Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Метод контент-аналізу веб-сайту муніципалітету із використанням алгоритмів штучного інтелекту

(тема)

Виконав:

студент II курсу, групи СПм-22-4

Хрустальов Є.К.
(прізвище, ініціали)

Спеціальність 123-Комп'ютерна інженерія

(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма системне програмування

(повна назва освітньої програми)

Керівник: проф. Фесенко Т. Г.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ _____ Коваленко А.А.
(підпис) (прізвище, ініціали)

2024 р.

Факультет комп’ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123-Комп'ютерна інженерія
(код і повна назва)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма системне програмування
(повна назва)

Харківський національний університет радіоелектроніки

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
“____”_____ 20__ р

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Хрустальову Євгенію Кириловичу
(прізвище, ім’я, по батькові)

1. Тема роботи Метод контент-аналізу веб-сайта муніципалітету із використанням алгоритмів штучного інтелекту

затверджена наказом по університету від “ 01 ” квітня 2024 р. № 257 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 15 червня 2024

3. Вхідні дані до роботи

1) документація мови програмування C#

2) документація мови програмування TypeScript

3) інформація про фреймворку та бібліотек ML.NET

4) інформація про фреймворку та бібліотек Angular

5) інформація про інтеграційні технології AJAX

6) інформація про інтеграційні технології RESTful API

4. Перелік питань, що потрібно опрацювати в роботі

1. Аналіз предметної області

2. Огляд і вибір технологій та інструментів розробки

3. Програмна реалізація

4. Інструкція користувача

5. Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів)

Слайд-презентація – 10 слайдів: об'єкт, предмет, мета та завдання дослідження, схема розробленого застосунку, використанні технології, модель глибокої нейромережі, приклад веб-сайту муніципалітету, головна сторінка, сторінка з результатами, висновки.

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Отримання завдання	01.04.2024 - 01.04.2024	Виконано
2	Збір аналіз матеріалів для дослідження	01.04.2024 - 15.04.2024	Виконано
3	Вибір технологій розробки веб-додатків	16.04.2024 - 07.05.2024	Виконано
4	Програмна реалізація	08.05.2024 - 20.05.2024	Виконано
5	Розробка опису програмної реалізації	21.05.2024 - 27.05.2024	Виконано
6	Відлагодження програмних модулів	27.05.2024 - 31.05.2024	Виконано
7	Оформлення кваліфікаційної роботи	31.05.2024 - 04.06.2024	Виконано
8	Рецензування проекту керівником	04.06.2024 - 07.06.2024	
9	Складання презентації	08.06.2024 - 12.06.2024	

Дата видачі завдання 01 квітня 2024 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Фесенко Т. Г.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 66 с., 20 рисунків, 1 додатки, 41 джерел.

КОНТЕНТ-АНАЛІЗ, ШТУЧНИЙ ІНТЕЛЕКТ, РЕКУРЕНТНА НЕЙРОННА МЕРЕЖА, НАЇВНИЙ БАЙЕС, ВЕБ-ПАРСІНГ, МУНІЦИПАЛІТЕТ, ВЕБ-ЗАСТОСУНОК, КЛАСИФІКАТОР ДАНИХ

У кваліфікаційній роботі представлено метод контент-аналізу веб-сайтів муніципалітетів із використанням алгоритмів штучного інтелекту. Основною метою дослідження є розробка та впровадження системи, яка автоматизує процес аналізу контенту веб-сайтів муніципалітетів із використанням машинного навчання.

У ході виконання кваліфікаційної роботи були розглянуті та проаналізовані різні інструменти з аналізу даних на основі штучного інтелекту, проведено дослідження існуючих методів контент-аналізу. Відповідно з встановленою задачею, розглянуто всі переваги та недоліки таких інструментів з обробки текстової інформації. Розроблено програмний застосунок, який виступає ефективним інструментом для контент-аналізу веб-сайтів муніципалітетів, що може бути корисним для муніципальних органів влади.

ABSTRACT

Master’s thesis: 66 pages, 20 figures, 1 appendices, 41 sources.

CONTENT ANALYSIS, ARTIFICIAL INTELLIGENCE, RECURRENT NEURAL NETWORK, NAIVE BAYES, WEB PARSING, MUNICIPALITY, WEB APPLICATION, DATA CLASSIFIER

This qualification paper presents the method of content analysis of websites of municipalities using artificial intelligence algorithms. The main purpose of the research is to analyze public opinion, complaints and general mood expressed in reviews and comments on websites of municipalities.

In the course of the qualification work, various data analysis tools based on artificial intelligence were considered and analyzed, research was conducted on existing methods of content analysis. In accordance with the established task, all the advantages and disadvantages of such tools for processing textual information are considered. A software application has been developed that acts as an effective tool for monitoring and analyzing public opinion, which can be useful for municipal authorities.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ 7
ВСТУП 8
1 ОГЛЯД ІСНУЮЧИХ МЕТОДІВ КОНТЕНТ-АНАЛІЗУ ВЕБ-САЙТІВ ТА ОГЛЯД ВИКОРИСТАННЯ АЛГОРИТМІВ ШТУЧНОГО ІНТЕЛЕКТУ В АНАЛІЗІ ДАНИХ 9
1.1 Існуючі методи контент-аналізу веб-сайтів 9
1.2 Застосування AI для аналізу рівня задоволеності громадян веб-послугами муніципалітету 15
1.3 Тенденції розвитку контент-аналізу за допомогою алгоритмів AI 20
2 ТЕХНОЛОГІЇ АНАЛІЗУ КОНТЕНТУ ВЕБ-РЕСУРСІВ 23
2.1 Особливості застосування нейронних мереж для аналізу контенту 23
2.2 Особливості застосування Теореми Байеса для аналізу тексту 33
3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАСТОСУНКУ 36
3.1 Опис використаних технологій 36
3.2 Web API 41
3.3 Клієнтський веб-застосунок 42
3.4 Реалізація методу контент-аналізу 43
4 ОПИС РОЗРОБЛЕНОГО ЗАСТОСУНКУ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ 49
4.1 Опис застосунку 49
ВИСНОВКИ 53
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 55
ДОДАТОК А ГРАФІЧНИЙ МАТЕРІАЛ КВАЛІФІКАЦІЙНОЇ РОБОТИ 61

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

AI – artificial intelligence

EHR – electronic health record

NLP – Natural Language Processing

RNN – Recurrent Neural Network

LSTM – Long Short-Term Memory

API – application programming interface

UI – User Interface

SDK – software development kit

LINQ – Language Integrated Query

ORM – Object-Relational Mapping

HTTP – HyperText Transfer Protocol

CLI – Command-Line Interface

URL – Uniform Resource Locator

ML – machine learning

ООП – об'єктно-орієнтоване програмування

SPA – Single Page Application

DI – Dependency Injection

MVC – Model-View-Controller

CSV – Comma-Separated Values

JSON – JavaScript Object Notation

ВСТУП

У сучасному інформаційному суспільстві веб-сайти муніципалітетів відіграють ключову роль забезпечення доступу громадян до інформації про діяльність державних органів на місцевому рівні. Вони є важливим інструментом взаємодії між владою та населенням, забезпечуючи прозорість та відкритість діяльності місцевих органів управління.

Однак, зі зростанням кількості інформації, що представлена на веб-сайтах муніципалітетів, виникає потреба в ефективних методах аналізу цієї інформації. Традиційні методи ручного аналізу стають неефективними через обсяг даних та їх динамічність. У зв'язку з цим виникає необхідність у розробці та застосуванні методів контент-аналізу, що дозволяють систематизувати та аналізувати інформацію на веб-сайтах муніципалітетів.

Актуальність даного дослідження обумовлена необхідністю оперативного та точного аналізу зворотного зв'язку від громадян для прийняття обгрунтованих управлінських рішень. Розробка методів контент-аналізу на основі алгоритмів штучного інтелекту сприяє покращенню комунікації між громадянами та муніципалітетами, підвищенню прозорості та підзвітності місцевих органів влади.

Метою даного дослідження є розробка та застосування методу контент-аналізу веб-сайтів муніципалітетів із використанням алгоритмів штучного інтелекту. Розробка автоматизованої системи контент-аналізу з використанням алгоритмів штучного інтелекту дозволить значно підвищити швидкість та точність обробки великої кількості інформації, що є особливо важливим в умовах сучасного інформаційного суспільства.

1 ОГЛЯД ІСНУЮЧИХ МЕТОДІВ КОНТЕНТ-АНАЛІЗУ ВЕБ-САЙТІВ ТА ОГЛЯД ВИКОРИСТАННЯ АЛГОРИТМІВ ШТУЧНОГО ІНТЕЛЕКТУ В АНАЛІЗІ ДАНИХ

AI революціонізує спосіб аналізу вмісту, полегшуючи та прискорюючи розуміння та використання даних для прийняття обґрунтованих рішень. У цьому розділі ми дослідимо роль штучного інтелекту в аналізі контенту та те, як він змінює наш підхід до контент-маркетингу, досліджень і аналізу. AI означає здатність машин імітувати людський інтелект, включаючи здатність вчитися, міркувати та адаптуватися. У контексті аналізу контенту AI можна використовувати для аналізу та інтерпретації різних типів контенту, наприклад тексту, зображень і відео.

Аналіз вмісту – це процес вивчення та інтерпретації різних типів вмісту для отримання значущої інформації та визначення закономірностей або тенденцій. Це можна зробити вручну або за допомогою технологій, таких як AI. Іншими словами, аналіз контенту передбачає використання різноманітних інструментів і методів для розуміння контексту та значення контенту, який потім можна використовувати для низки цілей, таких як маркетинг, дослідження та прийняття рішень.

Таким чином, поєднання штучного інтелекту та аналізу контенту дозволяє ефективно й точно інтерпретувати великі обсяги даних, дозволяючи компаніям і окремим особам отримати цінну інформацію про поведінку споживачів, ринкові тенденції тощо. З чітким розумінням цих термінів ми тепер можемо досліджувати переваги використання штучного інтелекту в аналізі контенту та те, як він змінює наш підхід до контент-маркетингу , досліджень і аналізу.

1.1 Існуючі методи контент-аналізу веб-сайтів

У сучасних дослідженнях активно застосовуються різні методи контент-аналізу веб-сайтів, спрямовані на аналіз структури та змісту інформації. Наприклад, метод тематичного аналізу спрямован на виділення основних тематик та категорій інформації на веб-сайтах. Зазвичай використовується визначення основних напрямів діяльності організації чи установи з урахуванням аналізу текстів контенту.

Структурний аналіз дозволяє вивчати організацію та взаємозв'язки між різними розділами та сторінками веб-сайту. Він включає аналіз ієрархії сторінок, систем навігації і взаємозв'язків між різними елементами сайту.

Наступний – аналіз частотності та ключових слів. Даний метод заснований на аналізі частотності слів і фраз на веб-сайті. Він допомагає виявити ключові теми та терміни, які використовуються на сайті, а також визначити найбільш релевантні ключові слова для пошукової оптимізації.

Семантичний аналіз спрямований на вивчення значень слів та фраз на веб-сайті. Він дозволяє визначити смислові зв'язки між різними елементами контенту та виявити семантичні групи слів.

Метод аналіз користувача поведінки ґрунтується на аналізі дій користувачів на веб-сайті, таких як кліки, перегляди сторінок, час перебування та інші метрики. Він допомагає зрозуміти, як користувачі взаємодіють із контентом та навігацією сайту.

Метод аналіза медіа-контенту спрямований на аналіз медіа-контенту на веб-сайті, такого як зображення, відео та аудіофайли. Він дозволяє визначити основні теми та характеристики медіа-контенту, що використовується на сайті.

Існує кілька переваг використання AI в аналізі контенту. Однією з найбільших переваг є те, що штучний інтелект може обробляти величезні обсяги даних набагато швидше, ніж коли-небудь могла б людина. Це означає, що аналіз вмісту можна виконувати в масштабі, надаючи цінну інформацію про великі набори даних, для аналізу яких вручну в іншому випадку знадобилося б багато часу.

Ще одна перевага штучного інтелекту в аналізі контенту полягає в здатності визначати закономірності та тенденції в даних, які можуть бути не відразу очевидними аналітику. Алгоритми штучного інтелекту можуть виявляти кореляції та закономірності, які люди можуть пропустити, і можуть надавати більш точні та надійні результати.

Штучний інтелект також може допомогти з аналізом настроїв, який передбачає аналіз емоційного тону частини контенту. Це може бути корисним для розуміння того, як споживачі ставляться до бренду, продукту чи послуги, і може допомогти компаніям відповідним чином адаптувати свої повідомлення.

Крім того, штучний інтелект може допомогти у створенні та куруванні контенту, допомагаючи визначати теми та теми, які резонують з аудиторією. Це може допомогти компаніям створювати більш привабливий контент, який з більшою ймовірністю поширюватиметься та переглядатиметься цільовою аудиторією.

Загалом, використання AI в аналізі контенту має численні переваги. Використовуючи можливості штучного інтелекту, компанії та окремі особи можуть отримати цінну інформацію про поведінку споживачів, ринкові тенденції тощо, що дозволить їм приймати більш обґрунтовані рішення та випереджати конкурентів.

AI може відіграти вирішальну роль у вдосконаленні стратегій контент-маркетингу. Через велику кількість контенту, що створюється в Інтернеті, підприємствам стає все важче виділитися та привернути увагу цільової аудиторії. Ось тут AI може допомогти.

Штучний інтелект може допомогти, аналізуючи дані про поведінку та уподобання клієнтів. Аналізуючи взаємодію клієнтів із попереднім контентом, алгоритми штучного інтелекту можуть визначити, які типи контенту є найбільш ефективними для залучення клієнтів і збільшення конверсій. Потім цю інформацію можна використати для створення більш цілеспрямованого та ефективного контенту, який резонуватиме з цільовою аудиторією.

Штучний інтелект також може допомогти з оптимізацією контенту, аналізуючи дані про продуктивність контенту та надаючи рекомендації щодо покращення. Наприклад, алгоритми штучного інтелекту можуть аналізувати такі показники, як рейтинг кліків, час, проведений на сторінці, і показники відмов, щоб визначити, які частини вмісту є найефективнішими, а які, можливо, потребують певних коригувань. Це може допомогти компаніям оптимізувати свій контент для максимального впливу та залучення.

Крім того, AI може допомогти з персоналізацією контенту, аналізуючи дані про вподобання та інтереси клієнтів. Аналізуючи дані про попередні взаємодії з брендом, алгоритми штучного інтелекту можуть визначити, які типи контенту найімовірніше сподобаються окремим клієнтам, і створити персоналізовані рекомендації щодо контенту.

Загалом штучний інтелект має потенціал для революції в стратегіях контент-маркетингу, надаючи цінну інформацію про поведінку клієнтів, уподобання та показники ефективності. Використовуючи можливості штучного інтелекту, компанії можуть створювати більш ефективний і привабливий контент, який резонуватиме з цільовою аудиторією та стимулюватиме розвиток бізнесу.

AI можна використовувати для аналізу широкого діапазону типів вмісту, включаючи текст, зображення, аудіо та відео. Кожен тип вмісту пропонує свій унікальний набір проблем і можливостей для аналізу.

Аналіз тексту, мабуть, найпоширеніша форма аналізу контенту. Алгоритми штучного інтелекту можна використовувати для аналізу великої кількості тексту, наприклад відгуків клієнтів, публікацій у соціальних мережах і статей новин, для виявлення закономірностей, тенденцій і настроїв. Потім цю інформацію можна використовувати для покращення маркетингових стратегій, обслуговування клієнтів та інших бізнес-процесів.

Аналіз зображень — це ще одна сфера, де штучний інтелект можна використовувати для отримання цінної інформації. Аналізуючи візуальний вміст зображень, алгоритми штучного інтелекту можуть ідентифікувати

шаблони та теми, які можуть бути пропущені аналітиками. Наприклад, штучний інтелект можна використовувати для ідентифікації об’єктів, людей і місць на зображеннях і навіть для визначення емоцій і настроїв.

Аналіз аудіо – це ще одна сфера застосування AI. Аналізуючи шаблони мовлення та визначаючи ключові фрази, алгоритми штучного інтелекту можуть допомогти компаніям краще зрозуміти своїх клієнтів і покращити обслуговування клієнтів. Аудіо-аналіз також можна використовувати для розпізнавання мовлення та перекладу, що може допомогти компаніям спілкуватися з клієнтами мовою, яку вони віддають перевагу.

Нарешті, аналіз відео – це сфера, де штучний інтелект стає все більш важливим. Аналізуючи візуальний і звуковий вміст відео, алгоритми штучного інтелекту можуть отримати цінну інформацію про поведінку та уподобання споживачів. Наприклад, штучний інтелект можна використовувати для визначення найбільш привабливих частин відео, що може допомогти компаніям оптимізувати свій вміст для максимального впливу та залучення.

Загалом штучний інтелект можна використовувати для аналізу широкого діапазону типів контенту, що дозволяє компаніям отримати цінну інформацію про поведінку, уподобання та настрої клієнтів. Використовуючи можливості штучного інтелекту, компанії можуть приймати більш обґрунтовані рішення та випереджати конкурентів.

SemEHR - універсальна система семантичного пошуку для виведення семантичних даних із клінічних записів для індивідуального догляду, набору випробувань і клінічних досліджень.

Розблокування даних, що містяться як у структурованих, так і в неструктурованих компонентах електронних медичних записів (EHR), має потенціал для забезпечення поетапної зміни даних, доступних для використання у вторинних дослідженнях, створення дієвих медичних ідей, управління лікарнею та набору дослідників. Щоб досягти цього, було впроваджено SemEHR, інструмент семантичного пошуку та аналітики з відкритим кодом для EHR.

SemEHR реалізує загальну інфраструктуру вилучення інформації (IE) та пошуку шляхом визначення контекстуалізованих згадок широкого спектру біомедичних концепцій у EHR. Анотації обробки природної мови додатково збираються на рівні пацієнта та доповнюються знаннями, що стосуються ЕЗД, щоб створити часову шкалу для кожного пацієнта. Семантичні дані обслуговуються через інтерфейси пошуку та аналітики на основі онтології.

SemEHR було розгорнуто або знаходиться в процесі розгортання в ряді систем NHS Trust EHR, включаючи Південний Лондон і Модслі, лікарню Королівського коледжу, лікарні Університетського коледжу Лондона та лікарню Гая. Результати та відгуки з кількох прикладів використання SemEHR показали його ефективність у автоматизації тривалих ручних завдань без шкоди для точності. Запити повертаються з достатньою швидкістю, щоб забезпечити можливість ітеративного адаптування для досягнення високої специфічності. Крім того, згідно з нашими тематичними дослідженнями в SLaM, SemEHR досяг точності, подібної до спеціально розроблених програм NLP, створених на TextHunter. Завдяки системі, що базується на онтологічній семантиці, дослідники можуть використовувати семантично пов’язані поняття для покращення результатів, наприклад, у дослідженні захворювань печінки на основі CRIS включення 8 препаратів, які використовуються для лікування захворювань печінки, допомогло знайти більше пацієнтів.

Тематичні дослідження показують, що створення уніфікованої структури, як-от SemEHR, реалізує більш економічно ефективний підхід до вирішення типових проблем IE та значно знижує бар’єр для дослідників, програмістів і клініцистів у доступі до знань, які містяться в неструктурованих клінічних нотатках. SemEHR має великий потенціал у забезпеченні ефективного та ефективного вторинного використання EHR для покращення медичних послуг. Крім того, системи, подібні до SemEHR, ініціюють платформу для спільного навчання, як пропагують Мозлі та ін., що дозволяє проводити дослідження у кооперативний спосіб, а не залишати ресурси в ізольованих бункерах.

SemEHR надає різні погляди на пацієнтів з метою представлення більш безперервного представлення графіка лікування пацієнта. Такі погляди можуть виявити дослідникам або клініцистам проблеми з якістю даних, щоб можна було вжити необхідних заходів, перш ніж робити висновки. Наприклад, подовжнє подання документів дає швидкий огляд того, наскільки повною та детальною є EHR пацієнта, що допомагає ідентифікувати пацієнтів, які мають неповні записи та яких, можливо, потрібно вилучити з досліджень. Однак проблеми з якістю даних, такі як неповнота, невідповідність і неточність даних, необхідно вирішувати систематично; інформування користувачів про можливі проблеми – це лише перший крок.

Результати численних тематичних досліджень демонструють ефективність SemEHR: у деяких випадках тижні або місяці роботи можна виконати протягом годин або хвилин. SemEHR надає більш повне уявлення про пацієнтів, приносячи більше та несподіване розуміння порівняно з орієнтованими на дослідження індивідуальними системами IE.

1.2 Застосування AI для аналізу рівня задоволеності громадян веб-послугами муніципалітету

Аналіз настроїв – це сфера, де штучний інтелект має значне застосування. Аналіз настроїв передбачає аналіз емоційного тону частини вмісту, наприклад твіту, огляду продукту чи публікації в блозі, щоб визначити, чи є він позитивним, негативним чи нейтральним. Це може бути цінним для підприємств і організацій, оскільки може допомогти їм краще зрозуміти, як цільова аудиторія сприймає їхній бренд, продукти чи послуги.

Один із способів використання штучного інтелекту в аналізі настроїв – алгоритми обробки природної мови (NLP). Алгоритми NLP можна навчити розпізнавати конкретні слова та фрази, які вказують на позитивні чи негативні настрої, а потім можуть призначити оцінку настрою фрагменту вмісту на основі частоти цих слів і фраз. Це можна зробити у великих масштабах,

дозволяючи підприємствам швидко й ефективно аналізувати великі обсяги відгуків клієнтів.

Інше застосування штучного інтелекту в аналізі настроїв – виявлення емоцій. Алгоритми виявлення емоцій можуть аналізувати тон голосу або вираз обличчя людини, щоб визначити її емоційний стан. Це можна використовувати, наприклад, для вимірювання задоволеності клієнтів під час взаємодії з клієнтами або для аналізу емоційного впливу певної маркетингової кампанії.

AI також можна використовувати для багатомовного аналізу настроїв. Використовуючи машинний переклад і алгоритми NLP, компанії можуть аналізувати настрої в різних мовах і культурах, що дає їм змогу краще зрозуміти настрої своєї глобальної клієнтської бази.

Загалом штучний інтелект має значні застосування в аналізі настроїв, дозволяючи компаніям отримати цінну інформацію про настрої клієнтів і відповідно покращити свої продукти, послуги та маркетингові стратегії. Використовуючи можливості штучного інтелекту, компанії можуть випереджати конкурентів і надавати більш персоналізований, привабливий і ефективний контент своїй цільовій аудиторії.

Однією з ключових переваг використання штучного інтелекту в аналізі контенту є його здатність визначати закономірності та тенденції у великих обсягах даних. Це особливо цінно для компаній, які хочуть отримати інформацію про поведінку та уподобання клієнтів, а також визначити можливості для зростання та вдосконалення.

Алгоритми штучного інтелекту можна навчити виявляти шаблони та тенденції в широкому діапазоні типів вмісту, включаючи текст, зображення та відео. Наприклад, штучний інтелект можна використовувати для визначення найбільш часто використовуваних ключових слів у відгуках клієнтів або публікаціях у соціальних мережах, а також для аналізу частоти та контексту, у якому ці ключові слова використовуються. Це може надати цінну інформацію про настрої, уподобання та больові точки клієнтів, які потім можна

використовувати для розробки маркетингових стратегій і розробки продукту.

AI також можна використовувати для виявлення закономірностей і тенденцій у візуальному вмісті, наприклад зображеннях і відео. Наприклад, штучний інтелект можна використовувати для виявлення повторюваних тем або мотивів у зображеннях або для аналізу візуальних елементів, які найбільше приваблюють глядачів. Потім цю інформацію можна використовувати для створення більш візуально привабливого та цікавого вмісту, який резонуватиме з цільовою аудиторією.

Крім того, штучний інтелект можна використовувати для виявлення закономірностей і тенденцій у різних каналах контенту, таких як трафік соціальних мереж, електронної пошти та веб-сайтів. Аналізуючи закономірності та тенденції в цих каналах, підприємства можуть отримати цілісне уявлення про поведінку та уподобання клієнтів, які можна використовувати для розробки маркетингових стратегій і розробки продуктів.

Загалом, роль штучного інтелекту у визначенні моделей і тенденцій у контенті полягає в тому, щоб дозволити компаніям отримати цінну інформацію про поведінку та уподобання клієнтів, а також визначити можливості для зростання та вдосконалення. Використовуючи можливості штучного інтелекту, компанії можуть випереджати конкурентів і надавати більш персоналізований, привабливий і ефективний контент своїй цільовій аудиторії.

AI значно впливає на те, як створюється та керується контент. Однією з ключових переваг використання штучного інтелекту для створення контенту є те, що він дозволяє компаніям створювати більш персоналізований і привабливий контент, який резонує з цільовою аудиторією. Аналізуючи дані та поведінку клієнтів, алгоритми штучного інтелекту можуть визначати типи контенту, які, швидше за все, зацікавлять різні сегменти цільової аудиторії, а потім можуть створювати контент, адаптований до цих уподобань.

AI також можна використовувати для автоматизації певних аспектів створення контенту, таких як створення заголовків або підписів або навіть

написання цілих статей чи описів продуктів. Це може заощадити час і ресурси компаній, а також дозволить їм створювати більший обсяг високоякісного вмісту.

Крім створення контенту, AI також впливає на керування контентом. Аналізуючи поведінку та уподобання клієнтів, алгоритми штучного інтелекту можуть визначати найбільш релевантний і привабливий контент для певної аудиторії, а потім підбирати цей контент відповідно до їхніх інтересів. Це може бути особливо цінним для компаній, які прагнуть залучити свою цільову аудиторію доречним і актуальним вмістом.

Загалом, вплив штучного інтелекту на створення та керування контентом полягає в тому, щоб дозволити компаніям створювати та керувати більш персоналізованим, привабливим та ефективним контентом, який резонує з їхньою цільовою аудиторією. Використовуючи можливості штучного інтелекту, компанії можуть випереджати конкурентів і надавати своїм клієнтам більш цінний і релевантний контент, з часом сприяючи залученню та лояльності.

Хоча існує багато потенційних переваг використання штучного інтелекту в аналізі контенту, є також кілька проблем і обмежень, які необхідно враховувати. Однією з основних проблем є якість даних, що аналізуються. Щоб алгоритми штучного інтелекту могли надавати точну та надійну інформацію, їх потрібно навчити на високоякісних даних, які репрезентують цільову аудиторію. Якщо дані неповні, неточні або упереджені, уявлення, надані алгоритмами штучного інтелекту, можуть бути помилковими або неповними.

Іншою проблемою є складність самих алгоритмів. Алгоритми штучного інтелекту настільки ж ефективні, наскільки ефективні дані та правила, які використовуються для їх навчання, і може бути важко переконатися, що алгоритми є достатньо надійними та точними, щоб надавати надійну інформацію.

Існують також етичні міркування, які слід взяти до уваги, зокрема щодо

питань конфіденційності та безпеки даних. Використовуючи штучний інтелект для аналізу даних клієнтів, компанії повинні переконатися, що вони роблять це таким чином, щоб поважати конфіденційність клієнтів і захищати їхні дані від неправомірного використання або крадіжки.

Варто зазначити, що хоча AI може бути неймовірно потужним, він не є панацеєю для всіх потреб аналізу вмісту. Завжди будуть ситуації, коли для повного розуміння та інтерпретації даних потрібні людська інтуїція та креативність, і важливо знайти баланс між використанням штучного інтелекту як інструменту для покращення прийняття рішень людиною та покладенням на нього як заміну людського досвіду.

Загалом потенційні проблеми та обмеження штучного інтелекту в аналізі контенту необхідно ретельно розглядати та керувати ними, щоб переконатися, що розуміння, яке надають алгоритми AI, є точним, надійним і етичним. Завдяки цьому компанії можуть використовувати потужність штучного інтелекту, щоб отримати цінну інформацію про поведінку та уподобання клієнтів, а також зберегти довіру та прозорість зі своїми клієнтами.

Використання штучного інтелекту в аналізі контенту викликає низку етичних міркувань, про які підприємства та організації повинні знати. Деякі з найважливіших міркувань включають проблеми конфіденційності, упередженості та прозорості.

Однією з ключових проблем є конфіденційність даних, що аналізуються. Щоб надати точну та релевантну інформацію, алгоритмам штучного інтелекту часто потрібен доступ до великих обсягів даних клієнтів, які можуть включати особисту інформацію. Компанії мають бути прозорими для своїх клієнтів щодо того, які дані збираються, як вони використовуватимуться та як вони будуть захищені.

Ще одним етичним міркуванням є можливість упередженості в алгоритмах, що використовуються. Алгоритми штучного інтелекту неупереджені настільки ж, наскільки об’єктивні дані, на яких вони

навчаються, і якщо дані містять упередження або неточності, ці упередження можуть бути посилені алгоритмами. Це може призвести до небажаних наслідків, як-от несправедливе націлювання на певні групи клієнтів або увічнення стереотипів.

Прозорість також є важливим фактором, коли йдеться про використання AI в аналізі контенту. Клієнти мають право знати, як використовуються їхні дані, і компанії повинні бути прозорими щодо алгоритмів, які вони використовують, як вони працюють і інформації, яку вони надають. Це може допомогти зміцнити довіру та прозорість з клієнтами, що важливо для підтримки довгострокових відносин.

Компаніям необхідно враховувати потенційний вплив їхніх зусиль з аналізу контенту на ширшу спільноту. Алгоритми штучного інтелекту можуть мати непередбачені наслідки, і компанії повинні вжити заходів, щоб гарантувати, що вони не завдають шкоди окремим особам або групам. Це може включати взаємодію із зацікавленими сторонами та широким співтовариством, щоб краще зрозуміти потенційний вплив їхніх зусиль з аналізу вмісту та вжити заходів для пом’якшення будь-яких негативних наслідків.

Таким чином, етичні міркування при використанні штучного інтелекту для аналізу контенту є важливими для побудови довіри та прозорості з клієнтами, уникнення небажаних наслідків і забезпечення того, що підприємства діють відповідально та етично. Ретельно розглядаючи ці проблеми, компанії можуть використовувати можливості штучного інтелекту, щоб отримати цінну інформацію про поведінку та уподобання клієнтів, дотримуючись при цьому своїх етичних і соціальних обов’язків.

1.3 Тенденції розвитку контент-аналізу за допомогою алгоритмів AI

Майбутнє штучного інтелекту в аналізі контенту яскраве, на горизонті багато захоплюючих розробок та інновацій. Оскільки технології штучного

інтелекту продовжують розвиватися, компанії зможуть використовувати ці технології, щоб отримати ще глибше розуміння поведінки та уподобань клієнтів, а також створювати більш персоналізовані та ефективні стратегії контент-маркетингу.

Однією з ключових сфер, де AI, ймовірно, матиме великий вплив, є сфера обробки природної мови (NLP). NLP — це галузь штучного інтелекту, яка зосереджена на аналізі та розумінні людської мови, і вона стає все більш складною завдяки прогресу в машинному навчанні та алгоритмах глибокого навчання. З удосконаленням технології NLP компанії зможуть використовувати її для аналізу та розуміння більш складних форм мови, таких як сарказм, іронія та метафори, що дозволить їм ще глибше зрозуміти настрої та уподобання клієнтів.

Штучний інтелект – аналіз зображень і відео. Оскільки алгоритми штучного інтелекту стануть більш вправними у розпізнаванні та інтерпретації візуальних даних, підприємства зможуть використовувати цю технологію для аналізу вмісту зображень і відео, а також для отримання цінної інформації про поведінку та уподобання клієнтів.

Вплив штучного інтелекту на промисловість також, ймовірно, буде значним. Оскільки підприємства стають більш вправними у використанні штучного інтелекту для аналізу та інтерпретації даних клієнтів, вони зможуть створювати більш персоналізовані та ефективні маркетингові стратегії, адаптовані до конкретних потреб і вподобань окремих клієнтів. Це дозволить підприємствам побудувати міцніші стосунки зі своїми клієнтами, а також підвищити лояльність і утримання клієнтів.

Крім того, AI, ймовірно, матиме великий вплив на такі галузі, як охорона здоров’я, фінанси та роздрібна торгівля, де здатність аналізувати та інтерпретувати великі обсяги даних стає все більш важливою. Використовуючи штучний інтелект для аналізу даних пацієнтів, наприклад, постачальники медичних послуг зможуть визначати моделі та тенденції в поведінці пацієнтів, що дозволить їм надавати більш персоналізовану та

ефективну допомогу. Подібним чином у галузях фінансів і роздрібної торгівлі штучний інтелект дозволить підприємствам отримати глибше розуміння поведінки та уподобань клієнтів, що дозволить їм приймати більш обґрунтовані рішення щодо розробки продуктів, маркетингу та стратегій продажів.

2 ТЕХНОЛОГІЇ АНАЛІЗУ КОНТЕНТУ ВЕБ-РЕСУРСІВ

Сучасні муніципалітети стикаються з необхідністю обробки великих обсягів текстової інформації, що надходить з різних веб-ресурсів, таких як соціальні мережі, офіційні сайти та онлайн-форуми. Аналіз цього контенту дозволяє виявити громадські настрої, ідентифікувати важливі проблеми та покращити комунікацію з мешканцями. У цьому розділі розглядаються технології аналізу текстового контенту, які можуть бути використані муніципалітетами, зокрема нейронні мережі та алгоритми Байеса. Ці методи дозволяють ефективно обробляти та інтерпретувати текстові дані, що сприяє прийняттю обґрунтованих рішень та підвищенню рівня обслуговування громадян.

2.1 Особливості застосування нейронних мереж для аналізу контенту

Використання методів глибокого навчання для класифікації тексту стало актуальним сьогодні, оскільки це може ефективно заощадити ресурси для отримання інформації. У цій галузі було розроблено нейронні мережі, які є обчислювальними математичними моделями, що імітують біологічні нейронні мережі. На рисунку 2.1 представлена базова модель нейрона, яка становить штучні нейронні мережі. Штучні нейронні мережі абстрагують симуляцію нейронів людського мозку з боку обробки інформації та моделюють зв’язки між нейронами мозку шляхом створення кількох моделей мікронейронів. Принцип його побудови полягає в тому, щоб імітувати структуру та склад біологічних нейронних мереж, створених через зв’язки між великою кількістю нейронів і нейронних синапсів. Ця модель може виконувати глибоке навчання та аналіз текстових даних, завдяки чому досягається більш точна та ефективна класифікація тексту.

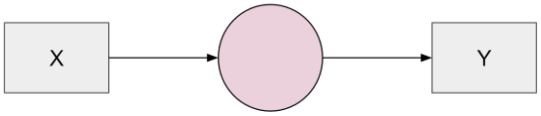


Рисунок 2.1 – Базова модель нейрона

Розробка моделей штучних нейронних мереж почалася з простих нейронних моделей, де кожен персептрон є простою бінарною лінійною моделлю класифікації. Персептрон обробляє вхідні дані шляхом лінійного перетворення та має структуру, подібну до біологічних нейронів, включаючи активований і неактивний стани. Нейронні мережі можна розглядати як складні мережеві моделі, що складаються з кількості персептронів. Поєднуючи кілька персептронів для формування різних структур нейронних мереж, нейронні мережі можуть гнучко вирішувати різні складні проблеми.

У нейронних мережах кожен персептрон є основною нейронною одиницею з такими атрибутами, як введення, вага та функція активації. Регулюючи значення ваги з’єднання між нейронами, нейронні мережі можуть навчатися та адаптуватися до різних режимів введення, таким чином вирішуючи такі завдання, як класифікація та розпізнавання вхідних даних. Персептрон є основою моделей нейронних мереж, тоді як нейронні мережі є складними моделями, що складаються з певної кількості персептронів, які представлені на рисунку 2.2.

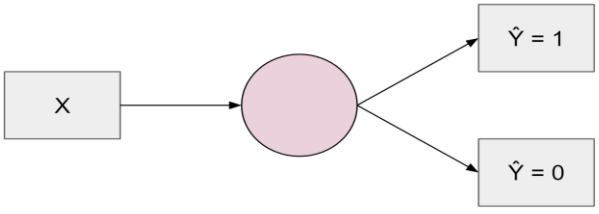


Рисунок 2.2 – Перцептрон

Регулюючи вагові значення з’єднань і використовуючи різні функції активації, нейронні мережі можуть обробляти та вивчати різні вхідні дані. На рисунку 2.3 показана базова модель прямої нейронної мережі, де сигнал поширюється в один бік від вхідного рівня до вихідного.

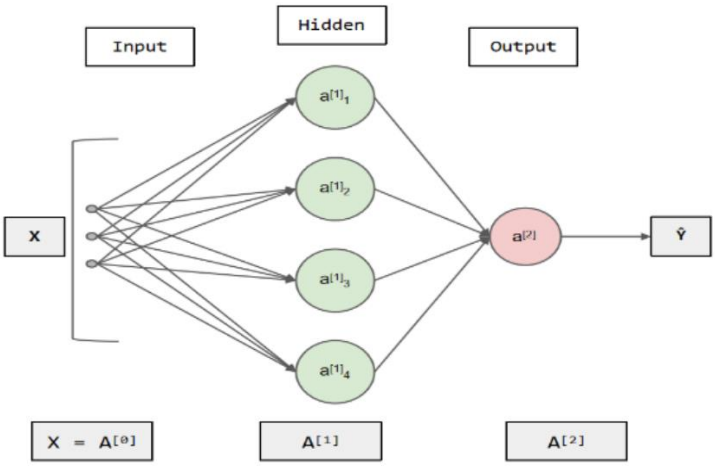


Рисунок 2.3 – Нейронна мережа прямого зв'язку

В останні роки, у зв’язку зі зростанням попиту на масивний аналіз текстових даних і розвитком та інноваціями глибокого навчання та структур нейронних мереж, технологія аналізу контенту на основі нейронних мереж стає ключовою технологією. На рисунку 2.4 показана модель глибокої нейронної мережі, яка має кілька прихованих шарів.

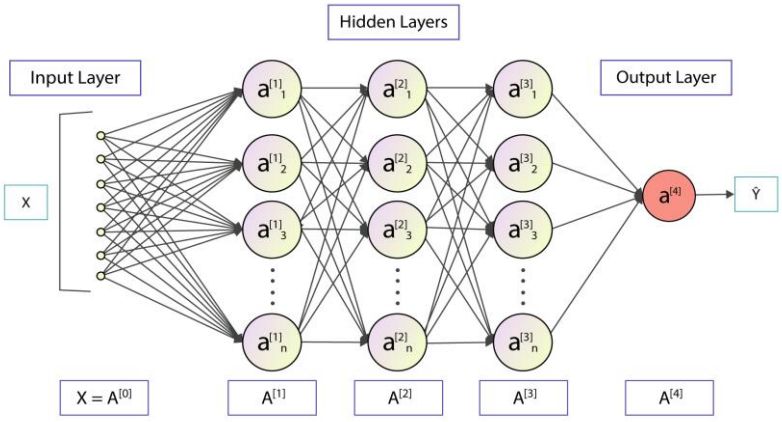


Рисунок 2.4 – Глибока нейронна мережа

Традиційна нейронна мережа не може забезпечити безперервність пам’яті. Традиційна нейронна мережа є мережею прямого зв’язку, кожен вхід обробляється незалежно, і немає пам’яті або передачі контекстної інформації. Це призводить до деяких проблем в обробці даних послідовності, таких як пропуск інформації. Щоб удосконалити традиційну нейронну мережу, дослідники запропонували рекурентні нейронні мережі. Рекурентна нейронна мережа (RNN) вводить механізм блоку пам’яті, який робить мережу здатністю пам’яті та показаний на рисунок 2.5. Прихований стан RNN може захоплювати попередню інформацію та передавати її в наступний стан, щоб реалізувати пам'ять і передача інформації.

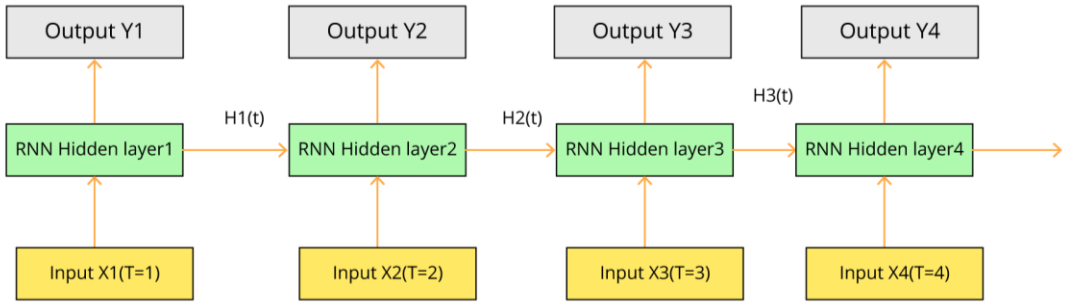


Рисунок 2.5 – Структура рекурентній нейронній мережі

У рекурентній нейронній мережі вхід і попередній прихований стан комбінуються в новий тензор, який потім обробляється через функцію активації для створення нового стану пам'яті, як показано на рисунку 2.6. Рекурентна нейронна мережа зазвичай використовує tanh() як функцію активації. Функція цієї активаційної функції полягає в тому, щоб допомогти відрегулювати розмір значення, щоб вихідне значення знаходилось в інтервалі (-1, 1), тим самим регулюючи вихід нейронної мережі.

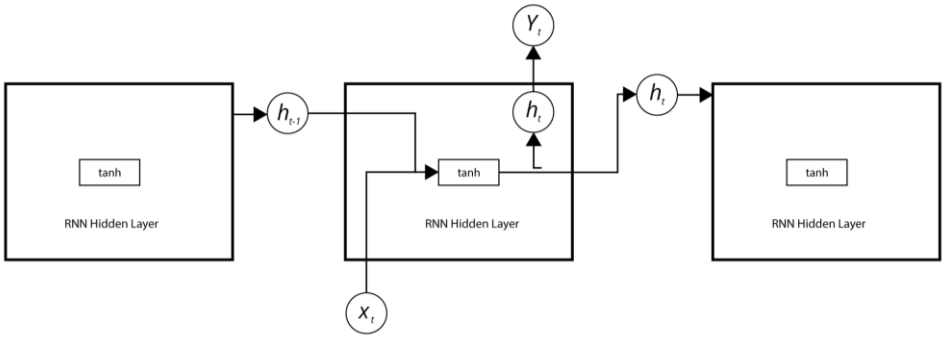


Рисунок 2.6 – Архітектура рекурентної нейронної мережі

$$\boldsymbol{h_t = tanh(W_{hx}X_t + W_{hh}h_{t-1} + b_h)} \ldots \quad (2.1)$$

$$\boldsymbol{Y_t = f(W_0 h_t + b_y)} \ldots \quad (2.2)$$

Конкретний внутрішній процес розповсюдження може бути виражений наступним чином: по-перше, вхід і попередній прихований стан будуть об'єднані в новий тензор. Потім цей тензор пройде через функцію активації, щоб створити новий прихований стан. Далі цей новий прихований стан буде передано до наступної комірки, а потім об’єднано з новим введенням для формування нового тензора, повторюючи описані вище кроки.

Завдяки цьому внутрішньому процесу розповсюдження рекурентна нейронна мережа може запам’ятовувати інформацію в послідовності текстових даних і використовувати її для моделювання. Рекурентна нейронна мережа може ефективно обробляти послідовність даних за допомогою комбінації вхідних і прихованих станів, а також ролі функцій активації та внутрішніх процесів розповсюдження, і має сильні передбачувані здібності.

У рекурентній нейронній мережі зазвичай використовується функція активації tanh() або relu(). Функція активації може підвищити здатність мережі та полегшити роботу мережі зі складними даними. Рекурентна нейронна мережа має деякі проблеми при обробці даних послідовності. Однією з них є

недолік короткочасної пам’яті, також відомий як проблема довгострокової залежності. Коли дані послідовності досить довгі, рекурентній нейронній мережі важко перенести інформаційну пам’ять із раннього стану в пізній, що призводить до легкого пропуску важливої інформаційної пам’яті.

Іншою проблемою є зникнення або вибух градієнта під час зворотного поширення. Зникнення градієнта відноситься до явища, коли значення градієнта падає надто мало, що призводить до того, що мережа не може ефективно навчатися та тренуватися. Вибух градієнта спричинений надмірним зростанням значень градієнта, що призводить до швидкого оновлення параметрів мережі та неможливості конвергенції до відповідних значень.

Усі ці проблеми впливатимуть на процес навчання рекурентної нейронної мережі, оскільки рекурентна нейронна мережа має тривалий і безперервний процес передачі інформації. Незважаючи на те, що рекурентна нейронна мережа має такі проблеми, як дефіцит короткочасної пам’яті та зникнення/розрив градієнта, її продуктивність і ефективність обробки даних послідовності можна покращити за допомогою деяких вдосконалених моделей і механізмів.

У 1997 році Hochreiter і Schmidhuber запропонували мережу довготривалої короткочасної пам'яті. Довга короткочасна пам'ять (LSTM) – це особливий тип нейронних мереж. Впроваджуючи механізм стробування, довготривала короткочасна пам’ять ефективно усуває проблему короткочасної пам’яті в традиційній нейронній мережі.

Рекурентна нейронна мережа — це проста рекурентна структура, яка має сильну здатність зберігати інформацію та запам’ятовувати, але не має здатності фільтрувати та відбирати інформацію.

Таким чином, рекурентна нейронна мережа буде зберігати всю отриману інформацію та обробляти її без різниці. Довга короткочасна пам'ять буде вибірково зберігати інформацію, оскільки вона має сильну здатність. У нього є ліберний пристрій, і він може вибирати скільки завгодно. На рисунку 8 показана комірка довгострокової пам'яті. Коротше кажучи, LSTM є більш

гнучким, ефективним і точним, ніж RNN. Завдяки впровадженню механізму стробування та оптимізації структури LSTM може досягти кращої продуктивності в різних завданнях.

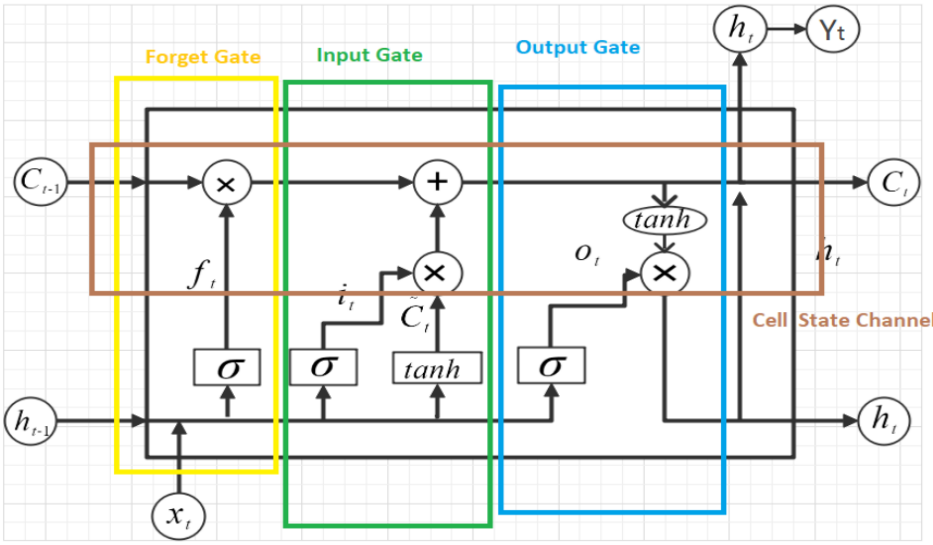


Рисунок 2.7 – Архітектура LSTM

Довга короткочасна пам'ять є потужною моделлю нейронної мережі, яка вводить структуру стробування для керування додаванням або видаленням інформації в стані клітини. Ця структура стробування складається з функції активації та операції множення точки. Зокрема, LSTM в основному включає наступні частини.

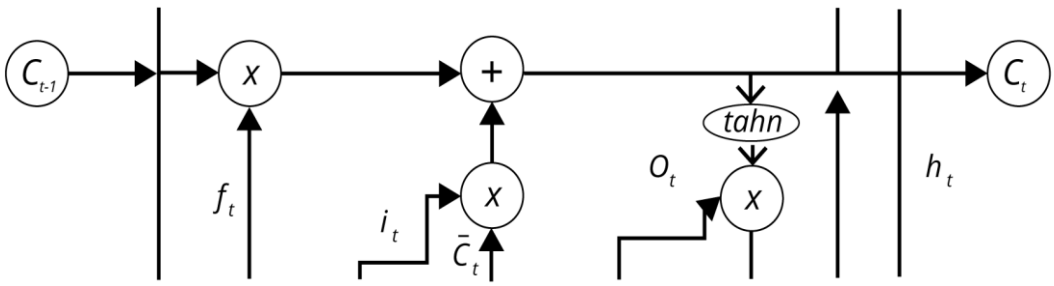


Рисунок 2.8 – Структура каналу стану комірки LSTM

$$C_t = f_t \times C_{t-1} + i_t \times InputGateC_t \ldots \quad (2.3)$$

Перший – це канал стану комірки, який включає дві лінійні операції, показані на рисунку 2.8. Перша лінійна операція використовується для керування пропорцією попередньої інформації комірки, а друга лінійна операція використовується для збільшення нової інформації, яку приносить вхід. Результати цих двох лінійних операцій використовуватимуться для оновлення стану комірки. Ми множимо тензор, створений попереднім вентилем забуття, і тензор зі станом основної клітинки, переданим попередньою клітинкою, а потім додаємо тензор, створений вхідним вентилем, щоб отримати новий тензор. Канал стану клітинки відповідає за передачу та збереження інформації протягом усього процесу. Він контролює потік і збереження інформації через налаштування цих воріт.

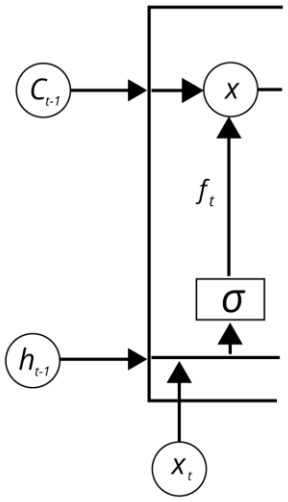


Рисунок 2.9 – Структура шлюза забуття

$$f_t = \sigma\left(W_f h_{t-1} + W_f X_t + b_f\right) \ldots \quad (2.4)$$

Другий — шлюз забуття, який виводить значення від 0 до 1 на основі

попереднього прихованого стану та поточного введення та показано на рисунку 2.9. Якщо вихідне значення дорівнює 0, це означає повне забуття історичної інформації, переданої раніше; Якщо вихідне значення дорівнює 1, це означає повне збереження переданої раніше інформації історії. Існування пропуску забуття може допомогти нейронній мережі уникнути надмірності та плутанини інформації. У довготривалій короткочасній пам'яті шлюз забуття є важливою частиною. Його роль полягає в тому, щоб вирішити, зберегти чи відкинути раніше отриману інформацію. Коли якась інформація в нейронній мережі збуджується функцією активації sigmoid(), її вихідне значення буде в діапазоні (0,1). Таким чином, існування воріт забуття може допомогти нейронній мережі краще навчатися та адаптуватися до різних вхідних даних. Розумно налаштувавши функцію активації, ефект нейронної мережі можна покращити.

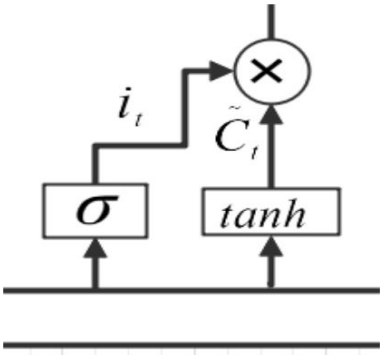


Рисунок 2.10 – Структура вхідного шлюза

$$\boldsymbol{i_t = \sigma(W_i \times h_{t-1} + W_i X_t + b_i)} \ldots \tag{2.5}$$

$$\boldsymbol{InputGateC_t = tanh(W_C \cdot h_{t-1} + W_C x_t + b_C)} \ldots \tag{2.6}$$

Третя частина — це вхідний шлюз, який також виводить дві відомості на основі попереднього прихованого стану та поточного введення: один

представляє, скільки інформації додано, а інший представляє силу інформації. Ця інформація буде використана для оновлення статусу комірки. Вхідний вентиль керує введенням нової інформації та визначає, яку інформацію слід додати до каналу стану комірки. На рисунку 2.10 показано структуру входу.

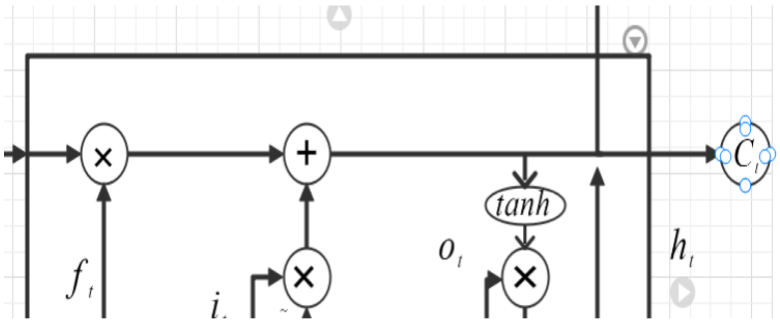


Рисунок 2.11 – Структура шлюза оновлення

Четверта частина — це шлюз оновлення, який виводить наступний стан комірки на основі попереднього стану комірки, шлюз забуття, шлюз введення та нову інформацію. Шлюз оновлення показаний на рисунку 2.11. Існування цієї структури стробування може допомогти нейронним мережам навчатися та міркувати ефективніше.

$$\boldsymbol{o_t = \sigma(W_o h_{t-1} + W_o x_t + b_o)}. \tag{2.7}$$

$$\boldsymbol{h_t = o_t * tanh(C_t)}... \tag{2.8}$$

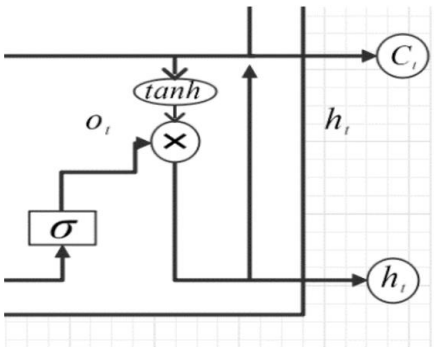


Рисунок 2.12 – Структура вхідного шлюза

Останньою частиною є вихідний шлюз, який додає вихідний шлюз на основі стану клітинки виводу шлюзу оновлення, щоб контролювати, скільки інформації виводиться як рівень наступного стану. Наявність цієї частини може допомогти нейронній мережі більш гнучко справлятися з різними вимогами завдань. На рисунку 2.12 показано структуру вихідного вентиля. Основна структура довготривалої короткочасної пам’яті складається з вхідних воріт, забутих воріт, вихідних воріт, воріт оновлення та каналу стану комірки. Ці ключові компоненти контролюють введення, збереження, вихід і передачу інформації за допомогою різних механізмів, щоб мережа довготривалої короткочасної пам’яті могла ефективно обробляти послідовні дані та полегшувати довгострокову залежність і проблему градієнта.

2.2 Особливості застосування Теореми Байеса для аналізу тексту

В основі даного методу лежить теорема Байєса для розрахунку ймовірності будь-якої події за умови, що сталося інше пов'язане з ним подія. Використовуючи формулу Байєса, можна розрахувати ймовірність події, беручи до уваги, як події, що вже відбулися, так і майбутні. Вона виглядає наступним чином:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2.9}$$

Де,

P(A) — безумовна ймовірність того, що подія A сталася, за відсутністю зворотної інформації, також це називають апріорною ймовірністю;

P(A|B) — ймовірність того, що подія A сталася з умовою, що також відбулася подія B (апостеріорна ймовірність);

P(B|A) — ймовірність настання події B при істинності гіпотези A;

P(B) — повна ймовірність настання події B.

У машинному навчанні існує ціла родина класифікаторів, заснованих на теоремі Байєса і припущенні, що ознаки об'єктів, які ми класифікуємо, не залежать один від одного. Вперше дані метод знайшов своє застосування ще у 1950 роках, його використовували для класифікації документів, де ознаками були частоти слів. Плюс цього алгоритму в тому, що ми можемо практично необмежено збільшувати кількість ознак, що наближатиме точність прогнозу до таких популярних методів, як метод опорних векторів.

Принцип роботи байєсовського методу нічим не відрізняється від інших класифікаторів, ми відносимо спостереження до того чи іншого класу з ознаки векторів. Але робимо це з важливим припущенням, вважаючи, що кожен ознака не пов'язані з іншими і впливає класифікацію. Якщо розглядати простий байєсовський класифікатор, то в його основі лежить навчання з вчителем. Також перевагою є те, що йому потрібний невеликий набір вхідні дані для навчання. Тому незважаючи на те, що його недарма називають «наївним», завдяки малоймовірному твердженню незалежності ознак, він уже довгий час утримує місце серед самих точних класифікаторів.

Ймовірна модель, що лежить в основі алгоритму. Є безліч подій (спостережень) $x = (x_1, x_2, \dots, x_n)$. Алгоритм співвідносить кожному спостереженню умовну ймовірність $p(C_k|x_1, x_2, \dots, x_n)$, де C_k-це клас. Скориставшись теоремою Байєса:

$$p(C_k|x_1, x_2, \dots, x_n) = \frac{p(C_k)p(x|C_k)}{p(x)} \tag{2.10}$$

Найбільший інтерес викликає чисельник, оскільки знаменник не бере участь у задачі класифікації та буде константою. Завдяки припущення, що це ознаки незалежні:

$$p(C_k|x_1, x_2, \dots, x_n) = p(C_k)p(x_1|C_k)p(x_2|C_k) \dots p(x_n|C_k) = \prod_n p(x_i|C_k) \quad (2.11)$$

Виходить простий байєсовський класифікатор надає кожному значенню мітку приналежності до класу, тобто. $y = C_k$ виглядає наступним чином:

$$y = arg_k max_{1\dots k} \prod_n p(x_i|C_k) \quad (2.12)$$

Клас y вибирається таким чином, щоб максимізувати функцію правдоподібності, яка є твір умовних ймовірностей ознаки x_i.

Наївний Байєс передбачає клас з найбільшою умовною ймовірністю для вектор ознак x.

3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАСТОСУНКУ

Для реалізації програмного застосунку було обрано сучасні технології. Під час розробки використовувалась .Net Core платформа. Для розробки аналізу контенту було використано ML.Net. Код був написаний мовою програмування C#.

Готовий проект складається з декількох частин, а саме: Web API та UI застосунку. Завдяки клієнтському застосунку дані передаються на веб-сервер. Також було розроблено истему, яка використовує нейронні мережі LSTM, для аналізу контенту веб-сайту муніципалітету. Клієнтський застосунок було реалізовано використовуючи фреймворк Angular. У файлах з розширенням .json зберігаються дані для тестування (20% записів від загальної кількості записів) та навчання (80% - від загальної кількості записів).

3.1 Опис використаних технологій

.NET Core – це кросплатформений, відкритий фреймворк для розробки різноманітних додатків. Він підтримує розробку на різних операційних системах, таких як Windows, Linux та macOS. Основні переваги .NET Core включають: можливість розробки та запуску додатків на різних операційних системах; оптимізований для швидкої роботи та високої ефективності; додатки можуть бути розгорнуті як у контейнерах, так і на серверах без необхідності повної установки фреймворку; дозволяє розробникам вивчати вихідний код, вносити власні зміни та використовувати готові рішення від спільноти; підтримка сучасних мов програмування, таких як C#, F# та VB.NET, а також різноманітних бібліотек і фреймворків для побудови веб, мобільних, десктопних та хмарних додатків.

.NET Core складається з кількох основних частин, які разом забезпечують його функціональність і гнучкість для розробки різноманітних

додатків. Основні частини .NET Core включають:

1) NET Runtime – основний компонент, який відповідає за виконання додатків;

2) .NET Libraries - колекція бібліотек, які надають багатий набір API для розробки додатків;

3) .NET SDK (Software Development Kit) — це набір інструментів і бібліотек, необхідних для розробки додатків на платформі .NET. SDK надає всі необхідні компоненти для створення, компіляції, налагодження та публікації додатків;

4) ASP.NET Core — це кросплатформений, високопродуктивний фреймворк для створення сучасних веб-додатків та веб-API;

5) Entity Framework Core - сучасний, кросплатформений ORM для роботи з базами даних. Забезпечує високорівневий API для доступу до даних з підтримкою LINQ (Language Integrated Query);

6) CLI (Command-Line Interface) - інтерфейс командного рядка для керування життєвим циклом додатків .NET. Забезпечує команди для створення нових проектів, відновлення залежностей, компіляції коду, запуску додатків і тестування.

7) ML.NET — це кросплатформений фреймворк з відкритим вихідним кодом від Microsoft, призначений для створення моделей машинного навчання (ML) у додатках .NET. Він дозволяє розробникам .NET інтегрувати машинне навчання без необхідності переходу на інші мови програмування або платформи. ML.NET підтримує різноманітні сценарії машинного навчання, такі як класифікація (визначення категорій для даних, наприклад, спам/не спам), регресія (прогнозування числових значень, наприклад, ціна будинку), кластеризація (групування подібних елементів, наприклад, сегментація клієнтів), рекомендації (надання персоналізованих рекомендацій, наприклад, рекомендації товарів), аналіз тексту (обробка і аналіз текстових даних, наприклад, виявлення настроїв) та виявлення аномалій (виявлення нетипових патернів у даних, наприклад, виявлення шахрайства).

ML.NET включає в себе функціонал автоматизованого машинного навчання (AutoML), що дозволяє автоматично обирати кращі моделі та гіперпараметри для ваших даних. Плавна інтеграція з існуючими додатками .NET дозволяє розробникам використовувати знайомі інструменти та мови програмування, такі як C# і F#. ML.NET підтримує формат ONNX (Open Neural Network Exchange), що забезпечує сумісність із іншими фреймворками машинного навчання, такими як TensorFlow і PyTorch. Фреймворк оптимізований для високої продуктивності та здатний працювати з великими наборами даних, а також може взаємодіяти з різними джерелами даних, включаючи SQL Server, текстові файли, Azure Blob Storage та інші.

ML.NET надає можливість розширення та кастомізації, що дозволяє розробникам додавати власні алгоритми та моделі. Інструменти для розробників, такі як Model Builder у Visual Studio, допомагають створювати моделі машинного навчання за допомогою візуального інтерфейсу, а інтерфейс командного рядка (CLI) дозволяє швидко створювати, тренувати та оцінювати моделі. ML.NET надає розробникам .NET потужні інструменти для інтеграції машинного навчання у свої додатки, забезпечуючи високу продуктивність і гнучкість у роботі з даними та моделями.

C# – це сучасна, об'єктно-орієнтована мова програмування, розроблена компанією Microsoft. Вона є частиною платформи .NET і призначена для створення різноманітних додатків, включаючи веб-додатки, десктопні програми, ігри, мобільні додатки та хмарні сервіси. C# поєднує простоту синтаксису з потужними можливостями, що робить її легкою у вивченні для новачків і водночас достатньо сильною для створення складних систем. Ця мова підтримує сучасні парадигми програмування, такі як об'єктно-орієнтоване програмування (ООП), узагальнене програмування та асинхронне програмування. Однією з ключових особливостей C# є інтеграція з .NET, що забезпечує доступ до великої кількості бібліотек і фреймворків, які полегшують розробку додатків. Завдяки управлінню пам'яттю через автоматичний збір сміття (garbage collection) C# дозволяє зосередитися на

логіці програми, а не на управлінні ресурсами. C# широко використовується у розробці бізнес-додатків, а також у галузях, де потрібна висока продуктивність та надійність. Мова постійно розвивається, додаючи нові можливості та вдосконалення, щоб відповідати сучасним вимогам до розробки програмного забезпечення.

Angular – платформа для розробки веб-додатків, створена і підтримувана компанією Google. Вона дозволяє створювати складні, масштабовані та інтерактивні веб-додатки з використанням TypeScript, який є надмножиною JavaScript. Angular забезпечує структуру для розробки односторінкових додатків (SPA), де більшість логіки та рендерингу виконується на стороні клієнта. Однією з ключових особливостей Angular є модульність, що дозволяє розробникам розділяти додаток на невеликі, повторно використовувані компоненти. Це спрощує розробку, тестування та обслуговування коду. Angular також включає в себе потужну систему прив'язки даних (data binding), яка автоматично синхронізує модель даних і користувацький інтерфейс.

Angular використовує Dependency Injection (DI) для управління залежностями, що підвищує модульність та тестованість коду. Фреймворк забезпечує вбудовану підтримку для розробки форм, маршрутизації, обробки запитів HTTP і анімацій, що зменшує необхідність у використанні сторонніх бібліотек.

Angular CLI (Command Line Interface) – інструмент командного рядка, який спрощує створення, налагодження та розгортання Angular-додатків. CLI автоматизує багато рутинних завдань, таких як генерація компонентів, служб і модулів, що дозволяє розробникам зосередитися на логіці додатка. Angular активно підтримується та розвивається спільнотою розробників і має велике екосистемне середовище, включаючи бібліотеки, інструменти та розширення, що допомагають у розробці високоякісних веб-додатків.

TypeScript – мова програмування, розроблена компанією Microsoft, яка є розширенням JavaScript з підтримкою статичного типізації. TypeScript дозволяє розробникам писати більш структурований і безпечний код,

виявляючи помилки на етапі компіляції і покращуючи продуктивність та читабельність коду.

Основною особливістю TypeScript є можливість визначення типів для змінних, параметрів функцій, об'єктів та інших конструкцій мови JavaScript. Це дозволяє визначити очікувані типи даних і зробити код більш зрозумілим для інших розробників та для самого розробника.

Крім того, TypeScript підтримує сучасні функції JavaScript, такі як стрілкові функції, розгортання об'єктів (spread operator), деструктуризацію та інші, що дозволяє використовувати нові можливості мови безпосередньо в TypeScript.

TypeScript компілюється в JavaScript і може використовуватися в будь-якому проекті, де використовується JavaScript. Він інтегрується з популярними інструментами розробки, такими як Visual Studio Code, та має широку підтримку у веб-спільноті. Ця мова дозволяє розробникам писати більш безпечний і стабільний код, що полегшує процес розробки великих та складних проектів.

Git – це розподілена система керування версіями, призначена для ведення історії змін у програмному коді. Вона дозволяє розробникам зберігати, відслідковувати та об’єднувати зміни в програмному коді в різних гілках розробки. Git використовується для спільної роботи над проектами, керування конфліктами, відстеження змін та відкату до попередніх версій коду.

Однією з основних переваг Git є його розподілена природа, що означає, що кожен розробник має повний доступ до історії змін та може працювати офлайн. Крім того, Git має швидкий та ефективний алгоритм комітів, що дозволяє швидко зберігати зміни та перехід між версіями коду.

Git використовується в різних сферах розробки програмного забезпечення, включаючи веб-розробку, розробку мобільних додатків, вбудовані системи та інші. Він є стандартом для керування версіями в багатьох компаніях та проектах, оскільки дозволяє ефективно керувати різноманітними

командами розробників та великими обсягами коду.

3.2 Web API

Проект було розділено на декілька частини. Перша частина – веб-клієнт за допомогою якого у користувача є можливість передавати адресу сайту муніципалітету, контент якого потрібно проаналізувати та переглядати результати роботи алгоритму. Наступними частиною є Web API, задача якого полягає в обробці запитів клієнта, в виконанні аналізу контенту веб-сайту та відправки результатів.

Веб-сервер складається з клієнтського застосунку, який реалізовано як Single Page Application (SPA) сервіс, контролерів та сервісів, де описани методі з аналізу контенту. Структуру проекту зображена на рисунку 3.1.

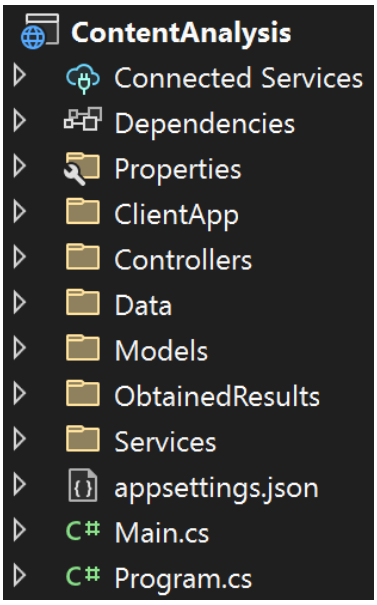


Рисунок 3.1 – Структура веб-сервера проекта

В директорії ClientApp розміщено клієнтський веб-застосунок. В наступній директорії Controllers знаходяться всі контролери для обробки запитів, які відправляє клієнтський застосунок. Сервіси для обробки даних та

виклику методів розташовані в директорії Services. Директорія ObtainedResults зберігає методи для обробки отриманих результатів з сервісів та конвертація у моделі (директорія Models), які будуть зрозумілі кінцевому користувачу на клієнтській частині.

Даний веб-застосунок, на додаток до фреймворку ASP.NET Core, використовує патерн Model-View-Controller (MVC). Цей патерн призначений для відокремлення бізнес-логіки застосунку від інтерфейсу користувача. Спочатку використовувався для розробки настільних GUI, але став популярним і в сфері веб-розробки. Багато популярних мов програмування мають власні фреймворки MVC, що спрощують впровадження цього шаблону.

3.3 Клієнтський веб-застосунок

Для розробки клієнтського веб-застосунку був обраний один з найпопулярніших та потужних фреймворків – Angular. Однією з переваг цього фреймворку є те, що код буде працювати в усіх популярних браузерах, а отже, на різних платформах. Звідси можна зробити висновок, що більша кількість користувачів матиме змогу використати систему для аналізу контенту веб-сайту муніципалітету. На рисунку 3.2 зображена структура клієнтського застосунку проекту.

Весь клієнтський код застосунку розміщено в директорії app, всі файли поза межами є конфігураційними. Всі компоненти розміщені в окремих директоріях, яких всього чотири: comparison, home, menu та analyze. В директорії models знаходяться інтерфейси, які є відображенням даних, отриманих і переданих до веб-сервера.

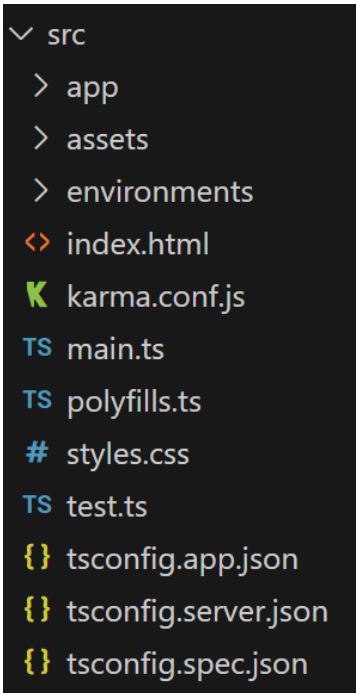


Рисунок 3.2 – Структура клієнтського застосунку

3.4 Реалізація методу контент-аналізу

Контент-аналіз включаю такі аспекти як:

1 структурний аналіз: вивчення структури сайту, його навігації та логіки організації контенту. Це включає перевірку того, наскільки легко користувачам знаходити необхідну інформацію;

2 аналіз змісту: вивчення текстового, графічного та мультимедійного контенту сайту щодо його актуальності, повноти, точності та відповідності потребам цільової аудиторії;

3 аналіз досвіду користувача (UX): оцінка зручності використання сайту, включаючи адаптивність під мобільні пристрої, розмір текстової інформації та наявність календаря подій;

4 аналіз доступності: перевірка доступності сайту для людей з обмеженими можливостями;

5 аналіз зворотного зв'язку: вивчення механізмів отримання зворотного зв'язку від користувачів, таких як форми зворотного зв’язку та наявності

соціальних мереж.

Для виконання структурного аналізу знадобиться парсинг HTML сторінок та оцінка їхньої структури та навігації. Для виконання цього завдання використовується бібліотека HtmlAgilityPack. Для визначення якості навігаційної панелі, будемо використовувати класифікаційну модель. Для цього потрібно розмітити набір даних веб-сторінок, вказавши, чи є вони прикладом хорошої або поганої навігації. Для цього було створено .json файл, який міститиме URL сторінки, HTML-розмітку та мітку (хороша або погана навігація і тд). Окрім мітки було додано оцінку критеріїв. Критерії структурного аналізу виступають: наявність основного меню – 30 балів, легкість доступа до головної сторінки – 20 балів, наявність хлібних крихт – 20 балів, наявність пошукової системи – 30 балів. Таким чином, кожен критерій оцінюватиметься від 0 до 100, і загальна оцінка обчислюватиметься як зважена сума цих критеріїв.

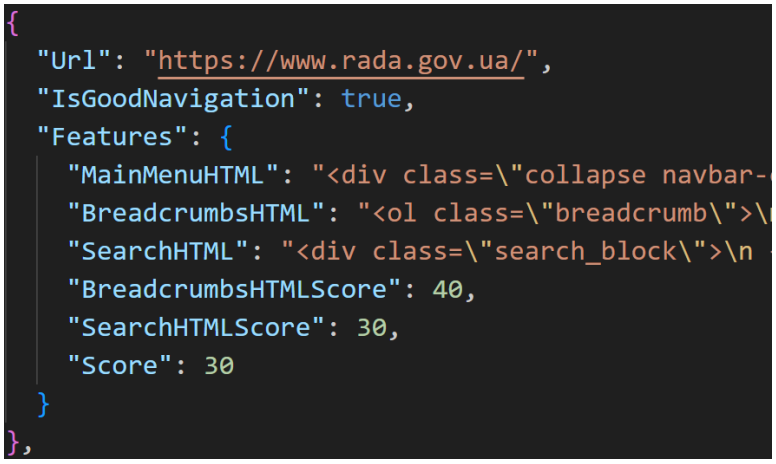


Рисунок 3.3 – Приклад тестових даних для тренування моделі

Нижче наведено реалізацію методу з навчанням моделі структурного аналізу веб-сайту на лістингу 3.1.

Лістинг 3.1 – Машинне навчання моделі для структурного аналізу

```
var data = mlContext.Data.LoadFromEnumerable(webPages);

            var pipeline =
mlContext.Transforms.Concatenate("Features",
                nameof(EvaluationData.HasMainMenu),
                nameof(EvaluationData.HasBreadcrumbs),
                nameof(EvaluationData.HasSearch))

.Append(mlContext.BinaryClassification.Trainers.LbfgsLogisticReg
ression(
                    labelColumnName:
nameof(WebPageData.IsGoodNavigation),
                    featureColumnName: "Features"));
            var model = pipeline.Fit(data);

            var predictions = model.Transform(data);
            var metrics =
mlContext.BinaryClassification.Evaluate(predictions,
labelColumnName: nameof(WebPageData.IsGoodNavigation));

            mlContext.Model.Save(model, data.Schema,
"navigation_model.zip");

            var predictionEngine =
mlContext.Model.CreatePredictionEngine<WebPageData,
NavigationPrediction>(model);
            var prediction =
predictionEngine.Predict(samplePage);
```

Для аналізу змісту веб-сторінки з точки зору актуальності контенту, а також відповідності потребам аудиторії було створено набор даних з оцінками для тренування моделі з наступними критеріями: актуальність веб-контенту (50 балів), повні контактні дані (30 балів), відповідність потребам аудиторії (20 балів).

Лістинг 3.2 – Навчання моделі для аналізу змісту

```
var contentData =
mlContext.Data.LoadFromEnumerable(contentDataWithFeatures);
            var pipeline =
mlContext.Transforms.Concatenate("Features",
                nameof(ContentRelevanceData.Score),
                nameof(ContentCompletenessData.Score),
                nameof(AudienceSuitabilityData.Score))

.Append(mlContext.BinaryClassification.Trainers.LbfgsLogisticReg
```

```
ression(
                    labelColumnName:
nameof(WebPageContentData.IsGoodContent),
                    featureColumnName: "Features"));
            var contentModel = pipeline.Fit(contentData);

            var contentPredictions =
contentModel.Transform(contentData);
            var contentMetrics =
mlContext.BinaryClassification.Evaluate(contentPredictions,
labelColumnName: nameof(WebPageContentData.IsGoodContent));
            mlContext.Model.Save(contentModel,
contentData.Schema, "content_model.zip");
            var contentPredictionEngine =
mlContext.Model.CreatePredictionEngine<WebPageContentData,
ContentPrediction>(contentModel);
            var contentPrediction =
contentPredictionEngine.Predict(sampleContentPage);
```

Для автоматизованого аналіз користувацького досвіду (UX) веб-сторінок, застосуєм ML.NET для навчання моделі на основі різних ознак, таких як розмір текстової інформації, адаптивність та наявність календаря подій. Для парсингу та аналізу CSS застосуємо бібліотеку AngleSharp.

Лістинг 3.3 – Навчання моделі для аналізу змісту користувацького досвіду

```
var dataView = mlContext.Data.LoadFromEnumerable(uxEvaluations);
            var pipeline =
mlContext.Transforms.Concatenate("Features",
                nameof(UXEvaluationFeatures.ClearHeadingsScore),

nameof(UXEvaluationFeatures.MobileAdaptationScore),

nameof(UXEvaluationFeatures.EventCalendarsScore))

.Append(mlContext.BinaryClassification.Trainers.LbfgsLogisticReg
ression(
                    labelColumnName:
nameof(WebPageUXEvaluation.IsGoodUX),
                    featureColumnName: "Features"));
            var model = pipeline.Fit(dataView);
            var predictions = model.Transform(dataView);
            var metrics =
mlContext.BinaryClassification.Evaluate(predictions,
labelColumnName: nameof(WebPageUXEvaluation.IsGoodUX));
            mlContext.Model.Save(model, dataView.Schema,
"ux_evaluation_model.zip");
```

```
            var predictionEngine =
mlContext.Model.CreatePredictionEngine<WebPageUXEvaluation,
UXEvaluationPrediction>(model);
        }
```

Для аналізу доступності, що включає текстову альтернативу до зображень, контрасту зображень та фону, а також наявність функцій для користувачів з обмеженими можливостями, будемо застосовувати бібліотеку HtmlAgilityPack та AngleSharp для аналізу HTML та CSS веб-сторінок.

Лістинг 3.4 – Навчання моделі для аналізу доступності

```
var dataView =
mlContext.Data.LoadFromEnumerable(accessibilityPages);
            var pipeline =
mlContext.Transforms.Concatenate("Features",

nameof(AccessibilityFeatures.TextAlternativesScore),
                nameof(AccessibilityFeatures.ContrastScore),

nameof(AccessibilityFeatures.AccessibilityFeaturesScore))

.Append(mlContext.BinaryClassification.Trainers.LbfgsLogisticReg
ression(
                    labelColumnName:
nameof(WebPageAccessibility.IsAccessible),
                    featureColumnName: "Features"));
            var model = pipeline.Fit(dataView);
            var predictions = model.Transform(dataView);
            var metrics =
mlContext.BinaryClassification.Evaluate(predictions,
labelColumnName: nameof(WebPageAccessibility.IsAccessible));

            mlContext.Model.Save(model, dataView.Schema,
"accessibility_model.zip");

            var predictionEngine =
mlContext.Model.CreatePredictionEngine<WebPageAccessibility,
AccessibilityPrediction>(model);

            var prediction =
predictionEngine.Predict(samplePage);
```

Аналіз зворотнього зв'язку розраховується на основі двох критеріїв: наявності форми зворотного зв'язку та посилань на соціальні мережі.

Лістинг 3.5 – Навчання моделі для аналізу зворотнього зв’язку

```
var dataView = mlContext.Data.LoadFromEnumerable(feedbackPages);
            var pipeline =
mlContext.Transforms.Concatenate("Features",
                nameof(FeedbackFeatures.FeedbackFormScore),
                nameof(FeedbackFeatures.SocialMediaScore))

.Append(mlContext.BinaryClassification.Trainers.LbfgsLogisticReg
ression(
                    labelColumnName:
nameof(WebPageFeedback.IsFeedbackAvailable),
                    featureColumnName: "Features"));
            var model = pipeline.Fit(dataView);
            var predictions = model.Transform(dataView);
            var metrics =
mlContext.BinaryClassification.Evaluate(predictions,
labelColumnName: nameof(WebPageFeedback.IsFeedbackAvailable));

            mlContext.Model.Save(model, dataView.Schema,
"feedback_model.zip");

            var predictionEngine =
mlContext.Model.CreatePredictionEngine<WebPageFeedback,
FeedbackPrediction>(model);
```

4 ОПИС РОЗРОБЛЕНОГО ЗАСТОСУНКУ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

В результаті виконання магістерської роботи було створено систему, яка здатна аналізувати контент веб-сайту муніципалітету. З метою демонстрації функціональності системи було розроблено два тестових проекту: Web-API та Angular-застосунок.

4.1 Опис застосунку

Після успішного завершення розробки та відсутності виявлених помилок було прийнято рішення про створення Angular-застосунку для зручної візуалізації результатів роботи системи. Схематичне представлення проекту наведено на рисунку 4.1

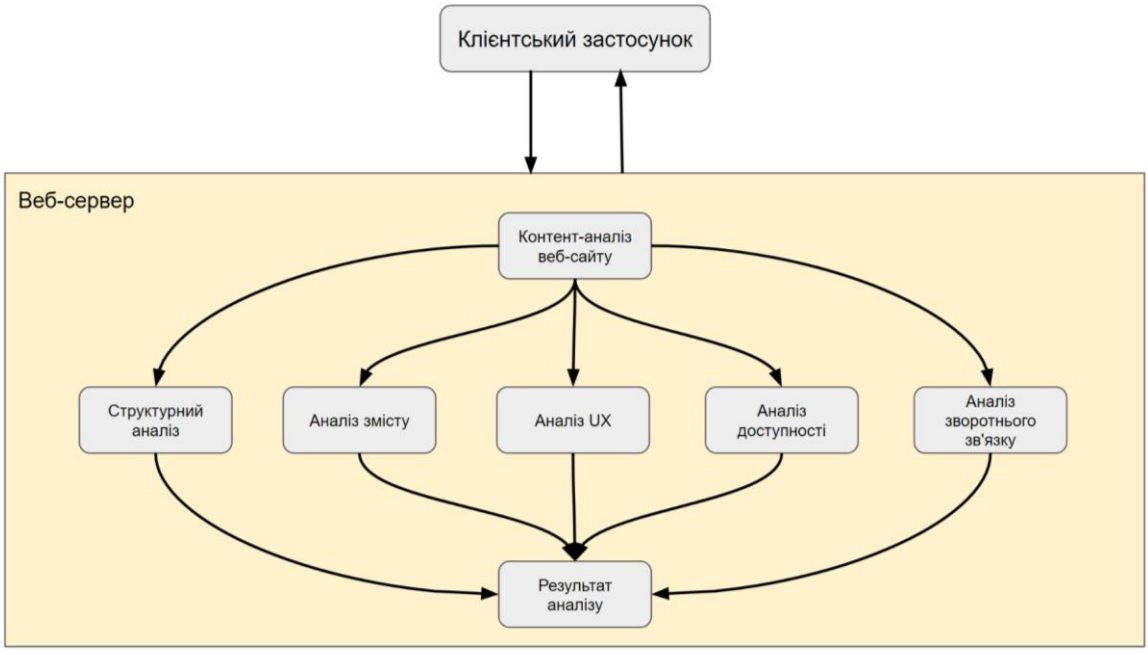


Рисунок 4.1 – Схема розробленого застосунку

Блок-схема процесу аналізу веб-сайту розділена на клієнтську та серверну частини. У верхній частині схеми знаходиться блок “Клієнтський застосунок”, який представляє клієнтську частину програми, з якою взаємодіє користувач. Далі, на серверній частині, показано обробку даних. Після чого повертаються результати аналізу на клієнтський застосунок.

Для того щоб перевірити метод контент-аналізу було обрано веб-сайт Харківської обласної ради (рисунок 4.2).

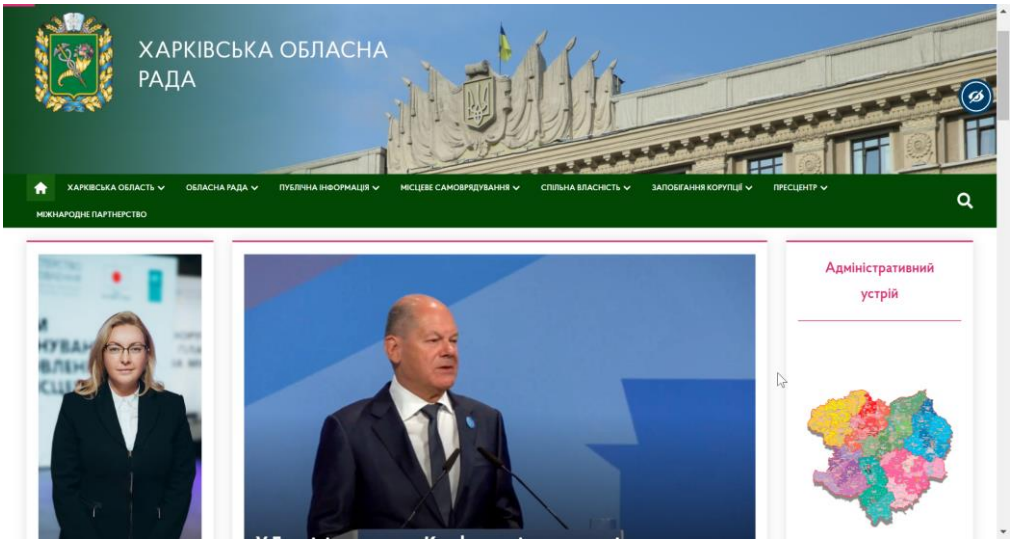


Рисунок 4.2 – Веб-сайт Харківської обласної ради для тестування роботи додатка

При активації розробленого застосунку на головному екрані користувач має можливість проаналізувати контент сайту муніципалітету. Для цього необхідно вести адресу сайту муніципалітету та обрати тип аналізу. Нижче на рисунку 4.3 можна побачити результат аналізу веб-сайту Харківської обласної ради.

Після проведеного аналізу, користувач бачить результати роботи алгоритмів з контент-аналізу. Загальна оцінка містить усереднений бал всіх показників. Списки показників містять результати по кожному з аналізів які було проведено.

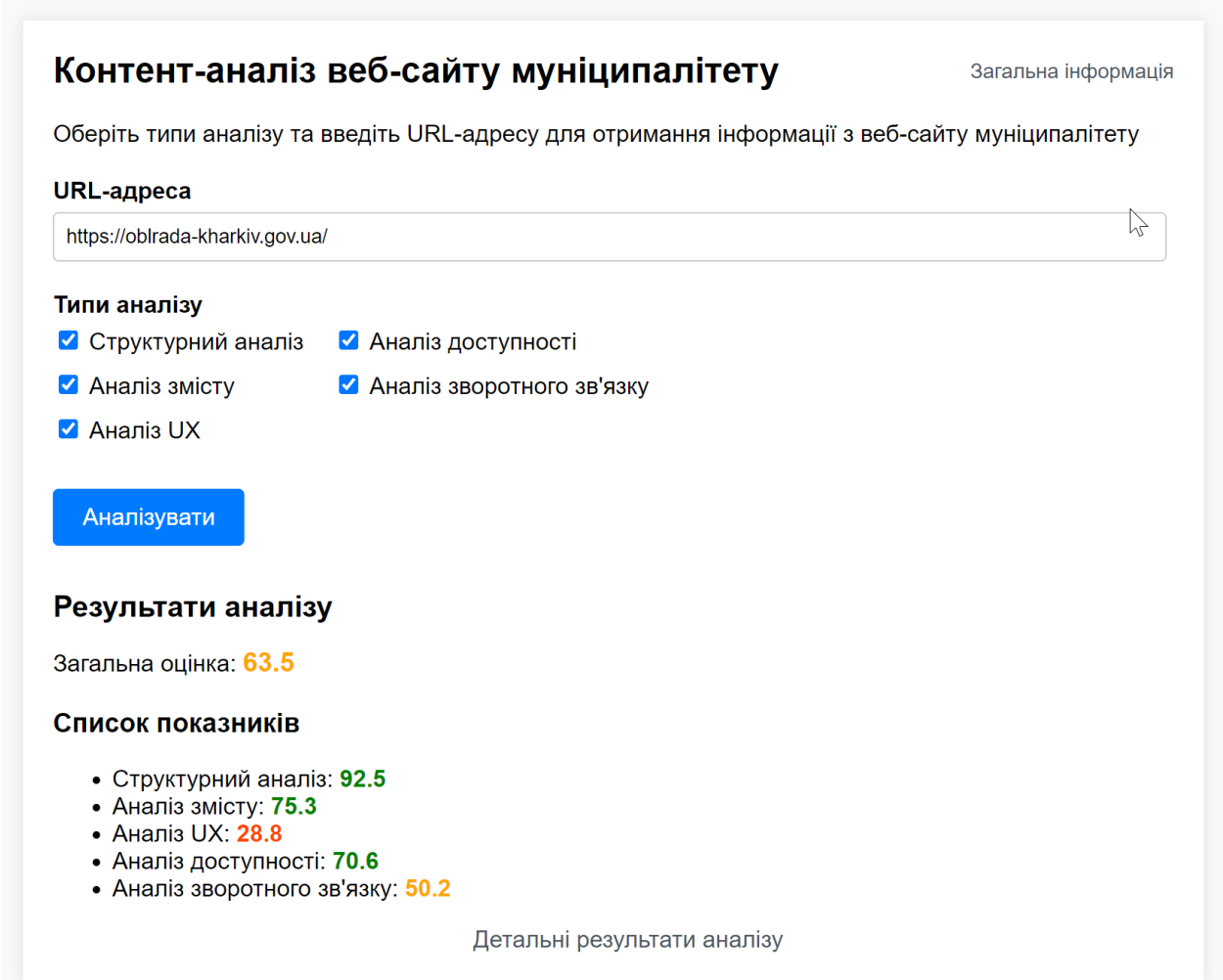
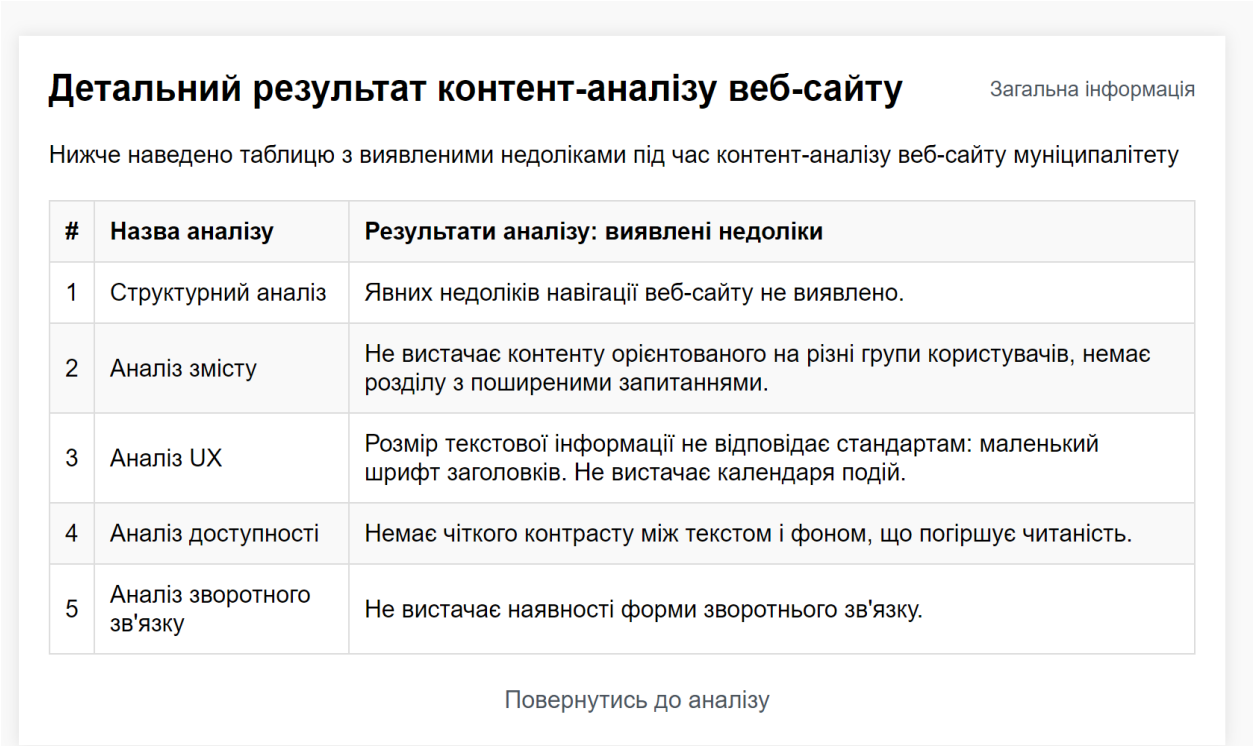


Рисунок 4.3 – Головна сторінка застосунку

Далі користувач може відкрити сторінку з детальними результатами аналізу. Сторінка детального аналізу містить таблицю в якій зазначено проведені аналізи та виявлені недоліки веб-сайта. Нижче на рисунку 4.3 наведено приклад сторінки детального аналізу.

Крім вищезазначених сторінок, є ще сторінка з загальною інформацією, яка містить види аналізу які розроблено та критерії згідно яких формується кінцевий результат. На рисунку 4.5 зображено сторінку з загальної інформацією.



Детальний результат контент-аналізу веб-сайту

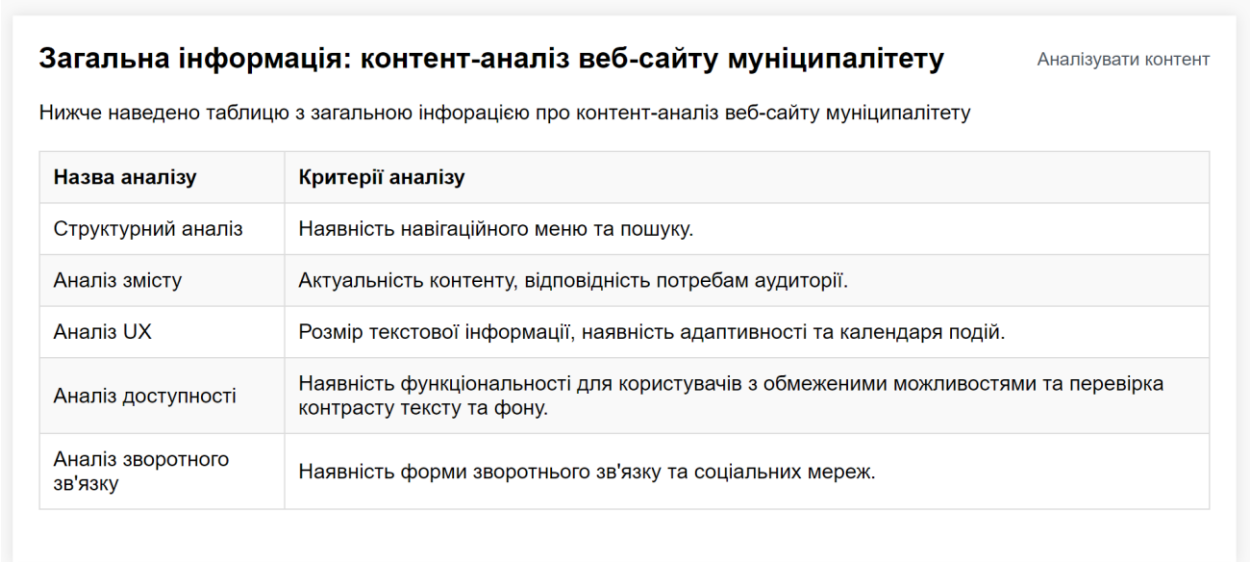
Загальна інформація

Нижче наведено таблицю з виявленими недоліками під час контент-аналізу веб-сайту муніципалітету

#	Назва аналізу	Результати аналізу: виявлені недоліки
1	Структурний аналіз	Явних недоліків навігації веб-сайту не виявлено.
2	Аналіз змісту	Не вистачає контенту орієнтованого на різні групи користувачів, немає розділу з поширеними запитаннями.
3	Аналіз UX	Розмір текстової інформації не відповідає стандартам: маленький шрифт заголовків. Не вистачає календаря подій.
4	Аналіз доступності	Немає чіткого контрасту між текстом і фоном, що погіршує читаність.
5	Аналіз зворотного зв'язку	Не вистачає наявності форми зворотнього зв'язку.

Повернутись до аналізу

Рисунок 4.4 – Сторінка з детальним результатом аналізу контенту



Загальна інформація: контент-аналіз веб-сайту муніципалітету

Аналізувати контент

Нижче наведено таблицю з загальною інфорацією про контент-аналіз веб-сайту муніципалітету

Назва аналізу	Критерії аналізу
Структурний аналіз	Наявність навігаційного меню та пошуку.
Аналіз змісту	Актуальність контенту, відповідність потребам аудиторії.
Аналіз UX	Розмір текстової інформації, наявність адаптивності та календаря подій.
Аналіз доступності	Наявність функціональності для користувачів з обмеженими можливостями та перевірка контрасту тексту та фону.
Аналіз зворотного зв'язку	Наявність форми зворотнього зв'язку та соціальних мереж.

Рисунок 4.5 – Сторінка з загальною інформацією

ВИСНОВКИ

В рамках кваліфікаційної роботи було розроблено та реалізовано систему для автоматичного аналізу якості веб-сайтів муніципалітетів з використанням методів машинного навчання. Основною метою дослідження було створення інструменту, здатного оцінювати різні аспекти веб-сайтів та надавати детальні оцінки, що дозволяє виявляти та покращувати проблемні галузі.

Для досягнення цієї мети було проведено збір даних, що включає отримання HTML-контенту веб-сайтів муніципалітетів. На основі цих даних були створені набори даних у форматі JSON, що містять HTML-фрагменти та мітки для навчання моделей машинного навчання.

Були розроблені та реалізовані алгоритми для аналізу різних аспектів веб-сайтів, таких як структурний аналіз, аналіз змісту, аналіз досвіду користувача (UX), аналіз доступності та аналіз зворотного зв'язку. Для кожного аспекту було створено методи оцінки, що дозволяють виділити ключові параметри та провести їх аналіз.

Використовуючи зібрані дані, були навчені моделі машинного навчання для прогнозування оцінок з різних аспектів. Моделі були валідовані, що підтвердило їхню точність і надійність.

Було розроблено веб-сервіс з API, що надає доступ до методів аналізу. Веб-сервіс дозволяє клієнтам, наприклад, веб-додаткам, надсилати запити на аналіз веб-сайтів та отримувати детальні оцінки.

Тестування розробленої системи на реальних даних веб-сайтів муніципалітетів показало її точність та ефективність. Це дозволило виявити як сильні сторони, і області подальшого поліпшення.

Навчені моделі показали високу точність у прогнозуванні оцінок різних аспектів веб-сайтів. Вони продемонстрували здатність детально оцінювати якість веб-сайтів, виявляючи як сильні сторони, так і області для покращення.

Розроблена система може бути адаптована для аналізу веб-сайтів в інших доменах або інших критеріїв оцінки, що підтверджує її універсальність. Система надає гнучкі можливості для розширення функціональності та додавання нових методів аналізу.

Практична цінність дослідження полягає у можливості автоматично оцінювати та покращувати якість веб-сайтів муніципалітетів, надаючи конкретні рекомендації. Впровадження системи може значно підвищити рівень обслуговування громадян, покращивши доступ до інформації та взаємодію з муніципальними службами через Інтернет.

Розроблена система контент-аналізу веб-сайтів муніципалітетів є потужним інструментом для автоматичного аналізу та покращення якості веб-ресурсів, забезпечуючи більш високий рівень зручності та доступності для користувачів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Хрустальов Є.К. (2024). Інструменти штучного інтелекту для створення контенту веб-ресурсів. Матеріали 28-го міжнародного молодіжного форуму. Харків: Т.5, 70. doi: https://doi.org/10.30837/IYF.PCEIP.2024.070

2. Андрусенко Ю.О. & Фесенко Т.Г. (2023). Нестаціонарність ресурсів та послуг хмарної інфраструктури. Системи управління, навігації та зв'язку. Збірник наукових праць. Полтава: ПНТУ, Т.4(74), 129–133. doi: https://doi.org/10.26906/SUNZ.2023.4.129.

3. Андрусенко Ю.О. & Фесенко Т.Г. (2023). Грід-технології в розподілених обчислювальних середовищах. Системи управління, навігації та зв'язку. Збірник наукових праць. Полтава: ПНТУ, Т.3(73), 148–151. doi: https://doi.org/10.26906/SUNZ.2023.3.148.

4. Rezanov, B., Semenova, A., Petrovska, I. & Fesenko T. (2021). Model for Providing the Second Factor of Authentication Into Authentication Services with Centralized Account Databases. Fifth International Scientific and Technical Conference "Computer and information systems and technologies", 46–47. https://doi.org/10.30837/csitic52021232201.

5. Андрусенко Ю.О. & Фесенко Т.Г. (2023). Планування ресурсів у розподілених обчислювальних середовищах. Проблеми інформатизації : тези доп. 11-ї міжнар. наук.-техн. конф., 16-17 листопада 2023 р., м. Баку, м. Харків, м. Бельсько-Бяла : [у 3 т.]. Т.1. Харків: Impress, С. 78.

6. Рибалов О.О. & Фесенко Т.Г. (2023). Дослідження засобів інтелектуального аналізу текстових документів. Збірник наукових праць XVI Міжнародної науково-практичної конференції «Академічна й університетська наука: результати та перспективи», 12–13 грудня 2023 року. Полтава: Полтавська політехніка, 330–331.

7. Gorbenko, I.D., Zamula, A.A. Information security and noise immunity of telecommunication systems under conditions of various internal and external

impacts Telecommunications and Radio Engineering (English translation of Elektrosvyaz and Radiotekhnika), 2017, 76(19), стр. 1705-1717.

8. Shafiq, M., Yu, X., Laghari, A. A., Yao, L., Karn, N. K., Abdessamia, F. (2016), "Network Traffic Classification techniques and comparative analysis using Machine Learning algorithms," Proceedings of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC), pp. 2451-2455. DOI: https://doi.org/10.1109/CompComm.2016.7925139

9. Salman, O., Elhajj, I.H., Kayssi, A., Chehab, A. (2020), "A review on machine learning–based approaches for Internet traffic classification", Annals of Telecommunications, No. 75(11), P. 673–710. DOI: https://doi.org/10.1007/s12243-020-00770-7.

10. Xie, J., Yu, F.R., Huang, T., Xie, R., Liu, J., Wang, C., Liu, Y. (2018), "A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges", IEEE Communications Surveys & Tutorials, No. 21(1), P. 393-430. DOI: https://doi.org/10.1109/COMST.2018.2866942

11. Gurney, K. (1997), An introduction to neural networks, CRC Press, 248 p

12. Bhadeshia, H. K. D. H. (1999), “Neural Networks in Materials Science”, ISIJ International, No. 39(10), P. 966-979. DOI: https://doi.org/10.2355/isijinternational.39.966.

13. Холл, П., Парк, Б., Семворт, Р.Дж. (2008), "Вибір порядку сусідів у класифікації найближчих сусідів", No. 5, С. 2135–2152.

14. Havrashenko, A., Barkovska, O.(2023), "Analysis of text augmentation algorithms in artificial language machine translation systems". Advanced Information Systems. No.7(1). P. 47–53. DOI: https://doi.org/10.20998/2522-9052.2023.1.08.

15. Barkovska, O, Havrashenko, A., Kholiev, V., Sevostianova, O.(2021), "Automatic text translation system for artificial languages", Computer systems and information technologies. No. 3. P. 21–30. DOI: https://doi.org/10.31891/CSIT-2021-5-3

16. Prabin S. M., Thanabal M. S. A repairing artificial neural network model-based stock price prediction. International journal of computational intelligence systems. 2021. Т. 14, № 1. С. 1337. DOI: https://doi.org/10.2991/ijcis.d.210409.002

17. Ramnani S. Revolutionising conventional marketing with AI: leveraging machine learning for marketing. Interantional journal of scientific research in engineering and management. 2024. Т. 08, № 01. С. 1–13. DOI: https://doi.org/10.55041/ijsrem28481

18. The impact of artificial intelligence on consumer behaviour and changes in business activity due to pandemic effects / T. Dias та ін. Human technology. 2023. Т. 19, № 1. С. 121 DOI: https://doi.org/10.14254/1795-6889.2023.19-1.8

19. Ziakis C., Vlachopoulou M. Artificial intelligence in digital marketing: insights from a comprehensive review. Information. 2023. Т. 14, № 12. С. 664. DOI: https://doi.org/10.3390/info14120664

20. Андрусенко Ю.О. Нестаціонарність ресурсів та послуг хмарної інфраструктури / Ю.О. Андрусенко, Т.Г. Фесенко // Системи управління, навігації та зв'язку. Збірник наукових праць. Полтава: ПНТУ, 2023. - Т.4(74). – С. 129–133. doi: https://doi.org/10.26906/SUNZ.2023.4.129.

21. Ткачов В.М. Критерії вибору стандарту безпровідної передачі даних у високомобільних комп'ютерних мережах / В.М. Ткачов, К.Р. Гальченко, А.А. Коваленко, О.А. Єрошенко // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 4 (66). – С. 63-68. – doi: https://doi.org/10.26906/SUNZ.2021.4.063

22. Tuan L. A survey of live Virtual Machine migration techniques // Computer Science Review, Volume 38, 2020, – С. 334-346, DOI: https://doi.org/10.1016/j.cosrev.2020.100304

23. Hummaida R., Norman W., Rizos S. A hierarchical decentralized architecture to enable adaptive scalable virtual machine migration // Concurrency and Computation: Practice and Experience 35.2, 2023. – С. 202-223, DOI: https://doi.org/10.1002/cpe.7487

24. Sanh V., Debut L., Chaumond J. & Wolf T. (2020). DistilBERT, a distilled

version of BERT: smaller, faster, cheaper and lighter. Cornell Universiti. doi: https://doi.org/10.48550/arXiv.1910.01108.

25. László Nemes & Attila Kiss (2021) Social media sentiment analysis based on COVID-19, Journal of Information and Telecommunication, 5:1, 1-15, DOI: 10.1080/24751839.2020.1790793.

26. M. Krommyda, A. Rigos, K. Bouklas and A. Amditis, "Emotion detection in Twitter posts: a rule-based algorithm for annotated data acquisition," in 2020 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2020 pp. 257-262. doi: 10.1109/CSCI51800.2020.00050

27. M. S. Munir Prince, A. Hasan and F. M. Shah, "An Efficient Ensemble Method for Cancer Detection," 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019, pp. 1-6, doi: 10.1109/ICASERT.2019.8934817.

28. Maryam Hasan, Elke Rundensteiner and Emmanuel Agu, “Automatic emotion detection in text streams by analyzing Twitter data”, International Journal of Data Science and Analytics, 2018, Doi:10.1007/s41060-018-0096-z.

29. N. Alswaidan and Mohamed El Bachir Menai, “A Survey of State-of-The-Art Approaches for Emotion Recognition in Text”, Knowledge and Information Systems, Vol. 62, pp. 2937-2987, 2020.

30. Nandwani P, Verma R. A review on sentiment analysis and emotion detection from text. Soc Netw Anal Min. 2021;11(1):81. doi: 10.1007/s13278-021-00776-6. Epub 2021 Aug 28. PMID: 34484462; PMCID: PMC8402961.

31. P. Ekman, “A methodological discussion of nonverbal behavior,” The Journal of Psychology, vol. 43, no. 1, pp. 141–149, 1957.

32. Robert Plutchik. 1980. A general psychoevolutionary theory of emotion. In Theories of emotion, pages 3–33. Elsevier Rout, J.K., Choo, K., Dash, A.K., Bakshi, S., Jena, S.K., & Williams, K.L. (2018). A model for sentiment and emotion analysis of unstructured social media text. Electronic Commerce Research, 18, 181-199.

33. Singh M, Jakhar AK, Pandey S (2021) Sentiment analysis on the impact of coronavirus in social life using the Bert model. Soc Netw Anal Min 11(1):1–11.

34. Suboh M Alkhushayni, Daniel C Zellmer, Ryan J DeBusk and Dua Alzaleq, “Text emotion mining on twitter”, IOP SciNotes, vol. 1, no. 3, pp. 1-9, 2020. Hasan A, Moin S, Karim A, Shamshirband S. Machine Learning-Based Sentiment Analysis for Twitter Accounts. Mathematical and Computational Applications. 2018; 23(1):11. https://doi.org/10.3390/mca23010011

35. D. Gupta, V. Kushwaha, A. Gupta and P. K. Singh, "Deep Learning based Detection of Water Bodies using Satellite Images," 2021 International Conference on Intelligent Technologies (CONIT), 2021, pp. 1-4, doi: 10.1109/CONIT51480.2021.9498442

36. Oliveira R and Yadollahi E. (2023). Robots in movies: a content analysis of the portrayal of fictional social robots. Behaviour & Information Technology. 10.1080/0144929X.2023.2196576. 43:5. (970-987). Online publication date: 3-Apr-2024. https://www.tandfonline.com/doi/full/10.1080/0144929X.2023.2196576

37. Choi S. (2023). Temporal Framing in Balanced News Coverage of Artificial Intelligence and Public Attitudes. Mass Communication and Society. 10.1080/15205436.2023.2248974. 27:2. (384-405). Online publication date: 3-Mar-2024. https://www.tandfonline.com/doi/full/10.1080/15205436.2023.2248974

38. Bankins S, Ocampo A, Marrone M, Restubog S and Woo S. (2023). A multilevel review of artificial intelligence in organizations: Implications for organizational behavior research and practice. Journal of Organizational Behavior. 10.1002/job.2735. 45:2. (159-182). Online publication date: 1-Feb-2024. https://onlinelibrary.wiley.com/doi/10.1002/job.2735

39. Porlezza C. (2023). Promoting responsible AI : A European perspective on the governance of artificial intelligence in media and journalism . Communications. 10.1515/commun-2022-0091. 48:3. (370-394). Online publication date: 24-Aug-2023.. Online publication date: 30-Aug-2023. https://www.degruyter.com/document/doi/10.1515/commun-2022-0091/html

40. Mao Y and Shi-Kupfer K. (2021). Online public discourse on artificial

intelligence and ethics in China: context, content, and implications. AI & SOCIETY. 10.1007/s00146-021-01309-7. 38:1. (373-389). Online publication date: 1-Feb-2023. https://link.springer.com/10.1007/s00146-021-01309-7

41. Li Q, Yang Y, Guo Y, Li W, Liu Y, Liu H and Kang Y. Performance Evaluation of Deep Learning Classification Network for Image Features. IEEE Access. 10.1109/ACCESS.2020.3048956. 9. (9318-9333). https://ieeexplore.ieee.org/document/9312591/

42. Фесенко Т.Г., Рибалов О.О., Хрусталев Є.К., Снігур А.Р. Методи і засоби інтелектуального аналізу текстових документів: бібліометричне дослідження. Збірник наукових праць. – Полтава: ПНТУ, 2024. Т. 3 (77). С. 148-151.