
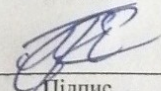


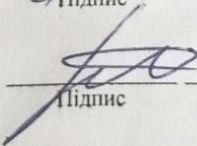
## КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

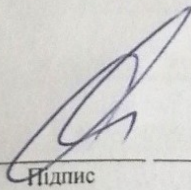
на тему Метод виявлення фейкових новин з використанням великих мовних моделей

Галузь знань 12 – Інформаційні технології  
Шифр і назва галузі знань  
Спеціальність 122 – Комп'ютерні науки  
Шифр і назва спеціальності  
Освітня програма Комп'ютерні науки  
Назва освітньої програми

Виконав: студент групи КН-21-1  Тімур МУСТАЄВ  
Група виконавця Підпис Ім'я, ПРІЗВИЩЕ

Керівник: док. філ., ст. викл. каф. КН  Павло РАДЮК  
Науковий ступінь, посада Підпис Ім'я, ПРІЗВИЩЕ

Нормоконтроль: к.т.н., доц. каф. КН  Руслан БАГРІЙ  
Науковий ступінь, посада Підпис Ім'я, ПРІЗВИЩЕ

До захисту допускаю:  
зав. кафедри КН, д.т.н., професор  Олександр БАРМАК  
Підпис Ім'я, ПРІЗВИЩЕ

09 06 2025 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій

Кафедра комп'ютерних наук

Освітній ступінь бакалавр

Галузь знань 12 – Інформаційні технології

Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук

(підпис)

д.т.н., професор Олександр БАРМАК

« 10 » 02 2025 року

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА**

1. Тема кваліфікаційної роботи бакалавра: «Метод виявлення фейкових новин з використанням великих мовних моделей»

2. Завдання видано студенту Тімуру Мустаєву

(ім'я, прізвище)

3. Керівник роботи старший викладач кафедри КН Павло Радюк

(посада, ім'я, прізвище)

4. Затверджено наказом університету від « 07 » 02 2025 р. № 023

5. Дата видачі завдання студенту: « 10 » 02 2025 р.

6. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета кваліфікаційної роботи бакалавра – підвищення точності виявлення фейкових новин з використанням великих мовних моделей. Метою роботи досягнуто внаслідок розв'язання таких завдань: спроектовано метод виявлення фейкових англomовних новин з використанням великих мовних моделей; розроблено архітектуру застосунку для виявлення фейкових англomовних новин; проаналізовано актуальні великі мовні моделі та обрано одну з них для розв'язання завдань роботи; створено програмну реалізацію у вигляді вебзастосунку; проведено експериментальне тестування реалізованого вебзастосунку за еталонними наборами даних.

7. Календарний план виконання кваліфікаційної роботи бакалавра:

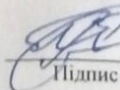
№	Назва етапів (розділів) кваліфікаційної роботи бакалавра	Термін виконання	Примітка
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи бакалавра з керівником, складання календарного графіка виконання	січень 2025	виконано
2	Ознайомлення з предметною областю, формулювання мети і задач дослідження, визначення об'єкта та предмета дослідження	лютий 2025	виконано
3	Проектування та розроблення методу вирішення завдання, загальної архітектури програмного забезпечення, інтерфейсу користувача, вибір засобів реалізації програмного забезпечення	березень 2025	виконано
4	Створення та тестування програмного забезпечення, дослідження ефективності, висновки з виконаної роботи	квітень 2025	виконано
5	Написання пояснювальної записки, урахування зауважень керівника, оформлення згідно з вимогами	травень 2025	виконано
6	Розробка презентаційних матеріалів та попередній захист кваліфікаційної роботи	травень 2025	виконано
7	Отримання відгуку керівника, рецензії, перевірка на плагіат, нормоконтроль	червень 2025	виконано
8	Підготовка до захисту та захист кваліфікаційної роботи	червень 2025	виконано

Виконавець: студент групи КН-21-1  
Група виконавця

  
Підпис

Тімур МУСТАЄВ  
Ім'я, ПРІЗВИЩЕ

Керівник: док. філ., ст. викл. каф. КН  
Науковий ступінь, посада

  
Підпис

Павло РАДЮК  
Ім'я, ПРІЗВИЩЕ

## Анотація

Тема кваліфікаційної роботи бакалавра: «Метод виявлення фейкових новин з використанням великих мовних моделей»

Виконавець кваліфікаційної роботи бакалавра: студент групи КН-21-1 Тімур МУСТАЄВ

Керівник кваліфікаційної роботи бакалавра: доктор філософії, старший викладач кафедри КН Павло РАДЮК

Кваліфікаційна робота бакалавра містить:

Пояснювальна записка				Кількість додатків
Сторінок	Рисунків	Таблиць	Джерел інформації	
57	22	17	47	2

Метою кваліфікаційної роботи бакалавра є підвищення точності виявлення фейкових новин. Метою роботи досягнуто в результаті створення методу виявлення фейкових англomовних новин з використанням великих мовних моделей. Запропонований метод реалізовано у вигляді вебзастосунку через використання мови програмування Python, фреймворку Django, бази даних PostgreSQL, платформи Docker, класифікатору XGBoost та великої мовної моделі Gemini text-embedding-004.

Розроблений вебзастосунок дає змогу виявляти фейкові новини та керувати моделями класифікації, що використовуються. Користувач також має можливість формувати аналітичні дані для опису точності роботи моделей.

Ключові слова: новини, фейк, великі мовні моделі, класифікація, вебзастосунок, база даних.

Виконавець: студент групи КН-21-1  
Група виконавця

  
Підпис

Тімур МУСТАЄВ  
Ім'я, ПРІЗВИЩЕ

## Зміст

Перелік скорочень .....	3
Вступ.....	4
Розділ 1 Аналіз методів, засобів та підходів до виявлення фейкових новин .....	6
1.1 Актуальність задачі виявлення фейкових новин засобами глибокого навчання .....	6
1.2 Огляд теоретичних підходів до виявлення фейкових новин.....	7
1.3 Аналіз наявних програмних засобів та наукових рішень для виявлення фейкових новин з використанням великих мовних моделей .....	11
1.4 Мета та основні завдання кваліфікаційної роботи бакалавра .....	15
Розділ 2 Проектування методу виявлення фейкових новин з використанням великих мовних моделей .....	17
2.1 Основна ідея методу виявлення фейкових новин.....	17
2.2 Кроки та схема методу виявлення фейкових новин.....	18
2.3 Процедура формування моделі для виявлення фейкових новин .....	19
2.4 Опис використаних великої мовної моделі та моделі машинного навчання в рамках спроектованого методу.....	21
2.5 Підготовка робочих вхідних даних для системи .....	23
2.6 Функціональна структура програмної реалізації запропонованого методу виявлення фейкових новин .....	23
2.7 Проектна архітектура вебзастосунку та взаємозв'язок його основних компонентів .....	25
2.8 Інформаційна структура вебзастосунку .....	27
2.9 Особливості використання спеціалізованих програмних компонентів .....	31
2.10 Метрики оцінювання .....	32
2.11 Висновки до розділу 2 .....	34
Розділ 3 Експериментальне дослідження методу виявлення фейкових новин з використанням великих мовних моделей.....	35

3.1	Опис програмної реалізації для дослідження методу виявлення фейкових новин з використанням великих мовних моделей.....	35
3.1.1	Визначення шляхів дослідження застосунку.....	35
3.1.2	Засоби розробки застосунку .....	36
3.1.3	Діаграма модулів застосунку.....	37
3.1.4	Особливості реалізації програмних складових застосунку.....	39
3.1.5	Експериментальне тестування вебзастосунку та вимоги до його розгортання.....	42
3.2	Результати класифікації новин за спроектованим методом .....	45
3.3	Висновки до розділу 3 .....	51
	Загальні висновки.....	52
	Перелік посилань.....	54
	Додатки	

**Перелік скорочень**

<b>Скорочення, термін, позначення</b>	<b>Пояснення</b>
NLP	Natural Language Processing
ЗМІ	Засоби Масової Інформації
LLM	Large Language Model
SVM	Support Vector Machine
RNN	Recurrent Neural Network
CNN	Convolutional Neural Network
API	Application Programming Interface
БД	База Даних
MOE	Mixture Of Experts

## Вступ

Кваліфікаційна робота бакалавра присвячена підвищенню точності виявлення фейкових новин через створення та реалізацію методу виявлення фейкових новин з використанням великих мовних моделей. Результати роботи покликані зменшити відсоток пропаганди, неправдивих фактів та історій в інформаційних середовищах.

**Актуальність.** Новини завжди були невід’ємною частиною життя будь-якого суспільства, а станом на сьогодні вони ширяться інформаційним простором як ніколи раніше: розвиток інтернету спричинив наявність дуже великої кількості джерел, усі з яких створюють статті чи пости у соціальних мережах та наголошують на своїй правдивості. Це призводить до збільшення кількості неправдивих новин, які станом на сьогодні є досить ефективним інструментом для дезінформації: у 2019 році в Україні було проведено дослідження «Ставлення населення до ЗМІ та споживання різних типів медіа у 2019 р.», згідно з яким лише 11% опитаних осіб, які знають про існування фейкових новин, змогли правильно відрізнити правдиву інформацію від неправдивої [1].

На сьогодні є низка рішень проблеми виявлення фейкових новин: одні використовують алгоритми машинного навчання, інші – нейронні мережі тощо. Проте в останні роки стрімко розвинулась технологія, відома як великі мовні моделі (LLM). Її перспективи полягають у високому рівні масштабування, продуктивності, підтримки багатьох мов, зручності у використанні [2]. LLM мають багато сфер застосування і однією з них є саме задача класифікації. Отже, проєктування методу виявлення фейкових новин є актуальним завданням, а використання великих мовних моделей – його перспективним розв’язком.

**Об’єкт дослідження** – процес оцінювання достовірності новинних джерел.

**Предмет дослідження** – алгоритми, засоби та підходи до виявлення фейкових новин з використанням великих мовних моделей.

**Мета кваліфікаційної роботи бакалавра** – підвищення точності виявлення фейкових новин з використанням великих мовних моделей.

**Завдання кваліфікаційної роботи бакалавра** – провести аналіз сучасних методів виявлення фейкових новин; спроектувати метод виявлення фейкових англomовних новин з використанням великих мовних моделей; виконати програмну реалізацію виявлення фейкових англomовних новин у вигляді вебзастосунку; провести експериментальне тестування реалізованого вебзастосунку за еталонними наборами даних.

## Розділ 1 Аналіз методів, засобів та підходів до виявлення фейкових новин

### 1.1 Актуальність задачі виявлення фейкових новин засобами глибокого навчання

Як у сучасності, так і за часів витоків цивілізації, новини відіграють важливу роль у житті суспільства, встановлюючи його настрої та надаючи нову інформацію. Протягом історії утворювались та розвивались різні джерела новин: від простого пліткування до різних форм оголошень, газет та інтернет-статей. Новини повинні бути неупередженими та просто переказувати інформацію, базуючись на фактах та об'єктивності, проте в реальності ж також існує поняття фейкових новин – неправдивої інформації, що є поданою під виглядом доповіді, якому можна довіритись [3].

Винахід мережі Інтернет призвів до того, що у 21 столітті новини отримали надзвичайно великий рівень впливу, оскільки більшість людей на Землі мають до них доступ незалежно від часу та локації. Це та те, що чимала частина сучасних людей виробила залежність від інтернету [4] сприяло тому, що фейкові новини набули неабиякої популярності як засіб маніпуляції суспільною думкою, просування власних наративів тощо. Сьогодні існує багато способів та технік, за допомогою яких можна поширювати неправдиву інформацію та відповідно до яких можна здійснити умовне розбиття фейкових новин на категорії, які конкретизовано у таблиці 1.1 [5].

Таблиця 1.1 – Категорії фейкових новин

Назва	Опис
Оманливий зв'язок	Новина, в якій заголовки, картинки або описи не відповідають реальному змісту
Оманливий контекст	Має правдивий зміст, проте поширюється з використанням оманливої контекстуальної інформації

Продовження таблиці 1.1

Маніпулятивний зміст	Новина маніпулює правдивою інформацією задля обману
Зміст-самозванець	Така новина імітує правдиві джерела інформації
Пародія чи сатира	Новина, яка наповнена гумором у різних його формах виключно для розваги своєї аудиторії
Сфабрикований зміст	Новина є повністю неправдивою, створена для обману

Фейкові новини є особливо дієвими у завданні обману не тільки через їх велике структурне різноманіття, а й через високий рівень інформаційного потоку в сучасних соціальних мережах, які є одним з основних джерел новин для багатьох людей та навіть для низки ЗМІ, зокрема, згідно дослідження Інституту масової інформації [6] 40 відсотків новин 10 найпопулярніших українських онлайн-видань мають за джерела інформації соціальні мережі. Велика кількість інформації сприяє тому, що навіть досвідчені журналісти припускаються помилок при класифікації новин на правдиві та фейкові, що у свою чергу призводить до збільшення масштабів дезінформації людей.

Отже, фейкові новини є серйозною проблемою для суспільства, відтак їх виявлення є актуальним завданням для рішення якого можна використати засоби комп'ютерних наук.

## 1.2 Огляд теоретичних підходів до виявлення фейкових новин

У межах комп'ютерних наук задачу класифікації тексту, чим і є виявлення фейкових новин, можна розв'язати, використовуючи різні інформаційні та програмні засоби, зокрема:

- машинне навчання;
- обробка природної мови (NLP);
- нейронні мережі.

Машинне навчання – це підкатегорія штучного інтелекту, яка полягає у тому, що система самостійно навчається і вдосконалює результати своєї роботи, приймаючи великі об’єми даних [7]. Наприклад, один з популярних алгоритмів, метод опорних векторів (SVM), полягає у визначенні оптимальної гіперплощини, яка максимізує відстань між кожним класом у N-вимірному просторі [8]. SVM часто використовують для вирішення завдань класифікації через її гнучкість, оскільки нею можна розв’язувати як лінійні (ті, де можливо чітко розділити дані на класи без додаткових видозмін), так і нелінійні задачі; даний алгоритм є більш ефективним за інші методи машинного навчання, такі як логістична регресія чи наївний Баєс, проте порівняно з нейронними мережами має гіршу гнучкість, масштабування, а також більш затратний з точки зору обчислень [8].

Обробка природної мови, NLP, займається питанням взаємодії комп’ютера та будь-яких природних мов [9], намагаючись покращити «розуміння» машинами людського синтаксису. До переваг NLP можна віднести швидкість роботи та варіативність сфер застосування, в той час як недоліками даного підходу є погане розуміння деяких мов та словесних конструкцій та сильна залежність від тренувального набору даних, яка може призвести до упереджених результатів [10].

Нейронні мережі встигли набути широкої популярності та використовуються для вирішення багатьох проблем, зокрема і для текстової класифікації. Суть нейромереж полягає в імітації діяльності людського мозку, взаємодії його нейронів між собою [11]. Класичним представником є рекурентна нейронна мережа, або ж RNN, яка застосовується для оброблення послідовних даних [12]. Особливістю RNN є те, що у ній, на відміну від нейромереж прямого розповсюдження, інформація рухається циклічно, тобто не тільки вперед, за рахунок чого рекурентні мережі здатні запам’ятовувати вхідні дані та розуміти їх контекст, через що даний підхід широко застосовується для вирішення задачі класифікації текстів [12]. Різниця між RNN та нейронними мережами прямого розповсюдження також продемонстрована на рисунку 1.1.

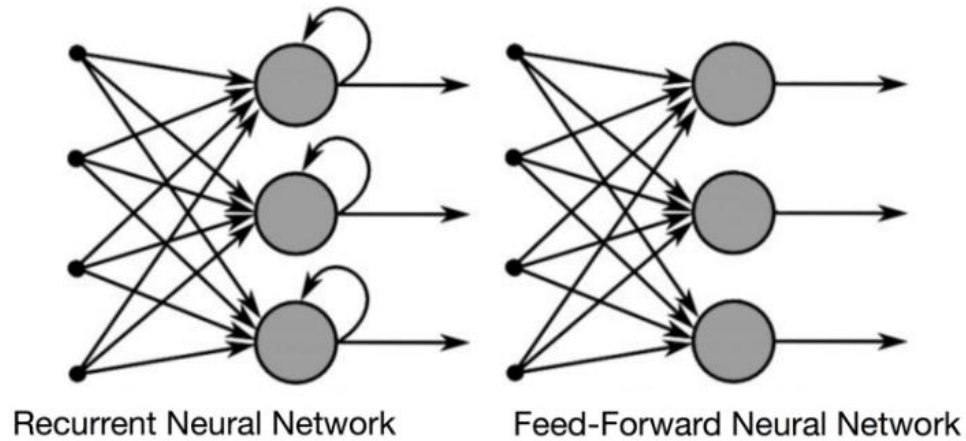


Рисунок 1.1 – Потік даних у RNN та у нейронній мережі прямого розповсюдження [12]

Внаслідок тривалого розвитку нейронних мереж з'явилась окрема підкатегорія штучного інтелекту – глибоке навчання. В рамках цієї галузі було створено нову архітектуру нейромереж, яка називається трансформером. Трансформери є базованими на RNN та виправляють їх недоліки [13]:

- архітектура дає змогу звертати увагу на слова навіть якщо відстань між ними у тексті є надто великою;
- покращено швидкість роботи.

Трансформери стали популярною архітектурою, які є основою для великих мовних моделей, LLM – на сьогодні вони є одним з найактуальніших підходів до вирішення низки різних завдань. Важливим для великих мовних моделей є концепт embeddings – вектори числових значень, утворених із вхідного тексту, призначені для розуміння їх моделлю [14]. Варто зазначити, що LLM також працюють з текстом через розділення його на токени – це найменша одиниця, на яку можна розбити текст, що отриманий моделлю на вході [15].

Однією з перших LLM, яку розробили у 2018 році, була BERT [16]. Дана LLM показала високий рівень ефективності обчислень та є винятковою у розумінні мовної структури та контексту, оскільки у випадку аналізу тексту вона враховує не тільки попередні слова, а й наступні по відношенню один до одного [17]. BERT дала поштовх для розвитку LLM та глибокого навчання в цілому, що

призвело до появи ChatGPT – одного з найпопулярніших інструментів для вирішення різноманітних завдань. У той час, як BERT потрібно використовувати програмно, ChatGPT розроблений у вигляді чат-боту, завдяки чому навіть недосвідчені користувачі можуть легко його використовувати, проте на відміну від першої LLM, GPT може аналізувати текст лише в одну сторону – справа на ліво [18].

Внаслідок проведеного аналізу описаних вище підходів було сформовано таблицю для підбиття підсумків (таблиця 1.2):

Таблиця 1.2 – Порівняння підходів до виявлення фейкових новин

Назва Підходу	Що розв’язує	Що не розв’язує
SVM	Може обробляти нелінійні дані; ефективний при великій кількості ознак; стійкий до появи аномальних значень	Вимогливість до часових та апаратних ресурсів; обмежена інтерпретація результатів; складність налаштування
NLP	Швидкість роботи; простий в імплементації	Залежність від навчальних даних; погане розуміння деяких мов; проблеми з розумінням контексту
RNN	Розуміння контексту; може обробляти послідовності різних довжин	Упередження до новіших даних; обмежена пам’ять; не є стійкою до перенавчання
BERT	Двосторонній аналіз тексту; високий рівень ефективності; наявність навчених моделей на різних мовах світу	Тривалий час роботи; необхідність впровадження на програмному рівні
ChatGPT	Зручність та простота використання; сильне розуміння контексту	Інтеграція у застосунки вимагає фінансових ресурсів; односторонній аналіз тексту

Переходячи від підходів до самої задачі класифікації тексту, при розв’язанні проблеми сам текст прийнято спочатку піддавати процесу нормалізації – перетворення тексту у стандартизовану форму, який може включати в себе такі кроки, як переведення усіх символів у нижній регістр, видалення розділових знаків тощо [19].

З огляду на переваги та недоліки кожного з описаного підходів, було обрано великі мовні моделі як технологію, що варто застосувати для вирішення проблеми виявлення фейкових новин.

### 1.3 Аналіз наявних програмних засобів та наукових рішень для виявлення фейкових новин з використанням великих мовних моделей

Завдання класифікації тексту та виявлення фейкових новин зокрема за час свого існування встигли отримати багато різних розв’язків у вигляді програмних засобів, але попри це навіть сьогодні розробляються нові рішення з метою поліпшення ефективності, швидкості роботи тощо.

Як одна з найбільш провідних компаній світу, Google також займається описаною проблемою, тож в якості рішення вони створили сервіс під назвою Google Fact Check Tools [20]. Даний сайт дає змогу переглядати перелік нещодавніх новин, достовірність яких вже було перевірено (рисунок 1.2).

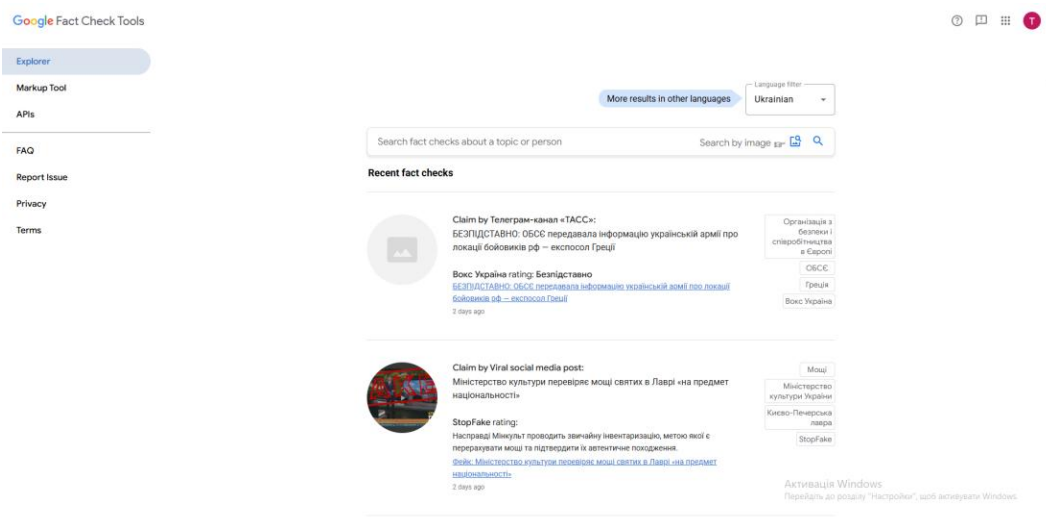


Рисунок 1.2 – Сторінка з переліком новин у Google Fact Check Tools [20]

Google Fact Check Tools можна назвати традиційним рішенням задачі класифікації новин, оскільки статті перевіряються реальними людьми – через це точність роботи є дуже високою. Ще однією перевагою є можливість перегляду новин на різних мовах, проте очевидним недоліком сервісу є великі витрати часових та людських ресурсів, у порівнянні з більш автоматизованими рішеннями, що є на ринку.

Прикладом більш сучасного рішення, яке не вимагає перевірки новин людьми, є інструмент для виявлення фейкових новин від Discourse Processing Lab [21]. Сайт пропонує 2 режими: класична перевірка ґрунтовності тексту новини на фактах, та класифікація жанру новини. Робота другого режиму продемонстрована на рисунку 1.3.

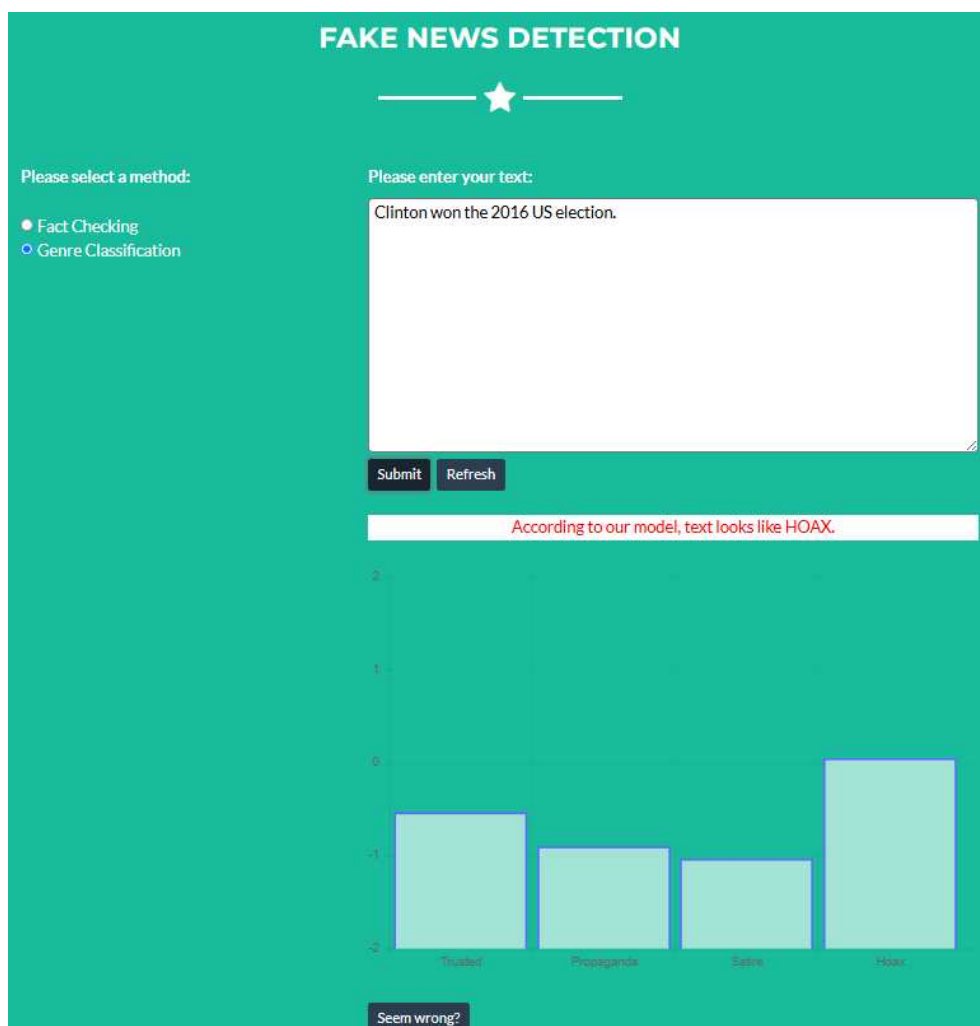


Рисунок 1.3 – Приклад класифікації жанру новини інструментом від Discourse Processing Lab [21]

Перевагами сайту є швидкість роботи, простий інтерфейс, а у випадку жанрової класифікації інструмент показує результати для всіх категорій, а не тільки назву тієї, що була обрана як найбільш ймовірна. До недоліків можна зарахувати можливість роботи лише з текстом новин та обмежену роботу з деякими мовами.

Oigetit Fake News Filter – це сервіс, який надає користувачу можливість переглядати новини, перевірені на достовірність [22]. Заявлено, що для перевірки застосовуються технології штучного інтелекту, при цьому кожній статті присвоєно відсоток достовірності та загальний настрій (сентимент) новини (рисунок 1.4).

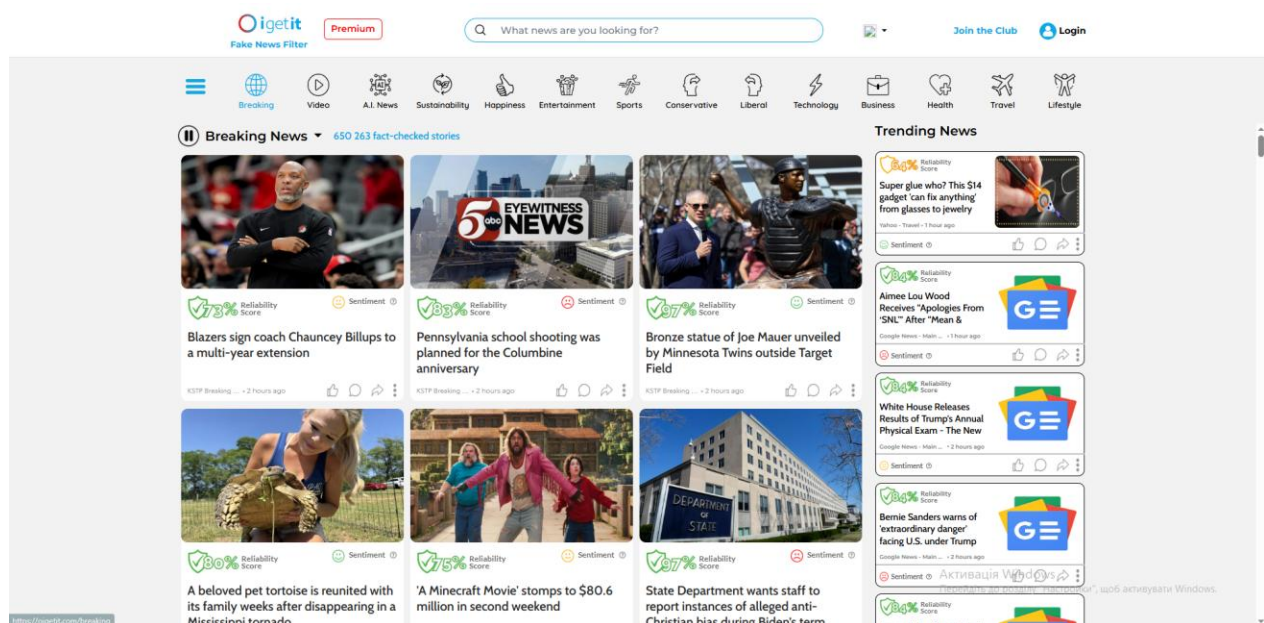


Рисунок 1.4 – Інтерфейс веб-версії Oigetit Fake News Filter [22]

До переваг сервісу можна віднести наявність версії для смартфонів, категоризацію доступних новин, зручність використання. Oigetit Fake News Filter також має і недоліки, серед яких наявність новин лише англійською мовою та відсутність можливості перевірити будь-яку новину власноруч.

Бібліотека transformers від Hugging Face [23] є open-source програмним рішенням для вирішення низки задач комп'ютерних наук, не виключаючи виявлення фейкових новин: вона містить у собі велику кількість тренуваних

моделей, які можна застосувати використовуючи, наприклад, мову програмування Python. Під open-source мається на увазі, що це бібліотека з відкритим доступом – це дає змогу вносити власні зміни у принципи роботи наданих мереж. Окрім цього до переваг даного рішення належить таке:

- висока точність запропонованих моделей;
- наявність єдиного API;
- відсутня потреба тренувати моделі самостійно.

Недоліками даного рішення є відсутність візуального інтерфейсу та необхідність навичок програміста для використання.

На тему виявлення фейкових новин також проводяться наукові дослідження: наприклад, у знайденій статті [24] в якості моделі, що використовуватиметься для розв’язку задачі, пропонується нейронна мережа OPCNN-FAKE, яка є оптимізованою версією CNN. Автори надали вичерпний опис структури розробленої ними системи та архітектури відповідної нейромережі, проаналізували точність останньої, зокрема за допомогою порівняння з іншими типовими алгоритмами машинного та глибокого навчання. Перевагою такого підходу є висока точність роботи, проте недоліками є комплексна архітектура OPCNN-FAKE та залежність від навчального набору даних.

Для узагальнення зібраної інформації сформовано таблицю 1.3.

Таблиця 1.3 – Порівняння рішень проблеми виявлення фейкових новин

<b>Назва рішення</b>	<b>Переваги</b>	<b>Недоліки</b>
Google Fact Check Tools	Точність виявлення, наявність новин на різних мовах	Необхідність за діяння експертів, менша швидкість у порівнянні з автоматизованими рішеннями
Інструмент від Discourse Processing Lab	Швидкість, можливість виявлення жанру фейкової новини, простий інтерфейс	Робота лише з текстом, обмежена підтримка певних мов

Продовження таблиці 1.3

Oigetit Fake News Filter	Наявність категоризації новин, зручність використання, існування мобільної версії	Є лише новини англійською мовою, відсутність можливості власної перевірки новини на фейковість
Бібліотека transformers	Наявні моделі є попередньо натренованими, висока ефективність моделей, наявність єдиного API	Відсутність візуального інтерфейсу; для використання потрібно розбиратись у програмуванні
OPCNN-FAKE	Висока ефективність	Комплексна архітектура, необхідність постійного оновлення тренувального набору даних та перенавчання моделі для збереження актуальності контекстів

У підсумку, проведено аналіз наявних підходів до рішення проблеми виявлення фейкових новин, в результаті якого було зроблено висновок, що використання великої мовної моделі є найкращим рішенням в рамках даної кваліфікаційної роботи.

#### **1.4 Мета та основні завдання кваліфікаційної роботи бакалавра**

Мета кваліфікаційної роботи бакалавра полягає у підвищенні точності виявлення фейкових новин з використанням великих мовних моделей. Для цього потрібно створити відповідний метод, після чого реалізувати його програмно, у вигляді вебзастосунку.

Для досягнення оголошеної вище мети є необхідним виконання наступного низки завдань:

- спроектувати метод виявлення фейкових англомовних новин з використанням великих мовних моделей;
- розробити архітектуру застосунку для виявлення фейкових англомовних новин;
- проаналізувати актуальні великі мовні моделі та обрано одну з них для розв’язання завдань роботи;
- виконати програмну реалізацію у вигляді вебзастосунку;
- провести експериментальне тестування реалізованого вебзастосунку за еталонними наборами даних;
- здійснити дослідження ефективності методу за допомогою створеного застосунку.

Отже, в першому розділі було окреслено мету кваліфікаційної роботи бакалавра та завдання, що необхідні для її досягнення. В результаті проведення аналітичних робіт було досліджено предмету область, визначено високий рівень актуальності вирішуваної проблеми – виявлення фейкових новин. Було розглянуто основні популярні підходи до рішення поставленого завдання, а також існуючі програмні рішення, в наслідок чого з’ясовано перспективність та переваги застосування великих мовних моделей. Також було чітко окреслено мету кваліфікаційної роботи та визначено перелік завдань, що є необхідними для виконання задля досягнення поставленої цілі.

## Розділ 2 Проектування методу виявлення фейкових новин з використанням великих мовних моделей

### 2.1 Основна ідея методу виявлення фейкових новин

У результаті формалізації мети кваліфікаційної роботи бакалавра та формування низки завдань, необхідних для її досягнення, було спроектовано узагальнений вигляд методу виявлення фейкових новин з використанням великих мовних моделей (рисунок 2.1):

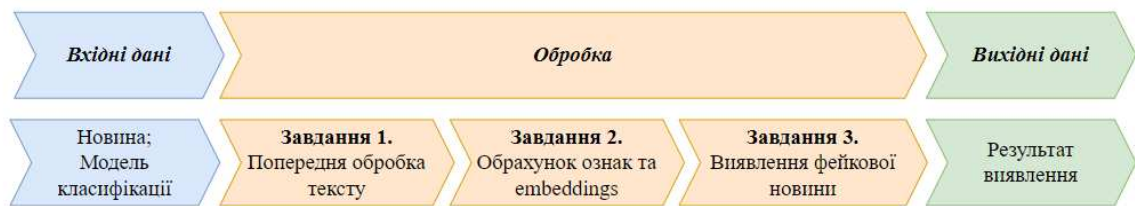


Рисунок 2.1 – Основна ідея запропонованого методу

Рисунок 2.1 демонструє, що вхідними даними для методу є, власне, сама новина та обрана конкретна модель класифікації.

Процес оброблення складається з трьох завдань:

а) на першому відбувається попередня обробка тексту з метою покращення його сприйняття моделлю;

б) на другому здійснюється обрахунок коефіцієнтів текстових ознак, наприклад суб'єктивності, та embeddings;

в) на третьому здійснюється процес виявлення фейкової новини, тобто класифікація.

Підсумком здійснення усіх кроків обробки є отримання вихідних даних – результату виявлення фейкової новини. Отже, було сформовано та описано основну ідею методу виявлення фейкових новин.

## 2.2 Кроки та схема методу виявлення фейкових новин

Після формування основної ідеї методу виявлення фейкових новин з використанням великих мовних моделей, її було деталізовано, у результаті чого розроблено схему методу, яку відображено на рисунку 2.2:

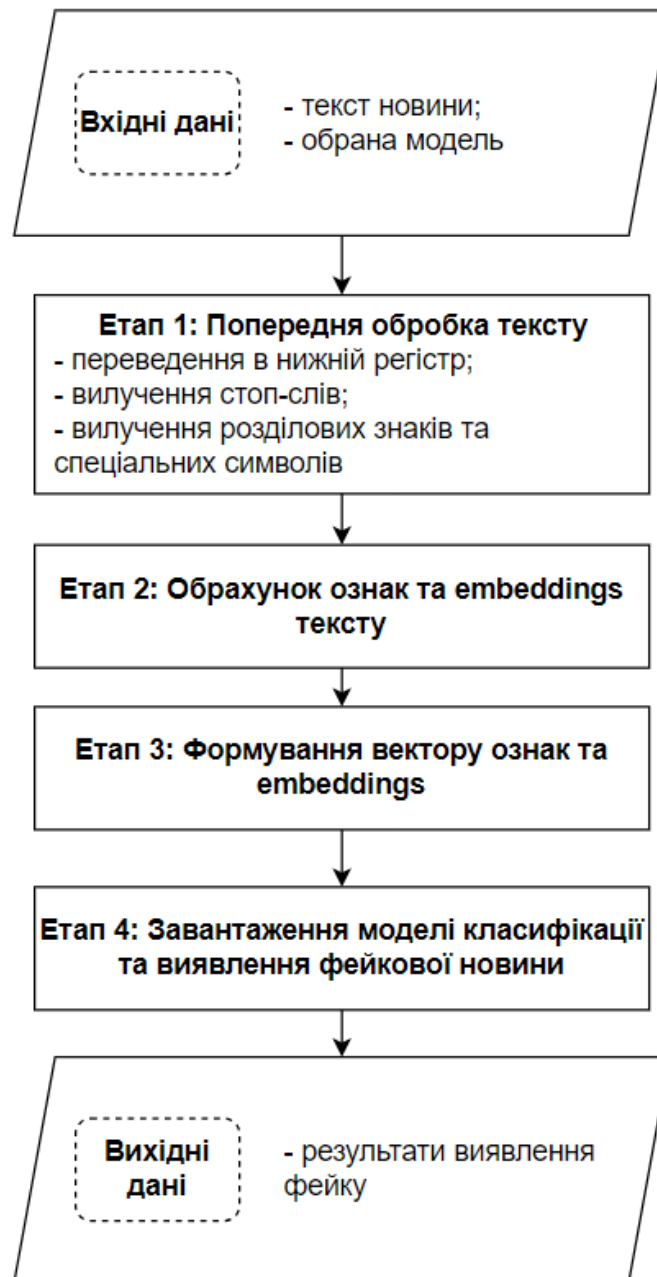


Рисунок 2.2 – Схема запропонованого методу виявлення фейкових новин з використанням великих мовних моделей

З рисунку 2.2 бачимо, що вхідними даними слугуватимуть власне текст новини та модель, що здійснюватиме класифікацію.

На етапі 1 здійснюється попередня обробка тексту, або ж по-іншому – його нормалізація, яка складається з переведення усіх слів у нижній регістр, вилучення стоп-слів, вилучення спеціальних символів та розділових знаків.

Під час етапу 2 для обробленого тексту обраховуються його ознаки, а також embeddings – останні формуються через використання великої мовної моделі.

На етапі 3 вже обраховані попередньо значення формуються у вектор ознак та embeddings, після чого фінальним кроком є завантаження моделі класифікації, якій передається вектор, та виявлення нею фейкової новини. Результатом роботи методу є числове значення від 0 до 1, яке фактично є коефіцієнтом фейковості новини: чим більше значення, тим більше новина походить на фейкову, та навпаки – чим менше значення, тим більше новина є правдивою у своєму змісті.

Під час проєктування даного методу було здійснено висновок, що процес формування моделі для виявлення фейкових новин є дуже комплексним процесом, тому його описано у наступному підрозділі.

Отже, було сформовано метод побудови моделі для виявлення фейкових новин з використанням великих мовних моделей, деталізовано усі його наявні етапи, вхідні та вихідні дані.

### **2.3 Процедура формування моделі для виявлення фейкових новин**

В результаті створення методу для виявлення фейкових новин було також розроблено процедуру формування моделі, що буде застосовуватись згаданим методом (рисунок 2.3).

Перед початком роботи необхідно зібрати та надати вхідні дані, до яких відносяться набір текстових новин, набір текстових ознак, які слугуватимуть

допоміжними вказівниками у процесі навчання моделі, набір параметрів навчання класифікатору та їхнє значення.

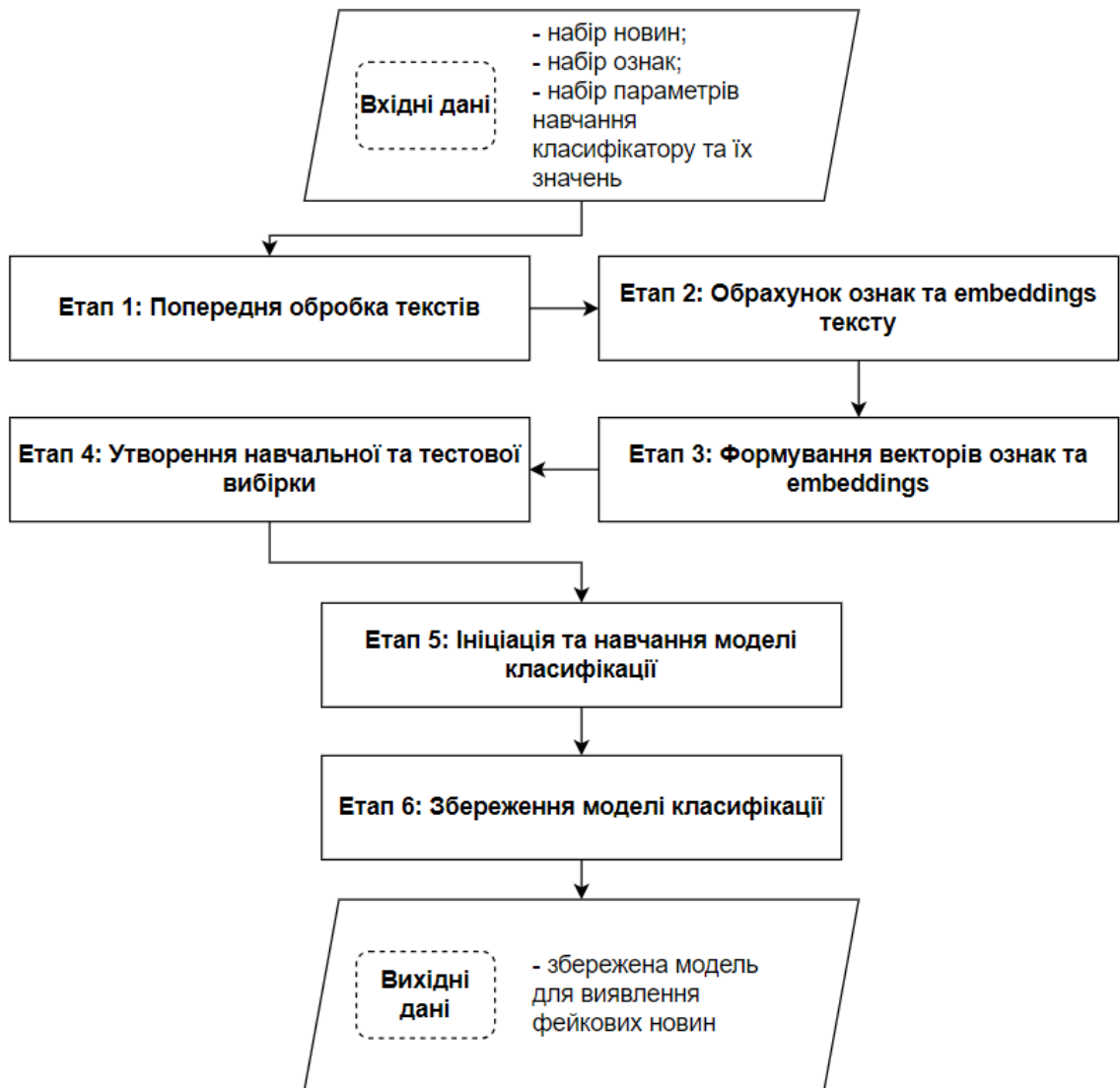


Рисунок 2.3 – Схематичне подання процедури формування моделі для виявлення фейкових новин з використанням великих мовних моделей

На етапі 1 відбувається попередня обробка текстів, яка є аналогічною до цього ж процесу у описаному попередньо методі.

Етап 2 полягає у обчисленні для усіх новин обраних ознак та embeddings, що є аналогічним до другого етапу методу.

На етапі 3 отримані числові значення ознак та embeddings формуються у єдиний вектор, і цей процес відбувається стосовно кожної новини у наборі.

Під час етапу 4 набір обрахованих векторів ділиться на відповідні навчальні та тестові вибірки, що будуть застосовуватись у процесі навчання класифікатору.

Етап 5 присвячений створенню та навчанню моделі, класифікатору машинного навчання, на основі обчислених раніше даних.

На останньому етапі відбувається збереження натренованої моделі у файлового вигляді – це уможлиблює її подальше використання у рамках методу виявлення фейкових новин з використанням великих мовних моделей.

Отже, описано процедуру формування моделі для виявлення фейкових новин з використанням великих мовних моделей, його кроки роботи.

## **2.4 Опис використаних великої мовної моделі та моделі машинного навчання в рамках спроектованого методу**

В результаті проведеного аналізу різних LLM-рішень було зроблено вибір використовувати існуючу модель, `text-embedding-004` – вона є ініціалізованою від більшої Gemini моделі, розробленої Google, та спеціалізується на обрахунку `embeddings` текстів [25]. Використання дистильованої моделі, оптимізованої під конкретну задачу, дасть можливість підвищити точність роботи методу виявлення фейкових новин.

Розробники та компанія не надають точної інформації про структуру Gemini, проте відомо, що перші версії будували на основі архітектури трансформерів, які лише розшифровують [26]. Із виходом версії Gemini 1.5 відбувся перехід до нової архітектури – суміш експертів (*mixture of experts*) [26]. Модель використовує низку нейронних мереж, кожна з яких спеціалізується на конкретній області краще за інші: коли виникає потреба виконати певне підзавдання, то воно спрямовується до найбільш підходящої мережі [27].

Що стосується конкретно `text-embedding-004`, то відомо, що у ній використовується трансформер, який аналізує текст в обидві сторони, для

вироблення послідовності токенів embeddings, а для генерації єдиного embedding по відношенню до всього вхідного тексту застосовується процес пулінгу [25].

В якості класифікатору, який здійснюватиме виявлення фейкових новин, прийнято рішення використати такий алгоритм, як XGBoost – він належить до методів ансамблевого навчання та використовує дерева рішень, інший алгоритм машинного навчання, для здійснення процесу навчання [28]. Запропонована процедура використовує низку дерев рішень, кожне з яких навчається на залишках (residuals) від попереднього, окрім першого. Отже, кожне наступне дерево в ряді навчається для виправлення помилок попереднього, що позитивно впливає на точність. У підсумку, результати навчання усіх дерев підсумовуються – це і є фінальним результатом класифікації.

У процесі аналізу XGBoost було також розглянуто і параметри навчання класифікатору, оскільки вони відіграють велику роль у побудові робочої моделі. Результатом стало звуження кола параметрів, що використовуватимуться – для їх опису сконструйовано таблицю 2.1.

Таблиця 2.1 – Обрані параметри навчання XGBoost

Назва	Опис
eta	Коефіцієнт навчання; визначає зменшення розміру кроку, який використовується для оновлення моделі під час навчання
n_estimators	Кількість дерев рішень, використовуваних для створення моделі
max_depth	Максимальна глибина одного дерева
lambda	Коефіцієнт L2-регуляризації ваг в листках дерева.

Отже, описано архітектуру обраної великої мовної моделі та машинного класифікатору, що будуть застосовані у інформаційній системі для виявлення фейкових новин.

## **2.5 Підготовка робочих вхідних даних для системи**

Для навчання моделі, що буде здійснювати виявлення фейкових новин, обрано набір даних WELFake [29]. Даний набір даних містить більше 72 000 записів та утворений через поєднання 4 інших популярних наборів даних новин [30], що забезпечує різноманітність тематик, а це у свою чергу сприятиме підвищенню точності виявлення фейкових новин.

WELFake містить таку інформацію про новини: порядковий номер (Index), заголовок (Title), текст (Text), категорія (Label).

Вирішено для вхідних даних для системи використати лише поля тексту та категорії через те, що вони є ключовими характеристиками для задачі виявлення фейкових новин. У процедурі формування моделі для виявлення фейкових новин з використанням великих мовних моделей одним із етапів роботи є попередня обробка текстів, тож при створенні нової моделі до кожної новини буде застосовано процес нормалізації.

У підсумку обрано набір даних WelFake для застосування у якості робочих вхідних даних для системи, описано процес підготовки даних.

## **2.6 Функціональна структура програмної реалізації запропонованого методу виявлення фейкових новин**

З огляду на поставлені задачі та специфіки процедури формування моделі для виявлення фейкових новин з використанням великих мовних моделей є доцільним поділ користувачів вебзастосунку на дві групи: гість та адміністратор.

Для демонстрації усіх функцій у системі було спроектовано діаграму варіантів використання (рисунок 2.4):

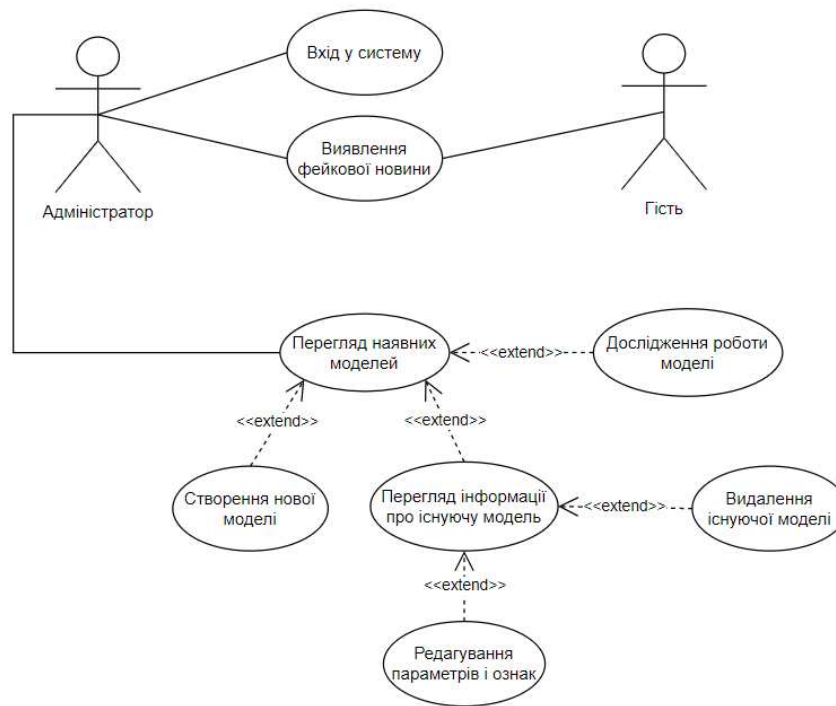


Рисунок 2.4 – Діаграма варіантів використання вебзастосунку

Як видно на рисунку 2.4, гість матиме доступ лише до однієї, ключової, функції – власне виявлення фейкової новини, у той час як головний фокус у функціоналі вебзастосунку буде на діях адміністратора, який зможе керувати моделями виявлення фейкових новин та досліджувати їх роботу. Далі розглянуто основні прецеденти для користувача-адміністратора.

*Назва прецеденту №1:* створення нової моделі.

*Діюча особа:* адміністратор.

*Мета:* зареєструвати нову модель для виявлення фейковості у системі.

*Передумови:* здійснено вхід у систему з використанням логіну та паролю.

*Послідовність:*

1. Адміністратор відкриває веб-сторінку для перегляду усіх наявних моделей у системі.

2. Адміністратор натискає на кнопку «Створити Модель».

3. У вікні, що з'явиться, адміністратор вказує назву моделі, обирає один за одним усі текстові ознаки, які необхідно застосувати для процесу навчання, обирає параметри класифікатору та вводить їх значення.

4. Адміністратор натискає на кнопку «Створити Модель».

5. Після певного очікування адміністратору відкриється сторінка для перегляду моделей із відображенням новоствореної моделі у переліку.

*Назва прецеденту №2:* видалення існуючої моделі.

*Діюча особа:* адміністратор.

*Мета:* видалити обрану модель для виявлення фейкових новин із системи.

*Передумови:* здійснено вхід у систему з використанням логіну та паролю.

*Послідовність:*

1. Адміністратор відкриває веб-сторінку для перегляду усіх наявних моделей у системі.

2. Адміністратор натискає на кнопку «Переглянути» у полі моделі, яку бажає видалити.

3. На сторінці, що відкриється, адміністратор натискає на червону кнопку із зображенням сміттєвого баку.

4. У спливаючому вікні адміністратор підтверджує свій вибір натисканням кнопки «ОК».

5. Адміністратор перенаправляється на сторінку перегляду моделей, де видалена модель вже відсутня у переліку.

Отже, розглянуто групи користувачів та доступний їм функціонал у інформаційній системі для виявлення фейкових новин із використанням великих мовних моделей.

## **2.7 Проектна архітектура вебзастосунку та взаємозв'язок його основних компонентів**

В результаті аналізу описаного раніше функціоналу, груп користувачів та методів побудови й застосування моделі для виявлення фейкових новин спроектовано схему архітектури вебзастосунку, що наведена на рисунку 2.5.



Рисунок 2.5 – Схема архітектури вебзастосунку

На рисунку 2.5 у зеленому прямокутнику знаходяться підсистеми, що належать до інтерфейсу адміністратора, у той час як третя підсистема є доступною і для гостей також.

Підсистема входу адміністраторів у систему дозволить авторизуватись завдяки введенню отриманих логіну та паролю. Окрім цього, без проходження через неї використання підсистеми, описаної наступною, є неможливим.

Підсистема роботи з моделями для виявлення фейкових новин є важливою частиною вебзастосунку, оскільки саме вона забезпечує створення класифікаторів, що використовуються для вирішення ключової задачі. Під процесом створення мається на увазі комбінація встановлення ознак, вибір параметрів та присвоєння їм певних значень, а також навчання моделей. Також дана підсистема надає змогу редагувати та видаляти наявні моделі й досліджувати їхню роботу.

Підсистема виявлення фейкової новини забезпечує здійснення класифікації наданої статті за допомогою моделей, створених із використанням попередньої підсистеми. При виборі моделі користувачу буде відображено параметри та ознаки, що було використано для її навчання.

Отже, було спроектовано архітектуру вебзастосунку виявлення фейкових новин із застосуванням великих мовних моделей, а також деталізовано принципи роботи її складових.

## 2.8 Інформаційна структура вебзастосунку

Для процесу формування інформаційної структури системи є важливим розуміння усіх сутностей системи та принципів їх взаємодій один з одним – саме тому в результаті проведеного аналізу було спроектовано діаграму «сутність-зв'язок», зображену на рисунку 2.6.

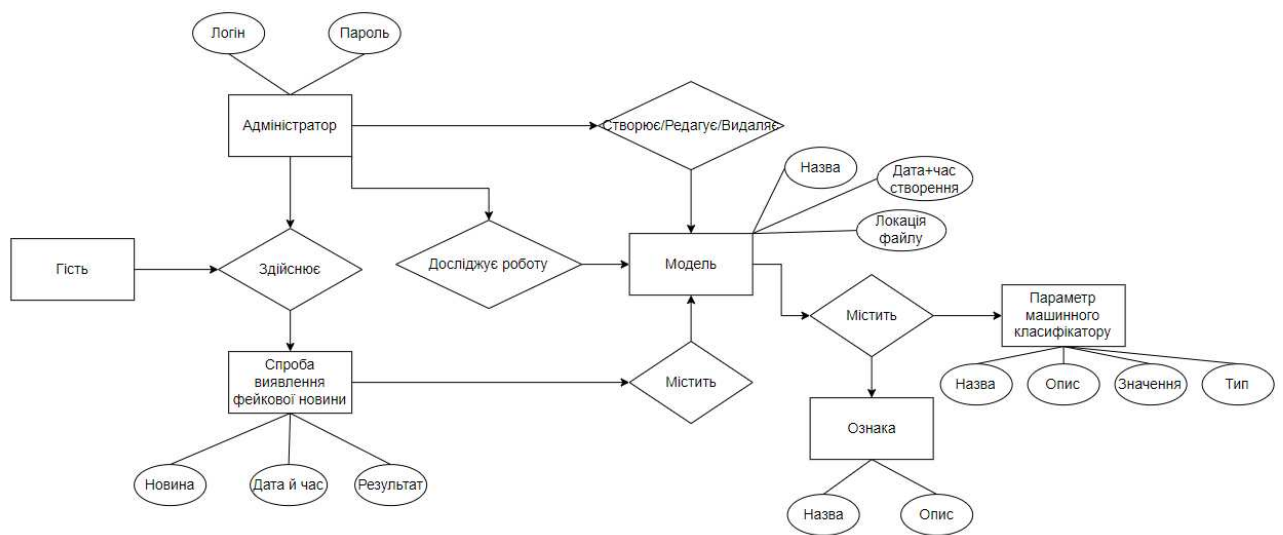


Рисунок 2.6 – Діаграма «сутність-зв'язок» вебзастосунку

Діаграма, що наведена на рисунку 2.6, демонструє, що кожен користувач може здійснювати спробу виявлення фейкової новини із такими характеристиками як текст самої новини, дата й час проведення та отриманий результат. Окрім цього, для кожної спроби необхідно обрати модель, що здійснюватиме класифікацію, відповідно її характеризуватимуть назва, час

створення, місце розташування файлу а також сутності параметрів та ознак, за допомогою яких вона навчалася. Адміністратори у свою чергу можуть створювати, редагувати, видаляти моделі класифікації та досліджувати їх роботу.

Після проведеного аналізу сутностей та їх взаємодій у межах вебзастосунку, було сформовано даталогічну модель БД, що застосовуватиметься (рисунок 2.7).

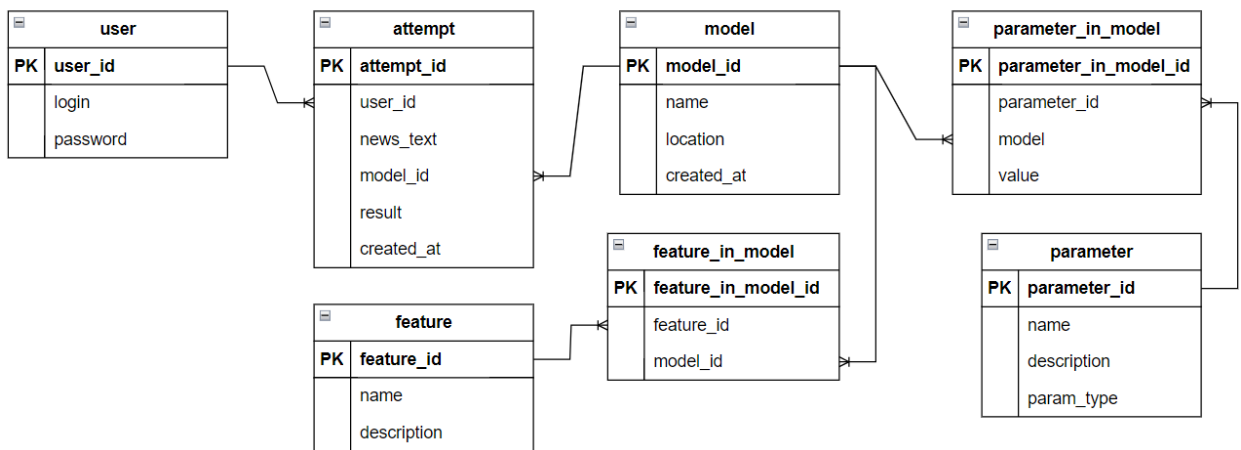


Рисунок 2.7 – Даталогічна модель БД вебзастосунку

Таблиця user призначена для збереження вхідних даних адміністраторів у систему, детальний опис полів таблиці наведено у таблиці 2.2.

Таблиця 2.2 – Атрибути таблиці «user»

Назва	Тип даних	Опис
user_id	bigint	Первинний ключ
login	character varying	Ім'я адміністратора
password	character varying	Захешований пароль

Таблиця attempt зберігає інформацію про спроби виявлення фейкових новин, здійснених користувачами. До атрибутів належать поля attempt\_id, user\_id, news\_text, model\_id, result та created\_at, які описано таблицею 2.3.

Таблиця 2.3 – Атрибути таблиці «attempt»

Назва	Тип даних	Опис
attempt_id	bigint	Первинний ключ
user_id	bigint	Вторинний ключ із посиланням на адміністратора
news_text	character varying	Текст новини, що пройшла класифікацію
model_id	bigint	Вторинний ключ із посиланням на модель
result	double	Числовий результат класифікації
created_at	timestamp	Дата та час створення запису

Таблиця model використовуватиметься для збереження базової інформації про моделі, які створюються адміністраторами, із наступними атрибутами: model\_id, name, location, created\_at. Таблиця 2.4 пояснює перелічені поля та їх призначення.

Таблиця 2.4 – Атрибути таблиці «model»

Назва	Тип даних	Опис
model_id	bigint	Первинний ключ
name	character varying	Назва моделі
location	character varying	Місце розташування файлу моделі
created_at	timestamp	Дата та час створення запису

Таблиця parameter зберігатиме відповідно параметри навчання класифікатору машинного навчання. Серед атрибутів наступні поля: parameter\_id, name, description та param\_type (таблиця 2.5).

Таблиця 2.5 – Атрибути таблиці «parameter»

Назва	Тип даних	Опис
parameter_id	bigint	Первинний ключ
name	character varying	Назва параметру
description	character varying	Опис параметру
param_type	character varying	Числовий тип параметру (Integer або Float)

Таблиця feature призначена для збереження текстових ознак новин, що будуть застосовані у процесі навчання моделей. Атрибутами даної таблиці є feature\_id, name та description (таблиця 2.6).

Таблиця 2.6 – Атрибути таблиці «feature»

Назва	Тип даних	Опис
feature_id	bigint	Первинний ключ
name	character varying	Назва ознаки
description	character varying	Опис ознаки

Розвідна таблиця у реляційних БД – це таблиця, призначена для утворення зв'язку багато-до-багатьох через збереження посилань на записи декількох таблиць одночасно [30]. У спроектованій БД вебзастосунку розвідними таблицями є feature\_in\_model для feature та model, parameter\_in\_model для parameter і model.

Таблиця feature\_in\_model зберігатиме записи про те, які текстові ознаки були використані для навчання конкретної моделі та з якими числовими значеннями. Атрибутами слугують поля feature\_in\_model\_id, feature\_id, model\_id, які описано у таблиці 2.7.

У таблиці parameter\_in\_model знаходитимуться записи про те, які параметри класифікатору машинного навчання та з якими числовими

значеннями були застосовані для навчання моделі. До атрибутів належать `parameter_in_model_id`, `parameter_id`, `model_id` та `value` (таблиця 2.8).

Таблиця 2.7 – Атрибути таблиці «feature\_in\_model»

Назва	Тип даних	Опис
<code>feature_in_model_id</code>	<code>bigint</code>	Первинний ключ
<code>feature_id</code>	<code>bigint</code>	Вторинний ключ із посиланням на ознаку
<code>model_id</code>	<code>bigint</code>	Вторинний ключ із посиланням на модель

Таблиця 2.8 – Атрибути таблиці «parameter\_in\_model»

Назва	Тип даних	Опис
<code>parameter_in_model_id</code>	<code>bigint</code>	Первинний ключ
<code>parameter_id</code>	<code>bigint</code>	Вторинний ключ із посиланням на параметр
<code>model_id</code>	<code>bigint</code>	Вторинний ключ із посиланням на модель
<code>value</code>	<code>double</code>	Числове значення параметру

У результаті описано сутності вебзастосунку виявлення фейкових новин з використанням великих мовних моделей, охарактеризовано їх принципи взаємодії, за допомогою чого було спроектовано даталогічну модель даних.

## **2.9 Особливості використання спеціалізованих програмних компонентів**

Програмна реалізація методу виявлення фейкових новин з використанням великих мовних моделей та процедури формування моделі, що здійснюватиме класифікацію, потребуватиме застосування низки спеціалізованих бібліотек, які розглянуто нижче.

Для застосування класифікатору машинного навчання XGBoost буде використано бібліотеку на мові програмування Python – xgboost [31]. Дана бібліотека надає можливість створювати, навчати та використовувати відповідний класифікатор для вирішення різних задач, у тому числі текстової класифікації. Зокрема варто відмітити, що бібліотека також надає можливість зберігати моделі у файловому форматі.

Обраний набір даних новин WELFake поданий у вигляді файлу із форматом csv: для зчитування файлу та збереження його даних буде застосовано бібліотеку pandas [32]. Ця бібліотека є частим вибором розробників при необхідності маніпулювання даними та їх аналізу. У рамках програмної реалізації методу виявлення фейкових новин з використанням великих мовних моделей зчитаний набір новин буде записано у датафрейм-структуру, яка є унікальним типом даних бібліотеки pandas.

Для розділення набору новин на навчальні та тестові вибірки використано відому бібліотеку машинного навчання – Scikit-learn [33]. Конкретизуючи, у даній бібліотеці є метод `train_test_split`, який дає змогу створити набори даних для навчання та тестування моделі у вказаному розмірному співвідношенні.

Задача обрахунку текстових ознак, пов'язаних із сентиментом тексту, буде вирішуватись через використання бібліотеки TextBlob [34]. Це бібліотека призначена для оброблення текстових даних. Вона надає API для роботи над виділенням фраз, аналізом сентименту, класифікацією тощо [34].

Отже, описано спеціалізовані бібліотеки, що буде використано для створення програмної реалізації методу виявлення фейкових новин з використанням великих мовних моделей, пояснено сфери їх застосування.

## **2.10 Метрики оцінювання**

Для того, щоб оцінити точність роботи методу виявлення фейкових новин з використанням великих мовних моделей, буде необхідно використати класичні метрики машинного навчання.

Accuracy є фундаментальною метрикою [35] – вона показуватиме відношення правильно передбачених фейкових новин до загальної кількості новин у наборі даних. У роботі також використано такі метрики, як Precision та Recall: перша описуватиме кількість правильно-виявлених фейкових новин по відношенню до загальної кількості виявлених фейкових новин, а остання – кількість точних передбачень фейків по відношенню до загальної кількості новин, що точно є неправдивими [35].

Також планується використати F1-SCORE, яка є середнім гармонійним Precision та Recall – дана метрика допоможе оцінити роботу моделі, навіть за умови, що набір даних для навчання не буде збалансованим [35]. Для вимірювання різниці між передбаченими коефіцієнтами та фактичними класами необхідно використати метрику log loss: на відміну від решти метрик, таких як Accuracy, log loss бере до уваги невпевненість у передбаченнях моделі, більш суворо караючи моделі за неправильні прогнози з високими коефіцієнтами [36].

Окрім числових метрик є доцільним застосувати графіки для більш детального дослідження ефективності моделі. Вирішено використати наступні графіки:

- Precision-Recall, який відобразатиме як змінюється співвідношення метрик Precision та Recall залежно від порогу класифікації [37];

- ROC-AUC, який показуватиме, як змінюється співвідношення між правильно класифікованими фейковими новинами (True Positive Rate) та хибно класифікованими правдивими новинами (False Positive Rate) при зміні порогу класифікації [37];

- Графік, що демонструватиме застосування техніки t-SNE для пониження розмірності даних, необхідний для вивчення питання роздільності класів [38].

У результаті, було обрано метрики та графіки, які планується використати для дослідження ефективності методу виявлення фейкових новин з використанням великих мовних моделей, та здійснено їхній опис.

## 2.11 Висновки до розділу 2

У результаті роботи над другим розділом було сформовано основну ідею методу виявлення фейкових новин з використанням великих мовних моделей, після чого спроектовано його повну версію та охарактеризовано етапи роботи. У ході аналізу моделі, що необхідно використати у методі, через її комплексність було створено схему процедури її формування та деталізовано усі кроки.

Описано моделі машинного навчання, що планується використати у програмній реалізації – велику мовну модель Google Gemini Flash 2.0 та класифікатор XGBoost, зокрема розібрано їх архітектуру, після чого було обрано набір даних для використання у якості вхідних даних та описано процес його підготовки для роботи програмної реалізації.

Проаналізовано функціональну структуру програмної реалізації згаданого методу, в результаті чого спроектовано діаграму варіантів використання та розглянуто її основні прецеденти.

Результатом аналізу функціональної структури вебзастосунку стало створення та пояснення схеми архітектури, її складових – підсистем. Було деталізовано сутності системи та їх зв'язки між собою, результатом чого стала діаграма «сутність-зв'язок». Завдяки діаграмі спроектовано даталогічну модель БД вебзастосунку, після чого розібрано усі її таблиці.

Наостанок, було описано спеціалізовані програмні компоненти – бібліотеки, що планується використати при розробці програмної реалізації спроектованої інформаційної системи, а також обрано й описано метрики та графіки, які планується використати для дослідження ефективності методу.

В результаті проєктування методу виявлення фейкових новин поставлено за подальшу мету розробити вебзастосунок для дослідження точності роботи методу для виявлення фейкових новин з використанням великих мовних моделей. Для того, щоб обґрунтувати результати дослідження, буде необхідно протестувати вебзастосунок та дослідити його функціонал.

## **Розділ 3 Експериментальне дослідження методу виявлення фейкових новин з використанням великих мовних моделей**

### **3.1 Опис програмної реалізації для дослідження методу виявлення фейкових новин з використанням великих мовних моделей**

#### **3.1.1 Визначення шляхів дослідження застосування**

Створення вебзастосування, який втілюватиме метод виявлення фейкових новин з використанням великих мовних моделей, потребує реалізації процедури формування моделі із забезпеченням користувацьких інтерфейсів для простих відвідувачів та адміністраторів, окрім цього необхідно інтегрувати БД для збереження інформації, що потім буде використана для проведення дослідження.

Вебзастосунок повинен надавати будь-якому користувачу можливість виявити фейкову новину, тим часом як адміністратор повинен мати доступ до функцій, пов'язаних з роботою з моделями класифікації. Для тестування функцій вебзастосування була проведена низка експериментальних тестувань.

Дослідження роботи методу виявлення фейкових новин з використанням великих мовних моделей буде здійснено по відношенню до таких метрик, як Accuracy, Precision, Recall, F1-SCORE, log loss та таких графіків, як Precision-Recall, ROC-AUC, t-SNE. Завдяки такому підходу планується дослідити вплив значень таких параметрів, як коефіцієнт навчання, максимальна глибина дерева, кількість дерев, коефіцієнту L2-регуляризації ваг в листках дерева а також таких ознак тексту, як сентимент та суб'єктивність. Окрім цього, планується використати такі текстові ознаки, як сентимент та суб'єктивність, та виявити їх вплив на якість класифікації. Для проведення дослідження буде використано окрему функцію, доступну адміністратору, яка полягає в обчисленні згаданих метрик та формуванні графіків.

У підсумку, описано яким повинен бути вебзастосунок, що реалізує метод виявлення фейкових новин з використанням великих мовних моделей, наявний у ньому функціонал, характер дослідження реалізованого методу, які ознаки й параметри планується застосувати.

### 3.1.2 Засоби розробки застосунку

Для створення вебзастосунку, що буде результатом програмної реалізації методу виявлення фейкових новин з використанням великих мовних моделей, заплановано використати мову Python, фреймворк Django, платформу Docker та БД PostgreSQL.

Станом на сьогодні Python досі залишається однією з найбільш популярних та використовуваних мов програмування, що робить її найкращим вибором для використання у розробці. До головних переваг можна віднести наступні [39]:

- простота та зрозумілість синтаксису (сприяють швидкості розроблення);
- наявність великої кількості бібліотек (спрощує та пришвидшує написання коду);
- велика спільнота (легко отримати підтримку та допомогу з технічними проблемами);
- гнучкість (забезпечує просте поєднання розроблення веб-елементів з засобами машинного та глибокого навчання).

Python дуже часто використовують для розроблення вебзастосунків – саме тому з часом з'явилося багато фреймворків, що виконують роль фундаменту. Одним із таких є Django – його і було обрано застосувати для побудови реалізації описаної раніше вебзастосунку. Даний фреймворк має низку переваг, серед яких наступні [40]:

- наявність великої та чітко-розгалуженої документації;
- наявність реалізацій ефективних механізмів безпеки;
- гнучка масштабованість;
- висока швидкість розроблення.

Для реалізації входу адміністраторів у систему буде використано бібліотеку `django-allauth` [41]. Ця бібліотека створена спеціально для фреймворку Django і надає фундамент для контролю користувачів вебзастосунку. Завдяки

django-allauth відпадає необхідність витратити зайвий час на розробку певних програмних компонентів, окрім цього є можливість змінювати запропоновану реалізацію під себе.

Для спрощення процесу розроблення було використано платформу Docker, яка дає змогу створювати застосунки з використанням контейнерів [42]. Контейнери призначені для можливості запуску програм у тимчасовому ізольованому середовищі, в якому зберігатимуться компоненти необхідні для завантаження. Docker було обрано через низку наявних переваг, зокрема гнучкий розподіл ресурсів, масштабованість, простота обслуговування застосунків, зменшення залежностей від програмного забезпечення [43].

В якості БД прийнято рішення використати PostgreSQL, оскільки вона підтримується фреймворком Django, що спростить розробку вебзастосунку. До переваг даної БД відносяться просунута підтримка SQL (мова запитів), надійність, підтримка масивів як типів даних, можливість роботи з форматом JSON (JavaScript Object Notation) [44].

У підсумку вирішено використати мову програмування Python, фреймворк Django разом з БД PostgreSQL для розробки програмної реалізації методу виявлення фейкових новин з використанням великих мовних моделей у вигляді вебзастосунку.

### 3.1.3 Діаграма модулів застосунку

Вебзастосунок, що реалізовує метод виявлення фейкових новин з використанням великих мовних моделей, розроблявся із врахуванням архітектури вебзастосунку, спроектованої у розділі 2. Для охарактеризування структури програмного рішення було створено діаграму модулів (рисунок 3.1).

Рисунок 3.1 демонструє, що застосунок складається з трьох модулів, які певним чином взаємодіють із БД:

- detection;
- allauth;

– admin\_control.

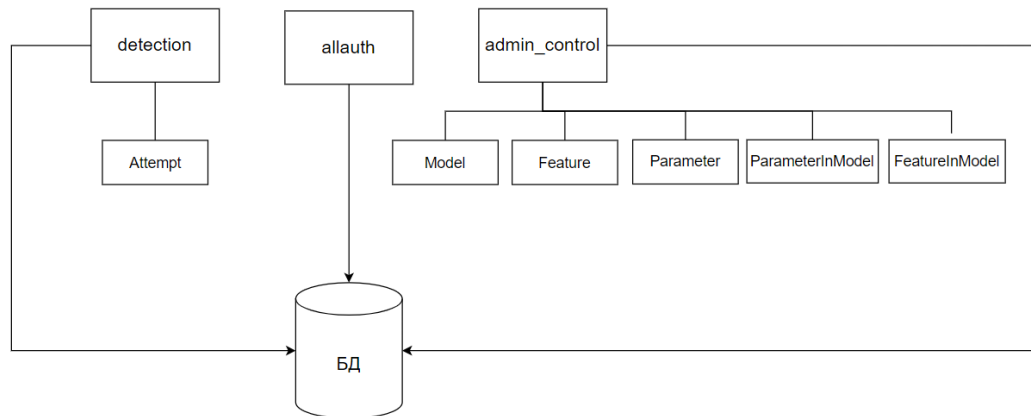


Рисунок 3.1 – Діаграма модулів розробленого вебзастосунку

Модулі detection та admin\_control також мають моделі, які пов’язані із їх функціоналом – це спеціальні класи у Django, які фактично відповідають таблицям БД [45].

Модуль allauth відповідає за вхід адміністраторів у систему та є реалізацією бібліотеки django-allauth.

Модуль detection є частиною вебзастосунку, яка відповідає за реалізацію проведення спроб виявлення фейкових новин. Відповідно, цей модуль є доступним для обох користувацьких груп та містить модель Attempt, яка зберігає інформацію про здійснені спроби.

Модуль admin\_control доступний лише адміністраторам системи та призначений для роботи з моделями для виявлення фейкових новин (не плутати з моделями Django). Відповідно, до даного модулю належать такі інформаційні моделі:

- Model – зберігає класифікатори, які використовуються для виявлення фейкових новин;
- Feature – текстові ознаки;
- Parameter – параметри навчання класифікатору;
- ParameterInModel – зберігає інформацію про те, які параметри пов’язані з якими моделями, а також числові значення перших;

– FeatureInModel – аналогічна до ParameterInModel, лише по відношенню до текстових ознак, а не до параметрів.

Admin\_control забезпечує адміністраторам можливість переглядати, створювати, редагувати видаляти та досліджувати точність роботи моделей для виявлення фейкових новин.

Отже, було описано структуру розробленої системи та її складові на основі відповідної діаграми, пояснено їх функціональне призначення.

### 3.1.4 Особливості реалізації програмних складових застосунку

Розроблений вебзастосунок, що реалізує метод виявлення фейкових новин з використанням великих мовних моделей, має низку функцій та компонентів, особливості яких варто описати.

Для адміністраторів системи важливою є можливість створювати нові моделі для виявлення фейкових новин – це комплексний процес, який складається з низки кроків. Користувач повинен ввести назву моделі, але ця дія може бути виконана на будь-якому етапі; далі адміністратор може обрати певну текстову ознаку з наведених та додати її до загального списку, натиснувши кнопку «Додати ознаку» (рисунок 3.2), що реалізовано завдяки механізму сесій Django, який зберігає дані на серверній стороні [46].

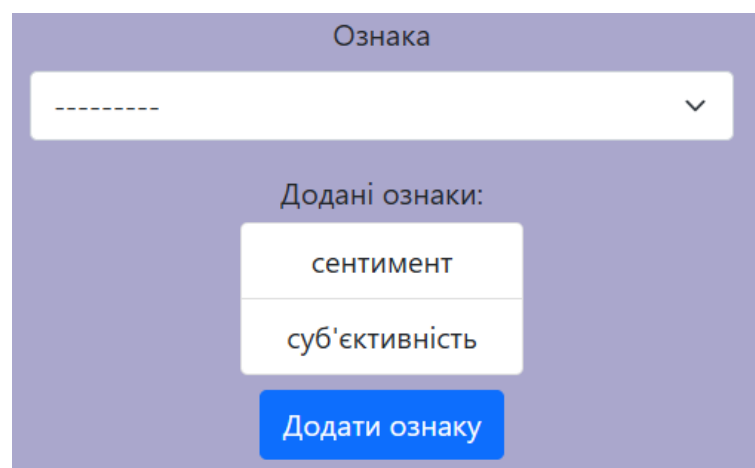
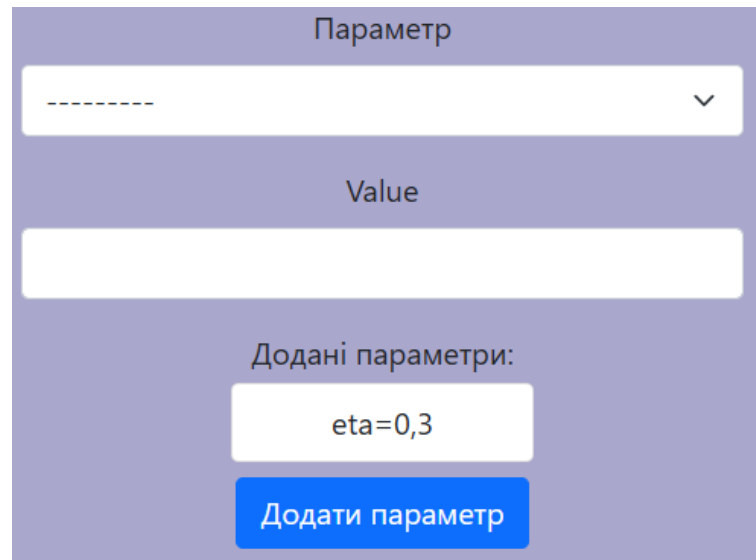


Рисунок 3.2 – Список доданих ознак при створенні моделі класифікації

Щоб додати параметр навчання, адміністратор має обрати один та ввести числове значення в окремому полі, після чого натиснути кнопку «Додати параметр», у результаті чого параметр буде відображено у відповідному переліку (рисунок 3.3).

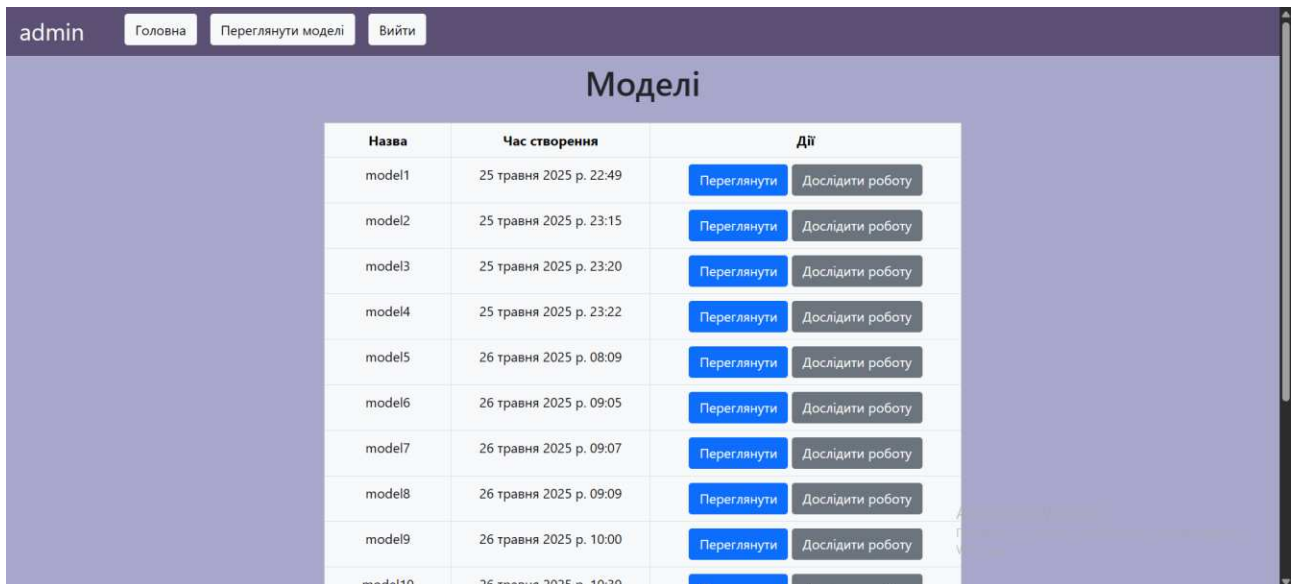


The image shows a web form with a purple background. At the top, the word "Параметр" (Parameter) is centered. Below it is a white dropdown menu with a dashed line and a downward arrow. Underneath is a text input field labeled "Value". Below the input field, the text "Додані параметри:" (Added parameters:) is centered. Below this text is a white box containing the text "eta=0,3". At the bottom of the form is a blue button with the text "Додати параметр" (Add parameter).

Рисунок 3.3 – Список доданих параметрів при створенні моделі класифікації

Отже, можна додавати низку ознак та параметрів по-черзі. Після того, як усі дані зібрано, форма надсилається методом POST на сервер, де витягуються усі значення параметрів та ознак, збережені у сесії, а далі здійснюється підготовка навчальної та тренувальної вибірок новин з обраного набору даних шляхом попередньої обробки та обрахунку числових значень embeddings тексту та обраних адміністратором ознак.

У результаті з утворених вибірок створюються об'єкти такої структури, як DMatrix, яка використовується обраним алгоритмом машинного навчання XGBoost [47]. Модель навчається на цих об'єктах, після чого її дані записуються у файл формату JSON, назвою якого є назва моделі, введена адміністратором. Після того, як усі ці кроки успішно виконано, дані про модель записуються у БД з використанням Django ORM, а користувач перенаправляється на сторінку з таблицею, що перелічує усі моделі класифікації включно із створеною. Ця сторінка зображена на рисунку 3.4.



Назва	Час створення	Дії
model1	25 травня 2025 р. 22:49	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model2	25 травня 2025 р. 23:15	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model3	25 травня 2025 р. 23:20	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model4	25 травня 2025 р. 23:22	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model5	26 травня 2025 р. 08:09	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model6	26 травня 2025 р. 09:05	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model7	26 травня 2025 р. 09:07	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model8	26 травня 2025 р. 09:09	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model9	26 травня 2025 р. 10:00	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>
model10	26 травня 2025 р. 10:39	<a href="#">Переглянути</a> <a href="#">Дослідити роботу</a>

Рисунок 3.4 – Сторінка переліку усіх моделей класифікації

Коли користувач починає виявлення фейкової новини, процеси, що відбуваються, є аналогічними до створення моделі з різницею лише у тому, що вони здійснюються по відношенню лише до одного екземпляру тексту: нормалізація тексту, обрахунок та формування числового вектору, перетворення вектору у DMatrix.

Далі ініціалізується об'єкт класифікатору машинного навчання, після чого функцією `load_model` завантажується обрана модель із збереженого раніше JSON-файлу. Останнім кроком є обчислення результату за допомогою функції `predict` – отримане числове значення виводиться на екран користувачу з висновком, чи є новина фейком (рисунок 3.5).

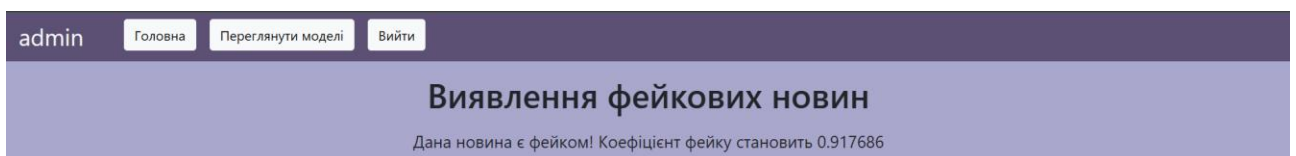


Рисунок 3.5 – Приклад результату виявлення фейкової новини

Отже, описано алгоритми ключових функцій розробленого вебзастосунку: створення моделі та виявлення фейкової новини.

### 3.1.5 Експериментальне тестування вебзастосунок та вимоги до його розгортання

У результаті проведеної роботи було створено вебзастосунок, що реалізує метод для виявлення фейкових новин з використанням великих мовних моделей. Тепер важливо перевірити коректність роботи програми та її компонентів окремо – для цього було проведено низку експериментальних тестувань.

Створення моделі для виявлення фейкових новин є однією з центральних функцій системи, тому її роботу було перевірено з використанням експериментального тесту №1, наведеного на таблиці 3.1.

Таблиця 3.1 – Експериментальний тест №1

<b>Експериментальний тест №1</b>		<b>Пріоритет: 1</b>	<b>Створено: 01.06.2025</b>
<b>Назва:</b> Перевірка функції створення моделі для виявлення фейкових новин			
<b>Передумови:</b> Адміністратор увійшов у систему			
<b>Кроки</b>		<b>Очікуваний результат</b>	
1. Натиснути кнопку «Переглянути моделі» 2. Натиснути кнопку «Створити Модель» 3. У формі, що з'явиться, ввести назву моделі – custom_model 3. Обрати текстову ознаку «сентимент» у відповідному випадаючому списку та натиснути кнопку «Додати ознаку» 4. Аналогічно до кроку 3 додати ознаку «суб'єктивність» 5. Обрати параметр класифікатору машинного навчання «eta» у відповідному переліку, ввести числове значення 0,5 у полі нижче та натиснути кнопку «Додати параметр» 6. Натиснути кнопку «Створити Модель» 7. У списку, що відобразився на екрані, натиснути кнопку «Переглянути» у рядку створеної моделі 8. Звернути увагу на інформацію, відображену на сторінці		Відображено сторінку з інформацією про модель з назвою custom_model, ознаками «сентимент» та «суб'єктивність», параметром «eta» з числовим значенням 0,5	
<b>Результат виконання тестування:</b> пройдено успішно			

Результат проходження наведеного вище експериментального тестування відображено рисунком 3.6.

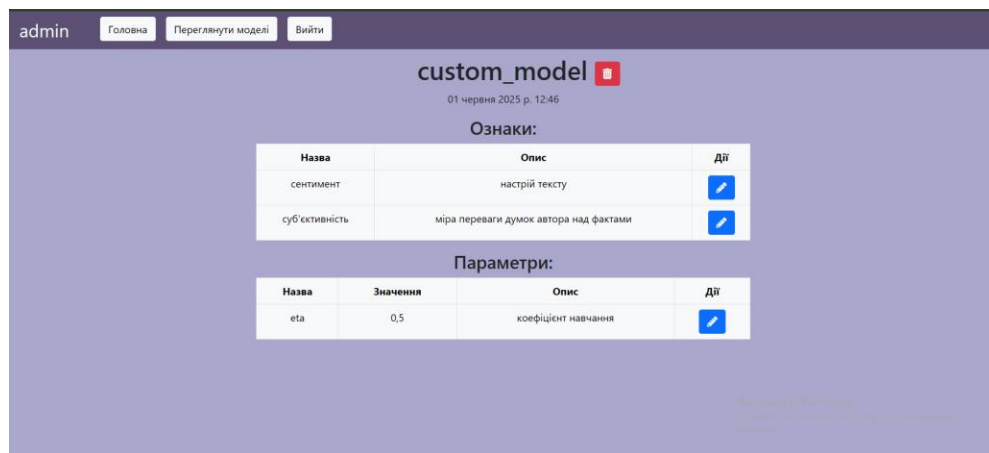


Рисунок 3.6 – Сторінка з інформацією про новостворену модель

Ще однією важливою складовою створеного вебзастосунку є можливість виявлення фейкової новини – її було протестовано завдяки дотриманню вказівок експериментального тесту №2 (таблиця 3.2).

Таблиця 3.2 – Експериментальний тест №2

<b>Експериментальний тест №2</b>	<b>Пріоритет: 1</b>	<b>Створено: 01.06.2025</b>
<b>Назва:</b> Перевірка функції виявлення фейкової новини		
<b>Передумови:</b> Відкрито базову сторінку		
<b>Кроки</b>		<b>Очікуваний результат</b>
<ol style="list-style-type: none"> <li>1. Ввести текст статті (англійською)</li> <li>2. Обрати модель для виявлення фейкової новини серед наявних у випадяючому списку</li> <li>3. Натиснути на кнопку «Виявити фейк»</li> <li>4. Звернути увагу на результат</li> </ol>		Відображення сторінки з висновком про фейковість новини
<b>Результат виконання тестування:</b> пройдено успішно		

Підсумок виконання кроків експериментального тесту №2 демонструє рисунок 3.7.

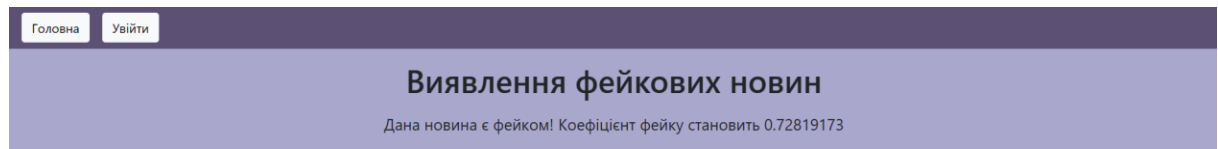


Рисунок 3.7 – Відображення висновку виявлення фейкової новини

Також перевірено функцію дослідження роботи моделі для виявлення фейкових новин із дотриманням кроків, описаних експериментальним тестуванням №3 (таблиця 3.3).

Таблиця 3.3 – Експериментальний тест №3

Експериментальний тест №3	Пріоритет: 1	Створено: 01.06.2025
<b>Назва:</b> Перевірка функції дослідження роботи моделі для виявлення фейкових новин		
<b>Передумови:</b> Адміністратор увійшов у систему		
Кроки		Очікуваний результат
1. Натиснути на кнопку «Переглянути моделі» у верхній панелі 2. У рядку конкретної моделі натиснути на кнопку «Дослідити роботу» 3. Закрити впливаюче повідомлення про необхідність очікування 4. Звернути увагу на сторінку, що з'явиться після певного часу		Відображення таблиці із значенням метрик Accuracy, Precision та log loss, а також графіків по відношенню до обраної моделі
<b>Результат виконання тестування:</b> пройдено успішно		



Рисунок 3.8 – Фрагмент сторінки дослідження роботи моделі для виявлення фейкових новин

У результаті проходження експериментального тестування №3 адміністратору відображається сторінка з інформацією про точність роботи моделі (рисунок 3.8).

Після того, як було проведено тестування складових програмної реалізації сформовано перелік рекомендованих вимог до апаратного та програмного забезпечення:

- встановлений сучасний браузер з підтримкою JavaScript;
- встановлений застосунок Docker Desktop для роботи з платформою Docker;
- процесор Intel Pentium 4 та краще;
- 8 гігабайт оперативної пам'яті.

Отже, перевірено роботу вебзастосунку, який реалізує метод виявлення фейкових новин з використанням великих мовних моделей через використання експериментального тестування та описано вимоги до розгортання системи.

### **3.2 Результати класифікації новин за спроектованим методом**

Для того, щоб дослідити роботу класифікаторів, створених відповідно до процедури формування моделі для виявлення фейкових новин з використанням великих мовних моделей, у вебзастосунку реалізовано окрему функцію для адміністраторів, яка полягає у формуванні аналітичних даних, до яких належать раніше зазначені метрики (Accuracy, Precision, Recall, F1-SCORE, log loss) та графіки (Precision-Recall, ROC-AUC, графік, що демонструє пониження розмірності даних через t-SNE). Кроки для використання зазначеної функції були описані у експериментальному тесті №3, у підрозділі 3.1.5.

Було вирішено у межах функції обмежити використання набору даних до 1000 записів та 1500 при тренуванні моделей, також варто зауважити, що для обрахунку усіх метрик, крім log loss, коефіцієнти фейковості новин, що були встановлені моделями, заокруглено до 1 у випадку, якщо значення сягало 0,5 та

вище. В якості текстових ознак для доповнення embeddings обрано суб'єктивність та сентимент.

У результаті було сформовано низку моделей з різними значеннями параметрів класифікатора та з'ясовано значення метрик Accuracy та log loss для кожної з них (таблиця 3.4).

Таблиця 3.4 – Вплив параметрів класифікатора XGBoost на метрики Accuracy та log loss

Параметри				Значення метрик	
eta	n_estimators	max_depth	lambda	Accuracy	log loss
0,003	700	5	0	0,87	0,38
0,01	400	10	3	0,89	0,31
0,1	200	15	5	0,92	0,21
0,3	150	20	8	0,92	0,21
0,6	100	25	12	0,91	0,22

З таблиці 3.4 можна побачити, що в загальному запропонований метод досягає високих значень вказаних метрик майже за будь-яких значень параметрів, проте можна чітко виділити межу, яка приносить найкращі результати та перехід за яку вже не приносить приросту метрик – це модель з коефіцієнтом навчання 0,1, кількістю дерев рішень 200, максимальною глибиною дерева 15 та значенням lambda 5.

Високе значення Accuracy свідчить про високу точність виявлення фейкових новин моделлю, а показник log loss, який характеризує величину розбіжності прогнозованого класу з фактичним, є найменшим серед решти протестованих конфігурацій.

З огляду на тісний зв'язок між метриками Precision, Recall та F1-Score, їх наведено в окремій таблиці 3.5.

Таблиця 3.5 – Вплив параметрів класифікатора XGBoost на метрики Precision, Recall та F1-SCORE

Параметри				Значення метрик		
eta	n_estimators	max_depth	lambda	Precision	Recall	F1
0,003	700	5	0	0,9	0,86	0,88
0,01	400	10	3	0,9	0,9	0,9
0,1	200	15	5	0,94	0,92	0,93
0,3	150	20	8	0,92	0,93	0,93
0,6	100	25	12	0,9	0,94	0,92

Таблиця 3.5 демонструє схожі результати до таблиці 3.4: та ж сама конфігурація, що отримала найкращі показники Accuracy та log loss, досягла також і найвищого значення F1-SCORE. Хоча можна помітити, що показник Recall має вищі значення з двома іншими наборами параметрів, проте це компенсується найвищим Precision. Отже, конфігурація із значеннями коефіцієнту навчання 0,1, кількості дерев рішень 200, максимальної глибини дерева 15 та коефіцієнту L2-регуляризації ваг 5 забезпечує найкращу точність моделі по розпізнаванню не тільки фейкових, а й правдивих новин.

З проведеного аналізу таблиць можна зробити висновок, що у запропонованого методу існує тенденція, яка полягає у отриманні кращих результатів через збільшення коефіцієнту навчання та зменшення кількості дерев рішень разом з усередненням решти метрик – проте коефіцієнт навчання повинен зберігати відносно низьке значення, оскільки згідно таблиці його збільшення, починаючи з 0,1, не приносить покращення результатів, а занадто велике значення може призвести і до погіршення точності навіть попри малу кількість дерев, що у свою чергу може свідчити про перенавчання.

Також було вирішено дослідити вплив текстових ознак на класифікацію: для цього було створено та протестовано окремі моделі на основі найкращих параметрів, одна з яких не використовує жодних текстових ознак, а решта

використовує лише суб'єктивність та лише сентимент відповідно. Результати було внесено у таблицю 3.6.

Таблиця 3.6 – Вплив текстових ознак на усі метрики моделі з найкращими параметрами

Використані текстові ознаки:	Значення метрик:				
	Accuracy	log loss	Precision	Recall	F1
Відсутні	0,9	0,24	0,88	0,94	0,91
Сентимент	0,91	0,24	0,93	0,9	0,91
Суб'єктивність	0,89	0,26	0,86	0,95	0,9

З таблиці 3.6 можна побачити, що як відсутність ознак, так і використання лише однієї з них впливає у погіршення ключових метрик Accuracy, F1-SCORE та log loss, з чого можна зробити висновок, що текстові ознаки є важливим компонентом у роботі моделі.

Графік Precision-Recall для оптимальної моделі наведено на рисунку 3.9.

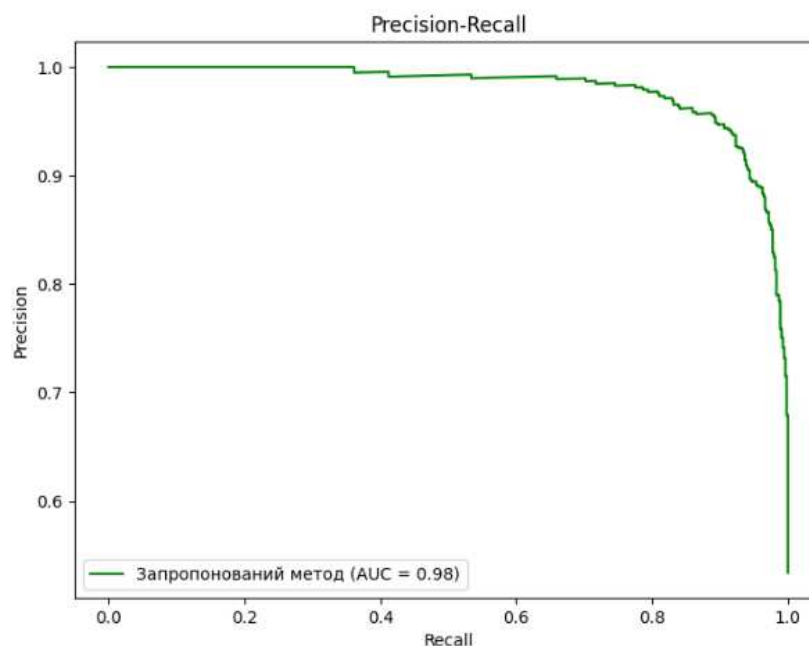


Рисунок 3.9 – Графік Precision-Recall для найкращої конфігурації параметрів класифікатора

На графіку рисунка 3.9 можемо бачити, що AUC, тобто площа під кривою, становить 0,98 – цей результат є наближеним до 1, що фактично є ідеальним значенням. Можемо також переконатися, що показники точності та повноти переважно залишаються наближеними до ідеальних значень, що свідчить про потенційно високий рівень точності роботи запропонованої моделі за широкого діапазону порогів (під порогом мається на увазі межу, яка розділяє правдиві новини від фейкових).

Також для згаданої конфігурації було сформовано графік ROC-AUC (рисунок 3.10).

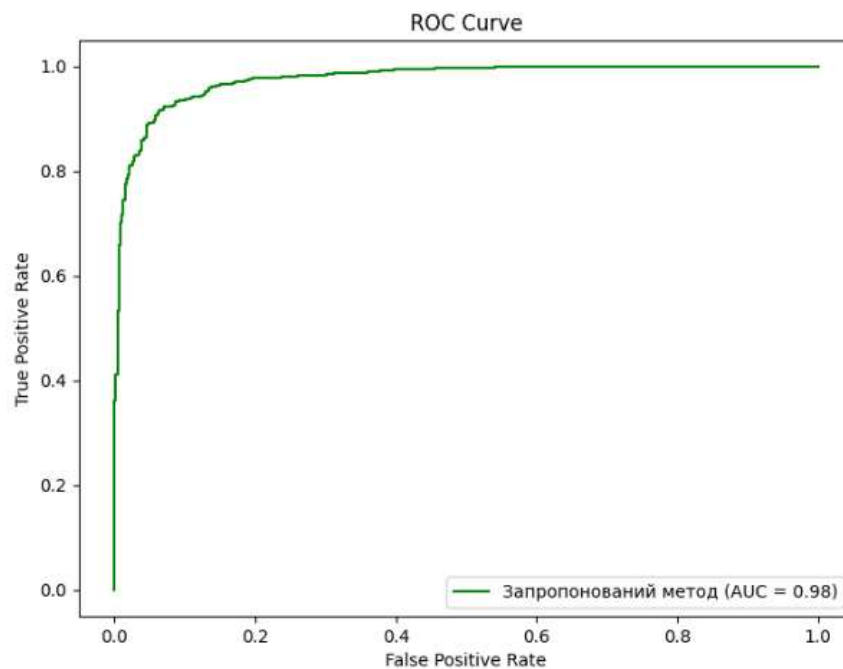


Рисунок 3.10 – Графік ROC-AUC для найкращої конфігурації параметрів класифікатора

Графік ROC-AUC також демонструє високу якість моделі, оскільки за низьких значень False Positive Rate зберігаються високі значення True Positive Rate – це означає, що модель виявляє велику кількість фейкових новин, допускаючи мінімум помилкових класифікацій правдивих новин як фейкових. До того ж, AUC є наближеною до ідеального значення та становить 0,98.

Для дослідження якості роздільності класів відповідно до згенерованих embeddings та обчислених ознак було сформовано графік пониження розмірності через t-SNE (рисунок 3.11).

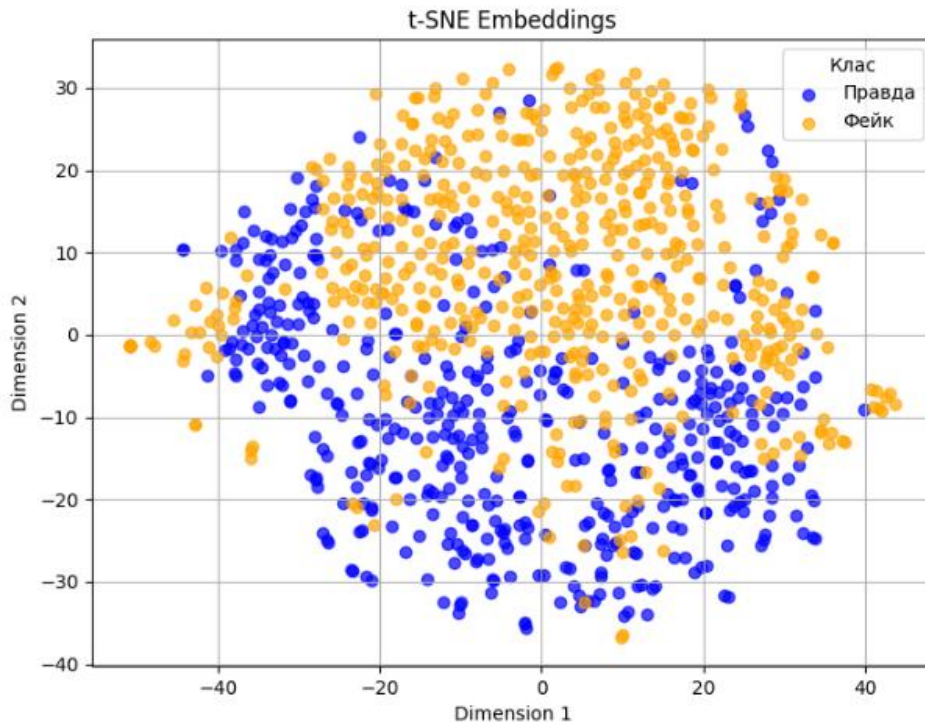


Рисунок 3.11 – Графік пониження розмірності embeddings та текстових ознак моделі з найкращою конфігурацією параметрів через t-SNE

З графіку вище можна побачити, що модель є досить точною щодо розділення новин на правдиві та фейкові за обрахованими текстовими ознаками та згенерованими LLM embeddings, оскільки є можливим проведення діагоналі, яка розділить кластери даних таким чином, що більшість фейкових новин перебуватимуть над лінією а більшість правдивих новин – під нею.

Отже, було проведено дослідження запропонованого методу для виявлення фейкових новин з використанням великих мовних моделей та його точності, протестовано різні конфігурації в наслідок чого отримано кращий результат з показниками метрик Accuracy – 0,92, log loss – 0,21, Precision – 0,94, Recall – 0,92, та F1-SCORE – 0,93. Також було проаналізовано графіки Precision-Recall, ROC-AUC та t-SNE для найкращої моделі виявлення, в результаті чого підтверджено високий рівень точності програмної реалізації поданого методу.

### 3.3 Висновки до розділу 3

У третьому розділі наведено детальний опис розроблення вебзастосунку, що реалізує поданий в роботі метод виявлення фейкових новин.

Спершу обрано способи дослідження результативності використання методу, до яких входить аналіз вказаних метрик та графіків. Після цього здійснено вибір засобів, які використовувались для розроблення програмної реалізації вебзастосунку: мова програмування Python, фреймворк Django, платформа Docker, БД PostgreSQL. Також було візуалізовано структуру застосунку, в ході чого описано принципи роботи його окремих компонентів; пояснено особливості реалізації деяких складових.

Розроблений вебзастосунок було провалідовано внаслідок виконання експериментальних тестувань: отримані результати продемонстрували адекватну роботу програмного забезпечення та його очікувану поведінку за низки стандартних ситуацій.

Насамкінець, було досліджено результативність роботи методу виявлення фейкових новин з використанням великих мовних моделей через використання розробленої спеціально для цього програмної складової; застосовано метрики Accuracy, Precision, Recall, F1-Score, log loss, та графіки Precision-Recall, ROC-AUC, t-SNE. У ході дослідження з'ясовано вплив параметрів та текстових ознак на роботу моделей, знайдено найкращу конфігурації з-поміж досліджених. Саме дослідження показало високу якість та точність роботи запронованого методу у межах розв'язуваної задачі.

## Загальні висновки

Результатом виконання кваліфікаційної роботи бакалавра стало успішне виконання поставленої мети: підвищено точність виявлення фейкових новин з використанням великих мовних моделей. Для цього було спроектовано метод виявлення фейкових новин з використанням великих мовних моделей та процедуру формування моделі у межах методу. Після проектування було розроблено програмну реалізацію методу у вигляді вебзастосунку.

Загалом, мету роботи було досягнуто завдяки виконанню наступних поставлених завдань:

- спроектовано метод виявлення фейкових англomовних новин з використанням великих мовних моделей;
- розроблено архітектуру застосунку для виявлення фейкових англomовних новин;
- проаналізовано актуальні великі мовні моделі та обрано Gemini text-embedding-004 для розв'язання завдань роботи;
- виконано програмну реалізацію у вигляді вебзастосунку;
- проведено експериментальне тестування реалізованого вебзастосунку за еталонними наборами даних;
- здійснити дослідження ефективності методу за допомогою створеного застосунку.

У результаті проведеного аналізу низки великих мовних моделей та засобів машинного навчання прийнято рішення використати LLM Gemini text-embedding-004 разом з класифікатором XGBoost для реалізації методу.

Для розроблення вебзастосунку використано мову програмування Python, фреймворк Django, платформу Docker та БД PostgreSQL. Програмна реалізація ділить користувачів на групи гостя та адміністратора, та надає їм можливість виконувати низку функцій: усі користувачі можуть проводити виявлення фейкової новини, використовуючи одну з наявних моделей, створених за спроектованим методом, у той час як адміністратори отримують повний

контроль над цими моделями, включаючи їх перегляд, створення із зазначенням використовуваних параметрів та текстових ознак, редагування із перетворенням, видалення та дослідження результативності роботи. Розроблений застосунок було протестовано із використанням експериментального тестування, внаслідок чого зроблено висновок про повну коректність роботи та високий рівень класифікації новин.

Також проведено дослідження методу для виявлення фейкових новин з використанням великих мовних моделей, застосовуючи функціонал розробленого вебзастосунку: завдяки метрикам Accuracy, Precision та log loss визначено вплив параметрів навчання та текстових ознак на точність роботи, окрім цього з низки протестованих конфігурацій знайдено найкращу та проаналізовано її графіки Precision-Recall, ROC-AUC та t-SNE. Внаслідок проведених тестувань підтверджено високий рівень точності методу з кращими показниками Accuracy, Precision, Recall, F1-Score, log loss, що становлять 0,92, 0,94, 0,92, 0,93 та 0,21, тобто точність виявлення фейкових новин при еталонному наборі даних перевищує 90%.

Отже, розроблений вебзастосунок повністю реалізує поставлену мету, працює справно та може використовуватись для виявлення фейкових англійських новин.

## Перелік посилань

1. Ms.detector.media. Лише 11 % українців змогли відрізнити фейки — дослідження. URL: [https://ms.detector.media/mediadoslidzhennya/post/23680/2019-10-23-lyshe-11-ukraintsiv-zmogly-vidriznyty-feyky-doslidzhennya/?utm\\_source=chatgpt.com](https://ms.detector.media/mediadoslidzhennya/post/23680/2019-10-23-lyshe-11-ukraintsiv-zmogly-vidriznyty-feyky-doslidzhennya/?utm_source=chatgpt.com)
2. Yellow.ai. Large language models: A guide on its benefits, use cases, and types. URL: <https://yellow.ai/blog/large-language-models/>
3. ESafetyComissioner. Fake news. URL: <https://www.esafety.gov.au/young-people/fake-news>
4. Barmak O., Radiuk P. Web-based information technology for classifying and interpreting early pneumonia based on fine-tuned convolutional neural network. *Computer systems and information technologies*. 2021. Vol. 3, no. 1. P. 12–18. URL: <https://doi.org/10.31891/CSIT-2021-3-2>
5. Radiuk P., Pavlova O., Hrypynska N. An ensemble machine learning approach for Twitter sentiment analysis. *Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2022). Volume I: Main Conference : CEUR-Workshop Proceedings, Gliwice, Poland, 12–13 May 2022 / ed. by V. Lytvyn et al. Aachen, 2022. P. 387–397. URL: https://ceur-ws.org/Vol-3171/paper32.pdf.*
6. Інститут масової інформації. Соціальні мережі, які диктують ритм новин. Дослідження ІМІ. URL: <https://imi.org.ua/monitorings/sotsialni-merezhi-yaki-dyktuyut-rytm-novyn-doslidzhennya-imi-i65389>
7. Google Cloud. What is Machine Learning? Types & Uses | Google Cloud. URL: <https://cloud.google.com/learn/what-is-machine-learning>
8. IBM. What Is Support Vector Machine? | IBM. URL: <https://www.ibm.com/think/topics/support-vector-machine>
9. Information system for public places and institutions visualization with opportunities of inclusive access and optimal routing / O. Pavlova et al. *Computer*

*systems and information technologies*. 2022. Vol. 1, no. 6. P. 62–68. URL: <https://doi.org/10.31891/CSIT-2022-1-8>

10. Profolus. Natural Language Processing: Advantages and Disadvantages - Profolus. URL: <https://www.profolus.com/topics/natural-language-processing-advantages-and-disadvantages/>

11. Investopedia. What Is a Neural Network? URL: <https://www.investopedia.com/terms/n/neuralnetwork.asp>

12. Built In. What Are Recurrent Neural Networks (RNNs)? | Built In. URL: <https://builtin.com/data-science/recurrent-neural-networks-and-lstm>

13. Datacamp. How Transformers Work: A Detailed Exploration of Transformer Architecture. URL: <https://www.datacamp.com/tutorial/how-transformers-work>

14. Melnychenko O., Savenko O., Radiuk P. Apple detection with occlusions using modified YOLOv5-v1. *Proceedings of the 12th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)* : Proceedings, Dortmund, Germany, 7–9 September 2023. New York, NY, USA, 2023. P. 107–112. URL: <https://doi.org/10.1109/idaacs58523.2023.10348779>

15. Explainable deep learning: A visual analytics approach with transition matrices / P. Radiuk et al. *Mathematics*. 2024. Vol. 12, no. 7. P. 1024. URL: <https://doi.org/10.3390/math12071024>

16. Datacamp. What is BERT? An Intro to BERT Models. URL: <https://www.datacamp.com/blog/what-is-bert-an-intro-to-bert-models>

17. Neurond AI. What Is BERT Language Model? Its Advantages And Applications. URL: <https://www.neurond.com/blog/what-is-bert>

18. Coursera. BERT vs. GPT: What's the Difference? URL: <https://www.coursera.org/articles/bert-vs-gpt>

19. Shupta A., Radiuk P., Krak I. Feature computation procedure for fake news detection: An LLM-based extraction approach. *Proceedings of the 6th International Workshop on Intelligent Information Technologies & Systems of Information Security*

(*IntelITSIS 2025*) : CEUR-Workshop Proceedings, Khmelnytskyi, Ukraine, 4 April 2025 / ed. by T. Hovorushchenko et al. Aachen, 2025. P. 112–124. URL: <https://ceur-ws.org/Vol-3963/paper10.pdf>

20. Toolbox.google. Fact Check Tools. URL: <https://toolbox.google.com/factcheck/explorer/search/list:recent;hl=uk>

21. Fakenews.research. URL: <http://fakenews.research.sfu.ca/>

22. Oigetit Fake News Filter. URL: <https://oigetit.com/>

23. GitHub. GitHub – huggingface/transformers. URL: <https://github.com/huggingface/transformers>

24. Radiuk P.M. Application of a genetic algorithm to search for the optimal convolutional neural network architecture with weight distribution. *Herald of Khmelnytskyi National University. Technical sciences*. 2020. Vol. 281, no. 1. P. 7–11. URL: <https://doi.org/10.31891/2307-5732-2020-281-1-7-11>

25. LEE, Jinhyuk, et al. Gemini embedding: Generalizable embeddings from gemini. arXiv preprint arXiv:2503.07891, 2025.

26. Wikipedia. Gemini (language model). URL: [https://en.wikipedia.org/wiki/Gemini\\_\(language\\_model\)](https://en.wikipedia.org/wiki/Gemini_(language_model))

27. Medium. From Multimodal Marvels to Mixing of Experts – Google’s Gemini Evolution. URL: <https://medium.com/happtiq-data-ai-hub/from-multimodal-marvels-to-mixing-of-experts-googles-gemini-evolution-e42622df65bf>

28. GeeksforGeeks. XGBoost – GeeksforGeeks. URL: <https://www.geeksforgeeks.org/xgboost/>

29. Kaggle. Fake News Classification. URL: <https://www.kaggle.com/datasets/saurabhshahane/fake-news-classification/>

30. AppMaster – The No-Code platform for building web & mobile apps. Junction Table. URL: <https://appmaster.io/glossary/junction-table>

31. PyPI. XGBoost. URL: <https://pypi.org/project/xgboost/>

32. Pandas - Python Data Analysis Library. URL: <https://pandas.pydata.org/>

33. Scikit-learn: machine learning in Python – scikit-learn 1.6.1 documentation. URL: <https://scikit-learn.org/stable/>

34. PyPI. textblob. URL: <https://pypi.org/project/textblob/>
35. V7 |. AI Document Processing & Data Labelling. Top Performance Metrics in Machine Learning: A Comprehensive Guide. URL: <https://www.v7labs.com/blog/performance-metrics-in-machine-learning>
36. Medium. Understanding Log Loss: A Comprehensive Guide with Code Examples. URL: <https://koshurai.medium.com/understanding-log-loss-a-comprehensive-guide-with-code-examples-c79cf5411426>
37. Machinelearningmastery. How to Use ROC Curves and Precision-Recall Curves for Classification in Python. URL: <https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/>
38. Datacamp. Introduction to t-SNE. URL: <https://www.datacamp.com/tutorial/introduction-t-sne>
39. ScaleupAlly. 18 Pros and Cons of using Python for Web Development. URL: <https://scaleupally.io/blog/pros-and-cons-of-python-for-web-development/>
40. CareerFoundry. 9 Pros and Cons of the Django Framework: A Coder's Guide. URL: <https://careerfoundry.com/en/blog/web-development/django-framework-guide/#advantages-and-disadvantages-of-django>
41. Django-allauth. URL: <https://docs.allauth.org/en/latest/>
42. Palo Alto Networks. What Is Docker?. URL: <https://www.paloaltonetworks.com/cyberpedia/docker>
43. Simplilearn.com. What Is and What Are the Benefits of Docker Container?. URL: <https://www.simplilearn.com/tutorials/docker-tutorial/what-is-docker-container>
44. Tessell. PostgreSQL - concepts, benefits and use cases | Tessell. URL: <https://www.tessell.com/blogs/postgresql-concepts-benefits-and-use-cases>
45. W3Schools.com. URL: [https://www.w3schools.com/django/django\\_models.php](https://www.w3schools.com/django/django_models.php)
46. Django Project. How to use sessions | Django documentation. URL: <https://docs.djangoproject.com/en/5.2/topics/http/sessions/>
47. Non-Brand Data |. Cornellius Yudha Wijaya |. Substack. Use XGBoost Like a Pro. URL: <https://www.nb-data.com/p/use-xgboost-like-a-pro>

# ДОДАТКИ

## Додаток А

### Програмний код

Програмний код створеного вебзастосунку доступний у репозиторії GitHub: <https://github.com/TimurMustaiev/Fake-News-Detection-App> (дата звернення: 02.06.2025).

На рисунку А.1 подано знімок екрана репозиторію на GitHub.

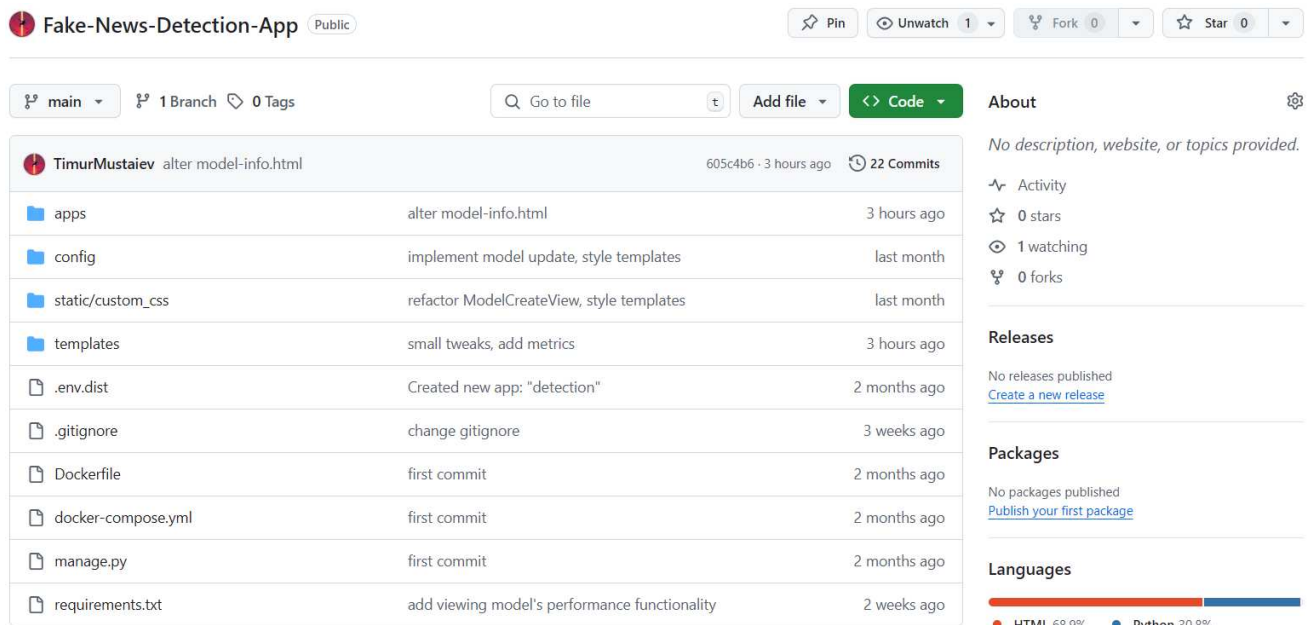


Рисунок А.1 – Світлина з екрана головної сторінки на репозиторії GitHub

Репозиторій містить такі основні компоненти:

- тека `apps` містить модулі `admin_control`, який включає весь адміністраторський функціонал керування моделями для виявлення фейкових новин та дослідження їх роботи;
- тека `config` базові файли, необхідні для запуску та загального налаштування застосунку;
- тека `static` призначена для зберігання статичних файлів та містить теку `custom_css`, в якій зберігаються створені файли формату `css`;
- тека `templates` містить `html`-файли модулю `allauth`.

## Додаток Б

### Презентаційний матеріал

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

## МЕТОД ВИЯВЛЕННЯ ФЕЙКОВИХ НОВИН З ВИКОРИСТАННЯМ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ

---



**Виконав:**  
студент 4 курсу, група КН-21-1  
Тімур **МУСТАЄВ**

**Керівник:**  
*Доктор філософії, старший викладач кафедри  
комп'ютерних наук*  
Павло **РАДЮК**



2

## Актуальність

У сучасному світі новини є головним джерелом інформації для мільйонів людей. Велика частина людей мають постійний доступ до будь-яких новин через соціальні мережі, месенджери та новинні сайти, що у свою чергу спричиняє високий рівень споживання контенту. Перенасичення інформаційного простору призводить до погіршення здатності розрізняти фейкову інформацію серед правдивої, тому проблема виявлення фейкових новин була і залишається актуальною.

Створення програмного застосунку для виявлення фейкових новин дозволить автоматизувати вирішення задачі та запобігти негативним наслідкам впливу фейків на суспільство та його думку. Використання великих мовних моделей для реалізації є перспективним через їх ефективність та стрімкий розвиток.

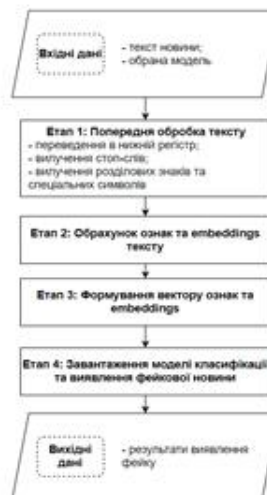
## Постановка задачі

Метою кваліфікаційної роботи бакалавра є підвищення точності виявлення фейкових новин з використанням великих мовних моделей.

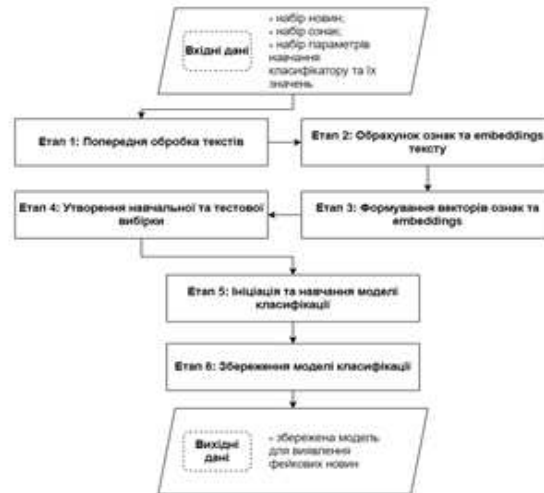
Для досягнення поставленої мети необхідно виконати наступні завдання:

- спроектувати метод виявлення фейкових англomовних новин з використанням великих мовних моделей;
- розробити архітектуру застосунку для виявлення фейкових англomовних новин;
- проаналізувати актуальні великі мовні моделі та обрати найбільш оптимальну з них;
- виконати програмну реалізацію у вигляді вебзастосунку;
- провести експериментальне тестування реалізованого вебзастосунку за еталонними наборами даних;
- здійснити дослідження ефективності методу за допомогою створеного застосунку.

## Схема методу виявлення фейкових новин з використанням великих мовних моделей



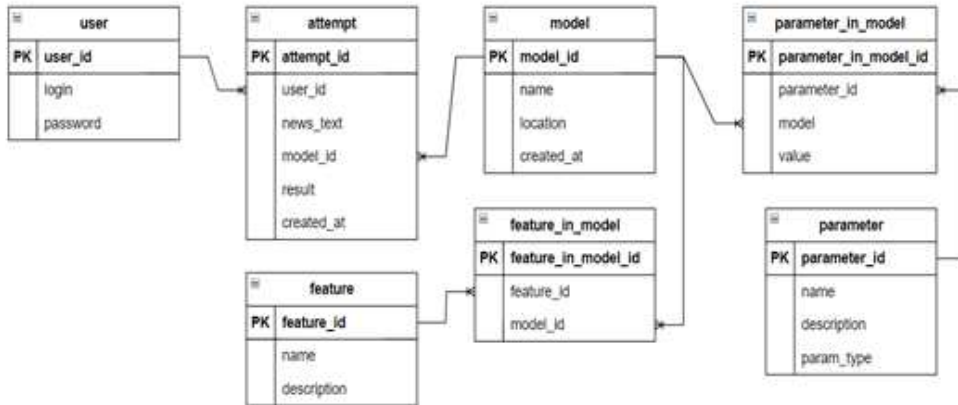
## Схема процедури формування моделі для виявлення фейкових новин з використанням великих мовних моделей



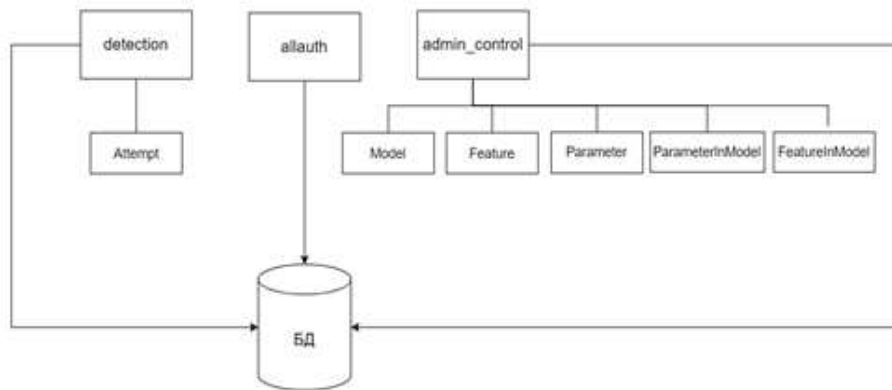
## Схема архітектури інформаційної системи виявлення фейкових новин із застосуванням великих мовних моделей

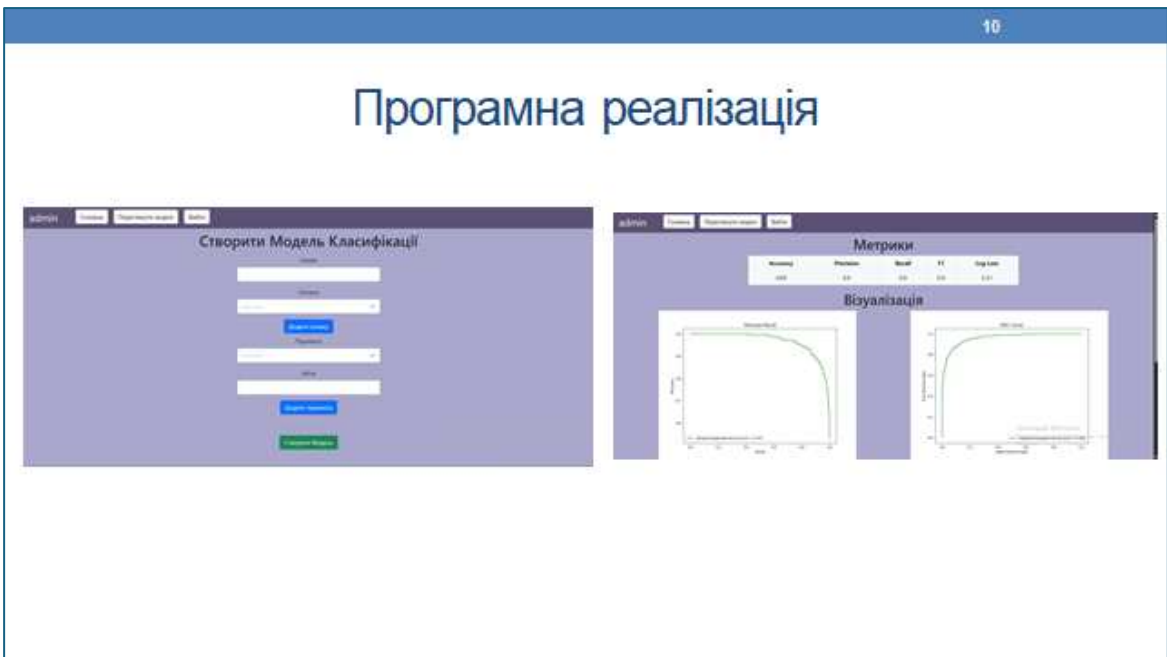
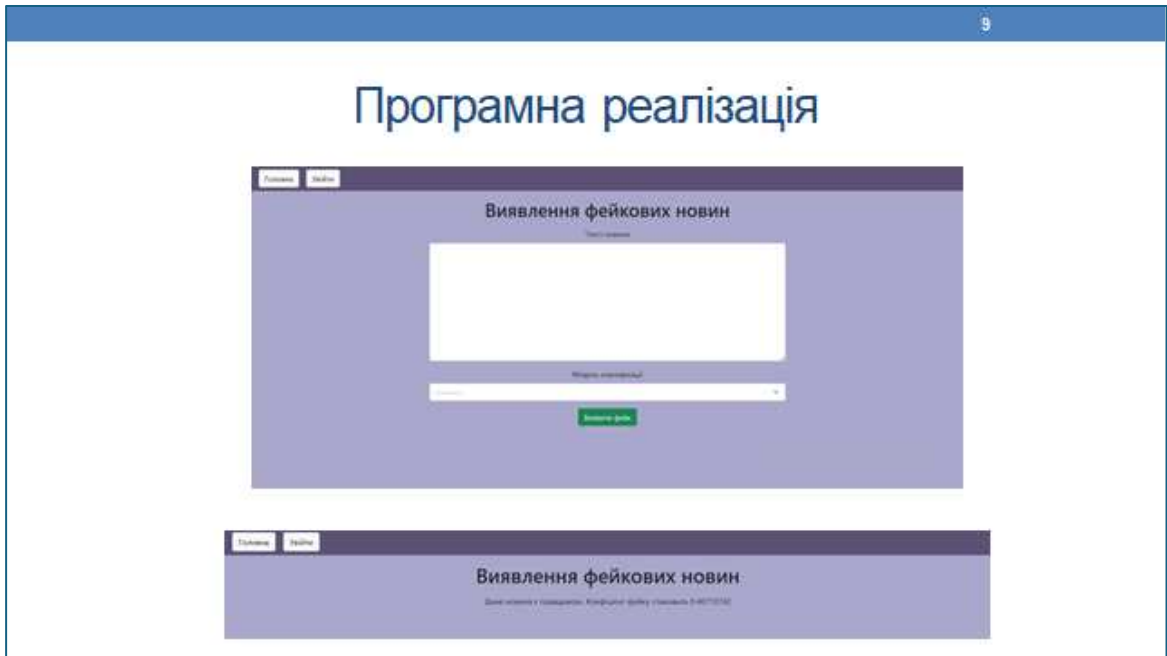


## Даталогічна модель бази даних



## Взаємодія модулів вебзастосунку





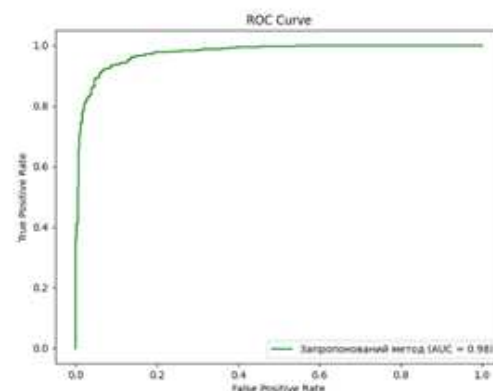
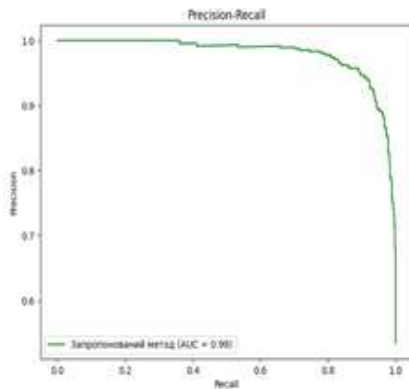
## Дослідження ефективності методу виявлення фейкових новин з використанням великих мовних моделей

Параметри:				Значення метрик:	
eta	n_estimators	max_depth	lambda	accuracy	log loss
0.003	700	5	0	0.87	0.38
0.01	400	10	3	0.89	0.31
0.1	200	15	5	0.92	0.21
0.3	150	20	8	0.92	0.21
0.6	100	25	12	0.91	0.22

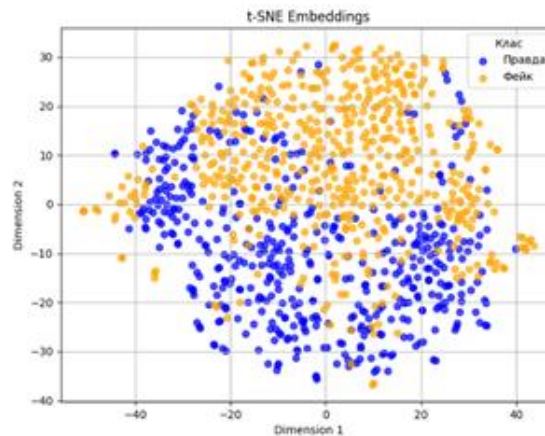
Параметри:				Значення метрик:		
eta	n_estimators	max_depth	lambda	precision	recall	F1
0.003	700	5	0	0.9	0.86	0.88
0.01	400	10	3	0.9	0.9	0.9
0.1	200	15	5	0.94	0.92	0.93
0.3	150	20	8	0.92	0.93	0.93
0.6	100	25	12	0.9	0.94	0.92

Використані текстові ознаки:	Значення метрик:				
	accuracy	log loss	precision	recall	F1
Відсутні	0.9	0.24	0.88	0.94	0.91
Сентимент	0.91	0.24	0.93	0.9	0.91
Суб'єктивність	0.89	0.26	0.86	0.95	0.9

## Дослідження ефективності методу виявлення фейкових новин з використанням великих мовних моделей



## Дослідження ефективності методу виявлення фейкових новин з використанням великих мовних моделей



## Висновки

Досягнуто мету кваліфікаційної роботи – підвищено точність виявлення фейкових новин з використанням великих мовних моделей. Для цього було спроектовано відповідний метод та алгоритм формування використовуваної моделі, які було втілено у розробленому вебзастосунку.

Вебзастосунок було створено з використанням таких засобів розробки, як мова програмування Python, фреймворк Django, база даних PostgreSQL, платформа Docker, класифікатор XGBoost та велика мовна модель Gemini text-embedding-004. Програмна реалізація дозволяє автоматизувати процес виявлення фейкових новин, надає можливість керувати моделями класифікації та формувати дані про їх роботу..

Було проведено дослідження ефективності програмної реалізації методу, в ході якого з'ясовано тенденцію впливу параметрів навчання та текстових ознак на якість класифікації моделі, окрім цього було знайдено найкращу конфігурацію для запропонованої моделі серед протестованих. Результати дослідження довели високий рівень ефективності роботи методу.

# Anti-Plagiarism (UA) v-15.281 Educational

**The maximum coincidence with one document 3.0%**

Dictionaries check: en\_US, ru\_RU, ua\_UA. **Errors in the documents: 13%**

ID: 244299 Title: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод виявлення фейкових новин з використанням великих мовних моделей Added in a DB: 2025-06-09 Authors: Тімур МУСТАЄВ Heads: Павло РАДЮК Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	65372	980	4924 (8%)	81 (8%)

## Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Тімур МУСТАЄВ

Співавтор:

Назва: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод виявлення фейкових новин з використанням великих мовних моделей

Науковий керівник: Павло РАДЮК, старший викладач кафедри, Ph.D.

Підрозділ: Кафедра комп'ютерних наук

Коефіцієнт подібності 1:5.9%

Коефіцієнт подібності 2:2.2%

Мікропробіли: 0

Заміна букв: 0

Інтервали: 0

Білі знаки: 3

Дата створення звіту: 2025-06-09 20:01:15.0

Після аналізу Звіту подібності констатую наступне:

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедур. Таким чином робота не приймається.

Обґрунтування:

2025-06-10

Дата

експерт

*Ст. Перевозчик Р.Р.*

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНИХ НАУК

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи Метод виявлення фейкових новин з використанням великих мовних моделей

Автор студент групи КН-21-1 Мустаєв Тімур

Освітня програма Комп'ютерні науки

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 – Комп'ютерні науки

Науковий керівник: док. філ., ст. викладач кафедри комп'ютерних наук Павло Радюк

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмними засобами комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	<i>відповідає</i>
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	<i>відсутні</i>

Підтвердження:

Запозичення, виявлені в роботі Тімура Мустаєва не є плагіатом, оскільки: запозичення розміщені в розділі огляду існуючих підходів, не описують безпосередньо авторську роботу і не стосуються її результатів; усі запозичення фрагментарні; до запозичень входять фрагменти, які не мають авторства і містять поширені конструкції та загальновідомі терміни, скорочення. Рівень подібності не перевищує допустимої межі. Таким чином, робота є законною та приймається до захисту.

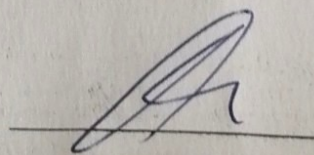
Обсяг запозичень, визначений системами виявлення збігів/ідентичності/схожості:

- за системою Anti-Plagiarism: 3%;

- за системою StrikePlagiarism КП1: 5.9%, КП2: 2.2%.

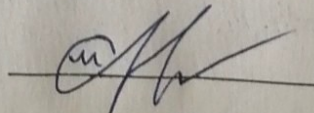
09.06.2025

Завідувач кафедри



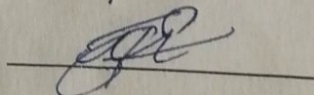
Олександр БАРМАК

Гарант освітньої програми



Олександр МАЗУРЕЦЬ

Керівник кваліфікаційної роботи



Павло РАДЮК



**ВІДГУК НАУКОВОГО КЕРІВНИКА  
на кваліфікаційну роботу бакалавра**

студента КН-20-1 Мустаєва Тімура Вікторовича

за темою Метод виявлення фейкових новин з використанням великих мовних моделей

**1. Актуальність теми**

У сучасному суспільстві проблема поширення фейкових новин є надзвичайно актуальною. Швидкість розповсюдження та обсяги неправдивої інформації в інтернеті мають здатність спричиняти серйозні негативні наслідки, як от маніпуляція громадською думкою, приховування фактів тощо. Тому виникає потреба у вдосконаленні наявних та проєктуванню нових методів виявлення фейкових новин. Створення відповідного методу з використанням великих мовних моделей сприятиме забезпеченню інформаційної безпеки та покращенню якості медіа-контенту.

**2. Відповідність роботи предметній області Стандарту спеціальності 122 Комп'ютерні науки**

За стандартом бакалавра вищої освіти України спеціальності 122 Комп'ютерні науки, а саме описом предметної області, об'єктом та предметом вивчення є процес виявлення фейкових новин з використанням великих мовних моделей. Метою роботи є підвищення точності виявлення фейкових новин з використанням великих мовних моделей. У процесі розв'язання поставленої задачі використано засоби машинного навчання для роботи з текстом. Отже, результати виконання кваліфікаційної роботи відповідають стандарту бакалавра спеціальності 122 Комп'ютерні науки.

**3. Професійні та особистісні якості бакалавра**

Під час виконання кваліфікаційної роботи студент Мустаєв Тімур Вікторович проявив себе кваліфікованим фахівцем, самостійним та дисциплінованим студентом. У процесі проєктування методу виявлення фейкових новин та розроблення на його основі вебзастосунку, а також під час написання записки до кваліфікаційної роботи здобувач продемонстрував необхідні компетенції галузі інформаційних технологій, що сприяло успішному досягненню мети роботи та отриманню позитивних результатів.

**4. Ступінь самостійності під час виконання кваліфікаційної роботи**

Отримані результати свідчать про самостійний характер виконання роботи. Усі етапи виконання кваліфікаційної роботи, від дослідження теоретичної частини до

реалізації вебзастосування, були виконані студентом особисто, що підтверджується якістю виконання завдань та обґрунтованістю отриманих висновків.

#### **5. Ступінь оволодіння методами дослідження**

У процесі виконання кваліфікаційної роботи студент Мустаєв Тімур Вікторович продемонстрував високий рівень компетентностей та володіння необхідними технологіями, засобами та методами спеціальності 122 Комп'ютерні науки.

#### **6. Повнота та якість розкриття теми роботи**

Тема роботи повністю обґрунтована й розкрита, здійснено аналіз актуальності предметної області, теоретичних підходів та сучасних програмних рішень у рамках тематики. Усі поставлені завдання було успішно виконано, зокрема, здійснено програмну реалізацію методу виявлення фейкових новин з використанням великих мовних моделей у вигляді вебзастосування, проведено його експериментальне дослідження та підтверджено підвищення точності класифікації фейкових новин за створеними методом та вебзастосуванням на його основі.

#### **7. Логічність, послідовність, аргументованість, літературна грамотність викладення матеріалу**

Кваліфікаційна робота бакалавра є чітко структурованою, матеріал викладено послідовно та логічно, з повним аргументуванням. Виклад відповідає поставленій меті та завданням. Стиль і мова роботи відповідають вимогам академічного письма, що забезпечує чітке сприйняття вмісту.

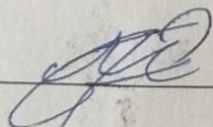
#### **8. Можливість практичного застосування кваліфікаційної роботи бакалавра, окремих її частин**

Програмна реалізація методу виявлення фейкових новин з використанням великих мовних моделей у вигляді вебзастосування може бути використана як інструмент протидії дезінформації журналістами-розслідувачами або фахівцями правоохоронних органів.

#### **9. Висновок про можливість допуску кваліфікаційної роботи бакалавра до захисту, на яку оцінку заслуговує робота**

З огляду на високий рівень виконання та забезпечення всіх необхідних вимог, вважаю, що кваліфікаційна робота бакалавра Мустаєва Тімура Вікторовича може бути допущена до захисту. Рекомендована оцінка – «відмінно».

Керівник



док. філ., ст. викл. каф. КН Павло РАДЮК



## РЕЦЕНЗІЯ

### на кваліфікаційну роботу бакалавра

студента КН-21-1 Мустаєва Тімура Вікторовича  
за темою: Метод виявлення фейкових новин з використанням великих мовних моделей

#### 1. Актуальність обраної теми

Проблема дезінформації набуває дедалі більших масштабів з розвитком інформаційних технологій, а здатність людей розрізняти неправдиву інформацію погіршується. Тому завдання виявлення фейкових новин є актуальним та важливим. Використання такої технології, як великі мовні моделі дає можливість створити перспективне рішення з високим рівнем точності виявлення фейкових новин.

#### 2. Повнота розкриття мети та завдань роботи

Автором кваліфікаційної роботи бакалавра було повністю розкрито мету та завдання. У роботі проаналізовано предметну область, внаслідок чого спроектовано та реалізовано метод виявлення фейкових новин з використанням великих мовних моделей, а також проведено відповідне дослідження ефективності.

#### 3. Зміст кожного розділу роботи

У першому розділі кваліфікаційної роботи проаналізовано інформаційні моделі предметної області, теоретичні підходи до розв'язання задачі та сучасні програмні засоби, результатом якого стали формалізація мети та завдань. У другому розділі наведено опис методу виявлення фейкових новин, алгоритму формування запропонованої моделі та деталізовано проектування програмної реалізації у вигляді вебзастосунку. У третьому розділі охарактеризовано створений вебзастосунок та проведено експериментальне дослідження поданого у роботі методу.

#### 4. Оцінка розробленого методу та його практична цінність

Створені метод та його програмна реалізація можуть застосовуватись для автоматизації процесу виявлення фейкових новин, що сприятиме зменшенню рівня дезінформації суспільства.

#### 5. Якість оформлення кваліфікаційної роботи бакалавра

Кваліфікаційна робота бакалавра виконана на високому рівні. Текст викладено чітко та структуровано, наявні візуальні матеріали, такі як таблиці, діаграми, схеми, графіки, сприяють кращому розумінню поданої інформації та підтвердженню висновків.

#### 6. Недоліки кваліфікаційної роботи бакалавра

Суттєві недоліки у кваліфікаційній роботі бакалавра відсутні. Було б доцільно детальніше проаналізувати процес навчання запропонованої моделі, проте це не впливає на загальну якість роботи та отримані результати.

7. Загальний висновок (допускається чи не допускається до захисту), та оцінка на яку заслуговує кваліфікаційна робота.

З огляду на високий рівень виконання та забезпечення усіх необхідних вимог, кваліфікаційна робота бакалавра Мустаєва Тімура Вікторовича може бути допущена до захисту. Рекомендована оцінка «відмінно».

Рецензент \_\_\_\_\_

Т. Говоруцек