

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ПОЛІСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет обліку та фінансів
Кафедра комп'ютерних технологій
і моделювання систем

Кваліфікаційна робота
на правах рукопису

Рубана Владислава Олександровича
(прізвище, ім'я, по батькові здобувача освіти)

УДК: 004.89

КВАЛІФІКАЦІЙНА РОБОТА

WEB-орієнтована технологія розпізнавання текстового контенту
інформаційних ресурсів англійською мовою

(тема роботи)

КН-20-22ст

(шифр і назва спеціальності)

Подається на здобуття освітнього ступеня бакалавр

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

(підпис, ініціали та прізвище здобувача вищої освіти)

Керівник роботи
Николюк Ольга Миколаївна
(прізвище, ім'я, по батькові)

доктор економічних наук, професор
(науковий ступінь, вчене звання)

Висновок кафедри _____

за результатами попереднього захисту: _____

Протокол засідання кафедри _____

№ _____ від « _____ » _____ 20 _____ р.

Завідувач кафедри _____

(науковий ступінь, вчене звання)

(підпис)

(прізвище, ім'я, по батькові)

« _____ » _____ 20 _____ р.

Результати захисту кваліфікаційної роботи

Здобувач вищої освіти _____ захистив (ла)

(прізвище ,ім'я, по батькові)

кваліфікаційну роботу з оцінкою:

сума балів за 100-бальною шкалою _____

за шкалою ECTS _____

за національною шкалою _____

Секретар ЕК

(науковий ступінь, вчене звання)

(підпис)

(прізвище, ім'я, по батькові)

АНОТАЦІЯ

Рубан В.О. WEB-орієнтована технологія розпізнавання текстового контенту інформаційних ресурсів англійською мовою.

Кваліфікаційна робота на здобуття освітнього ступеня бакалавра за спеціальністю 122 – комп'ютерні науки. – Поліський національний університет, Житомир, 2024.

Кваліфікаційна робота присвячена розробці WEB-орієнтованої технології розпізнавання текстового контенту інформаційних ресурсів англійською мовою. Для досягнення мети роботи у теоретичній частині проведено дослідження предметної області, особливостей задач розробки WEB-систем, підходи до обробки природньої мови. Описано архітектуру WEB-системи розпізнавання текстового контенту інформаційних ресурсів, що включає діаграму бази даних, а також UML діаграму процентентів та послідовності. Розроблено алгоритм розпізнавання текстового емоційного контенту, що включає розробку власної нейронної мережі та застосування навчених моделей для додаткових функцій системи.

Ключові слова: WEB – технологія, БД, обробка природньої мови, англійська мова, алгоритм обробки, розпізнавання, парсинг.

SUMMARY

Ruban V.O. WEB - oriented technology for recognizing textual content of information resources in English.

Qualification work for obtaining a bachelor's degree in specialty 122 - Computer Science - Polissya National University, Zhytomyr, 2024.

Content of the summary

The qualification work is devoted to the development of a WEB-oriented technology for recognizing the textual content of information resources in English. To achieve the goal of the work, the theoretical part of the work includes a study of the subject area, features of the tasks of developing WEB-systems, approaches to natural language processing. The architecture of the WEB system for recognizing the text content of information resources is described, which includes a database diagram, as well as a UML diagram of percentages and sequences. An algorithm for recognizing textual emotional content has been developed, which includes the development of a proprietary neural network and the application of trained models for additional system functions. The results of the system are obtained. Recommendations for the implementation of the system and its use are given.

Keywords: WEB technology, database, natural language processing, English language, processing algorithm, recognition, parsing.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	6
ВСТУП	7
РОЗДІЛ 1. ТЕОРЕТИЧНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕНЬ.....	9
1.1 Аналіз предметної області досліджень	9
1.2 Моделювання бізнес-процесів предметної області.....	11
Висновки до першого розділу.....	13
РОЗДІЛ 2. ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ЗА ТЕМОЮ КВАЛІФІКАЦІЙНОЇ РОБОТИ.....	14
2.1 Структура інформаційної системи розпізнавання текстового контенту	14
2.2 Моделювання структури інформаційної системи за напрямом кваліфікаційної роботи.	16
2.3 Проектування структури бази даних інформаційної системи.	17
2.4 Проектування алгоритму розпізнавання текстового контенту.	19
Висновки до другого розділу	25
РОЗДІЛ 3. Реалізація MoodAnalyzer	26
3.1 Інтерфейс MoodAnalyzer	26
3.2 Тестування та інструкція користувачу інформаційної системи.	31
Висновки до третього розділу	33
ВИСНОВКИ.....	34
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	35

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ІС – інформаційна система

БД – база даних

API – програмний інтерфейс додатків

ОС – операційна система

ПК – персональний комп'ютер

СКБД – система керування базою даних

ВСТУП

Актуальність. За останні роки інформаційні технології пройшли великий шлях та зробили повсякденне життя надзвичайно насиченим потоком даних, де обмін інформацією відбувається з великою швидкістю і така інформація може містити не лише пізнавальну складову та бути корисною, а й містити негативний контент, який згідно дослідженням, шкодить людям не менше ніж фізичне перебування на місцях подій [1]. Тому, розуміння емоційності контенту стає важливим аспектом нашого життя. Використання подібних технологій має потужний потенціал розвитку починаючи з бізнесу, де потрібно аналізувати відгуки клієнтів, закінчуючи розпізнанням емоційного стану людини та наданням необхідної їй підтримки.

Мета кваліфікаційної роботи – розробка інформаційної системи, покликаної вдосконалити процес розпізнавання контенту англійською мовою.

Об'єктом дослідження є процес розпізнавання текстового контенту англійською мовою.

Предметом дослідження є методи, інструменти, моделі розпізнавання текстового контенту англійською мовою.

У процесі дослідження були використані методи аналізу та синтезу, моделювання та прийом порівняння. За темою дослідження опубліковано 2 наукові праці, а саме:

- Рубан В.О., Загрози та атаки для веб-сервісів. Матеріали Всеукраїнської науково-практичної конференції здобувачів вищої освіти і молодих вчених “Інформаційні технології та моделювання систем”. Житомир : Поліський національний університет, 2023. 39-41 с.
- Рубан В.О., Важливість технологій обробки природньої мови на психоемоційний стан людини. Матеріали Всеукраїнської науково-практичної конференції здобувачів вищої освіти і молодих вчених “Інформаційні технології та моделювання систем”. Житомир : Поліський національний університет, 2024. 24-25 с.

Практичне значення отриманих результатів дослідження:

WEB-технологія розпізнавання текстового англомовного контенту буде використовуватися під час інтернет серфінгу, з метою запобіганню негативного впливу на психоемоційний стан людини. Така система допомагає виявляти небажаний текстовий контент та надає користувачу об'єктивну оцінку, щодо текстових характеристик: знаходження негативних/позитивних нейтральних речень, ключових слів.

РОЗДІЛ 1. ТЕОРЕТИЧНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕНЬ

1.1 Аналіз предметної області досліджень

Предметною областю дослідження є процес розпізнання людиною текстового емоційного контенту, а об'єктом є процеси та механізми, завдяки яким людина сприймає, аналізує і розпізнає текстову інформацію. Серед функціональних вимог до інформаційної системи – MoodAnalzyer виділено наступні:

- впровадження підтримки різних форматів даних такі як: текстовий, посилання та файли (DOCS, TXT). Збільшення форматів потоку даних відіграватиме важливу роль у забезпеченні різноманітності та доступності для користувачів сервісу, що також підвищує рівень конкурентоспроможності;
- знаходження ключових слів та посилань на них надає необхідний контекст, що дасть відповідь на запитання – “чи хочу я це читати?”.
- підсумовування тексту додатково збільшує контекст, підвищуючи необхідний рівень розуміння суті. Це допомагає уточнити основні ідеї, а також робить інформацію більш доступною;
- фільтрація речень за рівнем сентиментного аналізу (позитив/негатив/нейтральність) надає конкретне розуміння, щодо настроїв у тексті;
- інформація, щодо знайдених негативних слів, цензури та спаму покращує досвід користувача, підкреслюючи небажаний контент, який може бути образливим або завдавати моральної шкоди користувачу.

У дослідженні проведено порівняльний аналіз аналогічних рішень, серед яких виділено *MonkeyLearn*, *Speak*, *ImentivAI*, *Text2Data*. Ці системи реалізовані у вигляді WEB-додатків, головна мета яких – аналіз текстових

даних, щодо наявності в них негативних настроїв, пошуку ключових слів, аналізу емоцій. Результати аналізу наведено в таблиці 1.1.

Таблиця 1.1 – Інформаційні системи аналізу текстових даних

Характеристика системи	MonkeyLearn	Text2Data	ImentivAI	Speak	MoodAnalyzer (власний продукт)
Формати отримання даних	4	2	5	5	3
Точність сентиментного аналізу	5	4	3	4	4
Пошук ключових слів	3	4	0	4	3
Пошук цензурних слів, спаму	2	0	0	2	3
Надання додаткової інформації	3	3	2	4	4
Разом	17(балів)	13(балів)	10(балів)	18(балів)	17(балів)

Формати отримання даних оцінює різноманіття інтерфейсу та адаптивності алгоритмів, що дозволяє інтегрувати інформацію з різних джерел (JSON, XML, CSV, DOCS, TXT, PDF). Шкала точності сентиментного аналізу відображає ступінь різноманіття використаної моделі:

- типи даних, на яких навчена модель (новини та статті, соціальні мережі);
- підтримка сленгу;
- автокорекція слів;

Оцінка пошуку ключових слів, демонструє точність їх виявлення та різноманіття можливих класифікацій, наприклад виявлене ключове слово – “Ukraine”, можна класифікувати, як геополітичну одиницю. Надання додаткової інформації оцінює загальний рівень наданого контексту поєднуючи пошук цензури, спаму, надання посилань по ключовим словам, та методу сумаризації тексту.

1.2 Моделювання бізнес-процесів предметної області

Бізнес-процеси осмислення процесу розуміння впливу емоційного контенту графічно представлені за допомогою методології SADT IDEF0, яка дає змогу візуалізувати загальну структуру, функції відповідної системи, а також інформаційні та матеріальні потоки. Результати процесного моделювання систематизовані у вигляді IDEF3. IDEF0 – це стандарт моделювання бізнес-процесів та функцій систем, під час використання якої створюється ієрархічна залежність функцій та їх взаємодія, що дозволяє краще розуміти роботу системи [1].

Стандарт IDEF0 включає такі елементи як функції, кожна з яких має вхідні та вихідні потоки, управління та механізми. Вхід функції відображає дані, які необхідні для виконання функції. Управління вказує на те, чим керується блок, а документація, в свою чергу – за допомогою чого відбувається функціонування системи. Виходом є результат роботи функціонального блоку.

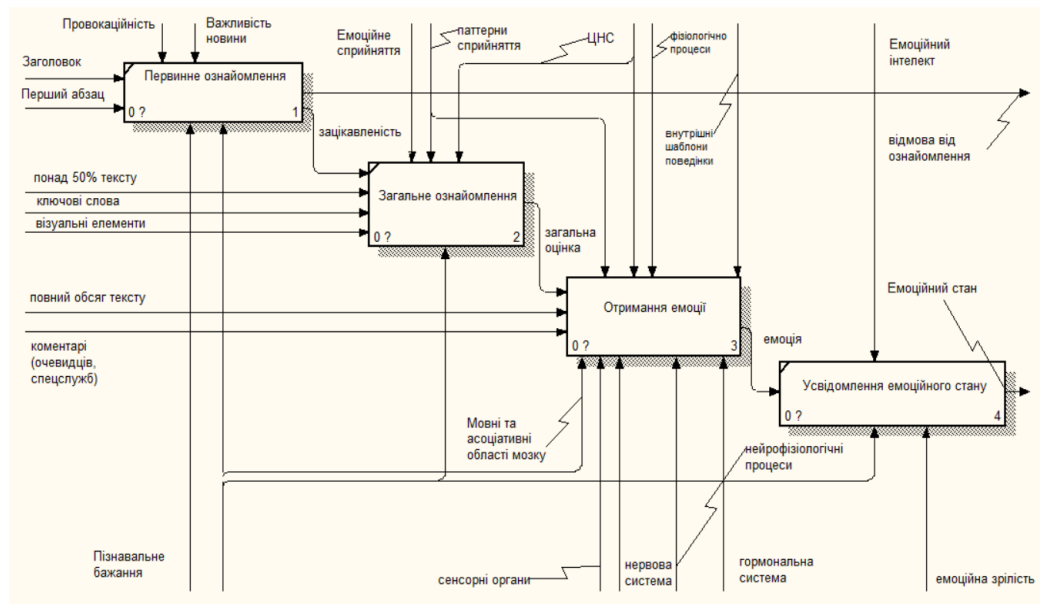


Рис 1.1 – IDEF0 діаграма бізнес процесу розуміння впливу емоційного контенту

Декомпозиція функціонального блоку “Отримання емоції” відображає процес розбиття складної функції на її зменшену ієрархічну структуру, яка

точно ілюструє логіку та послідовність блоку “Отримання емоції”, що дає змогу глибоко зрозуміти процес отримання емоції у людини (рис. 1.2)

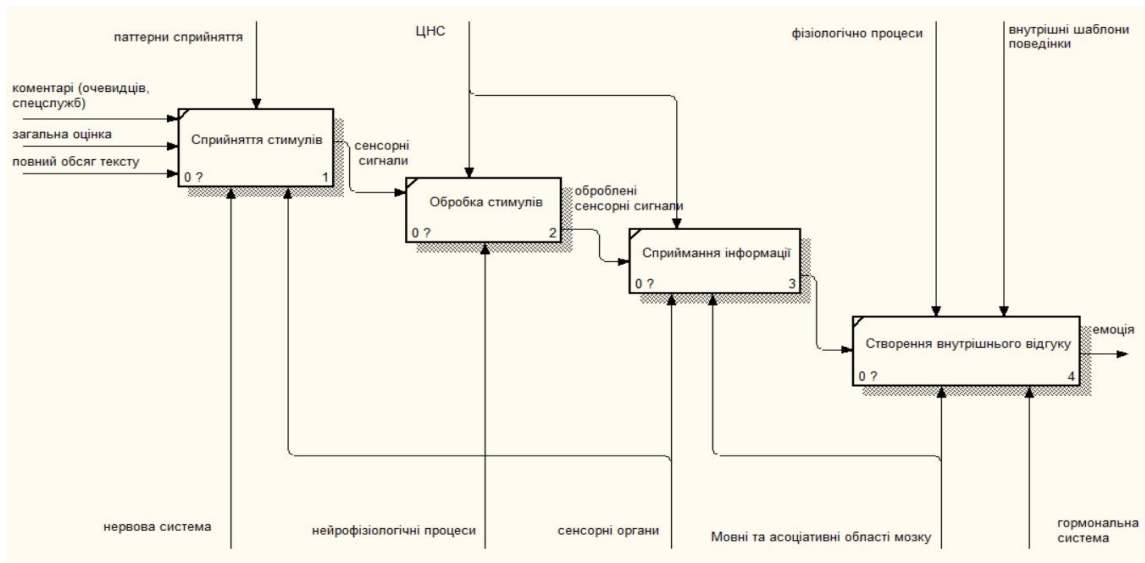


Рис. 1.2 – Декомпозиція функціонального блоку “Отримання емоції”.

Для першого блоку вхідними параметрами є коментарі та загальна суб’єктивна оцінка ситуації, а також повний обсяг тексту; механізмом – нервова система і сенсорні органи. Управлінням у даному випадку є паттерни сприйняття людини, а вихідними потоками – сенсорний сигнал, який потрапляє в блок обробки стимулів, що керується центральною нервовою системою завдяки нейрофізіологічним процесам. Після чого сигнал, потраплений в блок “Сприйняття інформації” оброблюється центральною нервовою системою та передається в блок “Створення внутрішнього відгуку”. Далі завдяки гормональній системі та внутрішніми шаблонами поведінки людина відчуває емоцію.

Для візуалізації потокової послідовності процесу розуміння впливу емоційного контенту розроблено діаграму типу PFDD, яка складається із процесів, зв’язків (стрілок), що демонструє потік даних та з рішень (Decision Points), від яких залежить подальші операції (рис. 1.3).

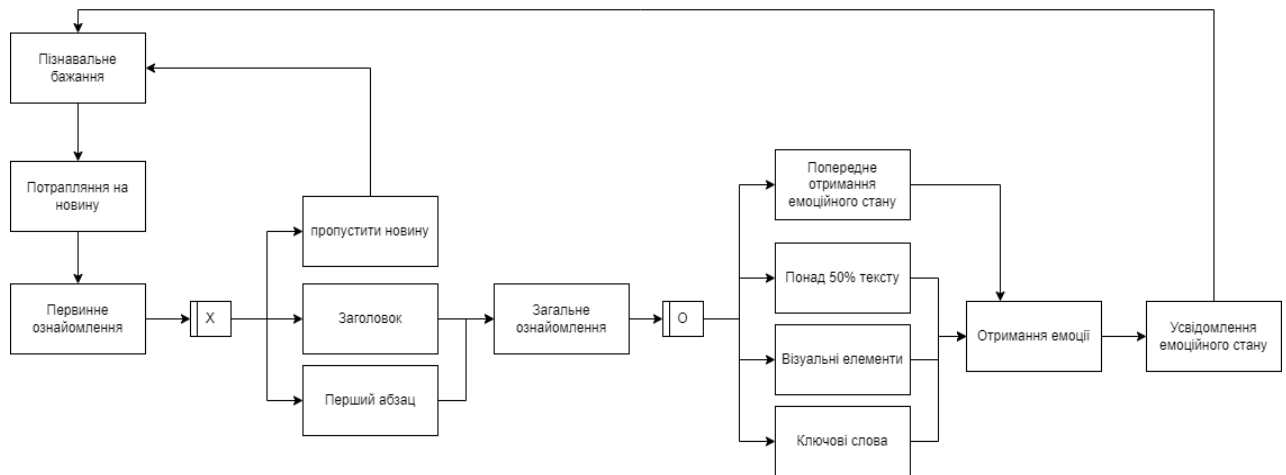


Рис. 1.3 – Діаграма опису потоків даних та рішень PFDD

Початковим процесом визначено Пізнавальне бажання, у межах якого визначається потреба людини в пізнанні. Наступним блоком є Первинне ознайомлення, під час якого вирішується, чи зацікавив текст людину і, якщо ні, то вона повертається до початкового стану. Останнім вирішальним процесом є Загальне ознайомлення, де постає певний вибір. Якщо людина вирішує не продовжувати ознайомлення з текстом, то вона отримає неповну картину події, та, можливо, сформує некоректне бачення ситуації, що вплине її емоційний стан. У випадку повного ознайомлення людина більш точно зможе оцінити текст та коректно сприйняти інформацію, що, можливо змінить бачення людини та змінить її емоційний стан іншим чином.

Висновки до першого розділу

У першому розділі встановлено, що предметною областю дослідження є процес розпізнавання людиною текстового емоційного контенту. Виділено основні функціональні вимоги до системи MoodAnalyzer а саме: підтримка різних форматів отримання даних (DOCS, TXT), знаходження ключових слів, підсумовування тексту, фільтрація речень за рівнем сентиментного аналізу. Також проведено порівняльний аналіз аналогічних рішень (зокрема, таких систем як ImentivAI, Speak, MonkeyLearn, Text2Data), на основі якого виділено їх переваги та недоліки. Моделювання бізнес-процесів виконано з використанням методологій IDEF0, IDEF3.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ЗА ТЕМОЮ КВАЛІФІКАЦІЙНОЇ РОБОТИ

2.1 Структура інформаційної системи розпізнавання текстового контенту

WEB-орієнтована система розпізнавання текстового контенту англійською мовою складається з клієнтської та серверної частини, що реалізовані за допомоги фреймворку Django. Серверна частина відповідає за зберігання, та керування даними, а також роботу із алгоритмом аналізу тексту. Клієнтська частина реалізована з використання HTML, CSS та JavaScript і забезпечує взаємодію із сервером за допомогою комбінованої архітектури REST API та MVT (рис. 2.1). Клієнтську частину можна умовно поділити на модуль авторизації/реєстрації, модуль оплати послуг, модуль профілю та модуль ведення/виведення даних.

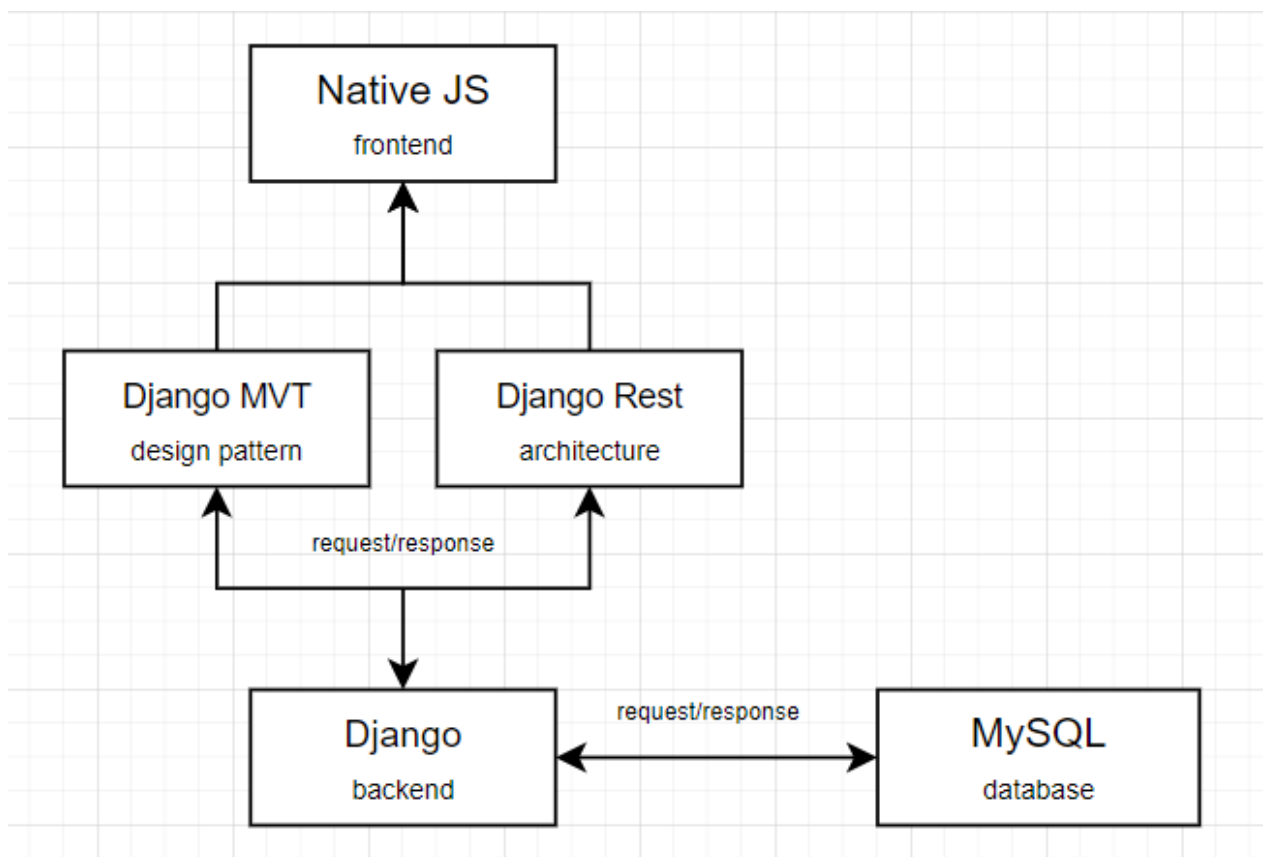


Рис 2.1 – Структура WEB-системи MoodAnalyzer

З метою поглибленого розуміння структури, функцій та особливостей роботи інформаційної системи, використано стандарт моделювання UML.

UML-діаграма прецедентів ілюструє функції (прецедентами) системи у розрізі акторів та, що дозволяє описати її на концептуальному рівні (рис 2.1)

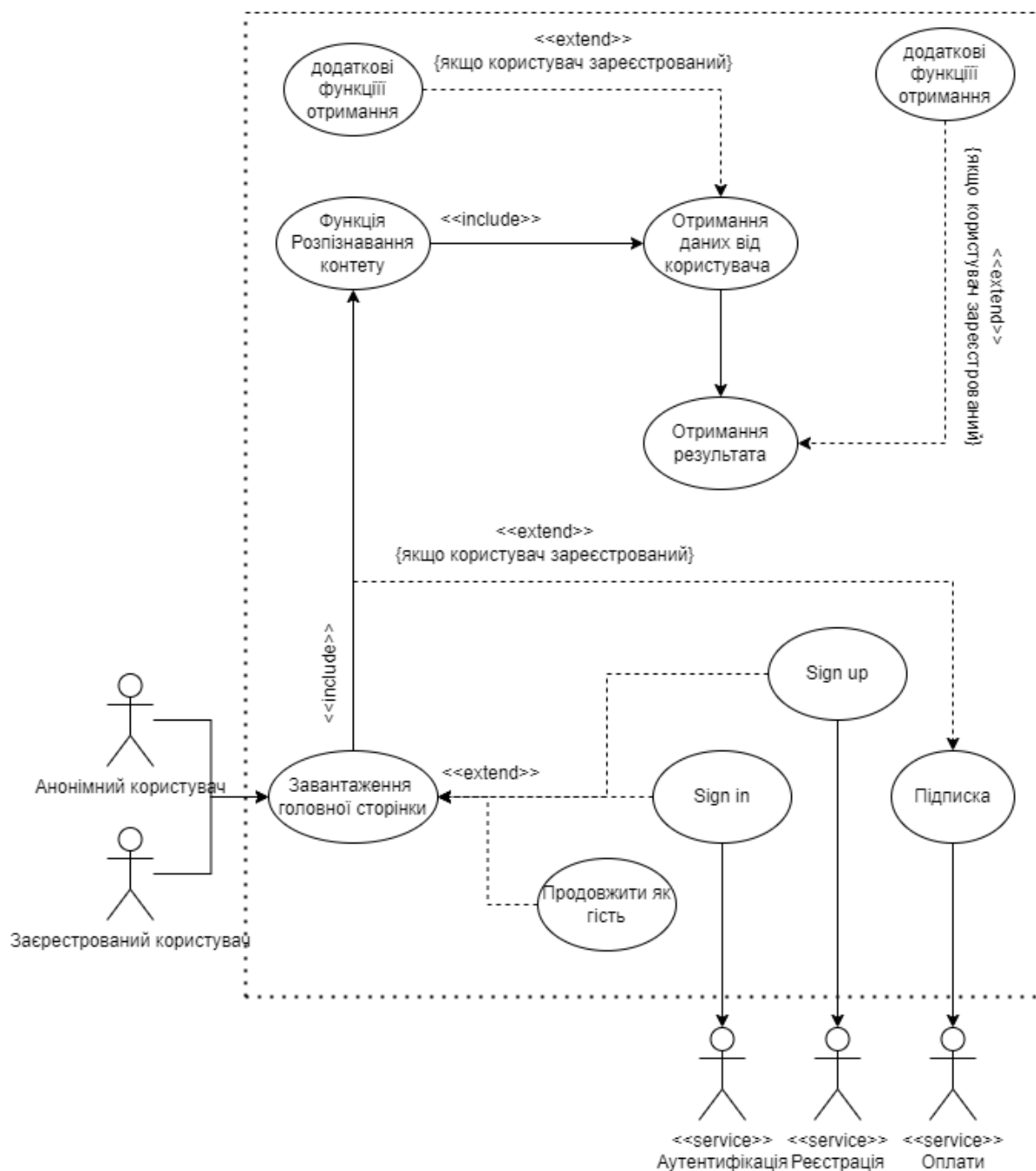


Рисунок 2.2 – UML-діаграма прецедентів WEB-сервісу MoodAnalyzer

Діаграма включає два типи акторів – анонімний користувач та зареєстрований. Відповідно до представленої UML-моделі зареєстрований користувач має низку переваг у вигляді доступу до додатковий функцій, а саме: додаткові функції передачі текстових даних (посилання, файли), а також більш детальний вивід результатів аналізу текстових даних.

2.2 Моделювання структури інформаційної системи за напрямом кваліфікаційної роботи.

UML-діаграма послідовності використовується для детального опису логіки та сценаріїв процесу роботи WEB-сервісу. Такі діаграми містять об'єкти, що взаємодіють у межах власного “сценарію”. Також об'єкти мають цикл життя, який додатково демонструє послідовність взаємодій між ними. Основними компонентами діаграми станів є: лінія життя, актори, повідомлення, приклад реалізації зображено на (рис 2.2).

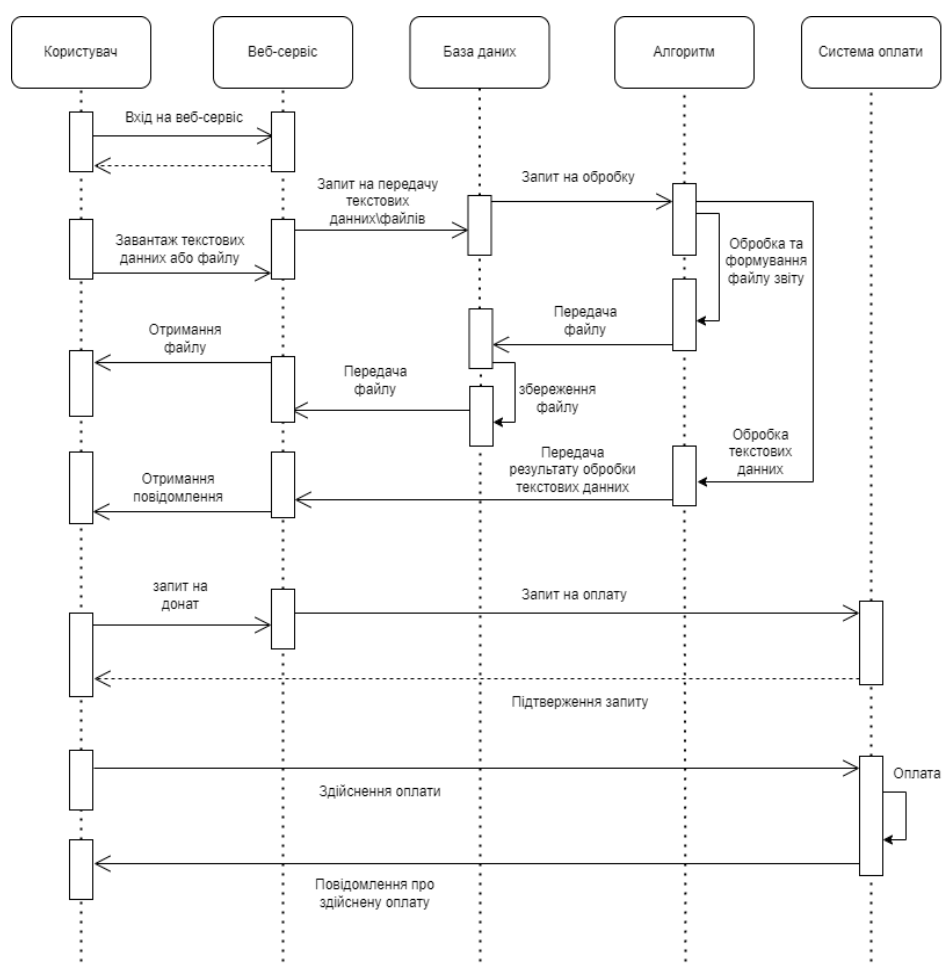


Рисунок 2.3 – UML-діаграма послідовності WEB-сервісу MoodAnalyzer

Діаграма послідовності включає 5 об'єктів, зокрема, Користувач, Веб-сервіс, База даних, Алгоритм, Система оплати. Кожен об'єкт виконує власну функцію. Наприклад веб-сервіс служить графічним інтерфейсом між користувачем та алгоритмом, а база даних відповідає за збереження та передачу файлів користувачеві.

2.3 Проектування структури бази даних інформаційної системи.

Для розв'язання поставлених у дослідженні задач при розробці WEB-сервісу прийнято рішення використовувати реляційну базу даних. Подібні бази даних зазвичай використовуються за необхідності структурованості та налаштуванням зв'язків між таблицями. Вибір реляційної бази даних для розробки WEB-сервісу обґрунтовується її спроможністю використання індексації та оптимізації запитів, що сприяє підвищенню швидкодії ефективності обробки даних (рис. 2.4).

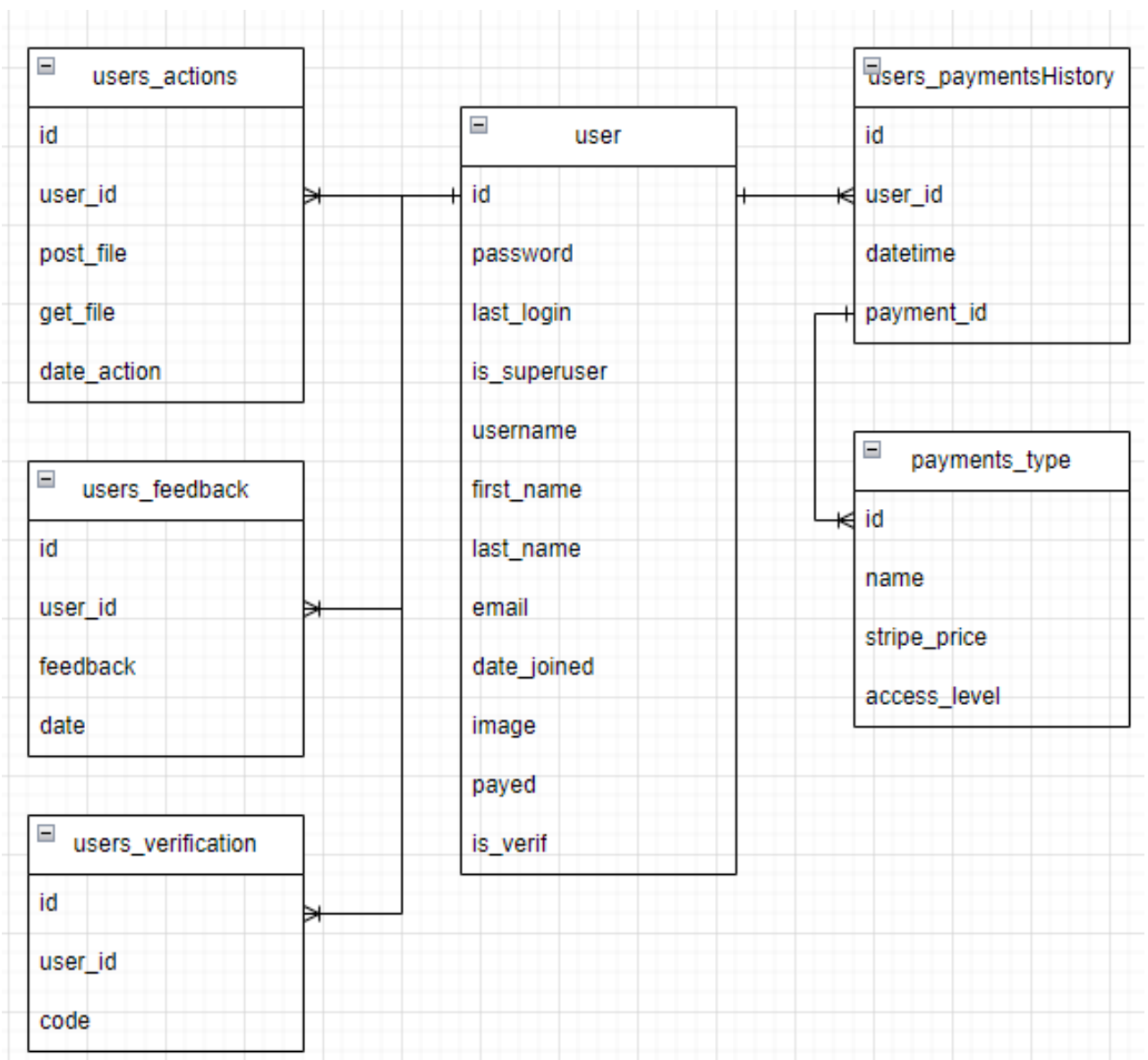


Рис. 2.4 – Структура схеми бази даних WEB-сервісу

Для покращеного розуміння логіки бази даних, створені описи масивів відповідних таблиць, які зображено в(табл. 2.1–2.6).

Таблиця 2.1 – Опис масиву таблиці БД “users”

Ключ/Тип	Поле	Опис поля
PK/int	id	Унікальний ідентифікатор
varchar(128)	password	Пароль користувача
bool=False	is_superuser	Перевірка на користувача з привілеями
UNIQUE/ varchar(128)	username	Логін користувача
varchar(128)	first_name	Ім'я користувача
varchar(128)	last_name	Фамілія користувача
UNIQUE/ varchar(128)	email	Електронна адреса
datetime	date_joined	Дата створення акаунту
blob	image	Аватар користувача
bool=False	payed	Флаг первинного аналізу перевірки статусу акаунта
bool=False	Is_verif	Флаг перевірки валідації акаунта

Таблиця 2.3 – Опис масиву таблиці БД “users_action”

PK/int	id	Унікальний ідентифікатор
FK/int	user_id	Ідентифікатор користувача
blob	post_file	Отриманий файл від користувача
blob	get_file	Результат аналізу наданого файлу

Таблиця 2.2 – Опис масиву таблиці БД “users_feedback”

PK/int	id	Унікальний ідентифікатор
FK/int	user_id	Ідентифікатор користувача

json	feedback	Відгук користувача
datetime	date	Дата надання відгуку

Таблиця 2.4 – Опис масиву таблиці БД “users_paymentsHistory”

PK/int	id	Унікальний ідентифікатор
FK/int	user_id	Ідентифікатор користувача
FK/int	payment_id	Ідентифікатор оплати
datetime	datetime	Дата платежу

Таблиця 2.5 – Опис масиву таблиці БД “payment_type”

PK/int	id	Унікальний ідентифікатор
varchar(128)	name	Назва
deceminal	price	Ціна
deceminal	stripe_price	Ціна для API
int	access_level	Рівень доступу

Таблиця 2.6 – Опис масиву таблиці БД “users_verification”

PK/int	id	Унікальний ідентифікатор
FK/int	user_id	Ідентифікатор користувача
uuid4	code	Унікальний код

2.4 Проектування алгоритму розпізнавання текстового контенту.

Алгоритм розпізнавання тексту базується на архітектурі нейронної мережі прямого поширення [18], що складається з трьох шарів, а саме: вхідного, прихованого та вихідного. Вхідний шар отримує введений текст у вигляді бінарного представлення [19], прихований шар виконує обчислення та аналіз, вихідний – надає фінальний результат розпізнавання символів. Завдяки застосуванню підходу, за якого кожен символ розбивається на біти, нейронна мережа може аналізувати кожен біт окремо, отримуючи інформацію про характеристики кожного символу. Завдяки використанню такого підходу, вхідний шар нейронної мережі складається із 8 нейронів, що дозволяє

збільшити кількість можливих розпізнаних символів до 256. Загальну архітектуру нейронної мережі прямого поширення зображено на рис. 2.5.

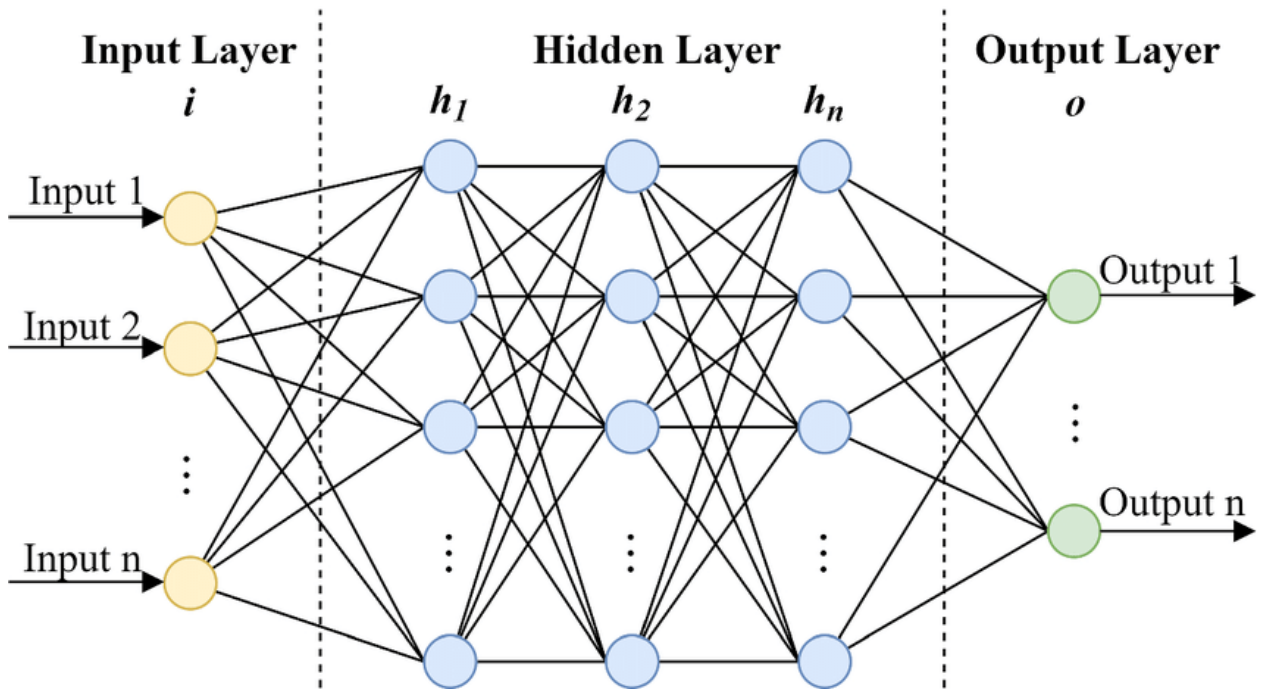


Рис 2.5 – Демонстративна архітектура нейронної мережі прямого поширення

У розробленому алгоритмі прихований шар нейронної мережі налічує 57 нейронів, що обумовлено потребою забезпечення достатньої потужності для вивчення закономірностей у вхідних даних. Вихідний шар нейронної мережі складається із 57 нейронів, що відповідає кількості літер у нижньому та верхньому регістрі, а також пунктуаційним символам: [? ! , . ‘ ‘]. Вибірка відображена на рис. 2.6.

```

1  alphabet_ = ["a", "b", "c", "d", "e", "f", "g", "h", "i", "j", "k", "l",
2             "m", "n", "o", "p", "q", "r", "s", "t", "u", "v", "w", "x",
3             "y", "z", " ", "?", "!", ",", ".:", "A", "B", "C", "D", "E",
4             "F", "G", "H", "I", "J", "K", "L", "M", "N", "O", "P", "Q",
5             "R", "S", "T", "U", "V", "W", "X", "Y", "Z"]
6

```

Рис 2.6 – Вибірка символів необхідних для розпізнання

Вибірка необхідна для перетворення абетки у числове векторне представлення для того, щоб навчити нейронну мережу розпізнавати та

працювати з різними символами, які можуть зустрічатися в тексті. Вибірка символів допомагає підготувати дані для подальшого використання у навчанні нейронної мережі. Процес перетворення реалізований в окремому класі – *BinaryAlphabet*, що налічує такі методи:

```
def create_bitStr(self, arg_text, arg_arr_bit):
    for val in arg_text:
        value = self.text_to_bits(val)
        arg_arr_bit.append(value)

def create_bitInt(self, arg_arr_bit, arg_arr_split):
    for val in arg_arr_bit:
        for values in val:
            values = int(values) * 0.98 + 0.01
            arg_arr_split.append(values)

def final_matrix(self, arg_matrix, arg_arr_bit):
    count = 0
    for i in range(len(arg_matrix)):
        for j in range(8):
            arg_matrix[i][j] = arg_arr_bit[count]
            count += 1
```

Метод “*create_bitStr*” отримує на вхід текстові дані (*arg_text*) та пустий список (*arg_arr_bit*), до якого потім будуть додаватися бінарні представлення кожного символу тексту. В цьому методі викликається метод *text_to_bits*, який перетворює кожен символ тексту у бінарний формат, і отриманий бінарний рядок додається до списку *arg_arr_bit*.

Метод “*create_bitInt*” отримує на вхід список бінарних рядків (*arg_arr_bit*) та пустий список (*arg_arr_split*), до якого потім будуть додаватися числові значення кожного біта бінарного рядка. В цьому методі кожен символ бінарного рядка перетворюється на числове значення від 0.01 до 0.99 для того, щоб забезпечити нормалізацію даних.

Метод “final_matrix” отримує на вхід матрицю (arg_matrix), яка є векторним представленням абетки, та список бінарних рядків (arg_arr_bit), які представляють символи. Метод заповнює матрицю arg_matrix значеннями з бінарних рядків arg_arr_bit.

Ці методи використовуються для перетворення текстових даних у числове векторне представлення для навчання нейронної мережі. Такий підхід важливий з позиції підготовки даних для подальшого їх використання у навчанні нейронної мережі та роботі з нею. Наступним важливим кроком є процес навчання нейронної мережі, що відбувається у декілька таких етапів:

1. На вхід подаються банарні векори, після чого кожен символ тексту перетворюється у відповідний бінарний вектор.
2. Процес прямого проходу, під час якого вхідні дані обчислюються із урахуванням вагів , які з'єднують кожен вхідний нейрон з прихованими та вихідними шарами, після чого подається в активаційну функцію.
3. Активаційна функція – сигмоїда [20], обчислює вихідний сигнал на кожному нейроні шарі. Вона отримує зважену суму вхідних сигналів для кожного нейрона шару і вираховує величину активації.
4. Після отримання вихідних сигналів на вихідному шарі розраховуються помилки, порівнюючи фактичні виходи мережі з очікуваними значеннями.
5. Застосовується алгоритм зворотного розповсюдження помилок “backpropagation” [21], який використовує знайдені помилки на вихідному шарі для оновлення ваг мережі.
6. Застосування епох для повторного навчання, задля коригування параметрів ваг.

Реалізація методів навчання розроблена у такий спосіб:

```
def traning(self, inputs_mass, goal_mass):
    # перетворити список вхідних значень у двовимірний масив
    global_inputs = np.array(inputs_mass, ndmin=2).T
    goal = np.array(goal_mass, ndmin=2).T

    # зважена сума на кожному нейроні прихованого шару
    hi_in = np.dot(self.w_i_h, global_inputs) + self.b_h
    hidden_out_sigma = self.sigma_activation(hi_in)

    # зважена сума на кожному нейроні вихідного шару
    out_in = np.dot(self.w_h_o, hidden_out_sigma) + self.b_o
    # вихід шару out після активації
    global_out_sigma = self.sigma_activation(out_in)

    err_out = goal - global_out_sigma
    err_hidden = np.dot(self.w_h_o.T, err_out)

    # оновлення ваг між прихованим та вихідним шаром
    self.w_h_o += self.ls * np.dot(err_out * global_out_sigma * (1 -
global_out_sigma), np.transpose(hidden_out_sigma))
    # оновлення ваг зміщення вихідного шару
    self.b_o += -1 * self.ls * err_out * global_out_sigma * (1 - global_out_sigma)
    # оновлення ваг між вхідним та прихованим шаром
    self.w_i_h += self.ls * np.dot(err_hidden * hidden_out_sigma * (1 -
hidden_out_sigma), np.transpose(global_inputs))
    # оновлення ваг зміщення прихованого шару
    self.b_h += -1 * self.ls * err_hidden * hidden_out_sigma * (1 -
hidden_out_sigma)

def episode(self):
    ep = 360
    for i in range(ep):
        for r in range(57):
            self.traning(self.alphabet_matrix[r], self.standartAlphabet[r])
```

Після реалізації методів навчання, які включають прямий прохід через мережу та алгоритм зворотного розповсюдження помилок, необхідно розглянути процес застосування мережі для передбачення результатів на

нових даних, завдяки методу – *questing*. Цей метод виконує прямий прохід через мережу, обчислюючи вихідні значення на основі вхідних даних. Після цього, він повертає результат у вигляді вектора вихідних значень. Приклад реалізації:

```
def questing(self, inputs_mass):
    # перетворити список вхідних значень у двовимірний масив
    global_inputs = np.array(inputs_mass, ndmin=2).T
    # зважена сума на кожному нейроні прихованого шару
    hi_in = np.dot(self.w_i_h, global_inputs) + self.b_h
    # вихід прихованого шару після активації
    hidden_out_sigma = self.sigma_activation(hi_in)
    # зважена сума на кожному нейроні вихідного шару
    out_in = np.dot(self.w_h_o, hidden_out_sigma) + self.b_o
    # вихід шару out після активації
    global_out_sigma = self.sigma_activation(out_in)
    return global_out_sigma
```

Після реалізації методу “*questing*”, який обчислює вихідні значення на основі вхідних даних, необхідно розглянути процес інтерпретації цих вихідних значень для розпізнавання конкретних символів. Цей процес здійснюється за допомогою методу “*identification_exitMatrix*”. Приклад реалізації неведений нижче:

```
def identification_exitMatrix(self, array):
    for j in range(len(array)):
        position_max = 0
        max_element = self.questing(array[j])[0][0]
        for i in range(self.onod):
            if self.questing(array[j])[i][0] >= max_element:
                max_element = self.questing(array[j])[i][0]
                position_max = i
        self.MASS_TEXT.append(self.__dict_symbols[position_max])
```

Метод “*identification_exitMatrix*” використовується для визначення символів на основі вихідних значень, отриманих від методу “*questing*”. Він

ідентифікує символи, які були розпізнані нейронною мережею, шляхом аналізу вихідного вектора значень, яке відповідає найбільш ймовірному символу.

Висновки до другого розділу

В другому розділі спроектовано архітектуру WEB-додатку розпізнавання текстового контенту MoodAnalyzer, розробленого із використання інструментів Django, Django REST, JS. Розподілений підхід при розробці дозволяє гнучко реалізовувати архітектурний стиль REST, завдяки якому легко масштабувати проєкт. Розроблено UML-діаграми прецедентів, за допомогою якої відображено WEB-орієнтовану систему на концептуальному рівні; та діаграму послідовності, яка представляє взаємодію між об'єктами у вигляді послідовності. Реалізована база даних, яка необхідна для збереження файлів користувачів, отримання відгуків, а також реалізації функції оплати. Додатково було розглянуто процес навчання навченої мережі та її використання для передбачення результатів на нових даних з використанням відповідних методів.

РОЗДІЛ 3. Реалізація інформаційної системи MoodAnalyzer

3.1 Інтерфейс MoodAnalyzer

Графічний інтерфейс веб-сервісу включає такі компоненти як кнопки, текстові поля, меню та графічні елементи для навігації та взаємодії користувача. Дизайн даної системи спрямований на забезпечення привабливого та зручного користувацького досвіду, а також ефективного використання її функцій. Головна сторінка WEB-сервісу “MoodAnalyzer”, зображена на рис. 3.1.

The screenshot displays the MoodAnalyzer web interface. At the top, the 'Mything' logo is on the left, and 'Support', 'My profile', and 'Log In' links are on the right. The main content area is organized into several functional blocks:

- Check emotion text:** A text input field containing a sample paragraph about a Tesla whistleblower, followed by 'Clear' and 'Check text' buttons.
- Organization:** A section with tags for 'Tesla' and 'BBC News'.
- Person:** A section with tags for 'Elon Musk', 'Cristina Balan', and 'Ms Balan'.
- Geo-Political Entity:** A section with tags for 'US', 'California', and 'Romania'.
- Text Analysis Tool:** A section titled 'Text Analysis Tool' with the subtitle 'Analyze your text files to identify sentiment, brands, topics and trends with analysis tool'. It shows a 'Main message' box with a sample text, followed by sentiment analysis results: 23.98% (positive), 58.13% (neutral), and 17.89% (negative). Below this are three boxes for 'Negativity', 'Censor', and 'Spam', each with a list of related terms and a 'Not found' status.
- Sentences Table:** A table with columns '№', 'Sentences', and 'Score'. It lists three sentences with their respective scores and sentiment icons.

№	Sentences	Score
1	She won a wrongful dismissal case but was later publicly accused by Tesla of using its resources for a "secret project" – accusations which amount to embezzlement which is a crime under US law.	-0.8462
2	But it failed – and Ms Balan went on to lose her job.	-0.8402
3	Ms Balan was worried the carpets were curling underneath some pedals – a simple but potentially lethal design flaw - and said customers had complained.	-0.631
- File processing:** A section with a flow diagram showing 'File processing' leading to 'Data processing', which then leads to 'Enjoy the result!'. Below the diagram are 'Upload file' and 'Enjoy the result!' buttons.

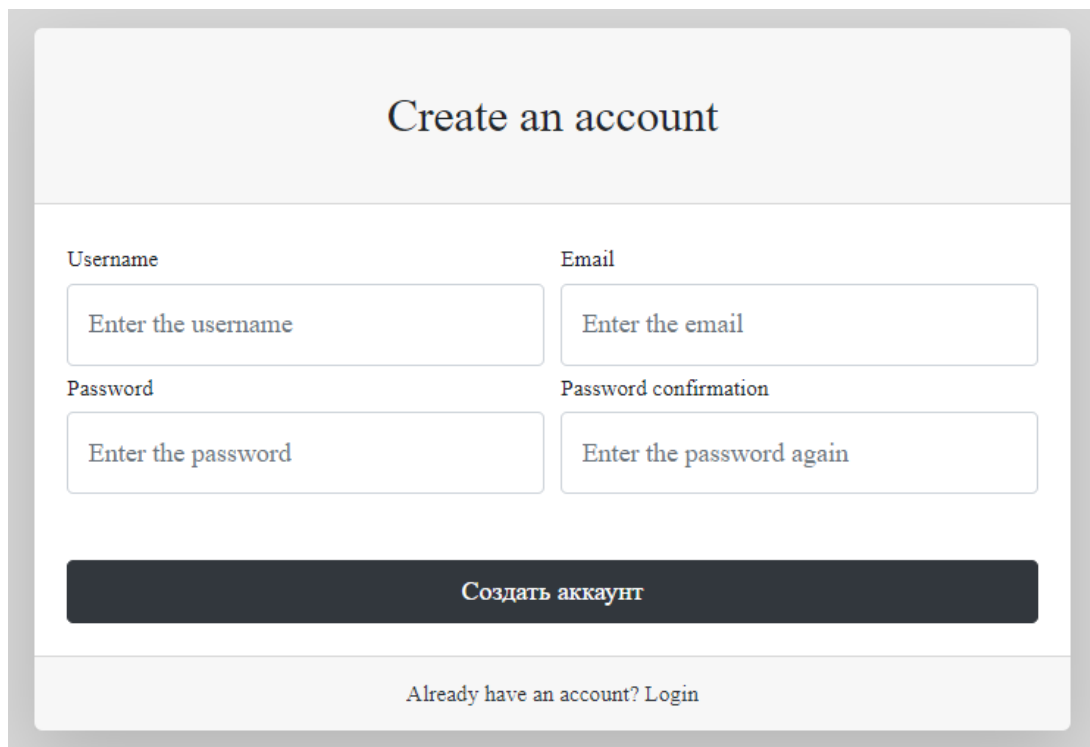
The footer of the page contains the name 'Ruban Vlad' and the affiliation 'Polissia National University'.

Рисунок 3.1 – Головна сторінка WEB-ресурсу

Головну сторінку WEB-сервісу можна умовно поділити на 3 блоки, а саме: блок отримання тексту та виводу ключових слів; блок виводу головного змісту тексту, виводу знайдених негативних слів, цензури та спаму, а також відображення речень та їх фільтрація (негатив/нейтральність/позитив); блок завантаження файлових даних. Також головна сторінка включає верхню панель управління з наступними посиланнями:

- Support – форма підтримки сайту через систему підписки;
- My profile – профіль користувача;
- Log in – форма авторизації користувача.

Вікно реєстрації включає такі поля вводу: логін, електронну адресу та два поля для коректного вводу паролю. Якщо користувач вже має акаунт, то йому необхідно натиснути “Sign In”, де буде відображено вікно авторизації (рис. 3.3, 3.4).



Create an account

Username

Enter the username

Email

Enter the email

Password

Enter the password

Password confirmation

Enter the password again

Создать аккаунт

Already have an account? Login

Рисунок 3.2 – Вікно реєстрації.

Рисунок 3.4 – Вікно авторизації.

Вікно авторизації містить два обов'язкових поля для введення – логін та пароль. Якщо користувач ввів некоректні дані, то буде виведено повідомлення про помилку, яке вказуватиме на необхідність повторного введення інформації. Після вдалої аутентифікації та авторизації користувач відкриває більше функцій (наприклад, доступними є функції завантаження файлових даних та персоналізації з можливістю створення відгуку про WEB-сервіс (рис. 3.5, 3.6).

№	Date	Type	Sent file	Received file	Delete
1	10.05.2024	DOCX	file2.docx	result3.docs	
2	8.05.2024	DOCX	file3.docx	result2.docs	
3	3.05.2024	TXT	file1.txt	result.docs	

Рисунок 3.5 – Сторінка профілю користувача та його файлів.

Профіль користувача налічує головну інформацію щодо користувача, зокрема, дату останнього платежу, електронний адрес а також повне ім'я. Для додаткового покращення інформаційної системи впроваджено блок фідбеку, що налічує головні пункти:

- Для чого використовується система?
- Звідки ви дізналися про нас?
- Чому саме ми?
- Який функціонал необхідно впровадити?

The screenshot shows a user profile page for 'Mything'. The page has a dark header with the logo 'Mything' on the left and 'Support', 'My profile', and a 'Log Out' button on the right. Below the header, there are two main sections: 'User profile' and 'User's files'. The 'User profile' section includes a 'Picture profile' with a green robot image and a 'Set photo' button. To the right of the image are input fields for 'Last payment' (10.05.2024), 'First name' (Vladyslav), 'Last name' (Ruban), and 'Email' (vladruban8@gmail.com), with a 'Submit' button below. The 'User's files' section contains a feedback form with the text 'Your feedback is very important to us, as it helps us improve ourselves'. It includes questions: 'What are you using us for?', 'Where did u here about us?', 'Why did u choose us?', and 'Which functionality are you still expecting?'. A 'Send feedback' button is at the bottom of the form. The footer of the page shows 'Ruban Vlad' on the left and 'Polissia National University' on the right.

Рисунок 3.6 – Сторінка профілю користувача та форми залишку відгука.

Для впровадження монетизації та інтеграції рівнів доступу, розроблено систему підписок із трьохрівневою ієрархією. Кожна підписка діє впродовж одного місяця та надає низку переваг. Підписка за 2\$ відкриває користувачу можливість завантажувати 10 файлів на місяць, підписка за 5\$ не має обмеження щодо кількості завантажених документів, а також відкриває можливість різноманітного фільтрування речень

(негатив/нейтральність/позитив) і додаткову функцію – знаходження спаму (рис. 3.7).

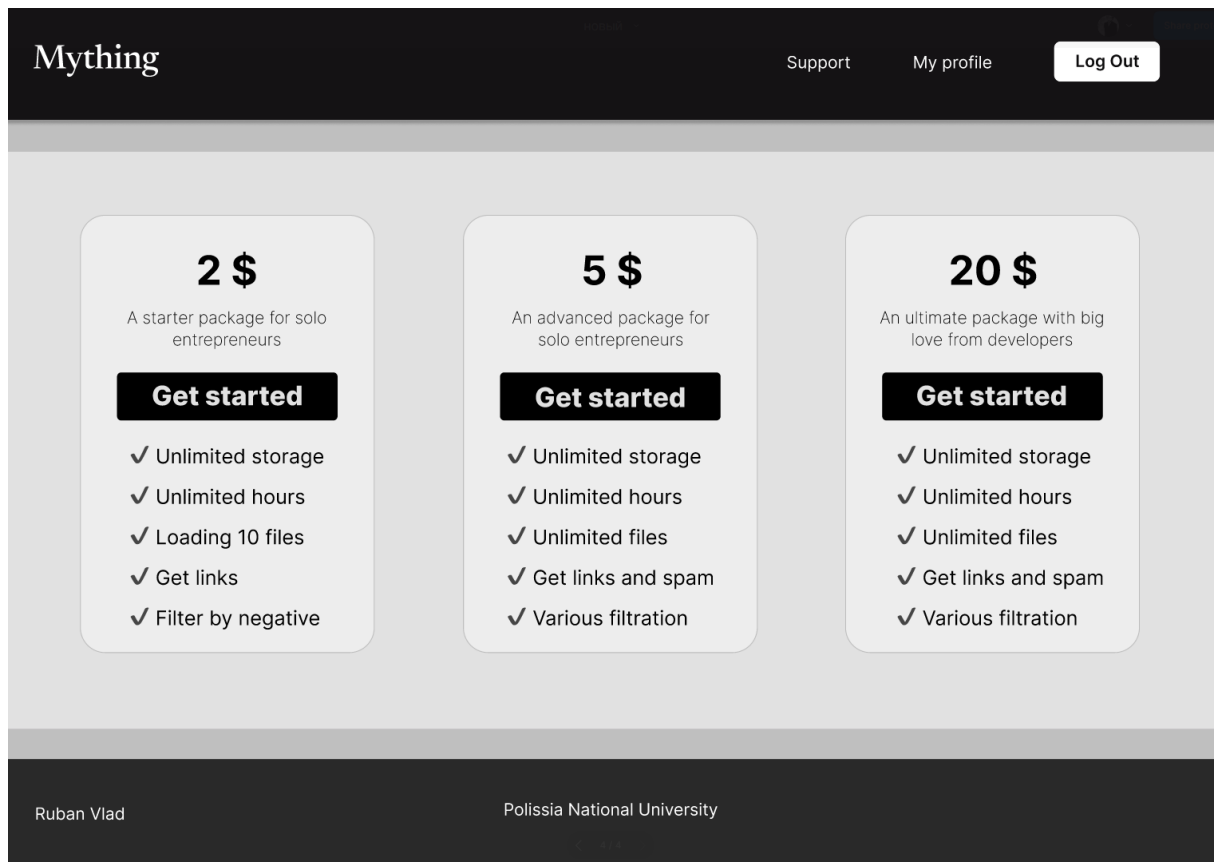


Рисунок 3.7 – Сторінка варіацій підписки

Для підключення оплати обрано онлайн-платформу Stripe, яка дозволяє зручно та ефективно працювати з онлайн-транзакціями. Stripe надає широкий спектр можливостей для інтеграції платіжних систем; однією із ключових переваг цієї платформи є простота у використанні та різноманітний функціонал.

Рисунок 3.8 – Сторінка оплати

Сторінка оплати налічує стандартну форму заповнення: електронну адресу покупця послуги, номер, код та ім'я власника картки, а також країна перебування.

3.2 Тестування та інструкція користувачу інформаційної системи.

Для використання повного функціоналу, користувачу слід створити обліковий запис, всі помилки під час реєстрації будуть виведені користувачу (рис. 3.8).

Рис. 3.9 – Приклад невдалої реєстрації

При виконанні умов, таких як унікальний логін, унікальна електронна адреса, а також пароль відповідного рівня складності користувач буде мати змогу створити облікову адресу. Після чого користувач отримує повний доступ до безкоштовного функціоналу WEB-сервісу: аналіз текстових даних у безлімітному режимі, а також можливість аналізу посилань (*kyivpost.com*, *unian.com*). Форма завантаження документів продемонстрована на рис. 3.10.

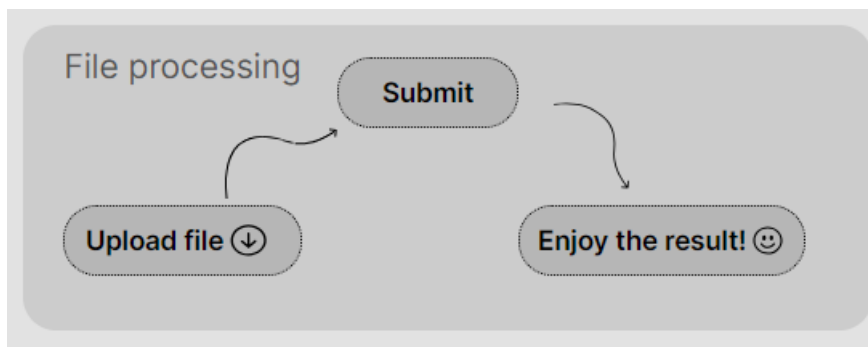


Рис 3.10 – Вікно завантаження документів

Після того, як користувач завантажив файл йому необхідно перейти до розділу “My profile” у секцію “User’s files”, де він зможе завантажити результат аналізу документу, приклад зображено на рис 3.10.

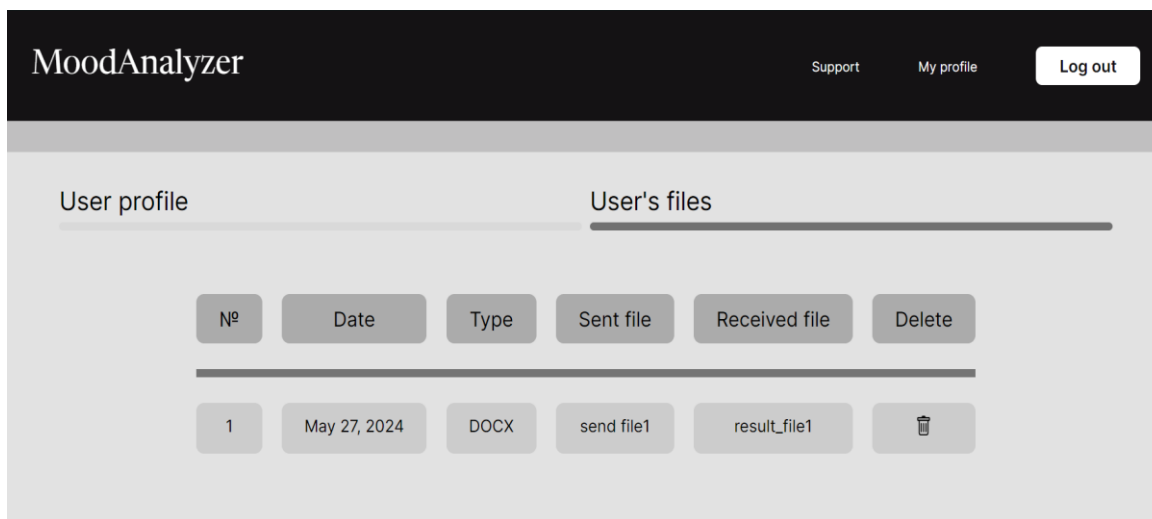


Рис. 3.11 – Сторінка профілю користувача секції з документами

Секція "Файли користувача" містить блоки, які надають додаткову інформацію про проведений аналіз: дата завантаження, тип отриманого файлу, назва переданого файлу та назва отриманого файлу, що містить префікс

"result", додатково містить кнопку для видалення файлів. У разі необхідності підвищення статусу акаунта, користувачу необхідно перейти до сторінки "Support" (рис. 3.7).

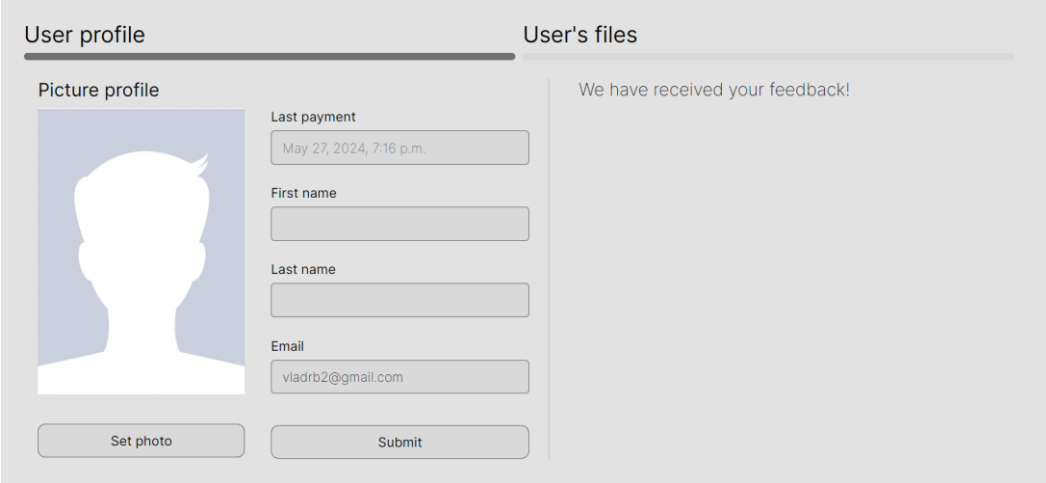


Рис. 3.12 – Сторінка профілю користувача

У випадку вдалої оплати користувач може переглянути останній платіж у профілю, а також залишити відгук, щодо функціоналу системи.

Висновки до третього розділу

В третьому розділі розроблено інтерфейс WEB-сервісу та продемонстровано інструкцію користувачу інформаційної системи, зокрема, відображено структуру його головної сторінки. Головна сторінка WEB-орієнтованої системи відображає три інформаційних вікна, а саме: блок отримання тексту та виводу ключових слів; блок виводу головного змісту тексту, виводу знайдених негативних слів, цензури та спаму, а також відображення речень та їх фільтрація; блок завантаження файлів. Проведено тестування відгуку графічного інтерфейсу на взаємодію із користувачем, також проведено тестування алгоритмів на попередньо підібраних текстових даних із відомим рівнем емоційного забарвлення та виявлених ключових слів.

ВИСНОВКИ

Розробка WEB-технології розпізнавання англomовного контенту є надзвичайно важливою, оскільки у період повномасштабної війни питання емоційної стабільності набуває надзвичайного значення. Особливо це стосується дітей та людей похилого віку, які є найбільш вразливими до негативного впливу агресивного інформаційного середовища. Така система дозволяє попередньо виявляти негативний контент та попереджує користувача щодо можливості його впливу.

Серед функціональних вимог виділено впровадження підтримки різних форматів даних такі як: текстовий, посилання та файли, знаходження ключових слів та посилань, підсумовування тексту, фільтрація речень за рівнем сентиментного аналізу і надання необхідних джерел. В рамках розробки WEB-технології розпізнавання англomовного контенту, було розроблено нейронну мережу прямого розповсюдження. Ця мережа є ключовим елементом системи, оскільки вона відповідає за розпізнавання текстової інформації. Реалізація нейронної мережі була здійснена з використанням архітектури з трьох шарів: вхідного, прихованого та вихідного. Процес навчання нейронної мережі включає етапи прямого проходу, де вхідні дані обчислюються з урахуванням ваг, та застосування активаційної функції. Після цього використовується алгоритм зворотного розповсюдження помилок для оновлення ваг мережі, навчання мережі здійснюється протягом 360 епох.

Перспективи подальших досліджень включають покращення точності розпізнавання контенту та розширення підтримки нових форматів даних – відео формат, для уможливлення аналізу відео контенту. Також планується інтеграція додаткових алгоритмів для аналізу емоційного забарвлення контенту та виявлення іронії чи сарказму. Розширення функціоналу системи дозволить збільшити її ефективність та адаптивність, що буде заохочувати нових користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Як перегляд травматичних новин впливає на психічне здоров'я? URL: <https://forbes.ua/lifestyle/depresiya-ptsr-ta-bezsonnya-yak-pereglyad-travmatichnikh-novin-vplivayut-na-psikhichne-zdorovya-14102022-9041>
(дата звернення: 02.06.2024)
2. What parsing is? URL: <https://themightyprogrammer.dev/article/parsing>
(дата звернення: 02.06.2024)
3. Create IDEF0 diagrams URL: <https://support.microsoft.com/en-us/office/create-idef0-diagrams-ea7a9289-96e0-4df8-bb26-a62ea86417fc>
(дата звернення: 02.06.2024)
4. Understanding IDEF Diagram: An In-depth Look URL: <https://boardmix.com/knowledge/idef-diagram/> (дата звернення: (дата звернення: 02.06.2024)
5. How to Create an IDEF0 Diagram for an Application Development URL: https://medium.com/@CSO_ConceptDraw/how-to-create-an-idef0-diagram-for-an-application-development-8c07654c9ae4 (дата звернення: (дата звернення: 02.06.2024)
6. IDEF3 Process Capture Method URL: https://link.springer.com/chapter/10.1007/978-3-540-89556-5_8 (дата (дата звернення: 02.06.2024)
7. IDEF3 URL: <https://en.wikipedia.org/wiki/IDEF3> (дата звернення: (дата звернення: 02.06.2024)
8. How to Model any Process or Structure in Your Business URL: <https://www.process.st/uml-tutorial/> (дата звернення: 02.06.2024)
9. The IDEF family URL: https://www.researchgate.net/publication/226325476_The_IDEF_family_of_languages (дата звернення: 02.06.2024)
10. Precedence Diagram URL: <https://www.edrawsoft.com/precedence-diagram.html> (дата звернення: 02.06.2024)

11. Activity Diagrams URL: <https://sparxsystems.com/resources/tutorials/uml2/activity-diagram.html>
(дата звернення: 02.06.2024)
12. What is Django? URL: <https://www.djangoproject.com> (дата звернення: 02.06.2024)
13. What is Django Rest Framework? URL: <https://www.django-rest-framework.org> (дата звернення: 02.06.2024)
14. Difference between Django and DRF? URL: <https://apidog.com/articles/difference-between-django-and-django-rest-framework/> (дата звернення: 02.06.2024)
15. What is REST? URL: <https://www.codecademy.com/article/what-is-rest> (дата звернення: 02.06.2024)
16. What is request in WEB? URL: <https://sourcedefense.com/glossary/web-request/#:~:text=A%20web%20request%20is%20a,on%20the%20World%20Wide%20Web.> (дата звернення: 02.06.2024)
17. What is relational database URL: <https://www.ibm.com/topics/relational-databases> (дата звернення: 02.06.2024)
18. Difference between Relational database and NoSQL: <https://www.geeksforgeeks.org/difference-between-relational-database-and-nosql/> (дата звернення: 02.06.2024)
19. Feed forward NN URL: <https://deepai.org/machine-learning-glossary-and-terms/feed-forward-neural-network> (дата звернення: 02.06.2024)
20. Representing data URL: <https://www.bbc.co.uk/bitesize/guides/zpfdwmn/revision/1> (дата звернення: 02.06.2024)
21. Sigmoid activation function: An introduction URL: <https://builtin.com/machine-learning/sigmoid-activation-function> (дата звернення: 02.06.2024)

22. Backpropagation in NN URL:
<https://www.geeksforgeeks.org/backpropagation-in-neural-network/> (дата
звернення: 02.06.2024)