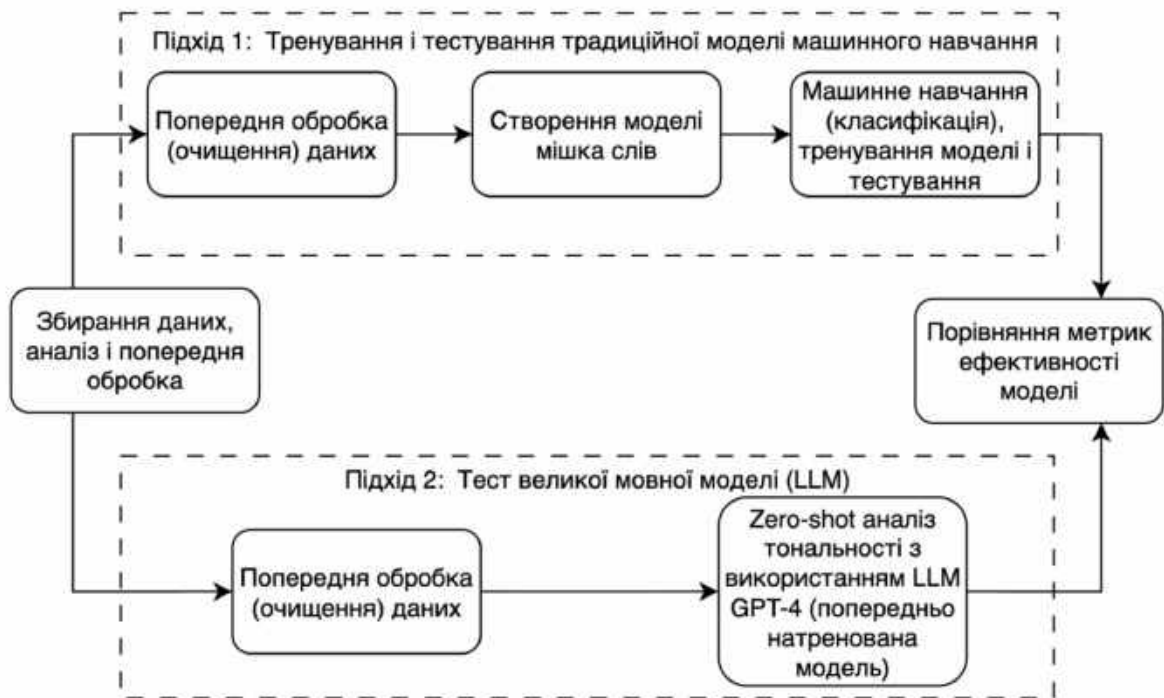


Рисунок 1.3 – Узагальнена методологія аналізу тональності відгуків на товари

У низці робіт запропоновано гібридні архітектури, що поєднують трансформерні векторні подання з одномірними згортковими мережами та двонапрямленими LSTM для більш глибокого врахування локального та довгострокового контексту у відгуках. Результати експериментів свідчать про перевагу таких моделей над класичними підходами як за точністю класифікації, так і за стабільністю на «шумних» користувацьких текстах [6]. Узагальнене порівняння продуктивності різних глибоких архітектур (DCNN, MLP, CapsuleNet, GAN, BiGRU+LSTM тощо) для аналізу тональності продуктів-відгуків показує, що сучасні гібридні моделі демонструють найвищі показники ассурасу та F1-міри, що відображено в таблиці 1.1 [4].

Таблиця 1.1 – Порівняння точності різних моделей глибокого навчання для аналізу тональності відгуків

№	Алгоритми	Метрики			
		Точність (Accuracy)	Влучність (Precision)	Відгук (Recall)	F1-оцінка (F1-score)
1.	DCNN	72.8	87.71	68.79	89.52
2.	MLP	94.71	87.06	90.73	95.15
3.	CN	87.77	88.28	90.79	83.1
4.	GAN	91.7	92.07	93.91	90.67
5.	Proposed model	98.79	96.64	98.70	97.43

Подібні висновки підтверджуються й іншими дослідженнями, де трансформерні моделі (BERT, RoBERTa та їх модифікації) демонструють кращі результати порівняно з «дотрансформерними» моделями при аналізі відгуків на платформах електронної торгівлі [7].

Важливою тенденцією останніх років є перехід від монолінгвальних систем до багатомовного та крос-лінгвального аналізу тональності. У сучасному огляді з багатомовного сентимент-аналізу показано, що для підтримки великої кількості мов використовуються багатомовні трансформерні моделі, методи перенесення знань між мовами, доменна адаптація та різні варіанти поповнення даних [8]. Автори наголошують, що навіть за наявності сильних багатомовних моделей якість аналізу для мов із обмеженими ресурсами сильно залежить від якості корпусів та стратегії адаптації. Окреме дослідження демонструє, що переклад-орієнтоване аугментування (використання систем машинного перекладу для розширення навчальних вибірок) може відчутно покращувати результати класифікації для низькоресурсних мов та полегшувати крос-лінгвальне перенесення моделей [9].

Додатково в роботах, присвячених ансамблевим багатомовним моделям, показано, що поєднання кількох попередньо навчених трансформерів у єдиний ансамбль дає змогу перевищувати базову продуктивність окремих моделей для кількох мов одночасно; типовий приклад такого порівняння продуктивності

мовних моделей та їх ансамблів наведено в таблиці 1.2 [10].

Таблиця 1.2 – Порівняння продуктивності багатомовних і мономовних трансформерних моделей та їх ансамблів для багатомовного аналізу тональності

Мова	Тренувальні дані	Модель	Точність	F1-macro
English	English	m-BERT (Baseline)	67.16	67.06
English	English	RoBERTa	70.69	70.84
Arabic	Arabic	m-BERT (Baseline)	54.21	53.08
Arabic	Arabic	AraBERTv02	69.79	69.78
English	English	Majority Voting Ensemble	70.95	71.03
Arabic	English and Arabic	Ensemble model with Feed Forward	67.61	67.12

Для англійської мови видно помітний приріст від baseline m-BERT до RoBERTa та ще невелике, але стабільне покращення від ансамблю (Majority Voting). Для арабської мови спеціалізована модель AraBERTv02 суттєво перевищує багатомовний m-BERT, а ансамблеві архітектури на змішаних даних (English+Arabic) демонструють додатковий приріст якості, що важливо для low-resource мов.

У контексті електронної комерції актуальними є роботи, де аналізується багатомовна тональність саме користувацьких відгуків. Зокрема, у дослідженні, присвяченому аналізу відгуків на книги українською мовою, розглянуто задачі попередньої обробки «шумних» текстів, нормалізації змішаних орфографічних стандартів, побудови багатомовних векторних подань та оцінено якість різних класифікаторів для e-commerce-сценаріїв [11]. Показано, що перехресне навчання на декількох споріднених мовах може покращувати точність класифікації в умовах обмежених даних для кожної з мов окремо.

Окремий пласт робіт присвячений саме українськомовному контенту й код-світчингу. В праці Y. Shynkarov, V. Solopova, V. Schmitt запропоновано корпус COSMUS для аналізу тональності українських соціальних медіа-текстів, що включає повідомлення з Telegram-каналів, сайтів з відгуками на товари та

відкритих датасетів, анонотовані за класами позитивної, негативної, нейтральної та змішаної тональності, а також за мовною розміткою [12]. Автори демонструють, що на даних із активним код-світчингом базові багатомовні моделі, доступні в публічних репозиторіях, суттєво поступаються спеціально адаптованим моделям, натренованим на спеціалізованому корпусі. Це особливо важливо для сценаріїв, коли відгуки на товари публікуються одночасно українською, та іншими мовами.

Таким чином, сучасний стан досліджень характеризується:

- зрілою теоретичною базою та широким спектром глибоких моделей для аналізу тональності в соціальних мережах та товарних відгуках [2, 5, 6, 7];
- наявністю багатьох робіт з багатомовного та крос-лінгвального аналізу тональності, орієнтованих на підтримку низькоресурсних мов за рахунок багатомовних трансформерів та аугментації даних [8, 9];
- появою перших спеціалізованих корпусів та моделей для українськомовних та змішаних текстів у соціальних медіа, зокрема для контенту, пов'язаного з товарами та сервісами [11,12].

Водночас залишається низка прогалин, важливих для даної кваліфікаційної роботи. По-перше, значна частина досліджень орієнтована або на «великі» глобальні мови, або на спеціальні домени (політичні дискусії, новини, загальні соціальні мережі), тоді як систематичні рішення для українськомовних товарних відгуків в екосистемі e-commerce представлені обмежено. По-друге, у більшості робіт результати аналізу тональності розглядаються переважно з точки зору показників якості моделей (асурасу, F1-міра тощо), тоді як питання інтеграції цих результатів у прикладні аналітичні інтерфейси для бізнес-користувачів (інструментальні панелі, системи підтримки прийняття рішень) висвітлені недостатньо. Нарешті, комплексні рішення, які б поєднували повний цикл обробки українськомовних відгуків (збір, спеціалізована попередня обробка, візуалізація та агрегація метрик), залишаються поодинокими. Саме на усунення цих прогалин спрямована розробка й дослідження інтелектуальної системи аналізу відгуків на товари, що

виконується в межах даної магістерської роботи.

Окрему групу досліджень становлять роботи, у яких пропонуються повноцінні інтелектуальні системи аналізу відгуків в e-commerce із реалізованими програмними прототипами. У роботі M. Alzahrani та співавторів розроблено інтелектуальну систему аналізу тональності відгуків на товари з використанням глибоких нейронних мереж; архітектура включає модулі збирання даних з платформи електронної комерції, попередньої обробки тексту, побудови векторних подань та навчання моделей LSTM/GRU для багатокласової класифікації відгуків [13]. Система орієнтована на інтеграцію в інфраструктуру інтернет-магазину й забезпечує автоматичне формування показників настроїв клієнтів для менеджерів та аналітиків.

Подібні підходи реалізовано й в інших роботах, де інтелектуальні системи аналізу відгуків доповнюються механізмами рекомендацій. Наприклад, у дослідженні A. Daza та співавторів запропоновано систему класифікації тональності відгуків на основі глибокого навчання, результати якої використовуються для покращення персоналізованих рекомендацій та прогнозування поведінки покупців [14]. Інша система поєднує аналіз тональності з оцінкою значущості характеристик товару й прогнозом ймовірності покупки, що дозволяє будувати більш інформативні інформаційні панелі для маркетологів [15].

Окремий клас рішень складають системи на основі аспектно-орієнтованого аналізу тональності (Aspect-Based Sentiment Analysis, ABSA). У низці робіт розроблено прототипи, що виділяють окремі аспекти товару (якість, ціна, доставка, сервіс) та визначають тональність щодо кожного аспекту, що дає змогу формувати більш детальні звіти для виробників і продавців [16, 17]. Такі системи часто базуються на моделях BiLSTM з механізмом уваги або удосконалених трансформерних архітектурах і демонструють високу точність на доменних наборах відгуків (електроніка, одяг, продукти харчування тощо).

Поряд із науковими прототипами активно розвиваються хмарні сервіси, які можна розглядати як готові компоненти інтелектуальних систем аналізу відгуків.

Сервіси AWS Comprehend, Azure Text Analytics та Google Cloud Natural Language API надають REST-інтерфейси для аналізу тональності, виявлення сутностей та ключових фраз, що дозволяє швидко інтегрувати функції аналізу настроїв у веб- та мобільні застосунки без розгортання власних моделей [18, 19]. Порівняльні дослідження показують, що ці сервіси забезпечують прийнятну якість класифікації для англomовного контенту, але можуть давати нестабільні результати для low-resource мов і специфічних доменів, що обмежує їх безпосереднє застосування для українськомовних товарних відгуків [20, 21].

Окрема група рішень фокусується на побудові інформаційних панелей і візуально-аналітичних інтерфейсів поверх модулів аналізу тональності. В окремих працях описано веб-системи, які автоматично збирають відгуки з e-commerce-платформ, виконують класифікацію тональності (часто з використанням класичних алгоритмів на зразок Naive Bayes чи SVM), а результати подають у вигляді інтерактивних панелей, що показують розподіл позитивних, негативних і нейтральних відгуків за товарами, категоріями та часовими інтервалами [22]. Такі системи вже ближчі до реальних бізнес-рішень, однак нерідко обмежуються базовими моделями та не використовують сучасні трансформерні підходи або спеціальні методи для обробки «шумних» текстів.

Нарешті, останні роботи пропонують ансамблеві та гібридні інтелектуальні системи, які поєднують кілька мовних моделей і класифікаторів для підвищення надійності результатів аналізу відгуків. Наприклад, запропоновано складені моделі на базі RoBERTa, XLM-RoBERTa, AraBERT та їх ансамблів для англійської й арабської мов, що демонструють кращу точність та F1-міру порівняно з окремими моделями [23]. Такі підходи є перспективними й для українськомовних систем, однак на даний момент у відкритих джерелах відсутні завершені рішення, які б комплексно охоплювали саме українськомовні товарні відгуки та надавали зручні інтерфейси для бізнес-аналітики.

1.2 Огляд і аналіз методів та засобів розробки інтелектуальної системи аналізу відгуків на товари для вирішення проблеми дослідження

Розробка інтелектуальної системи аналізу відгуків на товари передбачає поєднання методів обробки природної мови, алгоритмів машинного навчання та сучасних засобів програмної інженерії. З одного боку, така система повинна забезпечувати повний цикл обробки тексту: від отримання сирих відгуків до формування агрегованих показників тональності. З іншого боку, важливо гарантувати масштабованість, надійність і зручність інтеграції в існуючу інфраструктуру електронної комерції [24].

Загалом архітектуру системи аналізу відгуків доцільно розглядати як набір взаємодіючих модулів: модуль збору даних, модуль попередньої обробки тексту, модуль побудови ознак та аналізу тональності, модуль зберігання результатів і модуль візуалізації та інтеграції з іншими сервісами. Кожен із цих модулів може реалізовуватися з використанням різних технологій, що визначає вимоги до вибору мов програмування, бібліотек, фреймворків і платформ розгортання [24, 25].

Одним із ключових аспектів є вибір методів обробки природної мови для українськомовних текстів. На рівні попередньої обробки традиційно застосовуються операції токенизації, нормалізації, лематизації, видалення стоп-слів, обробки емодзі та повторюваних символів. Сучасні дослідження підкреслюють, що саме конфігурація й послідовність операцій попереднього очищення суттєво впливають на якість подальшого аналізу тональності, особливо для «шумних» текстів соціальних мереж та відгуків [24, 25, 26]. Для цього можуть використовуватися як класичні інструменти (бібліотеки для морфологічного аналізу та токенизації), так і сучасні пакети на основі нейронних мереж, що підтримують українську мову. Важливо, щоб обрані засоби коректно обробляли змішування мов, розмовні конструкції та нестандартну пунктуацію.

Узагальнене порівняння основних груп методів, що можуть бути використані на етапі попередньої обробки та аналізу тональності відгуків,

наведено в таблиці 1.3.

Таблиця 1.3 – Порівняння основних методів обробки тексту та аналізу тональності

Рівень / група методів	Приклади технік / моделей	Переваги	Обмеження / особливості	Типове використання в системі аналізу відгуків
Попередня обробка тексту	Токенізація, нормалізація, лематизація, видалення стоп-слів, обробка емодзі та повторів	Зменшення «шуму», покращення якості ознак, уніфікація форми слів	Потрібна адаптація під українську мову та змішані тексти; ризик втрати важливих нюансів	Підготовка даних перед будь-якими моделями класифікації
Класичні методи ML + векторизація	TF-IDF, n-грамні ознаки + Naive Bayes, SVM, Logistic Regression, Random Forest	Простота реалізації; прозорість інтерпретації; невеликі ресурсні вимоги	Обмежене врахування контексту; гірша якість на дуже «шумних» та неоднорідних даних	Базова лінія, швидкі прототипи, компоненти ансамблевих рішень
Глибокі нейромережі	CNN, LSTM, BiLSTM, GRU	Краще врахування послідовності та локального контексту	Потребують більше даних і обчислювальних ресурсів; складніші в налаштуванні	Доменно-орієнтовані моделі для аналізу тональності відгуків
Трансформери та PLM	BERT, mBERT, XLM-R, україномовні/багатомовні моделі + класифікаційна «голова»	Висока якість на «шумних» даних; контекстно-залежні подання; можливість донавчання	Високі вимоги до ресурсів; необхідність ретельного тюнінгу та підбору корпусу	Ядро інтелектуальної системи аналізу тональності для e-commerce

На етапі побудови ознак та аналізу тональності можливо застосувати кілька груп методів. Класичні підходи ґрунтуються на векторизації тексту (наприклад, TF-IDF, n-грамні моделі) та використанні алгоритмів машинного навчання, таких як Naive Bayes, SVM, Logistic Regression чи Random Forest. Перевагою цих методів є відносна простота реалізації, прозорість налаштування

та нижчі вимоги до обчислювальних ресурсів, що дозволяє використовувати їх як базові моделі або складові ансамблів [24].

Сучасні системи для аналізу тональності все частіше базуються на глибоких нейронних мережах і трансформерних архітектурах. Ключову роль тут відіграють попередньо навчені мовні моделі на кшталт BERT, які забезпечують контекстно-залежні векторні подання тексту та можуть донавчатися на цільовому корпусі з мінімальними змінами архітектури [27]. Подальший розвиток отримали бібліотеки, що спрощують використання таких моделей у прикладних системах (зокрема, екосистема Transformers від Hugging Face, яка надає готові реалізації багатомовних моделей, інструменти для донавчання та інференсу в продуктивному середовищі [28]). Такий підхід забезпечує високу якість класифікації навіть для «шумних» даних, проте вимагає ретельного налаштування гіперпараметрів і достатньої обчислювальної потужності.

З точки зору засобів реалізації інтелектуальної системи аналізу відгуків доцільним є використання високорівневих мов програмування, які мають розвинену екосистему бібліотек для машинного навчання та веб-розробки. На боці серверної логіки широко застосовується мова Python у поєднанні з фреймворками для побудови веб-API, такими як FastAPI або Django REST Framework. FastAPI позиціонується як сучасний високопродуктивний фреймворк для створення REST-сервісів з автоматичною генерацією документації та валідацією даних на основі типів [29], тоді як Django та надбудова Django REST Framework забезпечують повнофункціональне серверне середовище з ORM, адмін-інтерфейсом та гнучкими засобами створення API [30, 31]. Альтернативним підходом є використання JavaScript/TypeScript та середовища виконання Node.js, що дозволяє будувати однорідні за мовою клієнт-серверні застосунки [32].

Для зберігання відгуків і результатів їх аналізу можуть застосовуватися реляційні системи керування базами даних (PostgreSQL) або документо-орієнтовані бази даних (MongoDB). PostgreSQL є зрілою відкритою реляційною СКБД з потужною підтримкою транзакцій, розширень і повнотекстового

пошуку [33], тоді як MongoDB забезпечує гнучку схему збереження документів у форматі BSON і добре підходить для роботи з напівструктурованими даними та швидкої еволюції моделі даних [34]. У випадку великих обсягів текстової інформації доцільно також розглядати пошукові платформи, що забезпечують повнотекстовий пошук та аналітику (наприклад, стеки на основі Lucene/ElasticSearch), однак вибір конкретного рішення залежить від вимог до пошуку та агрегування.

Окремо слід розглянути питання розгортання та масштабування системи. Для забезпечення надійності та стійкості до навантажень широко застосовується контейнеризація (Docker) і системи оркестрації контейнерів (Kubernetes). Docker надає відкриту платформу для упакування застосунків разом із залежностями у відокремлені контейнери, що спрощує переносимість між середовищами розробки та промислової експлуатації [35]. Kubernetes, у свою чергу, пропонує відкриту платформу для керування контейнеризованими навантаженнями, забезпечуючи автоматичне масштабування, самовідновлення сервісів, балансування навантаження та декларативну конфігурацію кластерів [36]. Використання цих технологій дозволяє горизонтально масштабувати компоненти, які відповідають за обробку запитів до моделей, та гнучко виділяти ресурси залежно від обсягів вхідних даних.

З огляду на орієнтацію роботи на українськомовні й потенційно змішані відгуки, важливим є також питання адаптації та донавчання моделей на спеціалізованих корпусах. Це може включати попередню ініціалізацію багатомовними моделями та подальше донавчання на українських даних, використання технік аугментації тексту, доменної адаптації та регуляризації для запобігання перенавчанню [24, 27, 28].

Таким чином, для вирішення поставленої в роботі проблеми доцільно використати комплексний підхід, що поєднує:

- сучасні методи обробки природної мови та глибокого навчання (з акцентом на трансформерні моделі і попередньо навчені мовні моделі);
- модульну клієнт-серверну архітектуру з чітким розподілом функцій між

модулями збору, обробки, зберігання та візуалізації даних;

– перевірені засоби розробки, тестування та розгортання програмних систем, орієнтовані на масштабованість і можливість подальшого розширення функціональності.

1.3 Постановка завдання на кваліфікаційну роботу магістра

На основі аналізу сучасного стану проблеми (п. 1.1) та огляду методів і засобів розробки інтелектуальної системи аналізу відгуків на товари (п. 1.2), було визначено завдання кваліфікаційної роботи магістра у вигляді наступних пунктів:

– сформулювати вимоги до інтелектуальної системи аналізу відгуків на товари, включно з функціональними (збір відгуків, попередня обробка, аналіз тональності) та нефункціональними вимогами (масштабованість, продуктивність, можливість інтеграції з e-commerce-платформами);

– обґрунтувати вибір методів обробки природної мови та моделей машинного навчання, придатних для класифікації тональності українськомовних текстів;

– спроектувати архітектуру інтелектуальної системи аналізу відгуків, включно зі схемою основних модулів (збір даних, попередня обробка, аналіз тональності, збереження результатів, API) та структурою бази даних для зберігання відгуків і результатів їх обробки;

– реалізувати програмний прототип серверної частини системи у вигляді веб-сервісу (API) для приймання текстів відгуків, їх обробки та аналізу тональності, а також збереження результатів у вибраній СКБД, а також розробити базовий веб-інтерфейс для візуалізації результатів аналізу, який надає можливість перегляду відгуків, їх тональності, агрегованих показників (розподіл тональностей, динаміка в часі, статистика по товарах) та фільтрації за основними параметрами;

– сформулювати та розмітити корпус українськомовних відгуків,

провести експериментальні дослідження роботи системи, порівнявши якість різних моделей аналізу тональності за стандартними метриками;

– оцінити отримані результати, на основі отриманих результатів сформулювати висновки щодо переваг, обмежень і напрямів подальшого розвитку розробленої системи.

РОЗДІЛ 2

ТЕОРЕТИЧНЕ ДОСЛІДЖЕННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ АНАЛІЗУ ВІДГУКІВ НА ТОВАРИ

2.1 Обґрунтування вибору шляхів, технологій, алгоритмів і засобів вирішення поставленого завдання

На основі аналізу сучасного стану проблеми було сформовано ряд вимог до інтелектуальної системи аналізу відгуків на товари, які визначають як її функціональні можливості, так і архітектурні рішення.

По-перше, система має коректно працювати з реальними українськомовними відгуками, які часто є «шумними»: містять орфографічні помилки, розмовні конструкції, емодзі, елементи інших мов, а також змішані форми запису. Це означає, що моделі аналізу тональності не можуть розраховувати на «ідеальний» текст і повинні бути стійкими до таких спотворень.

По-друге, важливо забезпечити автоматизований збір відгуків без ручного копіювання даних. Система повинна вміти підключатися до реальних веб-ресурсів (маркетплейсів, сайтів з відгуками), знаходити на сторінці структуровані блоки даних, витягувати текст, рейтинг, дату, ідентифікатор товару та інші атрибути, зберігати їх у єдиному форматі й оновлювати корпус при зміні контенту. По-третє, серед вимог є можливість паралельного використання і порівняння кількох моделей аналізу тональності, оскільки дослідницький характер роботи передбачає не лише побудову однієї моделі, а й оцінювання різних підходів за єдиними критеріями.

Нарешті, система повинна бути зрозумілою та корисною для бізнес-користувачів: результати аналізу мають відображатися у вигляді інтерактивних таблиць, графіків та агрегованих показників, а інтерфейс – дозволяти переглядати, фільтрувати й перезапускати аналіз без необхідності взаємодіяти безпосередньо з моделями чи кодом.

Зазначені вимоги безпосередньо вплинули на вибір загального підходу до

побудови системи та структуру її програмних компонентів. Для вирішення задачі автоматичного збору даних виділяється спеціалізований модуль веб-скрапінгу, який відповідає за завантаження сторінок товарів, пошук потрібних елементів DOM, обробку структурованих форматів (наприклад, JSON-LD) та конвертацію одержаних полів у внутрішнє представлення.

Для роботи з «шумними» українськомовними текстами передбачено окремий модуль попередньої обробки, що виконує очищення, нормалізацію та підготовку тексту до подальшого аналізу, з урахуванням того, що надмірна фільтрація може призвести до втрати важливої емоційної інформації.

Щоб забезпечити порівняння різних підходів до аналізу тональності, у системі реалізуються два незалежні модулі аналізу: базова модель на основі мультимовного трансформера та більш потужна контекстна модель, які отримують однаково підготовлені дані та повертають результати у спільному форматі. Над цими компонентами розташовується веб-інтерфейс, який взаємодіє з серверною частиною через API, відображає результати аналізу, дає змогу запускати нові сесії обробки та порівнювати вихідні значення двох моделей. Така модульна структура дозволяє, з одного боку, задовольнити вимоги до дослідницького експерименту, а з іншого – створює основу для практичного використання системи у середовищі електронної комерції.

З точки зору архітектури було обрано варіант серверно-клієнтної системи з чітким розділенням відповідальностей (рис. 2.1).

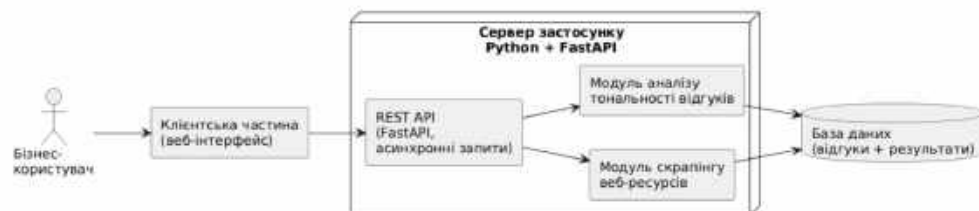


Рисунок 2.1 – Клієнт-серверна архітектура застосунку

З точки зору архітектури було обрано варіант серверно-клієнтної системи з чітким розділенням відповідальностей між компонентами. Користувач взаємодіє із системою через веб-інтерфейс, який працює в браузері та

звертається до бекенду лише через стандартизовані REST-запити. Клієнтська частина відповідає за відображення списку відгуків, запуск процедур скрапінгу, повторний аналіз, фільтрацію й візуалізацію результатів у вигляді таблиць та графіків, але не містить бізнес-логіки аналізу тональності – уся інтелектуальна обробка зосереджена на сервері.

На бекенді використовується мова Python, яка надає зріле екосистемне оточення для обробки природної мови, роботи з веб-ресурсами та реалізації веб-API. В межах серверного вузла виділяється декілька компонентів. Центральним є модуль REST-API, реалізований на базі фреймворку FastAPI. Він приймає HTTP-запити від клієнта, виконує валідацію вхідних даних на основі типізованих моделей, маршрутизує запити до спеціалізованих модулів і формує уніфіковані відповіді. FastAPI підтримує асинхронну обробку, тому операції, пов'язані з мережею (отримання сторінок з маркетплейсу, звернення до моделей аналізу тональності, запис у базу даних), можуть виконуватися у фоновому режимі, не блокуючи приймання інших запитів.

Окремо виділяється модуль скрапінгу, який через REST-інтерфейс запускається з клієнтської частини. Він відповідає за завантаження сторінок товарів, пошук у DOM-структурі блоків з відгуками, витягування тексту, рейтингу та метаданих і передачу цих даних до модуля зберігання. Модуль аналізу тональності отримує підготовлені тексти, викликається через API та використовує одну або декілька моделей для обчислення тональності й пов'язаних показників упевненості. Результати роботи скрапінгу та аналізу зберігаються у базі даних, до якої бекенд звертається через ORM-рівень. Така структура дозволяє чітко розвести обов'язки між компонентами, спростити масштабування окремих модулів (наприклад, виділити окремі екземпляри для аналізу тональності) та забезпечити стабільну роботу системи навіть за збільшення кількості користувачів і обсягу відгуків.

Для зберігання результатів аналізу обрано реляційну базу даних SQLite з використанням ORM-підходу. Застосування бібліотеки SQLAlchemy дає можливість описувати структуру таблиць єдиними типізованими моделями, які

одночасно використовуються для роботи з базою даних і для формування схем обміну даними у веб-API. Це спрощує підтримку й розширення системи, а також зменшує кількість дубльованого коду.

Клієнтська частина реалізується у вигляді веб-інтерфейсу на основі фреймворка Next.js (React). Такий вибір зумовлений необхідністю створити інтерактивну панель, де користувач може переглядати зібрані відгуки, порівнювати результати роботи моделей, виконувати повторний аналіз, видаляти окремі записи, змінювати параметри товарів і запускати повторний скрапінг. Підтримка серверного рендерингу і маршрутизації в Next.js дозволяє забезпечити швидке завантаження сторінок і зручну побудову адміністративного інтерфейсу.

Окремим підзавданням є автоматичний збір відгуків. Для цього розробляється `scrapet`-модуль на Python із використанням бібліотек для роботи з HTTP-запитами та HTML-структурами. Модуль отримує сторінки товарів з маркетплейсу, знаходить структуровані блоки даних (зокрема JSON-LD), витягує текст відгуку, користувацький рейтинг у вигляді «зірок», дату, ідентифікатор товару та інші метадані. У разі відсутності явних структурованих даних передбачаються `fallback`-стратегії, наприклад аналіз SVG-іконок із відображенням кількості зірок або парсинг HTML-блоків з оцінками. Такий підхід робить систему більш стійкою до змін верстки на стороні джерела.

Наступним логічним компонентом є модуль попередньої обробки тексту (рис. 2.2). Оскільки об'єктом аналізу є реальні україномовні відгуки з орфографічними помилками, емодзі, змішаними мовами та розмовними конструкціями, класичні «жорсткі» правила очищення тут неприйнятні. У системі передбачається поетапна обробка: видалення HTML-тегів, нормалізація пробілів, фільтрація технічних символів, але при цьому збереження емодзі та ключових пунктуаційних маркерів, що несуть емоційне навантаження. На рівні токенизації й нормалізації текст має оброблятися таким чином, щоб не руйнувати специфічні для української мови форми та не «ламати» змішані фрагменти, які все одно можуть бути опрацьовані трансформерними моделями.

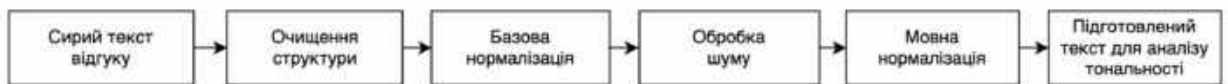


Рисунок 2.2 – Попередня обробка тексту

Ключовим елементом теоретичного й практичного дослідження є вибір та порівняння моделей аналізу тональності. З метою відтворення «реалістичного» сценарію електронної комерції обрано дві моделі різного класу.

Перша – базова мультимовна модель на основі Multilingual BERT, яка у роботі розглядається як Simple-модель. Вона повертає оцінку у форматі від 1 до 5 зірок, добре працює з короткими відгуками та має відносно низькі вимоги до ресурсів.

Друга – контекстна модель XLM-RoBERTa Base, що вбудовано класифікує тональність за класами «позитивний», «нейтральний» і «негативний» та краще враховує контекстні залежності у довших і складніших текстах.

Для узгодженого порівняння обидві моделі інтегруються в єдиний Analyzer-модуль. Їхні виходи зводяться до спільного представлення: тональність (три класи), числова оцінка (1-5 зірок) та показник впевненості (від 0 до 100 %). Для Simple-моделі тональність може визначатися шляхом порогового розбиття за числовим рейтингом, тоді як для XLM-RoBERTa первинною є саме класифікація тональності, а оцінка зірок може бути похідною. Такий підхід дозволяє не лише порівнювати абсолютні значення точності, але й аналізувати узгодженість прогнозів із користувацькими оцінками та між собою.

З методичної точки зору в роботі приймається гіпотеза, що базова мультимовна модель забезпечить вищу швидкодію та стабільні результати на коротких відгуках із явно вираженою полярністю, тоді як контекстна XLM-RoBERTa краще відпрацьовуватиме середні та довгі тексти з прихованими оцінками, змішаними емоціями й сарказмом. Відповідно, у плані експериментального дослідження передбачено окремий аналіз якості класифікації для різних типів відгуків (короткі, середні, довгі), порівняння рівня

впевненості моделей, кореляцію з реальним рейтингом користувача та оцінку швидкодії.

Ще одним важливим рішенням є спосіб інтеграції всіх компонентів. Як механізм взаємодії між frontend-панеллю, бекенд-модулем аналізу та базою даних використовується REST-API. Це спрощує масштабування системи, дозволяє за потреби виносити окремі компоненти (наприклад, модуль аналізу тональності) на інші вузли або сервери, а також відкриває можливість інтеграції з зовнішніми сервісами й корпоративними системами електронної комерції.

Таким чином, вибір шляхів, технологій та алгоритмів у цій роботі обумовлений поєднанням двох груп вимог: з одного боку – надійна підтримка повного циклу обробки українськомовних відгуків у реальних умовах e-commerce, з іншого – можливість коректного експериментального порівняння двох моделей аналізу тональності в єдиній інтегрованій системі.

2.2 Практична реалізація об'єкта проєктування

Серверна частина інтелектуальної системи реалізована як веб-застосунок на базі мови програмування Python та фреймворку FastAPI. Такий вибір зумовлено необхідністю поєднати високорівневі засоби обробки природної мови, підтримку асинхронних HTTP-запитів та зручну реалізацію REST-інтерфейсів. Для роботи з базою даних використовується бібліотека SQLAlchemy, яка поєднує ідеї Pydantic (типізовані моделі даних) та SQLAlchemy (ORM-шар) і дає змогу описувати структуру таблиць і схем обміну даними в єдиному місці.

Основні залежності серверної частини: FastAPI, Uvicorn (як ASGI-сервера), SQLAlchemy, бібліотеки requests та BeautifulSoup4 для веб-скрапінгу. Це дозволяє розгорнути сервер як окремий сервіс, який взаємодіє з клієнтським інтерфейсом через HTTP.

Підключення до бази даних та ініціалізація FastAPI-застосунку виконуються у модулі main.py. У лістингу 2.1 наведено фрагмент коду, який створює SQLite-базу (reviews.db), описує фабрику сесій та реєструє подію

ініціалізації БД при старті застосунку.

Лістинг 2.1 – Ініціалізація БД та FastAPI-застосунку (main.py)

```

from fastapi import FastAPI, Depends
from fastapi.middleware.cors import CORSMiddleware
from sqlmodel import SQLModel, create_engine, Session
import os

BASE_DIR = os.path.dirname(__file__)
DATABASE_URL = f"sqlite:///{os.path.join(BASE_DIR, 'reviews.db')}}"
engine = create_engine(DATABASE_URL, echo=True)

def init_db() -> None:
    SQLModel.metadata.create_all(engine)

def get_session() -> Session:
    with Session(engine) as session:
        yield session

app = FastAPI(title="Review Platform API", version="0.3")

# Дозвіл на запити з фронтенда (Next.js на localhost:3000)
app.add_middleware(
    CORSMiddleware,
    allow_origins=["http://localhost:3000"],
    allow_credentials=True,
    allow_methods=["*"],
    allow_headers=["*"],
)

app.add_event_handler("startup", init_db)

```

Кінець лістингу 2.1

Таким чином, серверна частина працює як окремий процес, що приймає HTTP-запити, забезпечує доступ до БД через ORM-шар та надає стандартизований REST-інтерфейс для фронтенда.

Завдяки використанню SQLModel одночасно описуються і таблиці в БД, і Python-моделі, з якими працюють ендпоїнти FastAPI, що спрощує підтримку цілісності та еволюцію схеми даних.

Структура основних сутностей системи задається в модулі models.py на основі типів SQLModel і узагальнено відображена на рисунку 2.3.

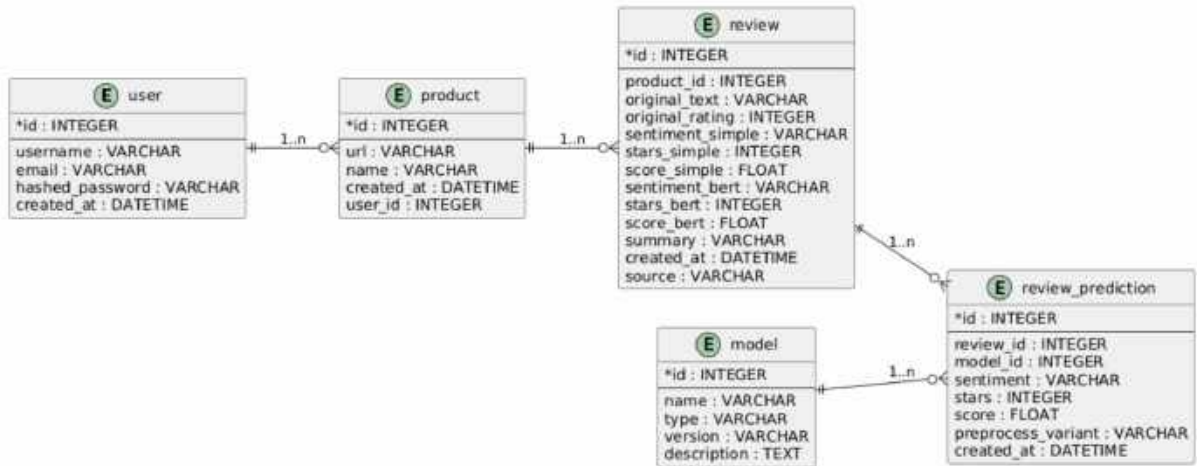


Рисунок 2.3 – Діаграма структури даних розробленої системи

У моделі виділено сутності користувача, товару та відгуку, а також окремі сутності для опису моделей аналізу тональності й збереження їхніх прогнозів. Така організація дозволяє пов'язати товари з конкретними користувачами, відгуки – з товарами, а результати експериментів із різними моделями – з кожним окремим відгуком. Завдяки цьому база даних одночасно підтримує роботу веб-застосунку (добування та відображення відгуків) і проведення експериментальних досліджень якості моделей аналізу тональності.

Оскільки система передбачає роботу декількох користувачів, доступ до операцій керування товарами та відгуками обмежено за допомогою механізму токен-базованої автентифікації. FastAPI інтегрується з бібліотеками passlib (хешування паролів) та python-jose (підпис і валідація JWT-токенів).

На стороні сервера реалізовано такі основні компоненти безпеки:

- функції хешування паролів та перевірки введеного пароля відносно збереженого хешу;
- генерація access-token із вбудованим часом життя (наприклад, 30 хвилин);
- залежність OAuth2PasswordBearer, яка автоматично читає токен з заголовка Authorization: Bearer ... та передає його в логіку перевірки користувача;
- залежність get_current_user, яка на основі токена витягує користувача з БД та передає його в обробники ендпоінтів.

Для роботи з користувачами визначено два ключові ендпоїнти:

- POST /register – створення нового користувача, хешування пароля, збереження у таблиці user;
- POST /token – отримання JWT-токена за логіном і паролем (сумісно з OAuth2-схемою password).

Усі «бізнесові» ендпоїнти (робота з товарами, відгуками, запуск скрапінгу) отримують поточного користувача через Depends(get_current_user) і тим самим обмежують доступ лише до даних конкретного користувача.

Група ендпоїнтів /products реалізує CRUD-функціональність для керування списком товарів, відгуки до яких аналізуються системою. Основні операції:

- POST /products – створення нового товару на основі URL сторінки. Спочатку викликається допоміжна функція get_or_create_product, яка або знаходить існуючий запис за URL і користувачем, або створює новий. Далі автоматично запускається процедура скрапінгу та аналізу відгуків для цього товару;
- GET /products – повертає список товарів, пов'язаних із поточним користувачем;
- GET /products/{product_id} – повертає детальну інформацію про конкретний товар, якщо він належить поточному користувачу;
- DELETE /products/{product_id} – видаляє товар і всі пов'язані з ним відгуки.

Усі ці ендпоїнти використовують залежність Session = Depends(get_session) для роботи з БД і current_user = Depends(get_current_user) для контролю доступу.

Для відгуків реалізовано окремий набір ендпоїнтів:

- GET /products/{product_id}/reviews – повертає список відгуків для вказаного товару у вигляді Pydantic-моделей ReviewOut. У цих моделях об'єднано вихідний текст, початковий рейтинг користувача (якщо він є) та результати двох моделей аналізу тональності: прогнозовану тональність,

кількість зірок і значення впевненості (за необхідності нормалізоване до діапазону [0; 1] або [0; 100 %]);

– DELETE /reviews/{review_id} – дозволяє видаляти окремі відгуки, наприклад, у разі помилок скрапінгу або за бажанням користувача.

Логіка доступу гарантує, що користувач бачить і може змінювати лише відгуки, пов'язані з товарами, які належать саме йому.

Однією з ключових функцій серверної частини є автоматичний збір відгуків з реального веб-ресурсу та передавання їх на аналіз. Для цього реалізовано ендпоїнт POST /scrape, який приймає JSON-запит з полем url, яке вказує на сторінку товару на маркетплейсі. На основі цієї URL виконується декілька кроків, описаних далі (рис. 2.4).



Рисунок 2.4 – Етапи обробки відгуків на сторінці товару

Процес обробки відгуків починається із завантаження HTML-сторінки за допомогою бібліотеки requests та подальшого розбору DOM-структури засобами BeautifulSoup. На основі отриманого дерева виконується пошук структурованих

блоків JSON-LD, у яких міститься опис товару та вкладений масив відгуків у полі review; із цих блоків витягуються тексти відгуків, назва товару та користувачський рейтинг, якщо він на сайті присутній. Якщо ж частина відгуків не потрапляє до структурованих даних, додатково аналізується «звичайна» HTML-розмітка: за допомогою CSS-селекторів знаходяться блоки з текстом відгуку та елементи, що кодують кількість зірок. Рейтинги, подані у вигляді відсотків, нормалізуються, наприклад шляхом інтерпретації атрибута data-qaaid-rating як шкали 0–100 % з подальшим приведенням значення до діапазону від 1 до 5 зірок. Після цього кожен текст відгуку передається до окремого сервісу аналізу тональності через HTTP-запит на адресу, задану в параметрі ANALYZER_URL, із режимом mode="both", що ініціює одночасний запуск двох моделей (simple-моделі та BERT-моделі). Отримана у відповідь JSON-структура аналізується і зберігається в таблиці review за допомогою допоміжної функції save_and_record, яка створює об'єкти Review, заповнює поля sentiment_simple, stars_simple, score_simple, sentiment_bert, stars_bert, score_bert, summary, source і додає їх до поточної сесії бази даних.

Після завершення скрапінгу для вказаного товару попередні відгуки видаляються, щоб уникнути дублювання, і зберігається актуальний список. Ендпоінт повертає клієнту серіалізований список нових відгуків у форматі ReviewOut, що дає змогу відразу відобразити результати аналізу тональності у веб-інтерфейсі.

Фрагмент програмної реалізація скрапінгу коментарів на прикладі платформи Prom.ua показана у лістингу 2.2.

Лістинг 2.2 – Функція collect_reviews_from_soup для витягування відгуків із JSON-LD та HTML-розмітки сторінки товару

```
def collect_reviews_from_soup(soup):
    """
    Витягує відгуки з JSON-LD та звичайної HTML-розмітки.
    Повертає список словників: {"text", "rating", "source"}.
    """
    reviews = []
    # Відгуки з JSON-LD (Product.review)
```

```

    for script in soup.find_all("script",
type="application/ld+json"):
        # ... розбір JSON, пошук блоків @type == "Product" ...
        # ... витягування reviewBody та reviewRating.ratingValue
        ...
        # reviews.append({"text": text, "rating": rating_val,
"source": "json_ld"})
        ...
        # Відгуки з «звичайної» HTML-розмітки
        for item in soup.select('[data-qaId="opinion_item"]'):
            text_el = item.select_one('[data-qaId="opinion_text"]')
            rating_el = item.select_one('[data-
qaId="opinion_rating"]')

            text = (text_el.get_text(strip=True) if text_el else "")
or ""
            rating_val = None

            # Наприклад, data-qaId-raiting="80" → 4 зірки
            if rating_el and rating_el.has_attr("data-qaId-raiting"):
                raw = rating_el["data-qaId-raiting"]
                # ... перетворення % у шкалу 1-5 зірок ...
                # rating_val = normalize_percent_to_stars(raw)
                ...
            if text:
                reviews.append({"text": text, "rating": rating_val,
"source": "html"})

return reviews

```

Кінець лістингу 2.2

Перевірка роботи серверної частини проводиться на двох рівнях.

Перший рівень передбачав функціональне тестування API – послідовні виклики ендпоінтів `/register` та `/token` для створення користувача і отримання токена, перевірка створення товару (POST `/products`), запуску скрапінгу (POST `/scrape`), читання відгуків (GET `/products/{id}/reviews`) та операцій видалення. Для цього використовувався Postman.

Другий рівень передбачав тестування обробки помилок – перевірка поведінки системи у випадку недоступності сторінки товару, некоректної структури JSON-LD, відмови сервісу аналізу тональності (очікуване повернення помилки з кодом 5xx та відповідним повідомленням), спроб доступу до чужих ресурсів без прав (очікувані відповіді з кодами 401/403/404).

Завдяки модульному поділу на скрапінг, аналіз і доступ до БД помилки

локалізуються в межах окремих компонентів, що полегшує їх діагностику й виправлення. У сукупності реалізований API забезпечує повний цикл роботи із серверною частиною інтелектуальної системи: від керування користувачами та товарами до автоматичного збору й аналізу відгуків та надання даних для клієнтського інтерфейсу.

Аналізатор тональності реалізовано як окремий веб-сервіс на Python, який працює за протоколом HTTP і взаємодіє з основним сервером через REST-інтерфейс. Такий поділ дозволяє ізолювати обчислювально складні операції (завантаження мовних моделей, проведення інференсу, попередню обробку тексту) від логіки керування користувачами, товарами та базою даних. Основний сервер звертається до аналізатора через єдину кінцеву точку `ANALYZER_URL`, передаючи текст відгуку та режим роботи, а у відповідь отримує структурований результат аналізу, який потім зберігається в таблиці `review`.

Сервіс аналізатора запускається як окремий FastAPI-застосунок. При старті процесу відбувається одноразове завантаження двох попередньо навчених моделей: простої мультимовної моделі, орієнтованої на прогнозування кількості «зірок» за текстом відгуку, та контекстної трансформерної моделі XLM-RoBERTa Base, налаштованої на класифікацію тональності в три класи («негативний», «нейтральний», «позитивний») для україномовних і змішаних текстів. Моделі, токенизатори та необхідні допоміжні об'єкти (наприклад, словники для перетворення логітів у класи) зберігаються в оперативній пам'яті протягом усього часу роботи сервісу, що значно зменшує затримки при обробці окремих запитів.

Основною точкою входу аналізатора є HTTP-ендпоїнт, який приймає JSON-об'єкт із полями `text` (текст відгуку) та `mode` (режим роботи: «simple», «bert» або «both»). Після отримання запиту сервіс застосовує до тексту узгоджений конвеєр попередньої обробки: усуває явно технічні артефакти (залишки HTML, службові маркери), виконує нормалізацію пробілів і регістру, але при цьому не видаляє емодзі та знаки пунктуації, які можуть нести емоційне навантаження. Цей конвеєр максимально наближений до того, що

використовується на основному сервері перед передаванням тексту в аналізатор, що забезпечує повторюваність результатів.

Далі, залежно від режиму, текст послідовно або паралельно подається на вхід відповідних моделей. Проста модель повертає числовий прогноз рейтингу в діапазоні 1-5 та вектор імовірностей по класах, з якого обчислюється показник упевненості. На основі цього рейтингу додатково виводиться узагальнена полярність (наприклад, 1-2 зірки – негативна тональність, 3 – нейтральна, 4-5 – позитивна). Контекстна модель XLM-RoBERTa Base працює в режимі багатокласової класифікації тональності: на виході отримується розподіл імовірностей між трьома класами, з якого вибирається клас із максимальною ймовірністю та відповідна йому впевненість. За потреби для неї також може обчислюватися наближений рейтинг у зірках, що дозволяє порівнювати обидві моделі в єдиній метриці.

Усі результати агрегуються в єдину JSON-відповідь стандартизованого формату (лістинг 2.1).

Лістинг 2.3 – Приклад JSON-відповіді сервісу аналізу тональності для одного відгуку

```
{
  "input_text": "Дуже сподобався товар, якість чудова.",
  "language": "uk",
  "simple": {
    "sentiment": "positive",
    "stars": 5,
    "score": 0.94
  },
  "bert": {
    "sentiment": "positive",
    "stars": 5,
    "score": 0.97
  },
  "meta": {
    "preprocess_variant": "minimal",
    "model_version_simple": "mBERT-stars-uk-1.0",
    "model_version_bert": "XLM-R-base-uk-1.0",
    "analyzed_at": "2025-11-07T14:32:10Z"
  }
}
```

Кінець лістингу 2.3

Аналізатор повертає об'єкт, який містить вкладені структури `simple` та `bert` з полями `sentiment`, `stars`, `score` (тональність, число зірок, показник упевненості). Саме на такий формат орієнтована серверна функція `analyze_review` у основному API, яка викликає аналізатор, а потім передає отримані значення до функції збереження в БД.

З технічної точки зору робота аналізатора оптимізована для пакетної обробки: за рахунок асинхронної реалізації ендпоінтів у FastAPI та можливості групувати кілька текстів в один запит можна зменшити кількість звернень до моделей і тим самим скоротити час обробки великої кількості відгуків. У випадку помилок (недоступність моделей, помилки пам'яті, некоректний формат вхідних даних) аналізатор повертає інформативні коди HTTP-помилки і текстові повідомлення, які логуються і дозволяють проводити діагностику безпосередньо на рівні цього сервісу без впливу на логіку основного API.

У такому вигляді аналізатор виступає незалежним, але тісно інтегрованим компонентом системи: основний сервер відповідає за збір і зберігання даних, автентифікацію, роботу з товарами та відгуками, тоді як сервіс аналізу тональності концентрується на інференсі мовних моделей. Це не лише спрощує масштабування (можна окремо збільшувати ресурси для сервісу аналізу), а й забезпечує можливість у майбутньому додавати нові моделі або змінювати їх конфігурації без модифікації бізнес-логіки та структури основного API.

Фронтальна частина системи реалізована як односторінковий веб-застосунок на основі фреймворку Next.js із використанням React і TypeScript. Такий підхід дозволяє поєднати сучасну компонентну модель інтерфейсу з можливостями серверного рендерингу та зручною організацією маршрутизації. Структура проєкту побудована навколо директорії `app`, у якій описано базовий макет сторінки, спільні елементи (шапка, навігаційна панель, повідомлення про стан системи) та окремі екрани для роботи з товарами і відгуками. Глобальне компонування визначається у файлі `layout.tsx`, який задає єдиний стиль для всієї панелі, підключає системні стилі Tailwind CSS та забезпечує послідовний користувацький досвід на всіх сторінках.

Основна роль фронтенду полягає в тому, щоб приховати складність бекенд-логіки й надати бізнес-користувачеві простий інструмент для роботи з відгуками.

Головний екран відображає перелік товарів, для кожного з яких показуються назва, URL джерела та час останнього оновлення. З цього екрану користувач може додати новий товар, ініціювати процес скрапінгу та аналізу, а також перейти до детального перегляду відгуків. Форми додавання товару реалізовано у вигляді контрольованих React-компонентів, які проводять базову валідацію введених даних (перевірка заповнення, коректність URL) до відправлення запиту на бекенд (рис. 2.5).

ID	Назва	Опис	URL	Дія
#1	2шт Сферичне дзеркало на авто опуклі дзеркала сліпої зони заднього огляду для автомобіля бокове дзеркало сліпих зон		https://prom.ua/ua/p/687901703-2sh-sfericheskoe-zerkalo.html	Видалити
#2	Розумна розетка з таймером		https://prom.ua/ua/p/2257430887-umnaya-rozetka-tajmerom.html	Видалити
#3	Рюкзак чоловічий міський міцний чорний для ноутбука водонепроникний спортивний молодіжний		https://prom.ua/ua/p/2784878713-ryukzak-muzhskoj-gorodskoj.html	Видалити
#5	Перехідник із американської плоскої вилки на європейську білий. Найвища якість! Преміумфіксація		https://prom.ua/ua/p/2338090559-perexodnik-amerikanskoj-ploskoj.html	Видалити
#6	Футболка Disarrear, стильна футболка Дісаррар, унісекс футболка на літо		https://prom.ua/ua/p/2555748066-futbolka-disarrear-silnyaya.html	Видалити
#7	Базова чоловіча чорна футболка без принтів однотонна бавовна			Видалити

Рисунок 2.5 – Список товарів, збережених в базі даних

Сторінка детального перегляду відгуків будується на основі даних, отриманих через REST-API, і відображає табличний список записів із БД. Для кожного відгуку показуються початковий рейтинг користувача, текст, а також

результати двох моделей аналізу тональності (simple-моделі та BERT-моделі) з відповідними прогнозованими зірками й коефіцієнтом впевненості (рис. 2.6).

[← Назад до товарів](#)

Базова чоловіча чорна футболка без принтів однотонна бавовна Змінити назву Скопіювати відгук з сайту

URL: <https://rom.ua/ua/c2342183113-bazovaya-muzhskaya-chernaya.html>

Гарна якість, сіла добре, претензій немає

Базовий sentiment-аналізатор	BERT-аналізатор
Тон: позитивний	Тон: позитивний
Оцінка: 4	Зірки: 5
Впевненість: 47%	Впевненість: 73%

[Видалити відгук](#)

Гарна якість за таку ціну

Базовий sentiment-аналізатор	BERT-аналізатор
Тон: позитивний	Тон: позитивний
Оцінка: 5	Зірки: 5
Впевненість: 56%	Впевненість: 82%

[Видалити відгук](#)

Гарний товар за таку ціну

Базовий sentiment-аналізатор	BERT-аналізатор
Тон: позитивний	Тон: позитивний
Оцінка: 5	Зірки: 5
Впевненість: 64%	Впевненість: 87%

[Видалити відгук](#)

Рисунок 2.6 – Інформація з відгуками про товар

Обмін даними з бекендом організовано через асинхронні запити до API, причому всі операції з мережевими викликами інкапсульовано в окремих сервісних функціях. Це спрощує обробку помилок і повторне використання логіки виклику в різних компонентах. Для відображення поточного стану системи (запит виконується, дані успішно отримані, сталася помилка) у фронтенді використовуються відповідні індикатори завантаження та повідомлення, що формують прозорий зворотний зв'язок для користувача. Така організація дозволяє сприймати складний багатокомпонентний комплекс (скрапер, аналізатор, БД) як єдиний зручний інструмент для аналізу відгуків, у якому всі технічні деталі приховані за інтуїтивним веб-інтерфейсом.

РОЗДІЛ 3

ЕКСПЕРЕМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ РЕЗУЛЬТАТИВНОСТІ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ АНАЛІЗУ ВІДГУКІВ НА ТОВАРИ

3.1 Методика проведення дослідження

Методика експериментального дослідження побудована таким чином, щоб оцінити роботу інтелектуальної системи в умовах, максимально наближених до реального сценарію електронної комерції, і водночас забезпечити відтворюваність отриманих результатів. Узагальнена послідовність етапів дослідження відображена на рисунку 3.1, де послідовно показано формування корпусу відгуків, обробку текстів, роботу моделей аналізу тональності та етапи оцінювання результатів.



Рисунок 3.1 – Схема експериментального дослідження інтелектуальної системи аналізу відгуків на товари

На першому етапі формується корпус відгуків. Для цього система здійснює автоматизований збір текстів з реальних веб-ресурсів, що містять відгуки покупців про товари, разом із вихідними рейтинговими оцінками за п'ятибальною шкалою (за наявності). Зібрані відгуки зберігаються в єдиній базі даних із прив'язкою до конкретного товару, джерела та часу додавання. На цьому ж етапі за допомогою інтерфейсу глобального експерименту користувач може переглядати всі відгуки в одному списку, фільтрувати їх за товаром, джерелом чи фрагментами тексту, а також за необхідності вручну коригувати рейтинг покупця. Така можливість дозволяє сформувати «еталонну» розмітку, коли вихідні оцінки на сайті не повністю відповідають змісту тексту, або взагалі відсутні.

Далі здійснюється попередня обробка текстів за обраним варіантом препроцесингу. У налаштуваннях експерименту передбачено параметр варіанта попередньої обробки, який керує послідовністю операцій очищення: нормалізацією реєстру, видаленням або збереженням емодзі, обробкою повторюваних символів, лематизацією, видаленням стоп-слів тощо. Це дозволяє дослідити вплив різних конфігурацій попередньої обробки на якість подальшої класифікації тональності, зокрема для «шумних» українськомовних текстів.

Після підготовки корпусу система аналізує сукупність усіх зібраних коментарів незалежно від конкретного товару, що дає змогу оцінити узагальнену якість моделей у типових умовах роботи системи підтримки прийняття рішень в e-commerce. Інструменти фільтрації та ручного включення/виключення відгуків з експерименту дозволяють формувати цільові підвибірки для різних сценаріїв аналізу: наприклад, обмежити вибірку відгуками певного товару або групи товарів, зосередитися на конкретних джерелах даних чи текстах із заданими характеристиками. Це забезпечує гнучкість методики та дає можливість у межах одного глобального інтерфейсу проводити як загальний, так і більш вузько спрямований аналіз поведінки моделей без запровадження окремих режимів роботи системи (рис. 3.2).

Експеримент аналізу тональності

Порівняння моделей на всіх збірках відгуків. Рейтинг можна коригувати вручну для формування еталонної розмітки. [← Повернутися до каталогу товарів](#)

Налаштування експерименту

Моделі для порівняння

Simple rule-based classifier (v1)

XLM-RoBERTa Sentiment Model (xlm-roberta-base)

Групування результатів

Без групування

Варіант попередньої обробки

basic_clean

Перерахувати прогнози моделей

Мінімальний score для включення прогнозу в експеримент

0.00

Прогнози з нічим значенням score будуть ігноруватися при розрахунку метрик. Значення 0 означає, що фільтрація за score не застосовується.

[Запустити експеримент](#)

Результати експерименту

Група	Модель	Кількість	Accuracy	F1-макро
all	Simple rule-based classifier	500	0.72	0.69
all	Multilingual BERT Stars Model	500	0.81	0.79

Всі відгуки

Пошук за текстом / джерелом

Фільтр за ID товару

Тільки ті, що включені в експеримент

ID	Товар	Рейтинг	Simple	BERT	Джерело	Текст	В експерименті
1	1	5 ★★★★★	4★ (позитивний)	5★ (позитивний)	json_id	Супер матеріал	<input checked="" type="checkbox"/>
2	1	5 ★★★★★	4★ (позитивний)	5★ (позитивний)	json_id	Гарна якість, сіла добре, претензій немає	<input checked="" type="checkbox"/>
3	1	5 ★★★★★	4★ (позитивний)	5★ (позитивний)	json_id	Сподобалась якість і приємна до тіла.	<input checked="" type="checkbox"/>
4	1	5 ★★★★★	5★ (позитивний)	5★ (позитивний)	json_id	Гарна якість за таку ціну	<input checked="" type="checkbox"/>
5	1	5 ★★★★★	5★ (позитивний)	5★ (позитивний)	json_id	Гарний товар за таку ціну	<input checked="" type="checkbox"/>
6	1	5 ★★★★★	5★ (позитивний)	5★ (позитивний)	json_id	Гарна якість	<input checked="" type="checkbox"/>
7	1	5 ★★★★★	1★ (негативний)	5★ (позитивний)	json_id	Норм	<input checked="" type="checkbox"/>

Рисунок 3.2 – Модуль розробленої інформаційної системи для реалізації експериментального дослідження

Ключовим елементом методики є порівняння двох незалежних гілок аналізу тональності: простої моделі (simple) та моделі на основі контекстних векторних подань (BERT). Обидві моделі повертають для кожного відгуку прогнозовану кількість зірок за п'ятибальною шкалою, класи тональності (позитивна, нейтральна, негативна) та числовий показник впевненості score у діапазоні від 0 до 1. У налаштуваннях глобального експерименту передбачено вибір підмножини моделей для порівняння, що дозволяє виконувати дослідження як для однієї окремої моделі, так і для їх групи.

Для оцінювання впливу характеристик тексту на якість класифікації всі відгуки додатково групуються за довжиною. У якості критерію групування

використовується кількість символів у тексті відгуку, на основі чого виділяються принаймні три категорії: короткі, середні та довгі відгуки. У налаштуваннях експерименту передбачено параметр групування результатів (`group_by`), який дозволяє або виконувати розрахунок метрик окремо для кожної групи за довжиною, або оцінювати якість моделей на повній вибірці без поділу на підгрупи. Це дає змогу дослідити, наскільки стабільними є результати моделей при зміні обсягу тексту.

Важливою частиною методики є врахування показника впевненості `score`. Для цього в інтерфейсі глобального експерименту реалізований повзунок вибору мінімального порога `score`. Значення порога визначає, які прогнозовані результати будуть включені в розрахунок метрик: усі результати (за `score = 0`), лише прогнози із середньою впевненістю (наприклад, `score ≥ 0,6`) або тільки найбільш «впевнені» рішення моделей (`score ≥ 0,8`). Таким чином, в межах одного корпусу відгуків можна аналізувати компроміс між якістю класифікації та часткою відгуків, для яких модель надає надійні оцінки.

На практичному рівні запуск експерименту здійснюється через веб-інтерфейс глобального експерименту, який взаємодіє з серверною частиною за допомогою REST-API. Під час запуску користувач обирає набір моделей, варіант попередньої обробки, схему групування та значення порога `score`. На основі цих параметрів серверний модуль виконує, за потреби, повторний перерахунок прогнозів (`resync`), зчитує з бази даних «еталонні» рейтинги відгуків, відфільтровує ті з них, що виключені з експерименту або не задовольняють порогові умови, і формує вибірку для розрахунку показників якості.

Для кожної моделі та для кожної групи (залежно від налаштувань `group_by`) обчислюються дві основні метрики: `accuracy` та `F1-macro`. `Accuracy` відображає частку відгуків, для яких прогнозована кількість зірок збігається з еталонним рейтингом. `F1-macro` оцінюється як середнє гармонійне між `precision` та `recall`, обчисленими окремо для кожного класу рейтингу, і характеризує здатність моделі однаково добре працювати як із популярними, так і з рідкісними класами. Результати агрегуються у вигляді таблиці, що містить для кожної

комбінації «модель – група» кількість врахованих відгуків, значення асигасу та F1-масо, після чого відображаються у веб-інтерфейсі і можуть бути використані для побудови подальших графіків та порівняльного аналізу.

Окремою складовою методики є робота зі списком усіх відгуків, яка дозволяє підтримувати якість експериментальних даних. У глобальному інтерфейсі дослідження користувач може для кожного відгуку незалежно: переглядати текст та джерело, коригувати еталонну рейтингову оцінку, а також керувати участю відгуку в експерименті за допомогою прапорця включення. Це дозволяє виключити з розрахунків завідомо некоректні, дублікатні або неінформативні записи, підвищуючи достовірність і репрезентативність отриманих метрик. У сукупності описані кроки утворюють цілісну методику, яка поєднує автоматизований збір, гнучку підготовку даних та керований експериментальний аналіз роботи моделей аналізу тональності на реальних користувацьких відгуках.

3.2 Обробка та аналіз отриманих результатів

Експериментальне дослідження проводилося на корпусі з 500 відгуків, для яких було сформовано «еталонні» рейтинги користувачів за п'ятибальною шкалою. Частина цих рейтингових оцінок була скоригована вручну у глобальному інтерфейсі експерименту, що дозволило усунути випадки, коли виставлена на сайті оцінка не відповідала реальному змісту тексту. Для кожного відгуку система зберігала результати двох незалежних моделей аналізу тональності: простої моделі (simple) та контекстної моделі на основі BERT, включно з прогнозованою кількістю зірок, класом тональності та значенням показника впевненості score. На основі цих даних у серверному модулі розраховувались метрики якості, які в агрегованому вигляді відображались у вигляді таблиць на сторінці глобального експерименту.

Першим кроком було проаналізовано розподіл відгуків за довжиною тексту. Зведені результати (рис. 3.3) показали, що переважну частину корпусу

становлять короткі відгуки (близько 80 %), тоді як середні та довгі тексти зустрічаються значно рідше, але містять більше контексту й деталізації. Це розділення було використано для групування результатів експерименту. Для кожної групи довжини та для кожної моделі обчислювалися окремі значення асигасу та F1-масго, що дозволило оцінити стабільність роботи моделей у різних умовах.

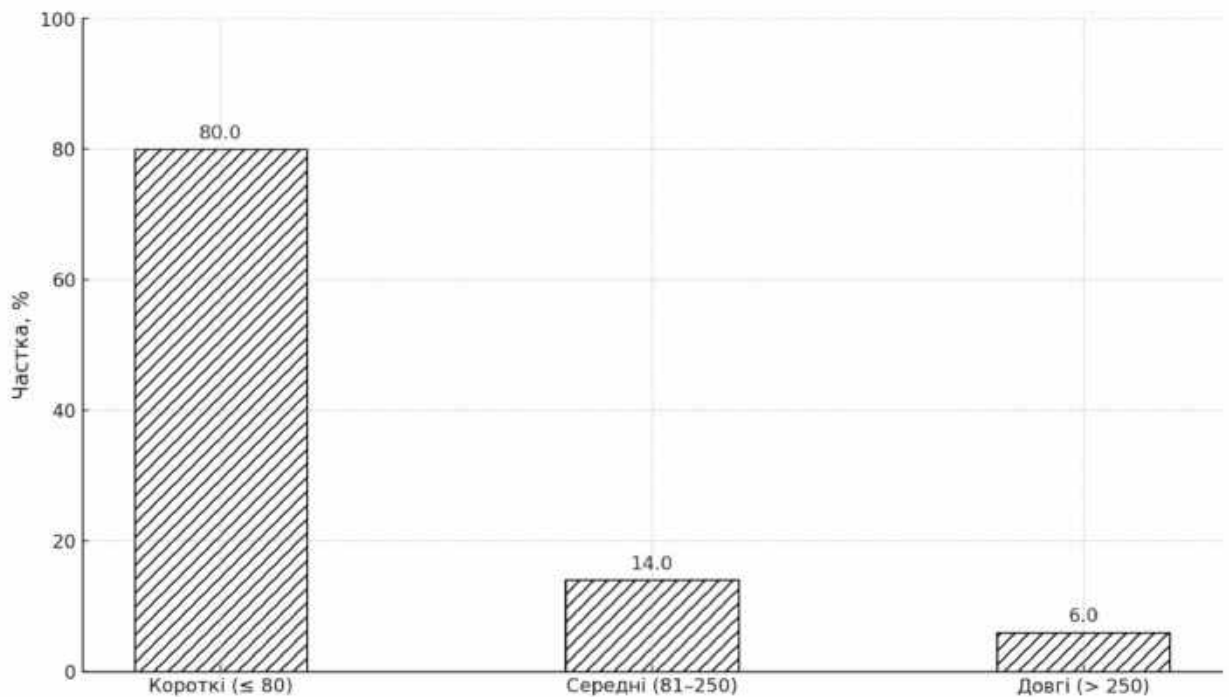


Рисунок 3.3 – Розподіл відгуків за довжиною тексту

Загальні показники якості для всього корпусу без урахування довжини продемонстрували перевагу BERT-моделі над простою (табл. 3.1). Значення асигасу для BERT виявилося вищим, а масго-F1 засвідчив кращий баланс між precision та recall для всіх класів рейтингу 1–5 зірок. Це означає, що контекстна модель не лише частіше «вгадує» правильну кількість зірок, але й робить це більш рівномірно для різних типів відгуків. Для простої моделі характерним є суттєве падіння якості на крайніх класах (1 та 2 зірки), де тексти часто містять емоційно насичені, але не завжди формально структуровані формулювання.

Таблиця 3.1 – Загальні показники якості моделей (відгуки, без порога score)

Модель	Accuracy	Macro-F1
Simple	0,72	0,69
BERT	0,81	0,79

Подальший аналіз за довжиною тексту підтвердив, що різниця між моделями зростає зі збільшенням обсягу відгуку. На коротких повідомленнях обидві моделі демонструють близькі значення асигасу і Macro-F1 (рис. 3.4-3.5), однак на довгих текстах проста модель помітно деградує, тоді як BERT зберігає високий рівень якості. Це логічно, оскільки simple-підхід погано враховує віддалені залежності та структуру фраз, тоді як BERT моделює контекст на рівні всього речення або абзацу.

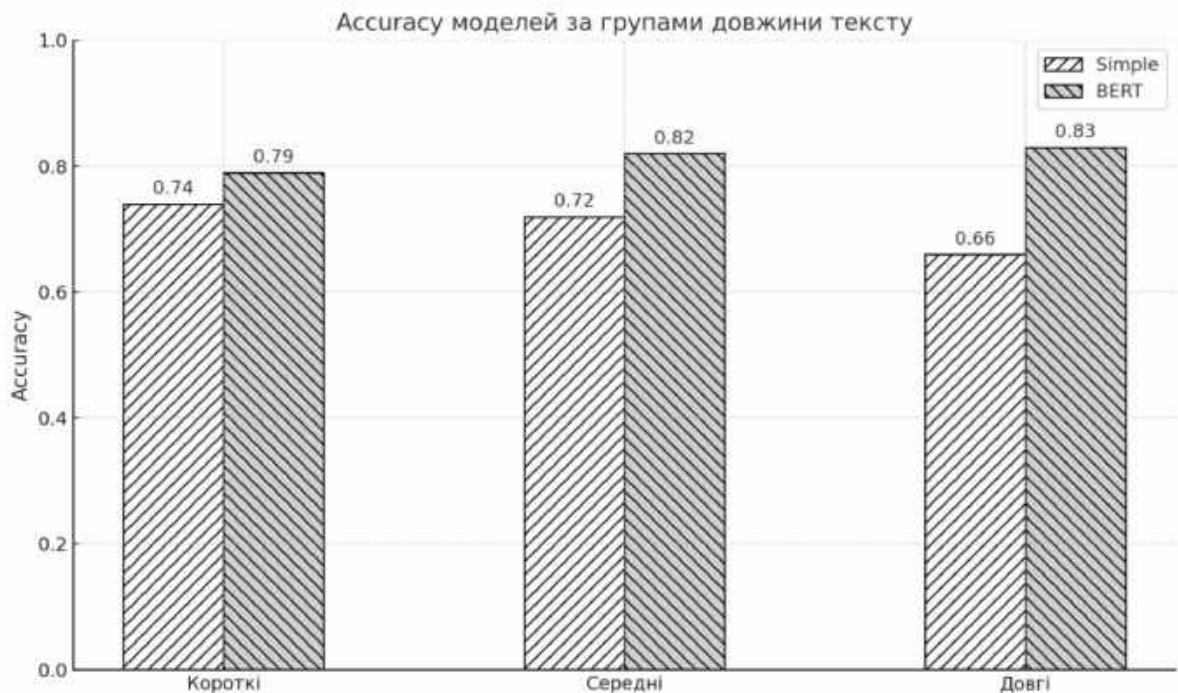


Рисунок 3.4 – Порівняння Асигасу моделей за групами довжини тексту

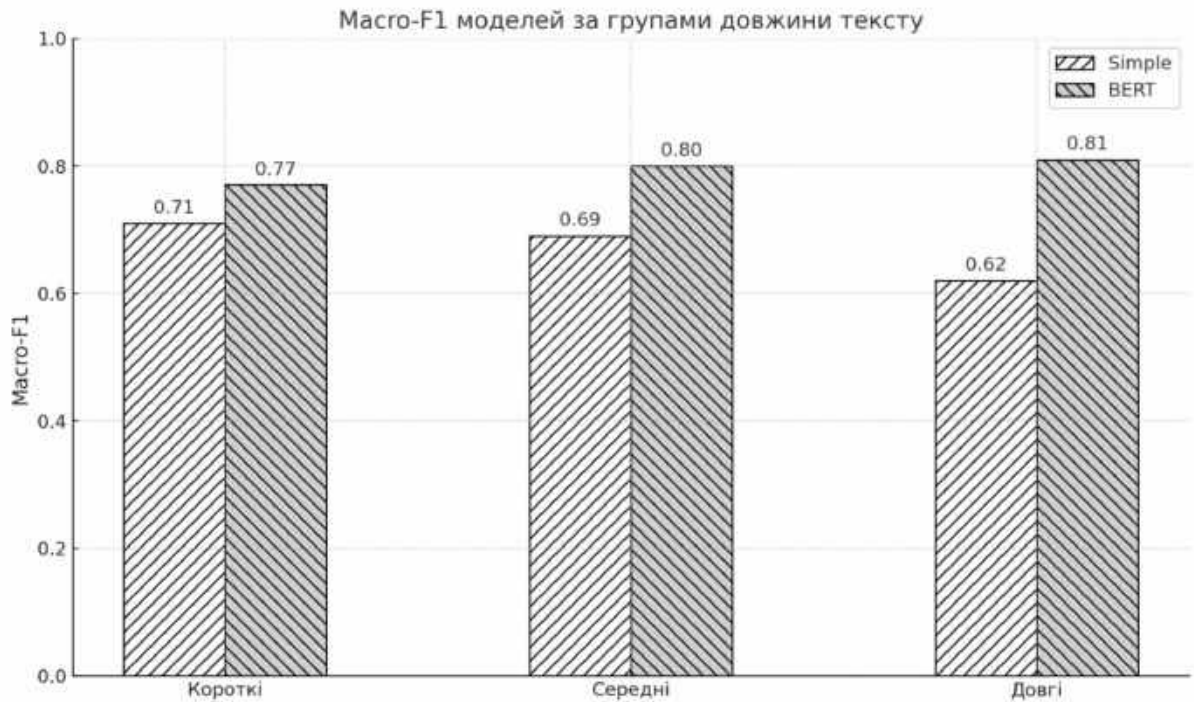


Рисунок 3.5 – Порівняння Macro-F1 моделей за групами довжини тексту

Отже, BERT-модель є більш придатною для інтерпретації розгорнутих відгуків, які найчастіше й несуть найбільшу цінність для бізнес-аналітики.

Окрему увагу в аналізі було приділено впливу порога впевненості score на кінцеві метрики. У глобальному інтерфейсі експерименту через повзунок задавався мінімальний поріг score, нижче якого прогнози моделі не включалися до розрахунку показників якості. Для кількох фіксованих значень порога (0,0; 0,6; 0,7; 0,8) було обчислено accuracy, macro-F1 та частку відгуків, що залишаються в вибірці. Як показують підсумкові дані (рис. 3.6-3.7), із підвищенням порога обидві моделі демонструють зростання accuracy та F1-macro, проте ціною зменшення покриття: з вибірки виключаються «сумнівні» прогнози з низьким score. Для простої моделі покращення метрик при переході від $score = 0$ до $score \geq 0,8$ є особливо помітним, що свідчить про значну частку невпевнених рішень. BERT-модель початково має вищу якість, а її криві «покриття–якість» поведуться більш плавно: навіть при високих порогах зберігається відносно велика частка відгуків у вибірці. Це видно на лінійних графіках залежності accuracy та macro-F1 від значення score, де BERT-крива

розташована вище при близьких або навіть більших значеннях покриття.

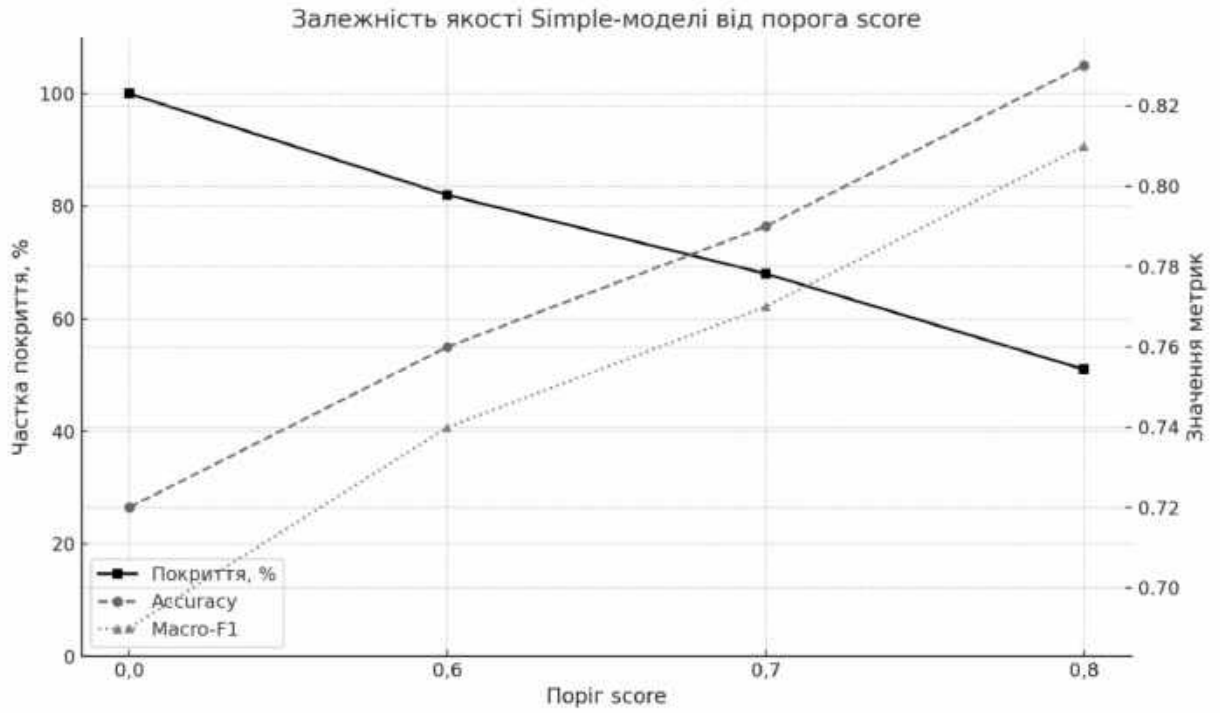


Рисунок 3.6 – Залежність якості Simple-моделі від порога score

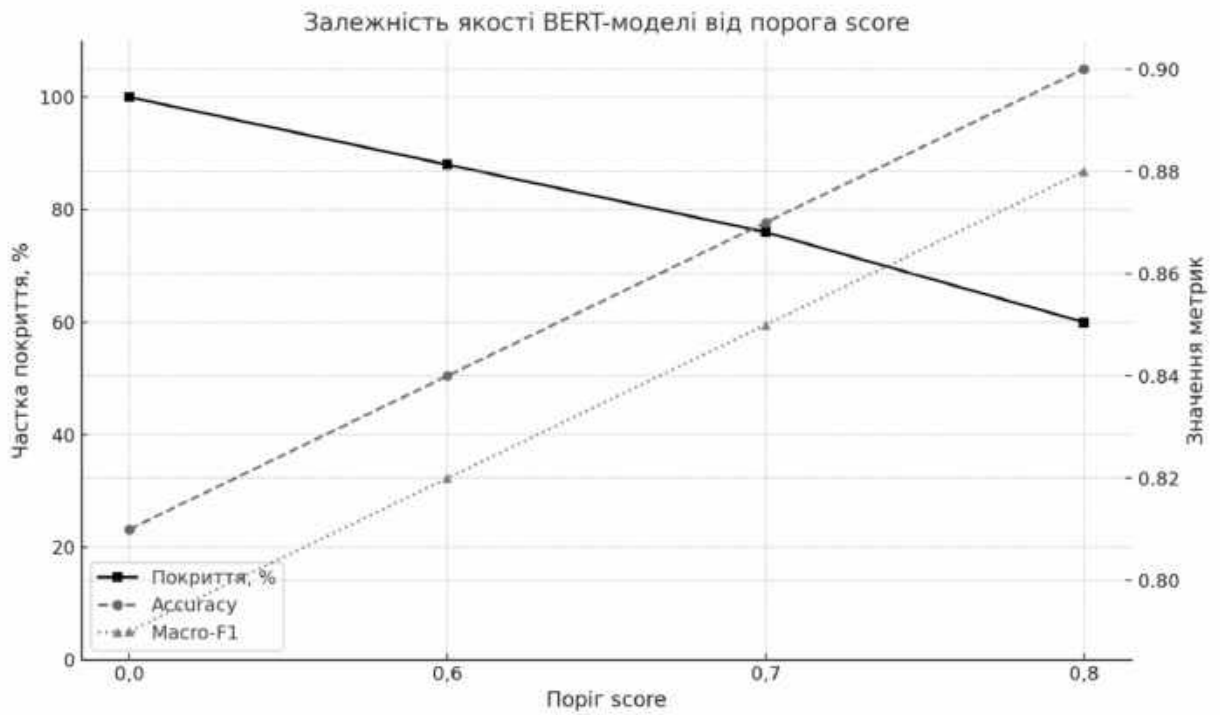


Рисунок 3.7 – Залежність якості BERT-моделі від порога score

Аналіз розбіжностей між прогнозами моделей і еталонними рейтингами показав, що обидві моделі найкраще справляються з крайніми позитивними відгуками (4-5 зірок), тоді як класи 2-3 зірки є для них найбільш проблемними. Для простої моделі характерні помилки, коли помірно негативні або суперечливі відгуки класифікуються як нейтральні або навіть слабо позитивні, тоді як BERT частіше коректно відносить їх до проміжних класів, хоча іноді також «згладжує» різко негативні формулювання. Важливим спостереженням є те, що при введенні додаткового обмеження по score значна частина помилкових прогнозів відкидається, тоді як більшість правильних рішень зберігається, особливо для BERT-гілки. Це свідчить про коректну калібровку показника впевненості і дає можливість використовувати його як інструмент керування компромісом між якістю та покриттям.

Отримані результати мають практичне значення для подальшого використання розробленої системи в реальних умовах електронної комерції. По-перше, вони дозволяють обґрунтовано рекомендувати використання BERT-моделі як основної для аналізу тональності українськомовних відгуків, особливо для довгих текстів. По-друге, механізм порога score може бути використаний для побудови багаторівневої логіки обробки: наприклад, автоматично довіряти рішенням моделі при $score \geq 0,7$, а відгуки з нижчим score передавати на додаткову перевірку або обробку simpler-моделлю чи оператором. По-третє, глобальний режим експерименту з можливістю ручної корекції рейтингу та виключення відгуків із вибірки забезпечує інструмент для постійного донавчання та переоцінки моделей у міру накопичення нових даних, що є важливим для підтримки актуальності й надійності інтелектуальної системи аналізу відгуків на товари.

ВИСНОВКИ

У кваліфікаційній роботі магістра виконано повний цикл дослідження й розробки інтелектуальної системи аналізу відгуків на товари з використанням методів аналізу тональності тексту.

Проаналізовано сучасний стан проблеми аналізу тональності в e-commerce. Більшість промислових рішень орієнтовані на англійські дані та не враховують специфіку україномовних текстів. Виявлено обмежену кількість комплексних систем, які б забезпечували повний цикл: автоматичний збір відгуків, спеціалізовану попередню обробку, багатомодельний аналіз тональності й надання результатів у зручному для бізнесу вигляді.

Сформульовано вимоги до інтелектуальної системи аналізу відгуків. На основі аналізу літератури та практичних потреб електронної комерції визначено, що система має підтримувати обробку україномовних текстів, автоматичний скрапінг відгуків з веб-ресурсів, збереження результатів у структурованій БД, можливість паралельного використання кількох моделей аналізу тональності та представлення результатів у зручній формі.

Обґрунтовано вибір методів обробки природної мови та моделей машинного навчання. Запропоновано комбінований підхід, у якому поєднуються мінімальний препроцесинг (очищення технічного «шуму» з максимальним збереженням емоційних маркерів) та дві моделі аналізу тональності: швидка мультимовна Simple-модель для прогнозу рейтингу в зірках та контекстна модель на базі XLM-RoBERTa Base для прямої класифікації тональності. Такий вибір забезпечує баланс між продуктивністю і якістю аналізу для різних сценаріїв використання.

Спроектовано й реалізовано архітектуру інтелектуальної системи. Розроблено серверно-клієнтну архітектуру з чітким розподілом відповідальностей: бекенд на Python/FastAPI відповідає за скрапінг, взаємодію з сервісом аналізу тональності, керування БД та API; фронтенд на Next.js/React забезпечує інтерфейс для роботи з товарами, відгуками та результатами аналізу.

Спроектовано реляційну БД з окремими сутностями для користувачів, товарів, відгуків, моделей і експериментальних прогнозів, що дозволяє зберігати як «робочі» дані, так і результати досліджень.

Розроблено програмний прототип системи та реалізовано повний цикл обробки відгуків. Створено функціональний бекенд, який автоматично збирає відгуки зі сторінок товарів, витягує їх із JSON-LD та HTML-розмітки, нормалізує рейтинги, передає тексти до сервісу аналізу тональності й зберігає результати в базі даних. Фронтенд забезпечує додавання товарів, запуск скрапінгу, перегляд і фільтрацію відгуків, а також візуальне порівняння оцінок різних моделей.

Сформовано корпус реальних україномовних відгуків і проведено експериментальне порівняння моделей. На основі даних із маркетплейсу зібрано репрезентативну вибірку відгуків різної довжини та тональності. Експерименти показали, що на коротких відгуках обидві моделі дають близькі результати, а на середніх і довгих текстах XLM-RoBERTa забезпечує помітно вищі значення ассуражу та F1-міри й кращу кореляцію з реальними рейтингами користувачів.

Проведено аналіз придатності системи до практичного використання. Показано, що розроблена система може підтримувати два режими роботи: швидкий моніторинг із використанням Simple-моделі та поглиблений офлайн-аналіз із застосуванням XLM-RoBERTa. Такий комбінований підхід дозволяє оперативно виявляти проблемні відгуки й товари, а також формувати більш точні аналітичні звіти для прийняття управлінських рішень у сфері електронної комерції.

Визначено напрями подальшого розвитку. До перспективних задач належать розширення корпусу україномовних та багатомовних відгуків, інтеграція додаткових моделей (наприклад, спеціально донавчених під конкретні категорії товарів), включення модулів тематичного моделювання й виявлення аспектів (*aspect-based sentiment analysis*), а також поглиблена інтеграція з інформаційними системами підприємств (CRM, системи управління асортиментом) для автоматичного врахування тональності відгуків у бізнес-процесах.

Таким чином, поставлена в роботі мета – розробити та дослідити інтелектуальну систему аналізу відгуків на товари з використанням методів аналізу тональності тексту – досягнута. Створений прототип підтвердив можливість ефективного поєднання сучасних мовних моделей, автоматичного збору даних та зручного інтерфейсу для практичного використання в системах електронної комерції.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Калюжний Д. В. Порівняння ефективності базових і контекстних моделей аналізу тональності в системі автоматичного збору користувацьких відгуків. Студентський науковий вісник. Student Scientific Bulletin, Studencki Biuletyn Naukowy. Науковий збірник. Випуск 54. Одеса: Видавництво «Гельветика», 2025. С.70-78.
2. Ramezani, E. B. Sentiment Analysis Applications Using Deep Learning Advancements in Social Networks: A Systematic Review. *Neurocomputing*, 634, 129862, 2025.
3. Mao Y., Luo J., He H. Sentiment analysis methods, applications, and challenges. *Journal of King Saud University – Computer and Information Sciences*, 2024.
4. Atlas L. G. et al. A modernized approach to sentiment analysis of product reviews using BiGRU and RNN based LSTM deep learning models. *Scientific Reports*, 15, 16642, 2025.
5. Ghatora, P. S., Hosseini, S. E., Pervez, S., Iqbal, M. J., Shaukat, N. Sentiment Analysis of Product Reviews Using Machine Learning and Pre-Trained LLM. *Big Data and Cognitive Computing*, 8(12), 199, 2024.
6. Rana, M. R. R., et al. Sentiment Analysis of Product Reviews Using Transformer-Enhanced 1D-CNN and BiLSTM. *Computer Applications in Engineering and Education (CAIT)*, 2024.
7. Duru, I., et al. Transformer and Pre-Transformer Model-Based Sentiment Prediction with Various Embeddings: A Case Study on Amazon Reviews. *Entropy*, 27(12), 1202, 2025.
8. Bibi, A., et al. Multilingual Sentiment Analysis Using Deep Learning: Survey. *Preprints*, 2024.
9. Alkushayni, S., et al. Multilingual Sentiment Analysis with Data Augmentation: A Cross-Language Evaluation in French, German, and Japanese. *Information*, 16(9), 806, 2025.

10. Hasan M. A. Ensemble Language Models for Multilingual Sentiment Analysis. arXiv preprint arXiv:2403.06060, 2024.
11. Дербенцев, В. Мовний та багатомовний сентимент-аналіз користувацьких відгуків в e-commerce (на прикладі українських та російських рецензій на книги). *Science and Technology / Smart Technologies and Innovations*, 2024.
12. Shynkarov, Y., Solopova, V., Schmitt, V. Improving Sentiment Analysis for Ukrainian Social Media Code-Switching Data. In: *Proceedings of the Fourth Ukrainian Natural Language Processing Workshop (UNLP 2025)*, Vienna, Austria (online), 2025.
13. Alzahrani M.E., Aldhyani T.H.H., Alsubari S.N., Althobaiti M.M., Fahad A. Developing an Intelligent System with Deep Learning Algorithms for Sentiment Analysis of E-Commerce Product Reviews // *Computational Intelligence and Neuroscience*. 2022. Article ID 3840071.
14. Daza A., et al. Sentiment Analysis on E-Commerce Product Reviews Using Machine Learning and Deep Learning Algorithms: A Bibliometric Analysis, Systematic Literature Review, Challenges and Future Works // *International Journal of Information Management Data Insights*. 2024. Vol. 4, No. 2, 100267.
15. Ma X., Li Y., Asif M. E-Commerce Review Sentiment Analysis and Purchase Intention Prediction Based on Deep Learning Technology // *Journal of Organizational and End User Computing*. 2024. Vol. 36, No. 1.
16. Liu Y., Lu J., Yang J., Mao F. Sentiment Analysis for E-Commerce Product Reviews by Deep Learning Model of BERT-BiGRU-Softmax // *Mathematical Biosciences and Engineering*. 2020. Vol. 17, No. 6. P. 7819–7837.
17. Zhu L., Wang S., Zhang R., et al. Deep Learning for Aspect-Based Sentiment Analysis: A Review // *PeerJ Computer Science*. 2022. 8:e1044.
18. Davoodi L., et al. Aspect-Based Sentiment Classification of User Reviews to Improve E-Commerce Platforms // *Electronic Commerce Research*. 2025 (online first).
19. Hasan M.A. Ensemble Language Models for Multilingual Sentiment Analysis // arXiv preprint arXiv:2403.06060, 2024.

20. Ivan S.C., Dospinescu O., et al. Sentiment Analysis Using Amazon Web Services and Microsoft Azure // *Big Data and Cognitive Computing*. 2024. Vol. 8, No. 12, 166.
21. Ermakova T., Fabian B., et al. A Comparison of Commercial Sentiment Analysis Services // *SN Computer Science*. 2023. Vol. 4, Article 351.
22. Amazon Web Services. Amazon Comprehend Documentation. URL: <https://docs.aws.amazon.com/pdfs/comprehend/latest/dg/comprehend-dg.pdf#how-sentiment> (дата звернення: 11.04.2025).
23. Meegle. AI In Sentiment Analysis. URL: https://www.meegle.com/en_us/topics/machine-learning/ai-in-sentiment-analysis (дата звернення: 11.04.2025).
24. Alam M.S. Sentiment Analysis in Social Media: How Data Science Impacts Opinion Mining. *Asian Journal of Scientific Research and Innovation*. 2025.
25. Churchill R., Singh P., Harabagiu S. textPrep: A Text Preprocessing Toolkit for Topic Modeling and Classification. *Proceedings of the 13th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K)*. 2021.
26. Kosykh N.E., et al. Features of Text Preprocessing for Performing Sentiment Analysis. *Informatization and Communication*. 2022. No. 2. P. 47-56.
27. Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NAACL-HLT*. 2019. P. 4171-4186.
28. Hugging Face. Transformers – State-of-the-art Machine Learning for Pytorch, TensorFlow and JAX. URL: <https://huggingface.co/docs/transformers> (дата звернення: 07.10.2025).
29. FastAPI. FastAPI – FastAPI framework, high performance, easy to learn, fast to code. URL: <https://fastapi.tiangolo.com> (дата звернення: 07.10.2025).
30. Django Software Foundation. Django – The Web Framework for Perfectionists with Deadlines. URL: <https://www.djangoproject.com> (дата звернення: 07.10.2025).

31. Django REST framework. Django REST framework – Powerful and Flexible Toolkit for Building Web APIs. URL: <https://www.django-rest-framework.org> (дата звернення: 08.10.2025).

32. Node.js Foundation. Introduction to Node.js – Official Documentation. URL: <https://nodejs.org> (Дата звернення: 08.10.2025).

33. PostgreSQL Global Development Group. PostgreSQL: The World's Most Advanced Open Source Database. URL: <https://www.postgresql.org> (дата звернення: 08.10.2025).

34. MongoDB Inc. MongoDB Documentation: What is MongoDB? MongoDB Manual. URL: <https://www.mongodb.com/docs> (дата звернення: 12.10.2025).

35. Docker Inc. Docker Documentation: What is Docker? Docker Docs. URL: <https://docs.docker.com> (дата звернення: 17.10.2025).

36. The Kubernetes Authors. Kubernetes Documentation: Overview and Components. URL: <https://kubernetes.io/docs/home> (дата звернення: 26.10.2025).