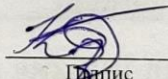
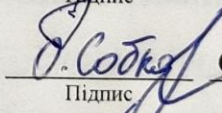
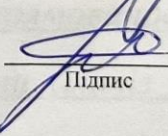


КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

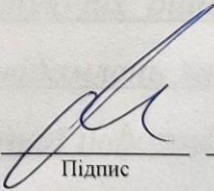
на тему Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

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

Виконав: студент 2 курсу, група КНм-22-1  В.А. Козакевич
Курс, група виконавця Підпис Ініціали, прізвище
Керівник: викладач кафедри КН  О.В. Собко
Науковий ступінь, посада Підпис Ініціали, прізвище
Нормоконтроль: к.т.н., доцент кафедри КН  Р.О. Багрій
Науковий ступінь, посада Підпис Ініціали, прізвище

До захисту допускаю:

Зав. кафедри КН, д.т.н., професор

 О.В. Бармак
Підпис Ініціали, прізвище

12 грудня 2023 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
Факультет інформаційних технологій
Кафедра комп'ютерних наук
Освітній ступінь магістр
Галузь знань 12 – Інформаційні технології
Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ
Завідувач кафедри комп'ютерних наук

(підпис)

д.т.н., професор О.В. Бармак

« 01 » вересня 2023 року

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА

1. Тема кваліфікаційної роботи магістра: Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам»

2. Завдання видано студенту Козакевичу Віталію Анатолійовичу
(прізвище, ім'я, по батькові)

3. Керівник роботи викладач кафедри КН Собко Олена Віталіївна
(прізвище, ім'я, по батькові)

4. Затверджені наказом університету від « 15 » серпня 2023 р. № 30

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

Метою кваліфікаційної роботи магістра є вирішення задачі генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам. Для цього необхідно провести аналіз предметної області, визначити особливості застосування n-грам для задач генерації текстів заданої семантичної спрямованості, виконати аналіз існуючих рішень щодо подібних задач, розробити метод генерації текстових повідомлень заданої семантичної спрямованості, спроектувати структуру відповідної інформаційної системи та створити застосунок, що використовує розроблений метод, виконати його тестування та використати його для того щоб дослідити ефективність розробленого методу щодо читабельності генерованих текстів.

Реферат

Кваліфікаційна робота магістра присвячена розв'язку науково-технічної задачі автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, яка за заданим семантичним напрямком дозволяє генерувати потрібні користувачу текстові повідомлення. Результатом роботи є метод, що призначений для генерації текстових повідомлень заданої семантичної спрямованості, що за навченою частотною n-грамною моделлю та заданою користувачем семантичною спрямованістю, генерує відповідне текстове повідомлення.

Актуальність теми. На сьогоднішній день організації в багатьох галузях промисловості використовують автоматизацію текстових повідомлень. Щодня з'являються нові варіанти використання інструментів автоматизації. Деякі з найпопулярніших застосувань включають в себе електронну комерцію, автоматизацію і упорядкування візитів, підтримку потенційних клієнтів і чат-ботів. Одним із вагомих переваг в автоматизації те, що вона так само добре працює як з 10 людьми, так і зі 100 000. Важливо пам'ятати, що потрібно персоналізувати кожне надіслане текстове повідомлення. Це призводить до кращого залучення та рівня відповіді. Також використання такої служби з генерації текстових повідомлень, дозволяє впоратися з важкою роботою, пов'язаною з підтримкою взаємодії з аудиторією, що інколи відіграє основну роль в роботі з клієнтами.

Мета і задачі роботи. Метою кваліфікаційної роботи магістра є розробка методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, яка дозволяє за заданою семантичною спрямованістю генерувати відповідні повідомлення, та програмної реалізації відповідного програмного застосунку. Задачі, які необхідно виконати для досягнення поставленої мети:

– провести аналіз предметної області, визначити особливості застосування n-грам для задач генерації текстів семантичної спрямованості;

- виконати аналіз існуючих рішень щодо подібних задач;
- розробити метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам;
- спроектувати структуру інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості й структуру відповідної бази даних;
- створити застосунок що використовує розроблений метод, виконати його тестування;
- дослідити ефективність розробленого методу.

Об’єкт дослідження – процес генерації текстових повідомлень заданої семантичної спрямованості.

Предмет дослідження – інформаційні технології, моделі та методи, алгоритми та засоби для генерації текстових повідомлень заданої семантичної спрямованості.

Методи дослідження, що застосовані для вирішення поставлених завдань, наступні: положення методів аналізу даних та теорії множин, методології проектування інформаційних систем для проектування інформаційної системи для генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, а також об’єктно-орієнтований підхід для розробки відповідної прикладної програмної реалізації.

Наукова новизна одержаних результатів. Результати виконання кваліфікаційної роботи магістра містять інновації та наукову новизну, зокрема було удосконалено метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, який дозволяє виконувати генерації текстових повідомлень заданого цільового обсягу.

Розроблений метод дозволяє генерувати текстові повідомлення як з використанням біграм, так і на основі триграм, із визначеним середнім рівнем читабельності тексту вище середнього. В ході дослідження ефективності було

зроблено висновок, що текстові повідомлення, які були згенеровані з використанням біграм мали гіршу читабельність, ніж згенеровані за допомогою триграм. Однак, переважна кількість згенерованих текстових повідомлень мала рівень читабельності Флеша середній та вище, а саме для біграм – 81% та для триграм 91%.

Практичне значення одержаних результатів. Було створено інформаційну систему для генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, яка є прикладною програмною реалізацією розробленого метода генерації текстових повідомлень і використовувалась для проведення експериментальних досліджень з метою апробації запропонованого метода.

Проведені дослідження ефективності розробленого в роботі метода генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам свідчать про те, що даний підхід можна використовувати для генерації текстових повідомлень для прикладних застосувань на кшталт чатботів та текстрайтингу.

Апробація результатів кваліфікаційної роботи магістра та публікації. Основні наукові і практичні результати кваліфікаційної роботи магістра доповідались у доповіді на тему «Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам» на XV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2023»; за темою роботи автором виконано наукову публікацію:

Козакевич В.А., Собко О.В., Тищенко О.О., Вознюк Л.О., Медведчук В.Ю. Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023».

Хмельницький, 2023. с. 136-141. URL: <https://kn.khmnu.edu.ua/wp-content/uploads/sites/18/apkn-2023-corporpaper.pdf>

Структура і обсяг роботи. Кваліфікаційна робота магістра складається з наступного: реферату, завдання, змісту, переліку скорочень, вступу, 4 розділів, висновків, переліку посилань з 43-х найменувань й 6-х додатків. Обсяг основного тексту кваліфікаційної роботи магістра становить 83 сторінки. В роботі наведено 34 світлин і 8 таблиць.

Ключові слова: текст, повідомлення, семантика, генерація тексту, програма, система генерації текстових повідомлень.

Зміст

Перелік скорочень	4
Вступ.....	5
Розділ 1 Дослідження предметної області генерації текстових повідомлень за семантичною спрямованістю	9
1.1 Аналіз предметної області генерації текстових повідомлень	9
1.2 Аналіз існуючих моделей та методів для генерації текстових повідомлень заданої семантичної спрямованості	12
1.3 Аналіз сучасних наукових публікацій з генерації текстових повідомлень .	14
1.4 Аналіз існуючих програмних засобів для генерації текстових повідомлень заданої семантичної спрямованості	16
1.5 Проблеми генерації текстових повідомлень	19
1.6 Постановка задачі.....	20
Висновки до розділу 1	21
Розділ 2 Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам	22
2.1 Загальний підхід до побудови частотної n-грамної моделі та прогнозування послідовності слів	22
2.2 Етапи виконання методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.....	23
2.3 Навчання n-грамної моделі в методі генерації текстових повідомлень заданої семантичної спрямованості	26
2.4 Генерація текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам	30
2.5 Способи оцінки якості згенерованих текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.....	33
Висновки до розділу 2	34
Розділ 3 Інформаційна система генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.....	35
3.1 Визначення функцій інформаційної системи.....	35
3.2 Проектування структури інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості.....	38

3.3 Вибір засобів розробки інформаційної системи генерації текстових повідомлень	41
Висновки до розділу 3	47
Розділ 4 Дослідження ефективності методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам	49
4.1 Структура інформаційної системи генерації текстових повідомлень та взаємозв'язок її компонентів	49
4.2 Особливості реалізації інформаційної системи	52
4.3 Тестування функціональних можливостей інформаційної системи генерації текстових повідомлень	53
4.4 Дослідження ефективності генерації текстових повідомлень заданої семантичної спрямованості.....	64
Висновки до розділу 4	75
Загальні висновки.....	77
Перелік посилань.....	80
Додатки	

Перелік скорочень

Скорочення, термін, позначення	Пояснення
API	Application Programming Interface
NLTK	Natural Language Toolkit
CORS	Cross-Origin Resource Sharing
GAN	Generative adversarial networks
NLP	Natural language processing
SC	Semantic Communication
GPT	Generative Pre-trained Transformer
WEB	World Wide Web
Git	розподілена система керування версіями файлів
IT	Інформаційні технології
КН	Комп'ютерні науки
КРБ	Кваліфікаційна робота магістра
КН	Комп'ютерні науки

Вступ

Кваліфікаційна робота магістра присвячена розв'язку науково-технічної задачі автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, яка за заданим семантичним напрямком дозволяє генерувати потрібні користувачу текстові повідомлення. Результатом роботи є метод, що призначений для генерації текстових повідомлень заданої семантичної спрямованості, що за навченою частотною n-грамною моделлю та заданою користувачем семантичною спрямованістю, генерує відповідне текстове повідомлення.

Актуальність теми. На сьогоднішній день організації в багатьох галузях промисловості використовують автоматизацію текстових повідомлень. Щодня з'являються нові варіанти використання інструментів автоматизації. Деякі з найпопулярніших застосувань включають в себе електронну комерцію, автоматизацію і упорядкування візитів, підтримку потенційних клієнтів і чат-ботів. Одним із вагомих переваг в автоматизації те, що вона так само добре працює як з 10 людьми, так і зі 100 000. Важливо пам'ятати, що потрібно персоналізувати кожне надіслане текстове повідомлення. Це призводить до кращого залучення та рівня відповіді. Також використання такої служби з генерації текстових повідомлень, дозволяє впоратися з важкою роботою, пов'язаною з підтримкою взаємодії з аудиторією, що інколи відіграє основну роль в роботі з клієнтами.

Мета і задачі роботи. Метою кваліфікаційної роботи магістра є розробка методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, яка дозволяє за заданою семантичною спрямованістю генерувати відповідні повідомлення, та програмної реалізації відповідного програмного застосунку. Задачі, які необхідно виконати для досягнення поставленої мети:

- провести аналіз предметної області, визначити особливості застосування n-грам для задач генерації текстів семантичної спрямованості;

- виконати аналіз існуючих рішень щодо подібних задач;
- розробити метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам;
- спроектувати структуру інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості й структуру відповідної бази даних;
- створити застосунок що використовує розроблений метод, виконати його тестування;
- дослідити ефективність розробленого методу.

Об’єкт дослідження – процес генерації текстових повідомлень заданої семантичної спрямованості.

Предмет дослідження – інформаційні технології, моделі та методи, алгоритми та засоби для генерації текстових повідомлень заданої семантичної спрямованості.

Методи дослідження, що застосовані для вирішення поставлених завдань, наступні: положення методів аналізу даних та теорії множин, методології проектування інформаційних систем для проектування інформаційної системи для генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, а також об’єктно-орієнтований підхід для розробки відповідної прикладної програмної реалізації.

Наукова новизна одержаних результатів. Результати виконання кваліфікаційної роботи магістра містять інновації та наукову новизну, зокрема було удосконалено метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, який дозволяє виконувати генерації текстових повідомлень заданого цільового обсягу.

Розроблений метод дозволяє генерувати текстові повідомлення як з використанням біграм, так і на основі триграм, із визначеним середнім рівнем читабельності тексту вище середнього. В ході дослідження ефективності було

зроблено висновок, що текстові повідомлення, які були згенеровані з використанням біграм мали гіршу читабельність, ніж згенеровані за допомогою триграм. Однак, переважна кількість згенерованих текстових повідомлень мала рівень читабельності Флеша середній та вище, а саме для біграм – 81% та для триграм 91%.

Практичне значення одержаних результатів. Було створено інформаційну систему для генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, яка є прикладною програмною реалізацією розробленого метода генерації текстових повідомлень і використовувалась для проведення експериментальних досліджень з метою апробації запропонованого метода.

Проведені дослідження ефективності розробленого в роботі метода генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам свідчать про те, що даний підхід можна використовувати для генерації текстових повідомлень для прикладних застосувань на кшталт чатботів та текстрайтингу.

Апробація результатів кваліфікаційної роботи магістра та публікації. Основні наукові і практичні результати кваліфікаційної роботи магістра доповідались у доповіді на тему «Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам» на XV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2023»; за темою роботи автором виконано наукову публікацію:

Козакевич В.А., Собко О.В., Тищенко О.О., Вознюк Л.О., Медведчук В.Ю. Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023».

Хмельницький, 2023. с. 136-141. URL: <https://kn.khmnu.edu.ua/wp-content/uploads/sites/18/apkn-2023-corporpaper.pdf>

Структура і обсяг роботи. Кваліфікаційна робота магістра складається з наступного: реферату, завдання, змісту, переліку скорочень, вступу, 4 розділів, висновків, переліку посилань з 43-х найменувань й 6-х додатків. Обсяг основного тексту кваліфікаційної роботи магістра становить 83 сторінки. В роботі наведено 34 світлин і 8 таблиць.

Розділ 1 Дослідження предметної області генерації текстових повідомлень за семантичною спрямованістю

1.1 Аналіз предметної області генерації текстових повідомлень

Текст є основним засобом людського спілкування та передачі інформації. Він використовується для вираження думок, ідей, почуттів, а також для забезпечення зв'язку між особами. Текст може приймати різні форми, такі як письмовий, усний чи електронний, і відігравати ключову роль у різних аспектах людського життя [1].

Повідомлення, в свою чергу, може бути визначене як структурована одиниця тексту, спрямована на передачу конкретної інформації або вираження певного змісту. В контексті спілкування та інформаційного обміну, повідомлення є ключовим засобом вираження думок та сприйняття ідеї читачем чи слухачем. Воно може містити текстовий, графічний чи аудіовізуальний зміст.

Що вважається повідомленням може залежати від контексту та мети комунікації. У сучасному світі, особливо в інтернеті та соціальних мережах, повідомлення може бути текстовим коментарем, електронним листом, чатовим повідомленням, аудіо-чи відеозаписом. Зазначено, що форма повідомлення може варіюватися від коротких та зрозумілих фраз до довгих та деталізованих текстів [2]. Варто також згадати про розмір повідомлення, багато засобів комунікації встановлюють обмеження, особливо у випадку коротких текстових повідомлень чи соціальних мереж, де обмежений обсяг тексту може бути встановлений для зручності користувачів та оптимізації мережевих ресурсів. Наприклад, Twitter обмежує довжину одного твіта до певної кількості символів.

Текст та повідомлення грають велику роль у спілкуванні та обміні інформацією в сучасному суспільстві. Спрощений доступ до цифрових засобів комунікації робить текстові повідомлення основним засобом взаємодії для мільйонів людей по всьому світу [3]. Соціальні мережі, такі як Facebook, Instagram, Twitter та інші, використовують текстові повідомлення як основний інструмент для спілкування між користувачами. Короткі та інформативні

повідомлення дозволяють ефективно обмінюватися ідеями та враженнями. За допомогою сучасних технологій, таких як штучний інтелект та обробка природної мови, створюються нові методи генерації текстового контенту [4].

Текст складається з ряду компонентів, які взаємодіють для передачі інформації:

- слова – основні будівельні блоки тексту, вони представляють собою лексичні одиниці, що несуть семантичне значення;
- фрази та речення – групи слів, що утворюють конструкції для вираження повних ідей та змісту;
- абзаци – групи фраз і речень, що об'єднуються в логічні секції для організації змісту;
- пунктуація – знаки, які визначають структуру і граматику тексту, вказують на паузи та відносини між різними частинами;

Також важливо роль у генерації тексту відіграє семантика. Семантика – це галузь лінгвістики, що вивчає значення слів та їхній вплив на мовлення [5]. Вона займається вивченням того, як слова та структури мови виражають значення та взаємовідношення між ними.

Семантична спрямованість вказує на визначення конкретного змісту, теми чи сфери, яку текст виражає. Це означає, що текст спрямований на передачу певного значення чи інформації, яка відповідає конкретному контексту чи меті. Семантична спрямованість є ключовим аспектом в мовному аналізі та текстовій генерації, спрямованих на визначення конкретної теми, контексту чи інформаційної області. Цей концепт відноситься до визначення змісту, напряму та цільової аудиторії, які впливають на вибір слів, виразів та структури тексту з метою ефективного спілкування та передачі інформації.

Визначення семантичної спрямованості починається з вибору слів та виразів. Для цього використовуються слова, які не тільки точно відображають тему, але й виражають необхідні концепції [6]. Джерела такого підходу можуть включати лексикони, тезауруси та словники, які допомагають розуміти семантичні відношення між словами.

Специфічна термінологія галузі чи тематики може бути використана для точного визначення семантичної спрямованості. Джерела термінології можуть включати спеціалізовані словники, глосарії та технічні довідники, які визначають та пояснюють терміни, що використовуються в конкретній області. Врахування контексту дозволяє не лише визначити тематичну орієнтацію, але й надати тексту додаткову семантичну глибину [7]. Джерела контексту можуть включати природні мовленнєві дані, корпуси текстів та бази даних для аналізу вживання слів у реальних ситуаціях.

Використання комбінації цих підходів та джерел дозволяє точніше задавати семантичну спрямованість тексту, надаючи йому відповідну глибину і контекст для максимально ефективного спілкування та передачі інформації.

Семантична спрямованість використовується в різних сферах. У сфері медіа та журналістики семантична спрямованість визначає тематику новин, статей та інших інформаційних матеріалів. Журналісти використовують точну лексику та фрази для передачі специфічної семантики новин, репортажів та інших медійних випусків [8].

У сфері маркетингу та реклами семантична спрямованість визначає, як ефективно спілкуватися з цільовою аудиторією. Рекламні кампанії використовують семантичні елементи для створення враження та взаємодії з клієнтами.

У наукових дослідженнях семантична спрямованість дозволяє точно визначити область та специфіку дослідження. Академічні тексти використовують семантичні елементи для ясності та точності у вираженні ідей.

У технічній сфері семантична спрямованість є критично важливою для чіткого вираження технічних концепцій та інструкцій. Технічні письменники використовують семантичні засоби для уникнення непорозумінь та помилок в інтерпретації інструкцій.

Реалізація методу з використанням лексичних n-грам важлива для вдосконалення автоматичного перекладу та генерації контенту. Збільшення контексту допомагає уникнути мовних казусів і забезпечити точніші переклади

та зміст [9]. У контексті автоматизованої обробки природної мови, методи з використанням лексичних n-грам є ключовими для розробки ефективних моделей NLP. Ці методи дозволяють враховувати лексичні одиниці на різних рівнях, покращуючи розуміння тексту та забезпечуючи кращу ефективність обробки мовленнєвих даних.

В сфері генерації текстових описів та сценаріїв важливо використовувати методи, які дозволяють враховувати семантичні зв'язки між словами та виразами. Лексичні n-грам допомагають створювати більш змістовні та логічні текстові матеріали [10]. В контексті глибокого навчання, використання лексичних n-грам є важливим елементом для покращення якості моделей, зокрема у завданнях генерації тексту. Здатність моделей «запам'ятовувати» та враховувати лексичні контексти сприяє досягненню більшої точності та креативності.

1.2 Аналіз існуючих моделей та методів для генерації текстових повідомлень заданої семантичної спрямованості

Дослідження в галузі генерації текстових повідомлень заданої семантичної спрямованості є досить популярним і активно розвивається. Це пов'язано з тим, що генерація текстових повідомлень має широкий спектр застосувань. На даний час існує багато різних методів та інформаційних технологій для генерації текстових повідомлень.

На сьогоднішній день створені моделі для генерації текстових повідомлень до таких моделей відносяться GPT-3, Bard, Turing NLG тощо.

GPT-3 – це велика мовна модель, розроблена компанією OpenAI. GPT-3 може генерувати текст, перекладати мови, писати різні види творчого контенту та інформативно відповідати на ваші запитання.

Bard – це велика мовна модель, розроблена компанією Google AI. Bard може генерувати текст, перекладати мови, писати різні види творчого контенту та інформативно відповідати на ваші запитання.

Turing NLG – це платформа для генерації тексту, розроблена компанією Google AI. Turing NLG може використовуватися для створення різних видів текстового контенту, таких як новинні статті, веб-сторінки, електронні листи тощо.

Широко застосовуються методи засновані на використанні правил, які генерують текст, використовуючи набір правил, що визначають, які слова та словосполучення можуть використовуватися в кожному конкретному випадку [10].

Методи засновані на використанні нейронних мереж генерують текст, використовуючи нейронні мережі, які навчаються на наборі даних текстових повідомлень. Нейронні мережі дозволяють генерувати текст, який є більш складним і природним, ніж текст, генерований методами, заснованими на правилах або статистичному підході.

Методи засновані на використанні статистичного підходу генерують текст, використовуючи статистичні дані про частоту використання слів та словосполучень. Наприклад, такий метод може використовуватися для генерації повідомлень для соціальних мереж або рекламних матеріалів. До методів заснованих на використанні статистичного підходу відносяться, наприклад, n-грами.

N-грами – це послідовності з n слів. Вони використовуються в обробці природної мови для моделювання частоти появи слів у тексті. Наприклад, біграми – це послідовності з двох слів, такі як "я люблю", триграми – це послідовності з трьох слів, такі як "я люблю тебе".

Ці методи, моделі та інформаційні технології мають широкий спектр потенційних застосувань, включаючи:

- створення творчого контенту, такого як вірші, код, сценарії, музичні твори та електронні листи;
- трансляція мови з однієї мови на іншу;
- відповіді на запитання.

Наразі дослідження в галузі генерації текстових повідомлень заданої семантичної спрямованості тривають. Вчені працюють над розробкою нових методів, моделей та інформаційних технологій, які дозволять генерувати текстові повідомлення з більш високою семантичною спрямованістю, творчістю та інформативністю, і тому розробка методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам має сенс.

1.3 Аналіз сучасних наукових публікацій з генерації текстових повідомлень

Сучасний розвиток галузі обробки природної мови і штучного інтелекту визначає нові виклики та можливості у генерації текстового контенту з конкретною семантичною спрямованістю. Дослідження у цьому напрямку стає все більш важливим, оскільки воно визначає шляхи для розвитку ефективних та контекстно-орієнтованих методів генерації тексту. Подальший аналіз наукових публікацій дозволить глибше розібратися в досягненнях та перспективах цього заходу та визначити ключові тенденції у даній області.

Одне з таких досліджень є теорія Арун Ядава в публікації "Sentence generation from a bag of words using N-gram model" [12]. У цьому дослідженні розглядається метод генерації речень з заданого набору слів (bag of words). Завдання генерації речень корисне у сферах сумаризації тексту, систем відповідей на питання та інших. Основна мета полягає в тому, щоб створити всі можливі правильні речення з заданого мішка слів. У тексті зазначається, що для реалізації цього завдання застосовується N-грамна мовна модель. Модель навчається на текстовому корпусі для генерації лише кандидатських послідовностей з даного мішка слів. Замість розглядання всіх можливих $N!$ перестановок порядку слів як кандидатських послідовностей, застосовується техніка DFS (Depth First Search) для фільтрації кандидатських послідовностей під час виконання програми, що дозволяє генерувати лише послідовності, які мають менше $N!$ варіантів. Користується двома корпусами: текстовим і

анотованим корпусом частин мови (POS tags). З анотованого корпусу витягуються всі дійсні POS триграмні теги. Кожна створена кандидатська послідовність має ймовірнісний бал. Кандидатські послідовності ранжуються за збігом із валідним триграмним підписом частин мови та ймовірнісним балом. Попередні експерименти, проведені в цьому напрямку за використання вищезазначеної моделі, показують обіцяючі результати.

Також варто згадати публікацію “Supporting Writers with Continuous Automatic Text Summaries” [13], в якій була детально описана пропозиція створення текстового редактора, який допомагає користувачам планувати, структурувати та рефлексувати їхній процес написання. Редактор надає постійно оновлювані параграфні резюме у вигляді анотацій у полях, використовуючи автоматичну сумаризацію тексту. Рівні резюме варіюються від повного тексту до обраних (центральных) речень та до набору ключових слів. У двох користувацьких досліджах (N=4 і N=8), де учасники писали аналітичні есе на задану тему та статтю, виявлено, що резюме дали користувачам зовнішню перспективу на їхнє написання і допомогли їм переглядати зміст і обсяг їхніх написаних параграфів. Люди також використовували інструмент для швидкого отримання огляду тексту та розробляли стратегії для інтеграції висновків з автоматичних резюме. Загалом, дана робота досліджує та підкреслює цінність розробки інструментів штучного інтелекту для письменників, з можливостями обробки природної мови (NLP), які виходять за межі простої генерації та корекції тексту.

Ще однією важливою публікацією в даній темі була публікація “Text generation models in deep learning” [14]. У даній публікації обговорюється застосування глибокого навчання у галузі обробки природної мови (NLP). Зазначається, що методи глибокого навчання, з численними оброблюючими шарами, здатні розуміти стратифіковане представлення даних і досягають передових результатів у різних областях. Текст присвячений огляду розвитку глибоких генеративних моделей, зокрема у сфері генерації тексту, з 2015 року [15]. Автори рецензують різноманітні моделі глибокого навчання, які

використовуються для генерації тексту, та надають огляд минулих, сучасних і майбутніх напрямків розвитку цих моделей. Також в роботі розглядаються методи глибокого навчання, які вивчаються та оцінюються в різних областях застосування в NLP.

Отже, можна зробити висновок про те, що у останні роки спостерігається значний прогрес у галузі генерації текстових повідомлень заданої семантичної спрямованості. Це пов'язано з розвитком методів машинного навчання, зокрема нейронних мереж. Очікується, що дослідження в галузі генерації текстових повідомлень заданої семантичної спрямованості будуть продовжуватися і в майбутньому. Це пов'язано з тим, що генерація текстових повідомлень є важливою технологією, яка має широкий спектр застосувань.

1.4 Аналіз існуючих програмних засобів для генерації текстових повідомлень заданої семантичної спрямованості

В умовах сучасних інформаційних технологій, генерація текстових повідомлень здобуває все більше популярності та важливості, особливо в контексті розвитку штучного інтелекту та обробки природної мови. Засоби для генерації тексту, включаючи інноваційні нейромережі, відіграють ключову роль у вирішенні завдань автоматичного формулювання текстового змісту, що варіюються від створення креативних текстів до генерації програмного коду. Перш за все, варто розглянути передові нейромережі, такі як GPT, BERT, OpenAI Codex та інші, які виявили вражаючі здібності у генерації тексту [16]. Усе це дозволить отримати повніше розуміння сучасних технологій генерації текстових повідомлень та визначити перспективи їхнього розвитку.

GPT (Generative Pre-trained Transformer) – це модель глибокого навчання, яка використовує нейронні мережі для генерації текстових повідомлень. Вони навчаються на великих корпусах текстів та здатні генерувати високоякісні текстові повідомлення з урахуванням семантики та структури мовлення зображений на рисунку 1.1.

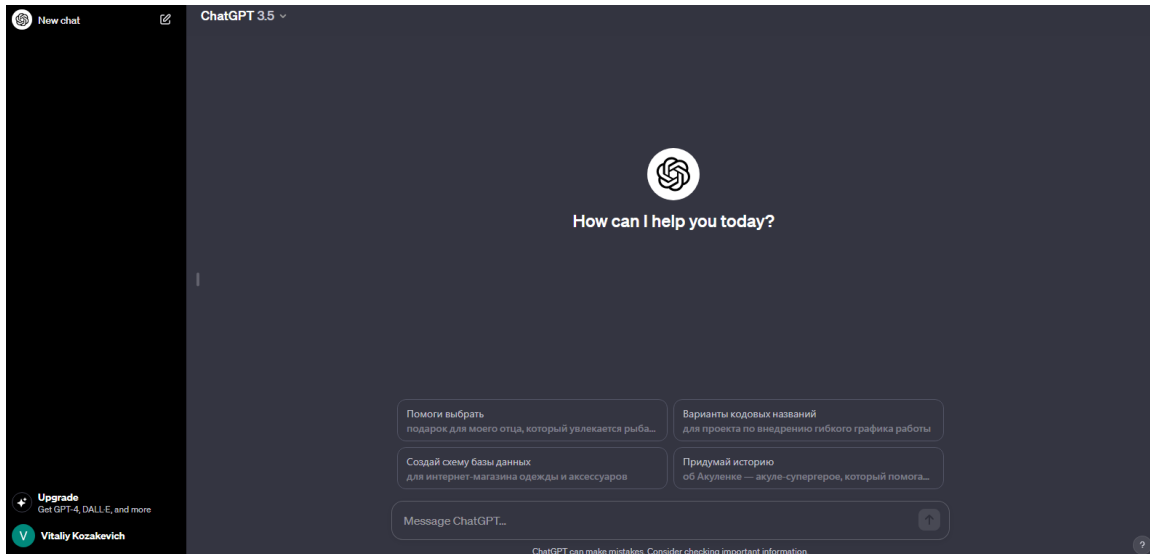


Рисунок 1.1 – ChatGPT [17]

Одним з основних недоліків ChatGPT є його здатність генерувати неточну або навіть шкідливу інформацію. Це пов'язано з тим, що ChatGPT навчається на величезній кількості текстових даних, які можуть містити помилки, упередження та неправдиву інформацію [17]. Ще одним недоліком ChatGPT є його обмежена здатність розуміти контекст. Це може призвести до того, що ChatGPT дасть неправильну або невідповідну відповідь на запитання. Наприклад, якщо користувач запитає ChatGPT про погоду, він може відповісти інформацією про погоду в іншому місці або про погоду в минулому. Нарешті, ChatGPT може бути повільним або навіть недоступним у пікові години. Це пов'язано з тим, що ChatGPT є ресурсомістким сервісом, який може бути перевантажений великими обсягами запитів.

OpenAI Codex. OpenAI Codex є моделлю, розробленою на основі GPT, але спеціалізованою на генерації програмного коду та текстів. Вона здатна створювати зразки коду з невеликої описової інформації. Дана модель зображена на рисунку 1.2.

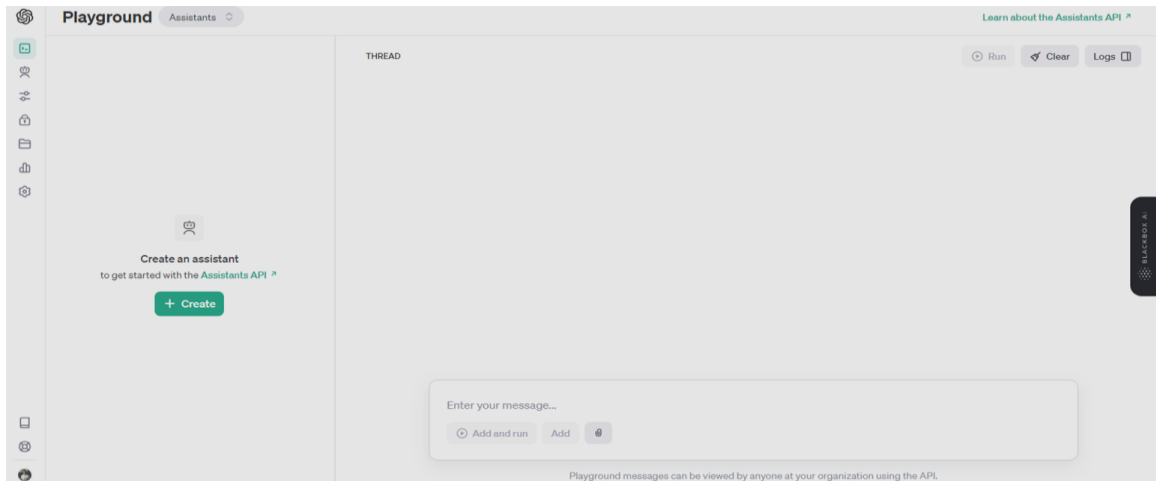


Рисунок 1.2 – OpenAI Codex [17]

OpenAI Codex, хоча і потужний, має свої недоліки [17]. По-перше, він не завжди точно розуміє контекст і може створювати неправильні або некоректні відповіді. По-друге, Codex може породжувати велику кількість тексту без необхідності, що робить важчим відділення корисної інформації. Нарешті, відсутність повного контролю над генерацією може становити проблему для вибіркового використання

BARD – це генеративна мовна модель, також відома як розмовний штучний інтелект або чат-бот, навчений бути інформативним і всеосяжним [17]. Він навчений на величезному наборі даних тексту та коду, і він може спілкуватися та генерувати текст, подібний до людського, у відповідь на широкий спектр підказок і запитань. Дана модель зображена на рисунку 1.3 .

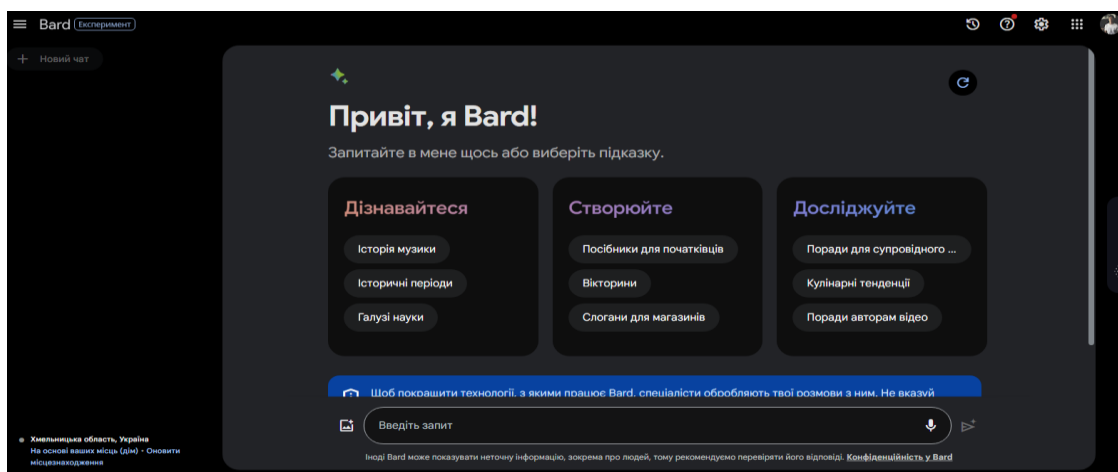


Рисунок 1.3 – Генеративна мовна модель BARD [17]

BARD має ряд недоліків. По-перше, його успішність залежить від доступності репрезентативних даних, що може обмежити його застосування в менше розвинених областях. По-друге, BARD не завжди ефективно враховує та вирізняє між моделями з різними рівнями якості. По-третє, алгоритм може виявити вразливість до атак, спрямованих на маніпулювання атрибутами. Нарешті, BARD може бути схильний до перенавчання, що може виникнути при недостатній увазі до різноманіття даних.

Отже, в даний час існує багато програмних засобів для генерації текстових повідомлень, які мають як переваги, так і недоліки. Вибір програмного засобу для генерації текстових повідомлень залежить від конкретного завдання та вимог. У майбутньому очікується, що програмні засоби для генерації текстових повідомлень будуть розвиватися і будуть мати більш високу точність, природність і оригінальність. Це дозволить використовувати їх для вирішення більш широкого спектру завдань і підвищить якість створюваного тексту.

1.5 Проблеми генерації текстових повідомлень

Генерація текстових повідомлень – це задача обробки природної мови, яка полягає в створенні нових текстових повідомлень на основі наявних даних. Ця задача має широкий спектр застосувань, включаючи створення творчого контенту, переклад мови, створення резюме текстових документів і відповіді на запитання. Незважаючи на те, що методи генерації текстових повідомлень досягли значних успіхів у останні роки, вони все ще мають ряд проблем.

Генеровані текстові повідомлення можуть бути семантично невідповідними заданим вимогам. Наприклад, якщо задано завдання створити позитивний текст, генератор може створити текст, який є негативним або нейтральним. Семантична невідповідність є однією з найсерйозніших проблем генерації текстових повідомлень [19]. Ця проблема виникає тому, що методи генерації текстових повідомлень зазвичай навчаються на наборах даних, які не

обов'язково містять текстові повідомлення з заданою семантичною спрямованістю. Наприклад, якщо генератор навчається на наборі даних новинних статей, він може бути не в змозі генерувати позитивні текстові повідомлення, оскільки в наборі даних може бути мало позитивних новинних статей.

Генеровані текстові повідомлення можуть бути нетворчими і неоригінальними. Наприклад, генератор може постійно генерувати схожі текстові повідомлення, або він може просто копіювати текст з наявних даних. Це пов'язано з тим, що методи генерації текстових повідомлень зазвичай навчаються на наборах даних, які містять текстові повідомлення, які є стандартними і шаблонними. Наприклад, якщо генератор навчається на наборі даних віршів, він може бути не в змозі генерувати оригінальні вірші, оскільки в наборі даних може бути багато віршів, які написані в одному і тому ж стилі.

Генеровані текстові повідомлення можуть бути низької якості, тобто вони можуть містити помилки, неточності або непотрібні деталі. Ця проблема може виникати через помилки в алгоритмі генерації текстових повідомлень, через неякісні дані, на яких навчається алгоритм, або через інші фактори. Наприклад, генератор може створити текст, який містить помилки в граматиці або пунктуації, або він може створити текст, який є нелогічним або несуперечливим.

Отже, враховуючи перелічені проблеми, можна зробити висновок про те, що задача генерації текстових повідомлень досі є актуальною, тому розробка методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам має перспективність.

1.6 Постановка задачі

Метою роботи є розробка та програмна реалізація методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, який дозволяє за заданою в вигляді фрази семантичною

спрямованістю генерувати відповідні повідомлення. Задачі, які необхідно виконати для досягнення поставленої мети:

- провести аналіз предметної області, визначити особливості застосування n-грам для задач генерації текстів семантичної спрямованості;
- виконати аналіз існуючих рішень щодо подібних задач;
- розробити метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам;
- спроектувати структуру інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості й структуру відповідної бази даних;
- створити застосунок що використовує розроблений метод, виконати його тестування;
- дослідити ефективність розробленого методу.

Висновки до розділу 1

У даному розділі кваліфікаційної роботи магістра було виконано аналіз предметної області генерації текстових повідомлень та встановлено, що на сьогоднішній день текстові повідомлення відіграють важливу роль у спілкуванні людей. На сьогоднішній день розроблено багато методів та моделей, які генерують текст, проте дослідження в цій області досі проводяться з метою покращенні існуючих підходів до генерації тексту.

Аналіз сфери генерації тексту показує, що її потенціал ще не реалізований у повному обсязі. Наявні підходи та архітектурні рішення для генерації текстових повідомлень мають недоліки, що спонукають до поглиблення в даній сфері. В розділі також визначено мету кваліфікаційної роботи, а саме розробку методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.

Розділ 2 Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

2.1 Загальний підхід до побудови частотної n-грамної моделі та прогнозування послідовності слів

N-грамна модель – це статистична мовна модель, яка використовує n-грами для прогнозування наступного слова в послідовності [21]. N-грама – це послідовність N слів. Наприклад, двограма (n=2) – це послідовність з двох слів, а триграма (n=3) – це послідовність з трьох слів. N-грамні моделі працюють, обчислюючи ймовірність появи кожного слова в послідовності на основі n-грам, які вже з'явилися в послідовності. Наприклад, якщо потрібно прогнозувати наступне слово в послідовності "я люблю", то потрібно обчислити ймовірність появи кожного слова в якості наступного слова, використовуючи двограми, які з'явилися в послідовності. N-грамні моделі можна використовувати для різних завдань обробки природної мови.

Прогнозування наступного слова – це одна з найпоширеніших задач обробки природної мови. N-грамні моделі можна використовувати для прогнозування наступного слова в послідовності, використовуючи ймовірність появи кожного слова в якості наступного слова [22]. Наприклад, якщо потрібно прогнозувати наступне слово в послідовності "я люблю", то потрібно обчислити ймовірність появи кожного слова в якості наступного слова, використовуючи біграми, які з'явилися в послідовності.

$$P("тебе" | "я люблю") = \frac{C("тебе", "я люблю")}{C("я люблю")}$$

$$P(тебе/"")$$

У цьому випадку ймовірність того, що наступним словом буде "тебе", дорівнює 1, оскільки в наборі даних, на якому навчалася модель, ця двограма зустрічається тільки один раз.

У тому випадку коли потрібно створити текст про любов, то потрібно почати з генерації випадкового слова, наприклад "любов". Наступним кроком потрібно продовжити текст, генеруючи слова, які, ймовірно, будуть наступними після слова "любов", це можуть бути "любов", "я", "кохаю", "тебе". Цей процес можна повторювати до тих пір, поки не буде створений текстовий документ потрібного розміру.

Отже, n-грамні моделі є потужним інструментом, який можна використовувати для різних задач обробки природної мови. Вони відносно прості у реалізації і можуть бути ефективними для прогнозування наступного слова в послідовності, розпізнавання мов і перекладу мов [23].

2.2 Етапи виконання методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам дає можливість генерувати текстові повідомлення заданої семантичної спрямованості, адже звичайна n-грамна модель не має такої можливості.

Схему етапів виконання методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам подано на рисунку 2.1.

Визначення вхідних даних для навчання моделі становлять важливий етап у процесі розробки методу генерації текстових повідомлень семантичної спрямованості. Ці дані можуть бути надані різними способами, проте в даному випадку метод передбачає введення текстового матеріалу для навчання користувачем [24]. Користувач може надавати власні дані, визначаючи семантичні категорії або теми для текстових повідомлень, що їх він подає на вхід. Це важливо для створення репрезентативної частотної n-грамної моделі.

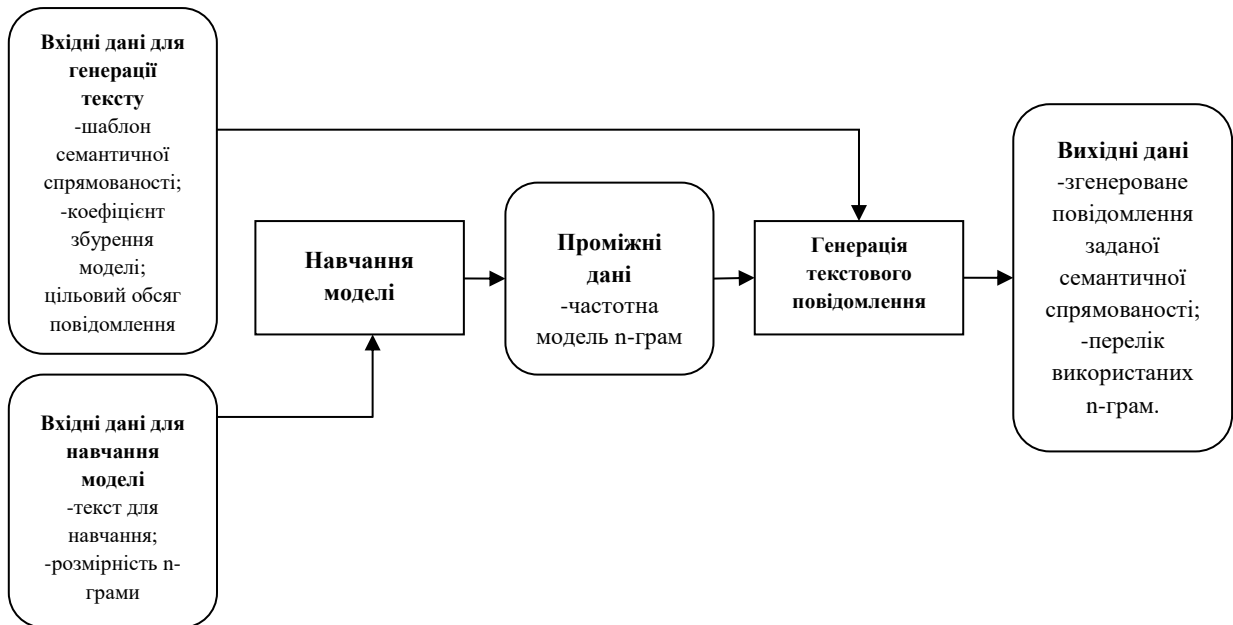


Рисунок 2.1 – Загальна схема етапів виконання методу генерації текстових повідомлень заданої семантичної спрямованості

Блок 1. Навчання моделі є ключовим етапом у процесі розробки методу генерації текстових повідомлень заданої семантичної спрямованості. Після збору вхідних даних, текст розбивається на біграми або триграми. Тоді потрібно побудувати наступну модель частот, яка матиме вигляд такий як в таблиці 2.1.

Таблиця 2.1 – Приклад таблиця частот n-грам

Н-грами	Частоти n-грам
я люблю	10
люблю тебе	5
ти любиш	2

Цей підхід дозволяє моделі аналізувати контекст у вигляді невеликих фрагментів, роблячи більш докладний внутрішній аналіз тексту.

Отримані лексичні одиниці та їх контекст зберігаються в словнику, де кожен ключ відповідає конкретній лексичній одиниці, а значення представляють собою зв'язані з нею контекстуальні фрагменти[25]. Під час навчання моделі

використовуються ці дані для побудови структури, яка враховує семантичні зв'язки та взаємодії між лексичними одиницями.

Після навчання частотної моделі є можливість взаємодіяти з нею, використовуючи збережений словник. Цей словник стає доступним для скачування, що дозволяє зберегти отримані знання та використовувати їх для інших завдань чи аналізу текстів. Крім того, є можливість доповнити або оновити словник, додаючи нові дані та розширюючи його для врахування нових лексичних особливостей або тематичних аспектів. Такий інтерактивний підхід дозволяє моделі неперервно вдосконалюватися та адаптуватися до змін у мовленнєвому потоці та вимогах користувача.

Окрім навченої моделі та отриманих векторних представлень для лексичних n-грам, які представляють проміжні дані, важливо також отримати вхідні дані для генерації текстового повідомлення семантичної спрямованості. Користувачу необхідно ввести текстове повідомлення, що відобразить семантичну спрямованість текстового повідомлення, яке буде згенероване. Також вхідними даними для подальшої генерації повідомлення є обмеження довжини згенерованого повідомлення. Це важливо, особливо при інтеграції з методами обробки та аналізу тексту, де фіксована довжина тексту може бути обмеженою або оптимальною для подальших обчислень.

Блок 2. Після успішного навчання моделі та підготовки проміжних даних, можна згенерувати текстове повідомлення семантичної спрямованості. Під час цього етапу, система використовує отримані векторні представлення та структуру лексичних зв'язків для створення нового тексту, який відповідає вказаному семантичному контексту, з урахуванням введеного користувачем коефіцієнта.

У рамках розробленого методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, враховуються індивідуальні вказівки, через введення коефіцієнта для конкретних tokenів, що може змінювати їхню вагу в процесі генерації. Це означає, що метод має можливість надавати певним лексичним одиницям, наприклад, конкретним

біграмам, більший пріоритет у порівнянні з іншими. Для прикладу вказавши коефіцієнт 2 для певної біграми, це може означати, що кожне появлення цієї біграми в тексті буде мати подвоєну вагу у виборі моделлю при генерації нового тексту. Такий підхід дозволяє забезпечити персоналізацію та керування увагою моделі відповідно до індивідуальних уподобань користувача.

Важливо враховувати, що такий підхід вимагає уважного використання, оскільки надання занадто великих коефіцієнтів може спричинити викривлення семантики текстів або приведе до переваги лише обраних токенів. Правильне калібрування коефіцієнтів та уважне спостереження за впливом на якість генерації забезпечують оптимальну роботу системи для конкретних потреб користувача.

Модель аналізує семантичний контекст чи тему, яку вказав користувач, та використовує збережені векторні представлення, щоб вибрати лексичні n-грами, які найкраще вписуються в цей контекст. З врахуванням коефіцієнтів, які користувач може надати для конкретних токенів, модель визначає вагу кожної лексичної одиниці при формуванні нового тексту. Для прогнозування наступного слова потрібно скористатися таблицею частот.

Отже, в даному розділі подано загальну схему розробленого методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, який поділяється на два блоки – навчання n-грамної моделі та блок генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.

2.3 Навчання n-грамної моделі в методі генерації текстових повідомлень заданої семантичної спрямованості

Структура вхідних даних для навчання n-грамної моделі є складним процесом, який включає в себе обробку текстового документа [26]. Цей текстовий документ може включати в себе різноманітні джерела інформації, такі як книги, статті, веб-сторінки або інші текстові документи.

Схема блоку навчання n -грамної моделі в методі генерації текстових повідомлень заданої семантичної спрямованості подана на рисунку 2.2.



Рисунок 2.2 – Схема навчання n -грамної моделі

Згідно рисунку 2.2 вхідними даними для навчання n -грамної моделі в методі генерації текстових повідомлень заданої семантичної спрямованості є текст, що може бути завантажений як текстовий документ формату .txt або .doc. та тип n -грами.

Варто також відзначити обмеження на розмір текстового документу у форматі .txt або .doc для оптимізації процесу навчання n -грамної моделі в методі генерації текстових повідомлень заданої семантичної спрямованості. З урахуванням великих обсягів тексту, які можуть використовуватися в таких

завданнях, введення максимальної кількості слів чи розміру файлу в байтах дозволить ефективно керувати вхідними даними та прискорити процес створення моделі.

Таким чином, вводячи обмеження на розмір текстового документу, можна додатково оптимізувати використання пам'яті та ресурсів, запобігти великим обсягам обчислень та зменшити час, необхідний для створення n-грамної моделі. Це дозволить забезпечити більш швидко і ефективну реалізацію методу генерації текстових повідомлень, зменшуючи при цьому вплив великих обсягів даних на виконавчий час. Важливо також зазначити, що обрані обмеження повинні враховувати баланс між точністю моделі та обмеженнями обчислювальних ресурсів, щоб досягти оптимального результату. Чим більший корпус, тим модель має більше можливостей вивчати ширший спектр мовних конструкцій та контекстів.

Зазвичай для ефективного навчання використовують великі тексти, такі як великі колекції книг, статей або новинних статей. Це дозволяє моделі вивчати різні стилі письма, теми та лексичні особливості. Однак важливо уникати перенавантаження моделі великим обсягом навчальних даних, оскільки це може призвести до великої кількості параметрів та зайвого шуму в моделі.

Першим кроком в блоці навчання n-грамної моделі є видалення надлишкових символів, які могли бути випадково залишені користувачем при формуванні тексту для навчання, або які не відіграють смислового навантаження у тексті. Це символи /, *,+,@, #, ^, ~ , лишні символи пробілу, які йдуть за підряд, перенос на наступний рядок тощо.

На *другому кроці* відбувається токенизація шляхом формування меж токенів, де кожен текст розбивається на окремі токени. Це створює основні елементи для подальшої аналізу та розпізнавання n-грам. Формування n-грам включає у себе створення послідовностей токенів фіксованої довжини, таких як біграми та триграми. Ці n-грами визначають контекст та взаємодію слів у тексті, що допомагає моделі аналізувати та уявляти мовні зв'язки.

Важливо, що при розробці методу генерації текстових повідомлень семантичної спрямованості на основі лексичних n -грам, в якості токенів розглядаються не лише окремі слова, але й розділові знаки. Урахування розділових знаків як токенів є важливим для покращення точності та натуральності генерованого тексту. Включення розділових знаків до моделі лексичних n -грам дозволяє зберігати правильність синтаксичної структури утвореного тексту. Це особливо актуально для генерації тексту з врахуванням семантичної спрямованості, оскільки розділові знаки можуть впливати на виразність і структуру повідомлення.

Межами токена вважається наявність одного пробілу перед та після слова. Якщо після слова стоїть розділовий знак, до прикладу «,» то він входить в токен, а не вважається окремим токеном (наприклад «слово,»). Якщо після слова стоїть знак пробілу та тире, то знак тире з пробілом перед ним є також одним токеном (наприклад «любов –»).

На *третьому кроці* формується вектор токенів з урахуванням меж токенів, що сформовані на другому кроці, що представляє собою список усіх токенів з вхідного тексту.

Четвертий крок – це побудова множини n -грам, що є одним із ключових кроків у реалізації методу генерації текстових повідомлень заданої семантичної спрямованості. N -грами представляють собою послідовності n слів або токенів, які дозволяють моделі вивчати взаємозв'язки та залежності між словами в тексті. Модель, пройшовши весь корпус тексту, розраховує частоту для кожної n -грами.

П'ятий крок включає в себе можливість збереження створеної моделі для подальшого використання та, в разі необхідності, її завантаження. Збереження моделі дозволяє уникнути повторного навчання при кожному використанні, що є часозберігаючим і дозволяє використовувати модель довільну кількість разів для генерації текстового повідомлення семантичної спрямованості.

Отже, було сформовано схему блоку навчання n -грамної моделі в методі генерації текстових повідомлень заданої семантичної спрямованості, а також детально розглянуто кожен крок в процесі навчання.

2.4 Генерація текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

Генерація тексту за допомогою n-грамної моделі – це процес створення послідовності слів на основі попередньо визначених n-грам, де n вказує на кількість слів у кожній групі.

Схема блоку генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам представлена на рисунку 2.3.



Рисунок 2.3 – Схема генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

Вхідні дані для методу генерації текстових повідомлень включають кілька ключових параметрів, які спрямовують роботу моделі та визначають характеристики згенерованого тексту.

Шаблон семантичної спрямованості визначає слова, які вводить користувач і використовує для визначення тематичної спрямованості генерованих повідомлень [28]. Важливо враховувати, що це може бути не просто одне слово, а й кілька слів, що дозволяє точніше визначити семантичний контекст для текстів.

Коефіцієнт збурення моделі визначає ступінь важливості слів, що складають шаблон семантичної спрямованості та використовуються в процесі генерації тексту. Це число помножується на частоту кожної n-грами, яка містить слова з шаблону семантичної спрямованості, «підсилюючи» n-грами, що містять слова з шаблону семантичної спрямованості, таким чином роблячи акцент на семантичну спрямованість текстового повідомлення, що буде згенероване.

Цільовий обсяг текстового повідомлення визначає бажану кількість слів у згенерованому повідомленні [29]. Важливо враховувати, що це число не завжди буде точно відповідати фактично згенерованому тексту, оскільки враховується необхідність логічного завершення тексту до розділового знаку, що позначає кінець речення.

Частотна модель n-грам є результатом навчання моделі та включає в себе структуру лексичних одиниць, їхні взаємодії та частоти входження в текст. Ця модель визначає, як система розуміє та використовує мовний матеріал при генерації текстових повідомлень.

Крок 1. Помноження частот n-грам, що містять слова з шаблону семантичної спрямованості, на коефіцієнт збурення моделі, що повинен бути більшим за 1 і може приймати також дробові значення. Це дає можливість враховувати важливість конкретних слів чи токенів, зазначених користувачем.

Крок 2. На цьому кроці визначається початкове слово, з якого буде генеруватися текстове повідомлення. У даному методі обирається перше слово з шаблону семантичної спрямованості.

Крок 3. Зворотна генерація передбачає визначення лексичних одиниць, які найкраще підходять для початку повідомлення, враховуючи заданий стартовий токен [30]. Для цього використовується частотна модель n-грам, яка була згенерована на попередньому кроці (після помноження на коефіцієнт збурення відповідних частот n-грам).

Алгоритм зворотної генерації тексту за допомогою n-грам реалізовується як функція, яка приймає як вхід початкове слово, довжину n-грамми та потрібну довжину генерованого тексту. Функція повинна почати з початкового слова і послідовно визначати попереднє слово серед списку n-грам. Потім функція повинна використовувати попереднє слово для генерації наступного слова. Функція повинна повторювати ці кроки, поки не буде досягнуто початку речення. Про початок речення говорить поява слова з великої літери.

Під час генерації тексту важливо враховувати чорний список (блекліст) використаних n-грам, щоб уникнути повторення та забезпечити розмаїтість згенерованого контенту. Чорний список визначає n-грамми, які вже були використані, і допомагає уникнути одноманітності у згенерованому тексті.

Крок 4. На цьому кроці відбувається пряма генерація тексту за допомогою n-грам, де від початкового токена, який був використаний на попередньому кроці, генерується кожне наступне слово за допомогою частотної n-грамної моделі.

На цьому кроці відбувається завершення текстового повідомлення, яке настає після досягнення показника цільового обсягу. Під час завершення тексту враховується логічне завершення речення, що може бути досягнуте за допомогою розділових знаків (".", ",", "!", "?").

Вихідними даними є згенероване текстове повідомлення семантичної спрямованості та перелік використаних n-грам для подальшого використання чи аналізу.

У висновку, наведений метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам є одним із інструментів для автоматичного створення текстів, де враховується семантична

спрямованість [31]. Цей процес дозволяє створювати текст, що відповідає заданим критеріям та може бути використаний у різних сферах, від генерації контенту до розробки програмних рішень для обробки природної мови.

2.5 Способи оцінки якості згенерованих текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

Оцінка якості згенерованих текстових повідомлень є важливою задачею в обробці природної мови. Вона дозволяє оцінити, наскільки добре генератор текстів відповідає заданим вимогам. Одним із підходів до оцінки якості згенерованих текстових повідомлень є використання лексичних n-грам.

Існує кілька способів оцінки якості згенерованих текстових повідомлень з використанням лексичних n-грам [32]. Один із способів полягає в тому, щоб порівняти частоту появи кожної лексичної n-грами в оригінальному тексті і в генерованому тексті. Якщо частота появи лексичної n-грами в генерованому тексті близька до частоти її появи в оригінальному тексті, то це означає, що генератор текстів добре розуміє семантику оригінального тексту.

Інший спосіб оцінки якості згенерованих текстових повідомлень з використанням лексичних n-грам полягає в тому, щоб використовувати статистичні методи для порівняння оригінальних текстів і генерованих текстів. Наприклад, можна використовувати метод k-ближчих сусідів, щоб знайти k найближчих оригінальних текстів до кожного генерованого тексту [33]. Якщо генеровані тексти схожі на оригінальні тексти, то це означає, що вони мають високу якість.

Лексичні n-грами мають ряд переваг у порівнянні з іншими методами оцінки якості згенерованих текстових повідомлень:

- є відносно простими у використанні. Вони можуть бути легко обчислені за допомогою частотних списків слів;

- дозволяють оцінити семантичну спрямованість генерованих текстових повідомлень. Це важливо для завдань, таких як генерація текстів про конкретну тему.

Лексичні n-грами також мають ряд недоліків:

- не можуть оцінити структуру генерованих текстових повідомлень. Це важливо для завдань, таких як генерація творчого тексту;
- можуть бути не дуже точними для оцінки якості генерованих текстових повідомлень, які містять незвичайні або рідкісні слова.

Лексичні n-грами є одним із ефективних методів оцінки якості згенерованих текстових повідомлень заданої семантичної спрямованості [34]. Вони прості у використанні і дозволяють оцінити семантичну спрямованість генерованих текстових повідомлень. Однак лексичні n-грами не можуть оцінити структуру генерованих текстових повідомлень і можуть бути не дуже точними для оцінки якості генерованих текстових повідомлень, які містять незвичайні або рідкісні слова.

Висновки до розділу 2

У цьому розділі було запропоновано метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам. Цей метод заснований на тому, що генератор текстів використовує збурену модель лексичних n-грам, щоб прогнозувати наступне слово в послідовності, а також зворотну генерацію тексту для генерації початку текстового повідомлення.

Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам має ряд перспектив розвитку. Одним із напрямків розвитку є використання більш складних моделей генерації текстів, які можуть враховувати структуру тексту. Це дозволить створювати більш якісні і творчі тексти.

Розділ 3 Інформаційна система генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

3.1 Визначення функцій інформаційної системи

При розробці інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам для визначення функцій інформаційної системи необхідно автоматизувати наступні бізнес-процеси:

- бізнес-процес «Робота з пошуком семантичної спрямованості»;
- бізнес-процес «Робота з обробкою тексту»;
- бізнес-процес «Робота з навчання моделі»;
- бізнес-процес «Робота з особистими даними»;

Бізнес-процес «Робота з пошуком семантичної спрямованості». Даний бізнес-процес надає можливість користувачу отримувати готову семантичну спрямованість, використовуючи для цього тільки декілька слів. Зокрема, автоматизації підлягають наступні функції:

- 1) пошук семантично спрямованого тексту;
- 2) виведення інформації про цей текст.

Бізнес-процес «Робота з обробкою тексту». Даний бізнес-процес надає можливість користувачу переглядати і зберігати словники n-грам.

Бізнес-процес «Робота з навчання моделі». Даний бізнес-процес надає можливість користувачу редагувати і видаляти збережені ним моделі навчання. Зокрема, автоматизації підлягають функції виведення, редагування інформації.

Бізнес-процес «Робота з особистими даними». Даний бізнес-процес надає можливість користувачу працювати з особистими даними. Зокрема, автоматизації підлягають наступні функції:

- 1) реєстрація в системі;
- 2) авторизація в системі.

На початку роботи з системою користувачу виводиться стартова сторінка, де він бачить інформацію для авторизації в продукті. Звідси він може перейти на сторінку «Головна сторінка», попередньо зареєструвавшись, де відображається інформація про користувача та весь функціонал додатку. Також з стартової сторінки користувач може залогінитись та зареєструватись за допомогою GitHub. На рисунку 3.1 зображена UML-діаграма сайту, що зображує описані бізнес-процеси.



Рисунок 3.1 – UML-діаграма варіантів використання сайту

Пройшовши авторизацію, користувач переходить на головну сторінку, де бачить наявний функціонал додатку. З цієї сторінки користувач може почати навчання своєї власної моделі, або завантажити вже готову.

Основними бізнес-процесами даної системи, являються «Генерація повідомлень заданої семантичної спрямованості», «Збереження попередніх моделей навчання», «Сортування тексту на n-грами», «Збереження даних для авторизації».

Бізнес-процес «Генерація повідомлень заданої семантичної спрямованості», відповідає безпосередньо за генерацію текстових повідомлень заданої семантичної спрямованості.

Бізнес-процес «Збереження попередніх моделей навчання», відповідає за збереження словників навчання моделі, для того щоб використовувати їх в майбутньому, або чітко вибираючи тему для навчання.

Бізнес-процес «Сортування тексту на n-грами», відповідає за збереження інформації у вигляді n-грам яку також можна використовувати для аналізу тексту, на найбільш вживані біграми або триграми.

Бізнес-процес «Збереження даних для авторизації», відповідає за збереження інформації користувача при авторизації в системі, а саме логін і пароль. Всі вище зазначені бізнес-процеси зображені на рисунку 3.2

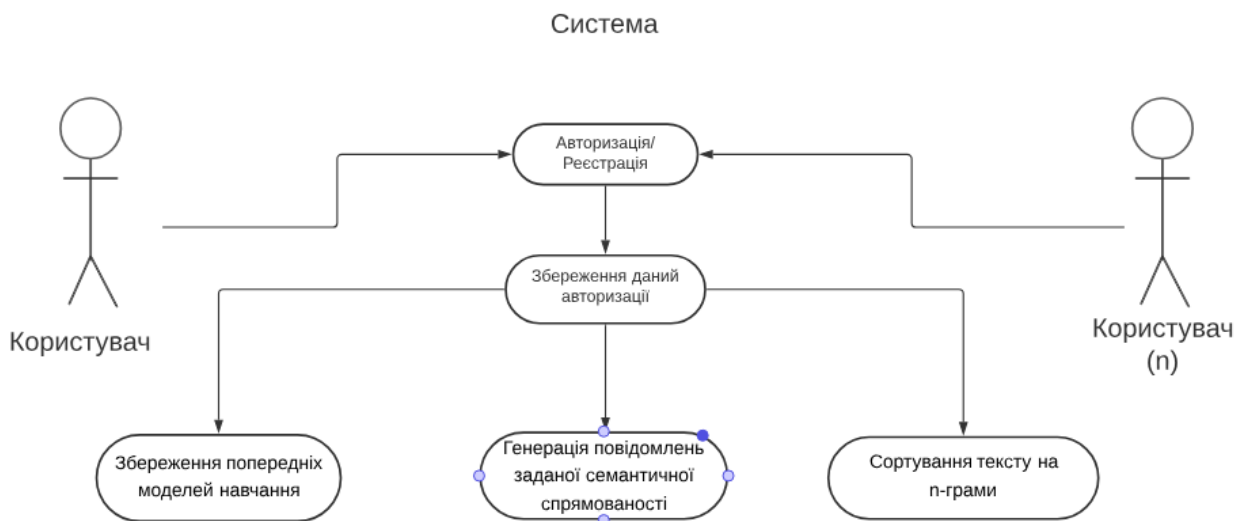


Рисунок 3.2 – UML-діаграма бізнес-процесів сайту

Таким чином, було розглянуто процеси генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, необхідно автоматизувати при розробці інформаційної системи.

3.2 Проектування структури інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості

Проектування структури інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам визначається низкою ключових функцій, що забезпечують ефективну взаємодію користувача з системою. Структура інформаційної системи зображена на рисунку 3.3.

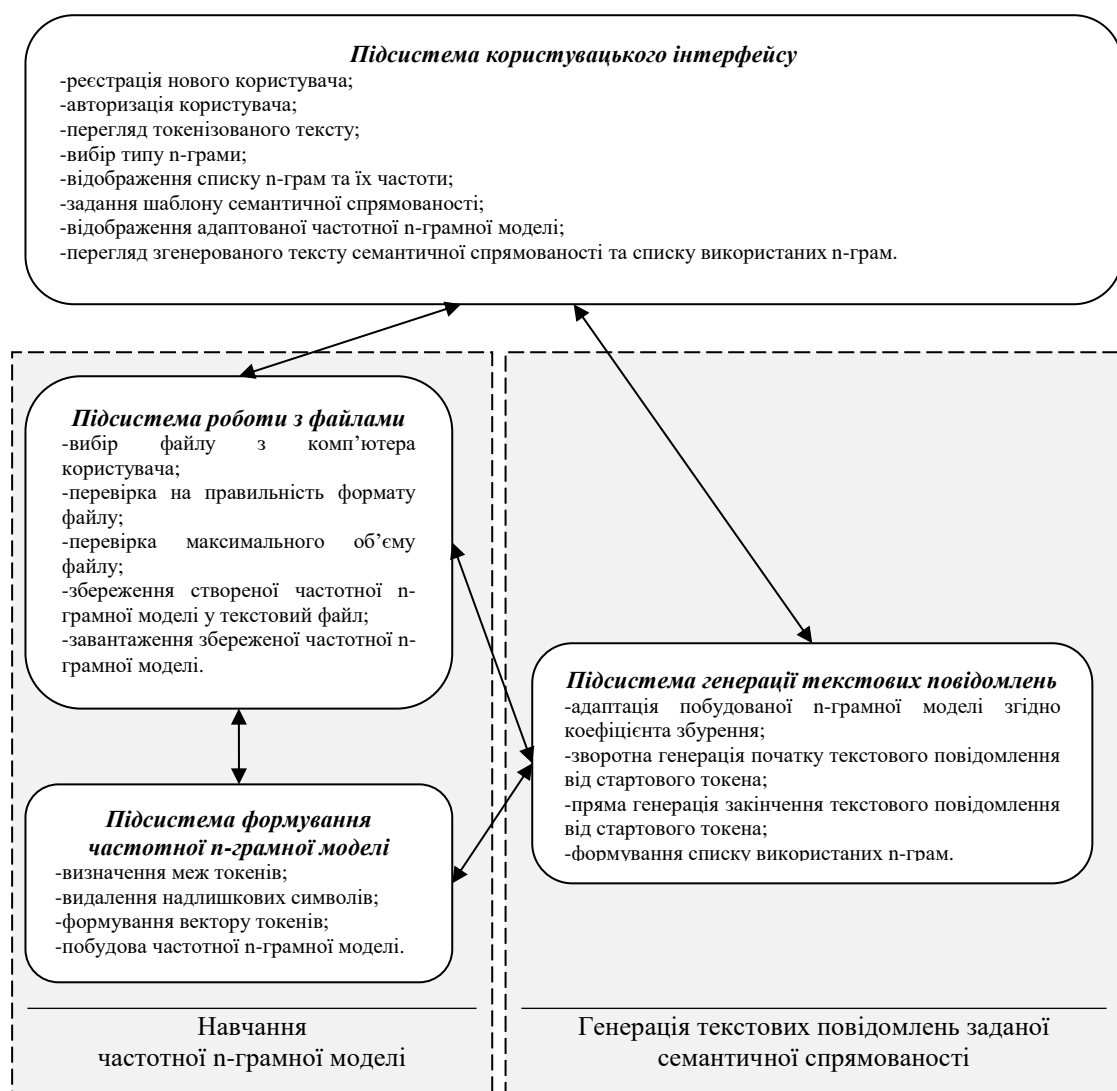


Рисунок 3.3 – Схема інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

Перша підсистема користувачького інтерфейсу – дозволяє користувачам створити обліковий запис, зберігаючи необхідну інформацію. Авторизація забезпечує вхід в систему, перевіряючи ідентифікаційні дані. Після реєстрації та входу, користувач може переглядати токенизований текст, визначати тип n-грам (від 1-грам до більш великих n-грам) та переглядати список n-грам та їх частоту у тексті.

Задання шаблону семантичної спрямованості дозволяє вказати параметри для генерації тексту. Система відображає адаптовану частотну n-грамну модель та дозволяє користувачеві переглядати згенерований текст семантичної спрямованості, а також список використаних n-грам для аналізу та зрозуміння структури тексту. Цей розділ ілюструє весь цикл взаємодії користувача з системою, від початкової реєстрації до аналізу результатів генерації тексту.

У наступній підсистемі визначається функціонал *підсистеми роботи з файлами*, яка спрощує обробку та використання зовнішніх текстових даних у веб-застосунку. Починаючи з вибору файлу користувачем з власного комп'ютера, система здійснює перевірку на правильність формату та максимальний об'єм файлу для запобігання можливим помилкам чи перевантаженню. Після валідації система здійснює збереження створеної частотної n-грамної моделі у текстовий файл, забезпечуючи легкий доступ та подальше використання результатів аналізу тексту. Також, враховуючи потреби користувача, існує можливість завантаження раніше збереженої частотної n-грамної моделі, що сприяє зручності та повторному використанню обраного текстового матеріалу у системі.

Також важливою підсистемою, є *підсистема формування частотної n-грамної моделі*, що є ключовою складовою процесу генерації текстових повідомлень. Визначення меж токенів відбувається для чіткого визначення текстових одиниць та виключення надлишкових символів, забезпечуючи правильну обробку вхідних даних. Після цього виконується етап видалення зайвих символів для оптимізації обробки та аналізу тексту. Формування вектору токенів визначає структуру даних, яка буде використовуватися для побудови

частотної n-грамної моделі. На завершальному етапі відбувається побудова самої частотної n-грамної моделі, що враховує частоту входження послідовностей токенів у вихідний текст. Ця підсистема визначає основні кроки підготовки та аналізу текстових даних для подальшого використання у процесі генерації тексту.

Останньою підсистемою є підсистема, в якій описується *ключовий функціонал генерації текстових повідомлень*, та яка базується на адаптації n-грамної моделі з урахуванням коефіцієнта збурення. Процес адаптації визначає, як система реагує на зміни у вихідних даних, оптимізуючи n-грамну модель для покращення результатів генерації. Здійснюється зворотна генерація текстового повідомлення від стартового токена, що дозволяє системі створювати зміст, виходячи з конкретного початкового слова чи фрази. Пряма генерація закінчення текстового повідомлення від стартового токена визначає, як система завершує структуру повідомлення. Окрім того, формується список використаних n-грам для подальшого аналізу та зрозуміння структури згенерованого тексту, створюючи комплексний інструмент для управління та оптимізації генерації текстового контенту.

Розроблена система генерації текстових повідомлень інтегрує в себе ефективні підсистеми для роботи з файлами, генерації текстового контенту та формування частотної n-грамної моделі. Підсистема роботи з файлами дозволяє користувачам легко завантажувати, зберігати та використовувати текстові дані, забезпечуючи високий рівень зручності та можливість повторного використання існуючих моделей. Ключова підсистема генерації текстових повідомлень використовує адаптовану n-грамну модель з урахуванням коефіцієнта збурення, оптимізуючи результати генерації та дозволяючи користувачеві контролювати структуру та зміст створеного тексту. Завершальна підсистема формування частотної n-грамної моделі визначає основні етапи обробки текстових даних, що забезпечує їхню коректну підготовку для подальшого використання у процесі генерації текстового контенту. Усі ці підсистеми спільно створюють

комплексний та ефективний інструмент для управління та оптимізації генерації текстового матеріалу.

3.3 Вибір засобів розробки інформаційної системи генерації текстових повідомлень

Для реалізації інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам вибір веб-застосунку є найбільш вдалим рішенням. Веб-застосунок дозволяє забезпечити широкий доступ до системи через браузер, що робить його зручним та легким у використанні для користувачів з будь-якого пристрою.

Такий тип застосунку надає можливості для взаємодії з користувачем, введення параметрів генерації тексту та отримання результатів через інтуїтивний веб-інтерфейс. Крім того, веб-застосунок легше підтримувати та оновлювати, адже зміни вносяться на серверному рівні, і користувачам не потрібно встановлювати або оновлювати додатки на своїх пристроях.

Для розробки системи генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам використовувалося інтегроване середовище розробки (IDE) Visual Studio Code [35]. Це легке та потужне середовище, забезпечене розширеннями для підтримки різноманітних мов програмування та інструментів. Visual Studio Code володіє зручним інтерфейсом, підтримкою вбудованих інструментів контролю версій, інтеграцією з терміналом та великим спектром розширень, що полегшує роботу з кодом. Серед його позитивних аспектів слід відзначити легку вагу та високу швидкість, широкий вибір розширень, вбудовану підтримку Git та крос-платформенність. Активна спільнота та відкритий код додають плюси у формі постійної підтримки та можливості адаптації. Однак, для деяких великих проектів може бути відчутне обмеження функціональності. Також, відсутність повноцінних інструментів для тестування та обмежена підтримка .NET можуть бути недоліками для певних розробників. Не всі розширення рівнофункціональні

на всіх платформах, і для деяких спеціалізованих завдань, таких як графічний дизайн чи мобільна розробка, інші інструменти можуть бути більш підходящими. Усупереч цим відмінностям, Visual Studio Code залишається одним із популярних інструментів у світі розробки завдяки своїй ефективності та гнучкості. Його широкі можливості в редагуванні тексту, відладці та взаємодії з системами контролю версій зробили його важливою частиною роботи над даним продуктом. Інтерфейс даного редактору коду, зображений на рисунку 3.1.

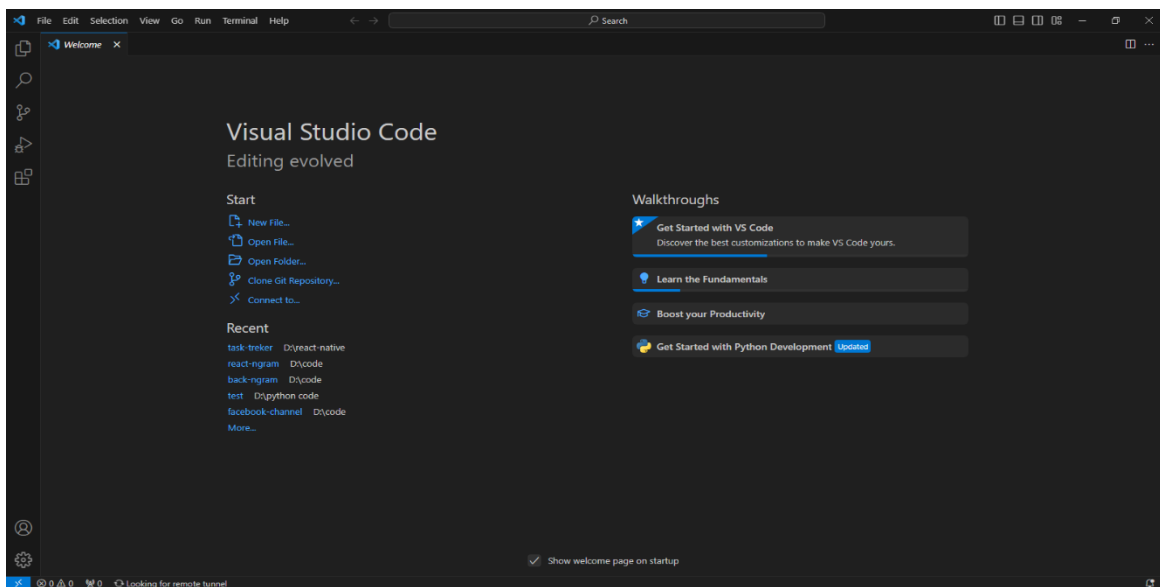


Рисунок 3.1 – Visual Studio Code

Для реалізації back-end частини даної роботи було обрано мову програмування Python [36]. Python є потужною та гнучкою мовою, яка дозволяє швидко розробляти та тестувати програмний код. Її читабельний синтаксис та велика екосистема бібліотек роблять її ідеальним вибором для веб-розробки та обробки даних. Крім того, наявність фреймворків, таких як Flask [37], спрощує створення стабільних та масштабованих веб-застосунків. Використання Python дозволило зосередитися на функціональності та швидкості розробки, сприяючи успішному завершенню розробки інформаційної системи. Мова програмування Python вирізняється своєю легкістю вивчення, читабельністю коду та великою спільнотою розробників. Завдяки безкоштовності, крос-платформенності та величезному вибору бібліотек та фреймворків, Python став популярним

інструментом для розробки веб-застосунків, аналізу даних та штучного інтелекту. Однак його обмежена швидкодія у виконанні обчислювально інтенсивних завдань та обмежена підтримка мобільної розробки можуть бути факторами вибору інших мов для конкретних завдань. Все ж, Python залишається важливою та впливовою мовою в сфері програмування, приваблюючи розробників своєю простотою та розширюючи свій вплив завдяки активній спільноті та ефективній інтеграції. Головна сторінка офіційного сайту Python, з якого можна завантажити актуальну версію, зображена на рисунку 3.2

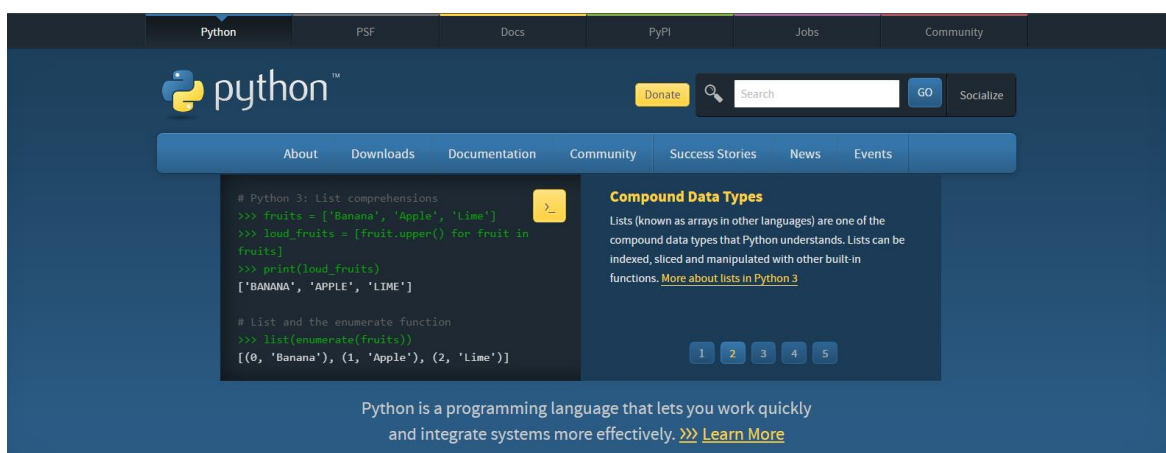


Рисунок 3.2 – Головна сторінка Python

В поєднанні з мовою програмування Python, для створення, back-end частини даного проекту використовувався фреймворк Flask.

Flask – це легкий та гнучкий веб-фреймворк для мови програмування Python. Його мінімалістична структура дозволяє ефективно розробляти веб-застосунки, забезпечуючи лише необхідний набір інструментів для створення потужних back-end рішень. Flask спрощує обробку маршрутів, керування HTTP-запитами та відповідями, що дозволило мені зосередитися на функціональності мого проекту. Завдяки простоті синтаксису та модульності, Flask став вибором для початківців та розробників, які цінують гнучкість у роботі. Вбудований розробницький сервер спрощує тестування, а велика активна спільнота забезпечує підтримку та безліч розширень. Незважаючи на свою легкість, Flask може виявитися менш практичним для великих проектів, де важлива більш

продумана структура та вбудовані опції, але він залишається потужним інструментом для швидкої та ефективної розробки веб-застосунків. Його розширювана архітектура також дозволила інтегрувати додаткові компоненти за необхідності, забезпечуючи гнучкість та простоту розробки. Головна сторінка Flask зображена на рисунку 3.3.



Рисунок 3.3 – Головна сторінка Flask

Вся робота з фронтенд-частиною даного продукту велась з бібліотекою React [38]. React – це декларативна та ефективна JavaScript бібліотека для створення інтерфейсів користувача. Її компонентний підхід дозволяє розбити інтерфейс на невеликі, повторно використовувані частини, що полегшує розробку та утримання коду. Virtual DOM оптимізує швидкодію оновлення інтерфейсу, покращуючи продуктивність веб-застосунків [39]. Визначений концепцією компонентного програмування, React дозволяє створювати веб-застосунки з ефективним та динамічним взаємодіючим інтерфейсом. Його основна перевага полягає в віртуальному DOM, що забезпечує оптимізовану роботу з DOM-елементами та покращує продуктивність. З React можна легко інтегрувати розробку фронтенду з іншими технологіями, а також використовувати у поєднанні з різними стеками технологій, роблячи його вибором для розробників великих та масштабованих проектів. Застосування

React у поєднанні з іншими бібліотеками, такими як Redux для керування станом [40], допомогло забезпечити ефективний та динамічний фронтенд для моєї роботи. На рисунку 3.4 зображено головну сторінку сайту React.

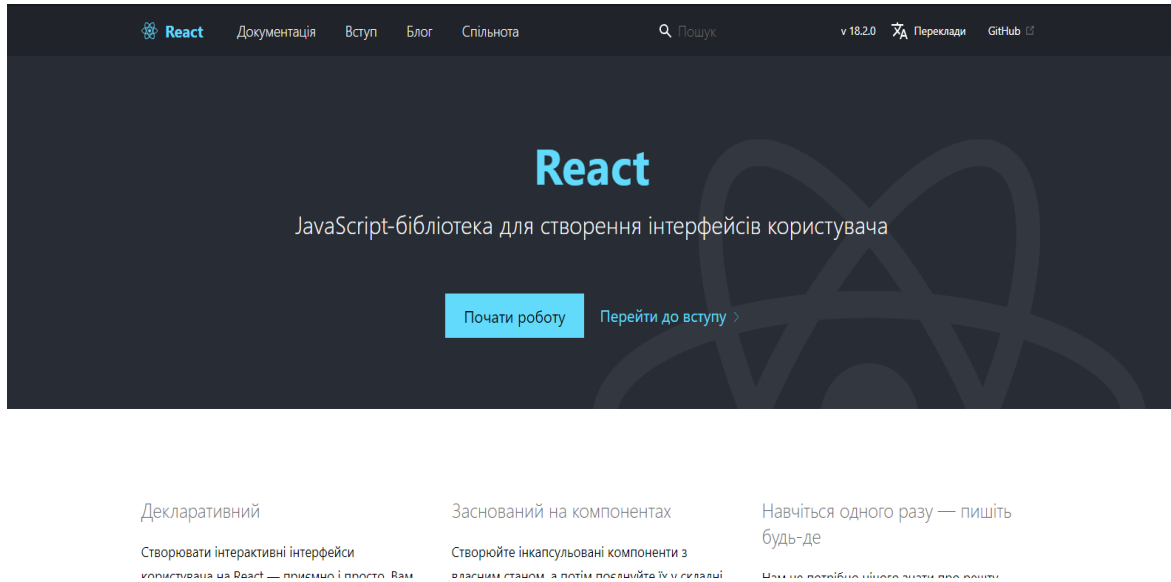


Рисунок 3.4 – Головна сторінка React

В поєднанні з React, для керування станом фронтенд-додатку в даному проєкті використовувалась бібліотека Redux.

Redux – це управління станом в застосунках на основі JavaScript. Він працює на принципі централізованого зберігання стану застосунку, який може бути змінений тільки за допомогою чистих функцій, відомих як "reducers". Actions і reducers взаємодіють так, щоб забезпечити однозначність та прогнозованість стану додатку. Redux полегшує відслідковування та управління станом великих та складних додатків, що робить його ідеальним вибором для фронтенд-розробки, особливо в поєднанні з бібліотеками, такими як React. Його головною ідеєю є централізація стану додатку, що полегшує управління складним станом компонентів та поліпшує передбачуваність додатка. Заснований на принципах одностайного потоку даних, Redux дозволяє ефективно відстежувати та змінювати стан застосунку, роблячи його особливо корисним для великих та складних проєктів. Використовуючи концепції дій, редукторів та магазинів, Redux надає чітку та ефективну архітектуру для

управління станом додатків, сприяючи впорядкуванню коду та покращенню його тестируемості. Використання Redux дозволило ефективно керувати станом компонентів та забезпечити консистентність даних у всьому додатку. На рисунку 3.5, зображена головна сторінка Redux.

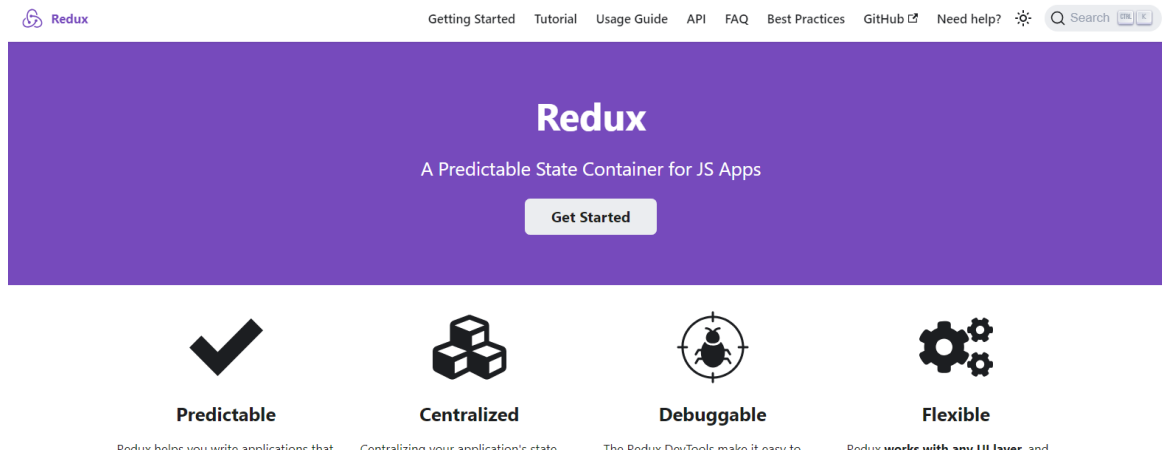


Рисунок 3.5 – Головна сторінка Redux

Також не менш важливою бібліотекою в проєкті є NLTK. NLTK (Natural Language Toolkit) – це бібліотека для мови програмування Python, яка надає інструменти для роботи з природною мовою. NLTK містить широкий спектр функцій, таких як токенізація, стемінг, лематизація, розпізнавання іменованих сутностей, аналіз синтаксису та інші. Ця бібліотека є корисною для обробки текстової інформації, в тому числі для вивчення та аналізу природної мови в проєктах машинного навчання або обробки текстових даних. NLTK також містить корпуси текстових даних та ресурси для навчання та експериментів у галузі обробки природної мови.

Використання NLTK спрощує велику частину завдань, пов'язаних з обробкою тексту, і робить його популярним інструментом серед дослідників та розробників. Головна сторінка бібліотеки NLTK зображена на рисунку 3.6.

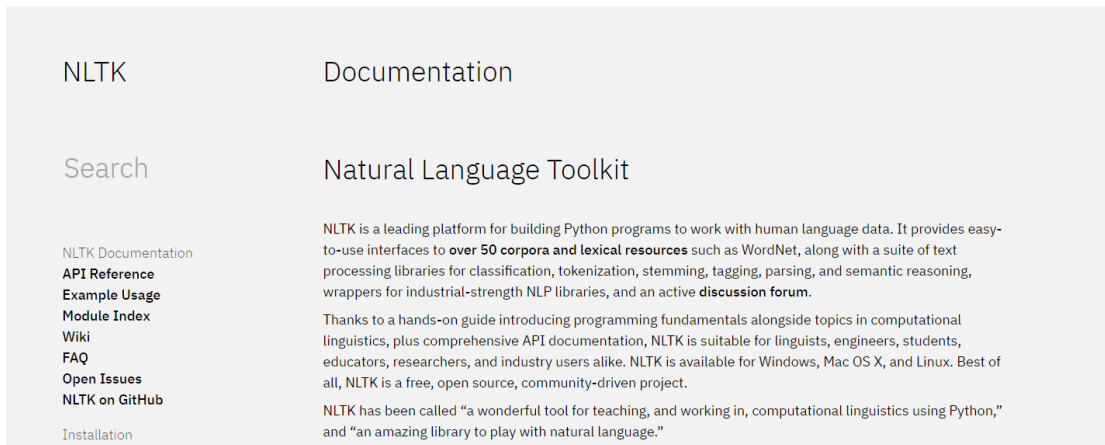


Рисунок 3.6 – Головна сторінка бібліотеки NLTK

Отже, обрана комбінація засобів розробки інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n -грам забезпечить швидкий та зручний процес розробки.

Висновки до розділу 3

Розроблена система генерації текстових повідомлень використовує розроблений метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n -грам та складається з трьох підсистем: підсистеми роботи з файлами, підсистеми генерації текстового контенту та підсистеми формування частотної n -грамної моделі.

Підсистема роботи з файлами забезпечує користувачам можливість легкого завантаження, зберігання та використання текстових даних. Це дозволяє користувачам використовувати існуючі моделі, а також створювати власні.

Підсистема генерації текстового контенту використовує адаптовану n -грамну модель з урахуванням коефіцієнта збурення. Це дозволяє оптимізувати результати генерації та контролювати структуру та зміст створеного тексту.

Підсистема формування частотної n -грамної моделі визначає основні етапи обробки текстових даних. Це забезпечує їхню коректну підготовку для подальшого використання у процесі генерації текстового контенту.

Усі три підсистеми спільно створюють комплексний та ефективний інструмент для управління та оптимізації генерації текстового матеріалу.

Впровадження передових технологій у даному проєкті, таких як Visual Studio Code для зручного кодування та Flask для побудови ефективного back-end, створює надійний та легко масштабований фреймворк для розробки веб-додатків. Вибір мови програмування Python для back-end гарантує гнучкість та потужність, а використання бібліотеки NLTK спрощує взаємодію з текстовою інформацією. Застосування React та Redux на фронтенді додає динамізм та спрощує управління станом, створюючи вразливий та ефективний інтерфейс. Всі ці технології гармонійно взаємодіють, формуючи сучасний та продуктивний додаток, готовий до викликів сучасного веб-середовища.

Розділ 4 Дослідження ефективності методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

4.1 Структура інформаційної системи генерації текстових повідомлень та взаємозв'язок її компонентів

Дана інформаційна система призначена для генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам використовує розроблений метод. Вона складається з двох основних частин: фронтенду та бекенду. Фронтенд реалізований за допомогою бібліотеки React, яка дозволяє ефективно створювати інтерактивні користувацькі інтерфейси. Використання бібліотеки Redux спрямоване на забезпечення централізованого управління станом додатку. Redux дозволяє зберігати та керувати одним станом, що полегшує взаємодію між різними компонентами та забезпечує послідовність та передбачуваність управління даними.

Механізм авторизації через Google, який використовується в додатку, надає додатковий рівень безпеки та зручності для користувачів. При використанні цього механізму, користувачі можуть зручно та безпечно авторизуватися за допомогою своїх облікових записів Google, що важливо для забезпечення довіри та конфіденційності даних.

Для здійснення взаємодії з сервером та обміну даними, в додатку використовується бібліотека Axios. Axios дозволяє легко виконувати HTTP-запити до сервера, отримувати та відправляти дані. Це особливо корисно для отримання або відправлення інформації з сервера, наприклад, при авторизації чи отриманні даних для відображення на сторінці. Користуючись Axios, додаток може ефективно взаємодіяти з серверною частиною, забезпечуючи швидкість та надійність в обміні даними між фронтендом та бекендом.

Для ефективної навігації між різними частинами додатку та управління маршрутами URL використовується бібліотека React Router. React Router дозволяє визначати сторінки та їхні маршрути, забезпечуючи відображення

відповідних компонентів при зміні URL. Це дозволяє користувачам легко переміщатися між різними частинами додатку та забезпечує взаємодію з різними відображеннями на основі їхнього контексту та потреб. Завдяки React Router, додаток стає більш динамічним та легко розширюється у відповідь на зміни вимог до навігації. Схема фронтенд частини додатку зображена на рисунку 4.1

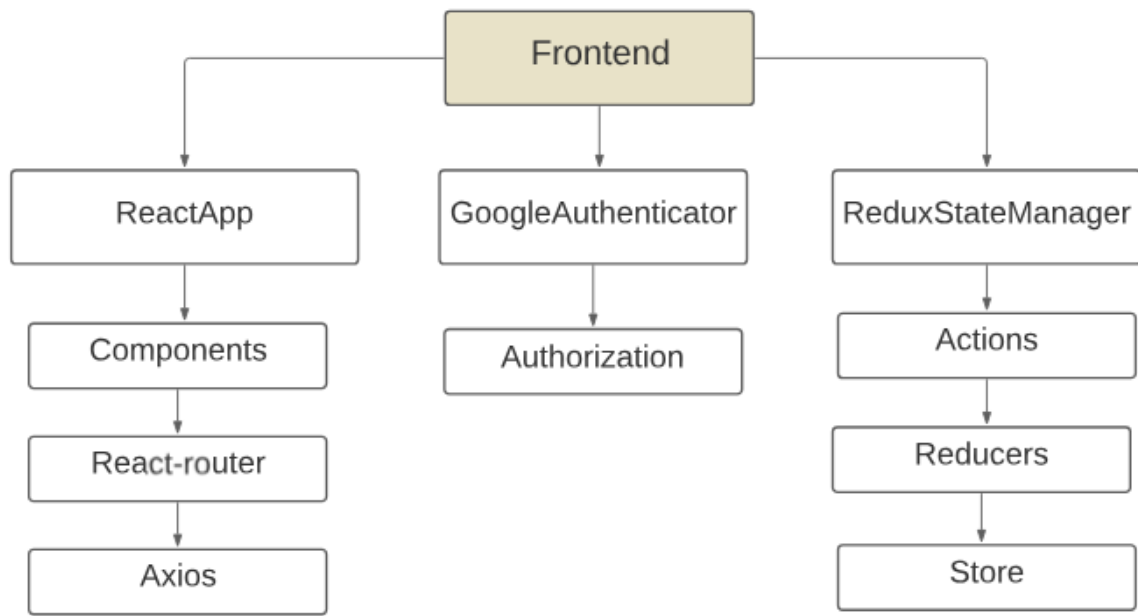


Рисунок 4.1 – Схема фронтенд частини додатку

Бекенд системи реалізований за допомогою мови програмування Python та фреймворка Flask. Flask дозволяє швидко створювати веб-застосунки та легко розширювати їх функціональність. Використання бібліотеки NLTK (Natural Language Toolkit) для обробки тексту дозволяє враховувати лексичні n-грами при генерації текстових повідомлень.

Взаємодія між фронтендом та бекендом здійснюється за допомогою API-запитів. Фронтенд взаємодіє з бекендом, виконуючи запити на отримання та збереження даних. Авторизація через Google здійснюється на фронтенді, а потім передається на бекенд для перевірки та отримання необхідних даних. API-запити обробляються в Flask за допомогою APIHandler, який визначає різні маршрути (routes). Наприклад, `route_1()` та `route_2()` можуть відповідати на різні типи запитів і взаємодіяти з іншими компонентами бекенду.

Важливою частиною роботи між фронтенд і бекенд частиною додатку є використання механізму CORS. Бекенд використовує механізм CORS, щоб дозволити взаємодію з фронтендом, який працює на іншому домені. Це забезпечує безпечну і ефективну взаємодію між різними частинами системи.

Текстові дані, які використовуються для генерації повідомлень, зберігаються у відповідному текстовому документі. Бекенд відповідає за обробку та аналіз цих даних, використовуючи бібліотеку NLTK для роботи з лексичними n-грамами. Схема бекенд частини додатку зображена на рисунку 4.2

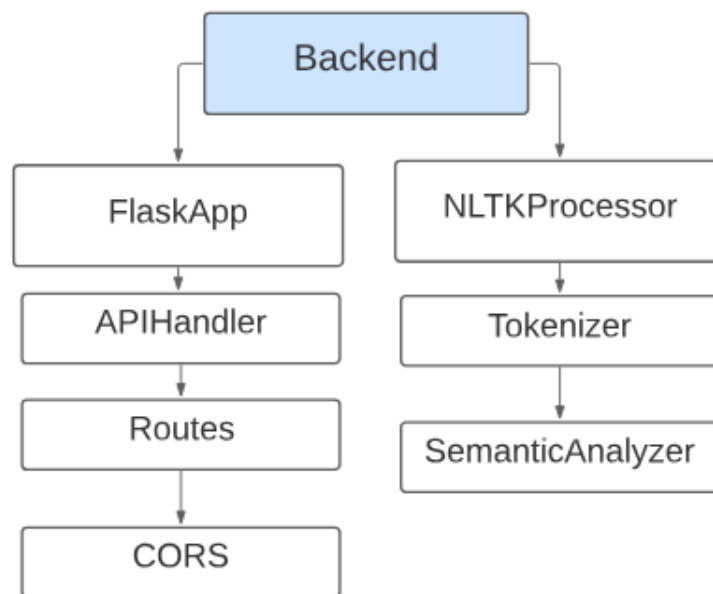


Рисунок 4.2 – Схема бекенд частини додатку

Тому у даному випадку використання Flask дозволяє ефективно створювати та розширювати веб-застосунки, забезпечуючи гнучкість та швидкість розробки. Важливим елементом функціональності бекенду є використання бібліотеки NLTK для обробки тексту та врахування лексичних n-грам при генерації текстових повідомлень. Це дозволяє враховувати семантичні зв'язки та забезпечує точність в генерації контенту. Описано важливий механізм взаємодії між фронтендом і бекендом за допомогою API-запитів, а також використання механізму CORS для забезпечення безпечної взаємодії між різними доменами.

Тому ці технології є важливою складовою для забезпечення ефективної та надійної взаємодії між фронтендом та бекендом, враховуючи сучасні вимоги до обробки текстового контенту при генерації текстових повідомлень заданої семантичної спрямованості.

4.2 Особливості реалізації інформаційної системи

Для реалізації інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості описуються технічні та архітектурні аспекти проекту, які дозволяють забезпечити ефективну та надійну функціональність системи. Далі будуть розглядатися ключові технологічні комбінації засобів розробки, структурні рішення та основні компоненти системи, які визначають її архітектуру та взаємодію між частинами.

В даному додатку використовується фреймворк Flask для реалізації серверної логіки та обробки HTTP-запитів. Flask визначає основну архітектурну основу системи, забезпечуючи легкість розробки та гнучкість у побудові веб-застосунків. Доступність широкого спектру розширень дозволяє легко вбудовувати додатковий функціонал.

Однією з ключових особливостей даного додатку, є використання бібліотеки NLTK для обробки природної мови. NLTK впроваджує техніки токенізації, визначення n-грам, аналізу семантики та інші, що дозволяє системі ефективно опрацьовувати текстові дані. Це особливо корисно для генерації текстових повідомлень з урахуванням семантичного контексту.

Токенізація визначає розділення тексту на окремі "токени" або слова, що дозволяє подальший аналіз та обробку. NLTK (Natural Language Toolkit) надає зручні функції для проведення токенізації тексту та визначення n-грам.

Фронтенд інформаційної системи побудований на бібліотеці React, яка надає зручний інструментарій для створення інтерактивних користувацьких інтерфейсів. Redux використовується для централізованого управління станом додатку, що полегшує керування станом та взаємодію між компонентами.

Важливою частиною фронтенд додатку, є його можливість динамічно переключатися між різними сторінкам. Для цього доцільно використовувати React Router. Ця бібліотека для навігації між сторінками в веб-додатках React. Забезпечуючи динамічну зміну контенту без перезавантаження сторінки, React Router дозволяє створювати багатосторінкові додатки зі зручним та інтуїтивно зрозумілим інтерфейсом.

Система генерації текстових повідомлень заданої семантичної спрямованості використовує механізм авторизації через Google для забезпечення безпеки та зручності користувачів. Користувачі можуть безпечно авторизуватися за допомогою своїх облікових записів Google, що підвищує рівень довіри та конфіденційності даних.

Взаємодія між фронтендом та бекендом реалізована з використанням CORS для дозволу взаємодії з веб-застосунком з різних доменів. Це забезпечує безпечну та ефективну взаємодію між різними частинами інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості.

4.3 Тестування функціональних можливостей інформаційної системи генерації текстових повідомлень

Тестування функціоналу інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості є критично важливою частиною розробки для забезпечення якості та надійності програмного забезпечення. В даному випадку, оскільки розробляється система генерації текстових повідомлень, тестування повинно зосередитися на перевірці правильності та ефективності цього процесу.

Для початку використання інформаційної системи генерації текстових повідомлень з визначеною семантичною спрямованістю, користувачу необхідно пройти авторизацію. Для цього необхідно використати обліковий запис Google або GitHub. Натиснути на відповідну кнопку "Увійти з Google" або "Увійти з GitHub" та слідувати вказівкам для введення облікових даних, що зображено на

рисунку 4.3. Цей етап не тільки забезпечить доступ до усіх функціональних можливостей системи, але й гарантує безпеку та конфіденційність інформації. Після успішної авторизації буде отримано повний доступ до інструментів генерації тексту та налаштувань системи. Також цей етап описаний текст-кейсом в таблиці 4.1.

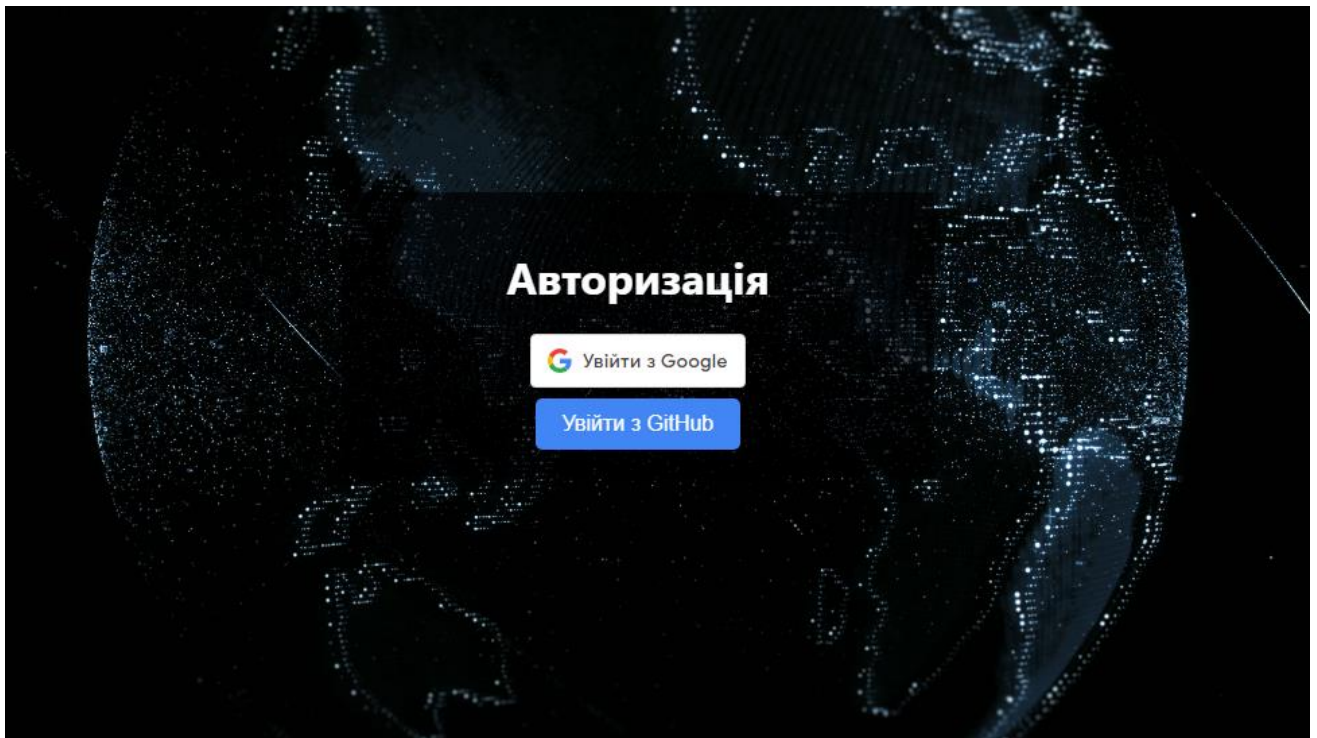


Рисунок 4.3 – Сторінка авторизації в системі

Після успішної авторизації користувач потрапляє на головну сторінку програми, де відображений весь функціонал інформаційної системи генерації текстових повідомлень з визначеною семантичною спрямованістю. Програма розглядається як система, розділена на два ключові блоки, що зображено на рисунку 4.4.

Перший блок це місце набору інструментів для ефективного навчання семантичної n-грамної моделі. Є можливість завантажувати тексти для навчання з файлів у форматах txt або doc, щоб розширити базу даних і поліпшити точність моделі, що описано тест-кейсом в таблиці 4.1. Також існує можливість ручного введення тексту в спеціальне поле для введення. Одна із функцій дозволяє користувачеві оглядати словник n-грам, що включає в себе усі вивчені та

збережені словосполучення. Також важливою функцією є скачування словника, що дозволяє користувачеві зберегти навчальні дані для подальшого використання чи аналізу. При виборі типу n-грами, користувач може оптимізувати навчання, визначаючи, чи використовувати біграми чи триграми для генерації тексту. Цей блок забезпечує користувача всіма необхідними інструментами для ефективного навчання моделі та налаштування параметрів генерації текстових повідомлень.

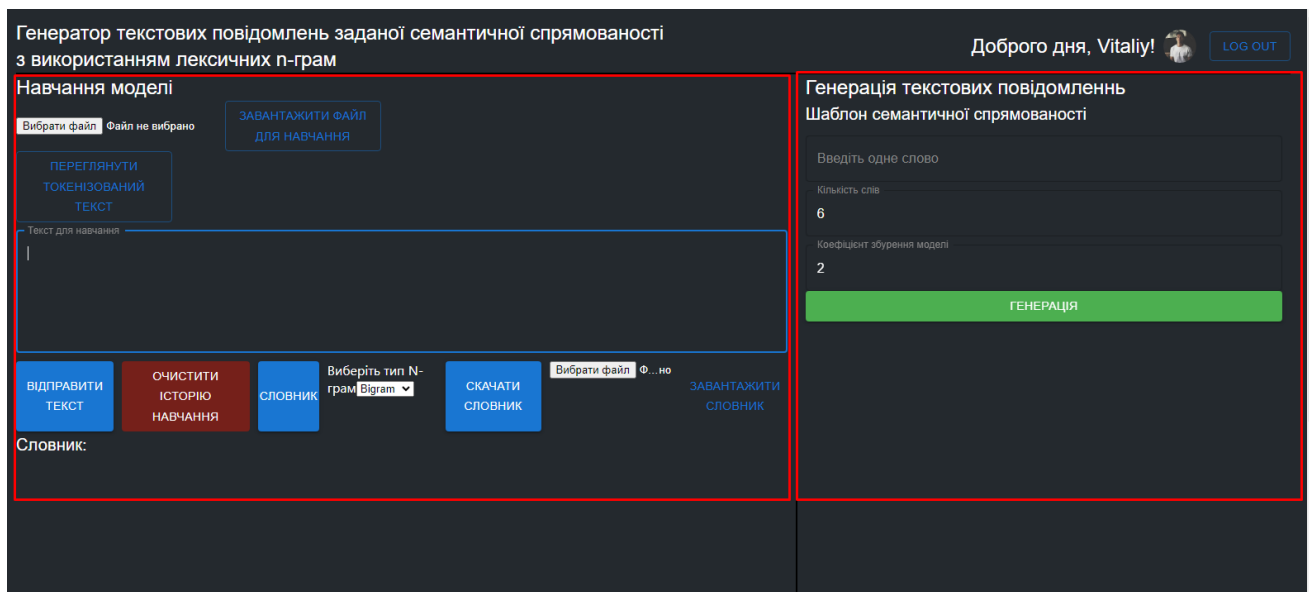


Рисунок 4.4 – Головна сторінка програми

Для того щоб згенерувати повідомлення заданої семантичної спрямованості, потрібно ввести або завантажити текст моделі, після чого відправити його на навчання, що зображено на рисунку 4.5.

Наступний блок відповідає саме за генерацію тексту заданої семантичної спрямованості та врахування встановлених параметрів користувача. Є можливість генерувати текстові повідомлення, використовуючи навчену n-грамну модель. Введення кількості слів для генерації дають можливість користувачеві контролювати структуру та обсяг створюваного тексту. Введення коефіцієнта збурення дозволяє користувачеві впливати на різноманітність та семантичну спрямованість генерованого контенту. Ключовий функціонал, який дозволяє користувачеві визначати шаблон семантичної спрямованості для

створюваних повідомлень, це введення ключових слів, що дозволяє користувачеві визначити тематику та стиль генерованого тексту.

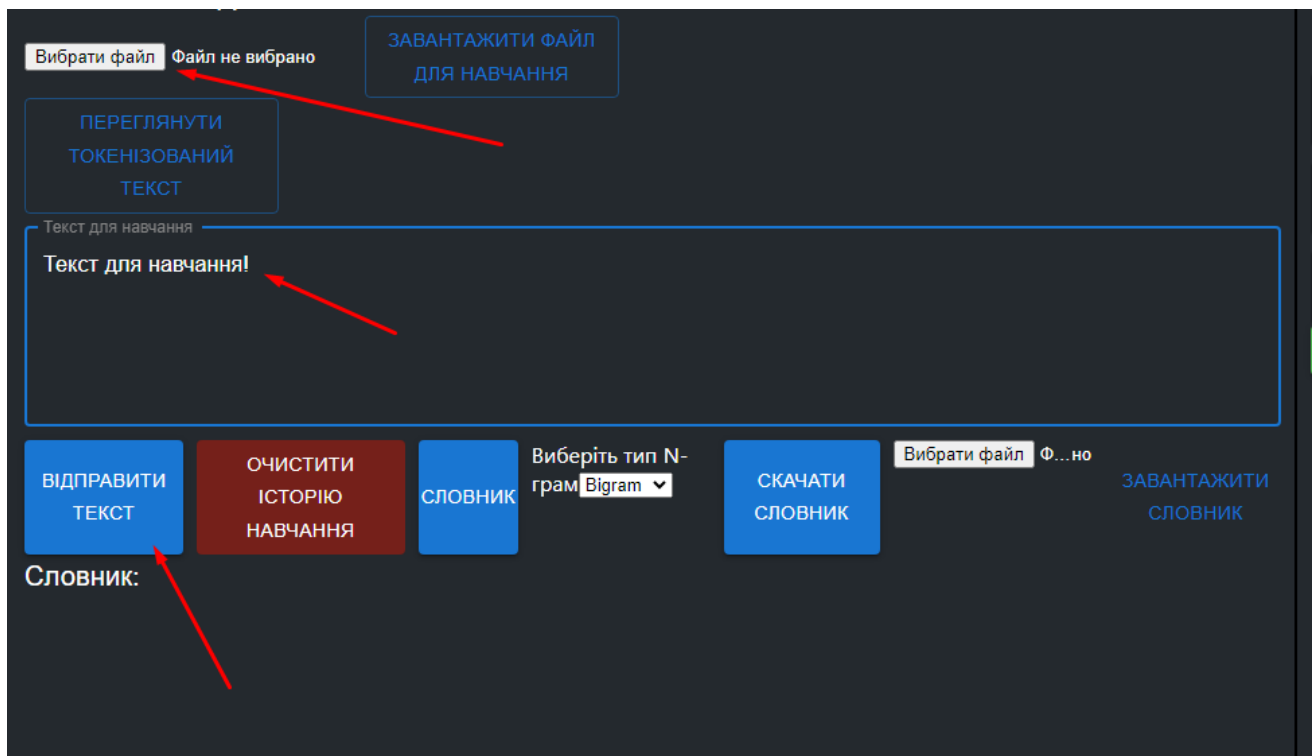


Рисунок 4.5 – Відправка даних для навчання

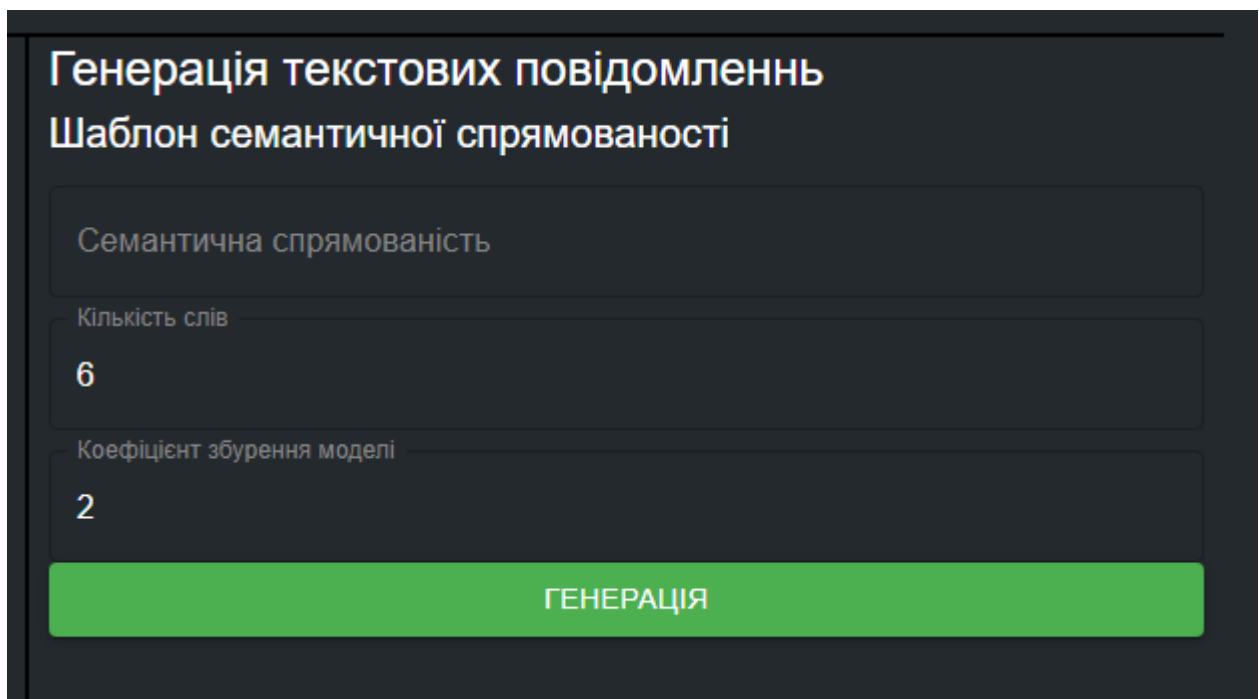


Рисунок 4.6 – Генерація текстових повідомлень

Тому для початку генерації тексту, потрібно вказати обмеження слів для генерованого повідомлення, а також коефіцієнт збурення. Після чого натиснути кнопку генерація, що зображено на рисунку 4.6.

В результаті взаємодії з інформаційною системою генерації текстових повідомлень з визначеною семантичною спрямованістю, користувач отримає можливість створювати текстовий контент, який точно відповідає його заданим параметрам семантичної спрямованості. Ця система дозволяє ефективно та гнучко генерувати текстовий контент, враховуючи специфічні потреби та стиль користувача.

Для тестування даного додатку використовувалися юніт-тести. Юніт-тести – це невеликі фрагменти коду, які призначені для перевірки окремих функцій або компонентів програми. Вони допомагають переконатися, що кожна частина коду працює правильно та відповідає визначеним специфікаціям.

```
def test_generate_message(self):
    seed_word = 'apple is mobile'
    num_words = 5
    n_gram_type = 'bigram'

    try:
        generated_message = generate_message(seed_word,
num_words, n_gram_type)
        self.assertIsInstance(generated_message, str)
        print("test_generate_message.....OK")
    except AssertionError as e:
        print(f"test_generate_message.....FAILED: {e}")
```

Результат проходження таких тестів відображається в консолі розробника із належним статусом до кожного із юніт тестів. Результат проходження юніт тестів зображений на рисунку 4.4.

```

PS D:\code\back-ngram> & C:/Python312/python.exe d:/code/back-ngram/unitTest.py
[nltk_data] Downloading package punkt to
[nltk_data]   C:\Users\Vitaliy\AppData\Roaming\nltk_data...
[nltk_data] Package punkt is already up-to-date!
test_delete_text....OK
.test_generate_message....OK
.test_get_bigram_dict....OK
..test_process_text....OK
.
-----
Ran 5 tests in 0.039s

OK
PS D:\code\back-ngram>

```

Рисунок 4.4 – Результат проходження Юніт тестів

Таблиця 4.1 – Тест-кейси завантаження файлу з текстом для навчання

Тест-Кейс ID: 1.1.1	Пріоритет: 1	Створено: 23.11.2023
Назва: Перевірка коректності завантаження тексту для навчання моделі		
Вхідні дані: Файл з текстом в форматі txt або doc		
Кроки		Очікуваний результат
<ol style="list-style-type: none"> 1. Натиснути кнопку «Вибрати файл» 2. Вибрати файл з текстом для навчання 3. Натиснути кнопку «Завантажити файл для навчання» 		Поява вікна із попередженням про успішне завантаження файлу
Результат виконання тест-кейсу: пройдено успішно		

Також було проведено три тест-кейса. В цих тест-кейсах виконується перевірка ключових функцій системи генерації тексту. Перший тест визначає правильність додавання тексту за допомогою файлу, до корпусу для навчання моделі. Другий тест підтвердив правильність додавання тексту за допомогою поля для вводу. Система демонструє здатність до точного відтворення семантичного контексту та спрямованості в згенерованих повідомленнях, що робить її високоефективною для задач генерації тексту з визначеною семантичною спрямованістю. Третій тест оцінює коректність очищення

збереженого тексту. Кожен тест враховує різні аспекти функціоналу, щоб переконатися в правильності роботи системи та її здатності до ефективної обробки даних. Загальний висновок вказує на успішний розвиток та оптимізацію системи генерації тексту для досягнення балансу між лаконічністю та збереженням семантичного змісту відповідно до конкретних вимог та уподобань користувача. Результат даних тест-кейсів зображено в таблиці 4.1. Результат проходження тест-кейсу зображено на рисунку 4.5.

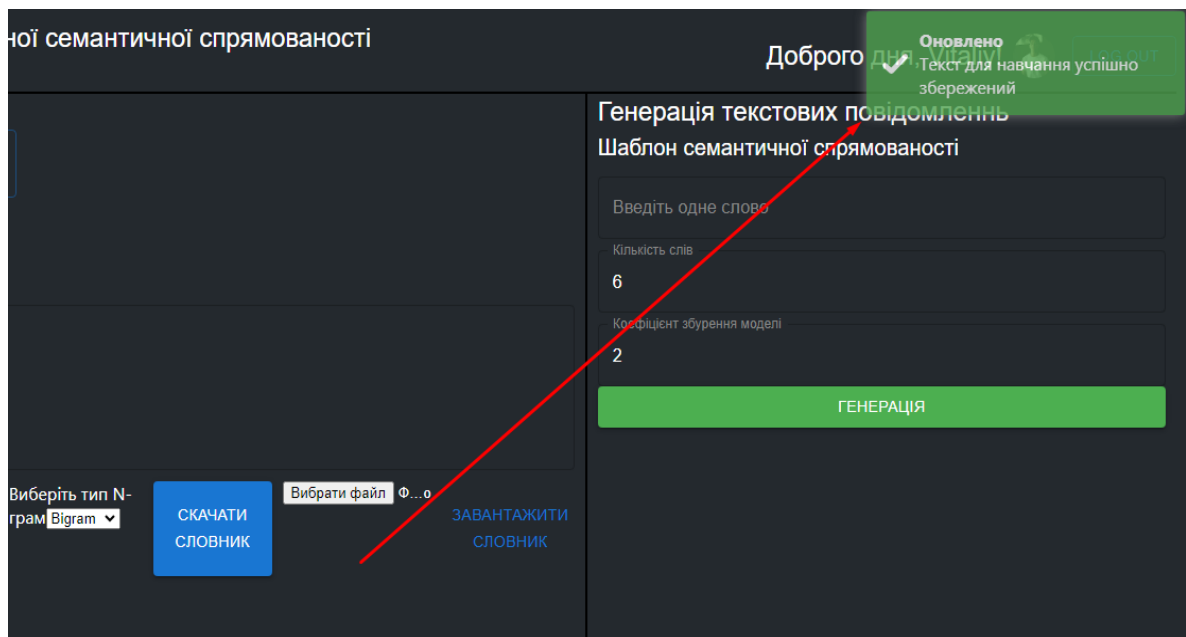


Рисунок 4.5 – Результат проходження тест-кейсу «Перевірка коректності завантаження тексту для навчання моделі»

В рамках тест-кейсу з таблиці 4.2, назва якого "Перевірка коректності завантаження тексту з поля для вводу" основною метою є перевірка правильності функціонування механізму завантаження текстового контенту з введеного користувачем поля. У цьому сценарії тестування вхідні дані обмежуються невеликим об'ємом тексту, що може бути введений або скопійований. Кроки тестування включають натискання на поле для вводу тексту "Текст для навчання", введення чи вставку тексту для навчання та подальше натискання кнопки "Відправити текст". Очікується, що після цих дій система повинна взаємодіяти із введеним текстом і виводити повідомлення про

успішне завантаження тексту. Успішне виконання тест-кейсу підтверджується результатом "пройдено успішно". Цей тест допомагає впевнитися в тому, що функціонал завантаження тексту працює належним чином, забезпечуючи користувачеві зручний та надійний інтерфейс для взаємодії з додатком. Результат виконання тест кейсу зображено на рисунку 4.6.

Таблиця 4.2 – Тест-кейс завантаження файлу з текстом для навчання

Тест-Кейс ID: 0.0.2	Пріоритет: 1	Створено: 23.11.2023
Назва: Перевірка коректності завантаження тексту з поля для вводу		
Вхідні дані: Невеликий об'єм тексту		
Кроки		Очікуваний результат
<ol style="list-style-type: none"> 1. Натиснути на поле для вводу тексту «Текст для навчання» 2. Ведіть текст для навчання або вставте скопійований 3. Натиснути кнопку «Відправити текст» 		<p>Поява повідомлення із попередженням про успішне завантаження тексту</p>
Результат виконання тест-кейсу: пройдено успішно		

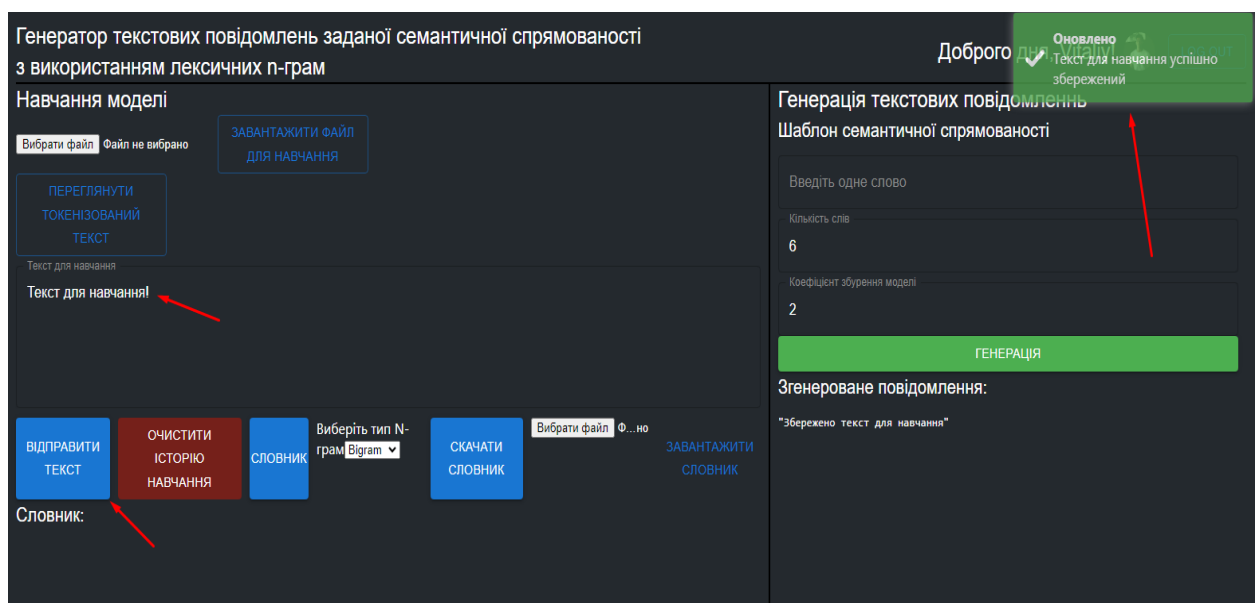


Рисунок 4.6 – Результат проходження тест-кейсу «Перевірка коректності завантаження тексту з поля для вводу»

У тест-кейсі з таблиці 4.3 виконується перевірка коректності очищення історії навчання. Даний тест визначає процедуру, яка спрямована на перевірки у правильності функціонування опції очищення історії навчання моделі.

Вхідні дані обмежуються невеликим об'ємом тексту, який може бути введений або скопійований у поле "Текст для навчання". Кроки тестування включають натискання на вказане поле, введення чи вставку тексту для навчання та подальше відправлення тексту за допомогою кнопки "Відправити текст". Після цього виконується крок, який передбачає натискання кнопки "Очистити історію навчання".

Очікується, що після виконання цих дій система повинна взаємодіяти із введеним текстом, виводити повідомлення про успішне завантаження тексту, а також підтверджувати успішне очищення історії навчання. Результат виконання тест-кейсу повинен бути позначений як "пройдено успішно". Цей тест допомагає переконатися, що опція очищення історії працює належним чином, надаючи користувачеві зручний та надійний інтерфейс для управління історією навчання моделі. Результат виконання тест кейсу зображено на рисунку 4.7.

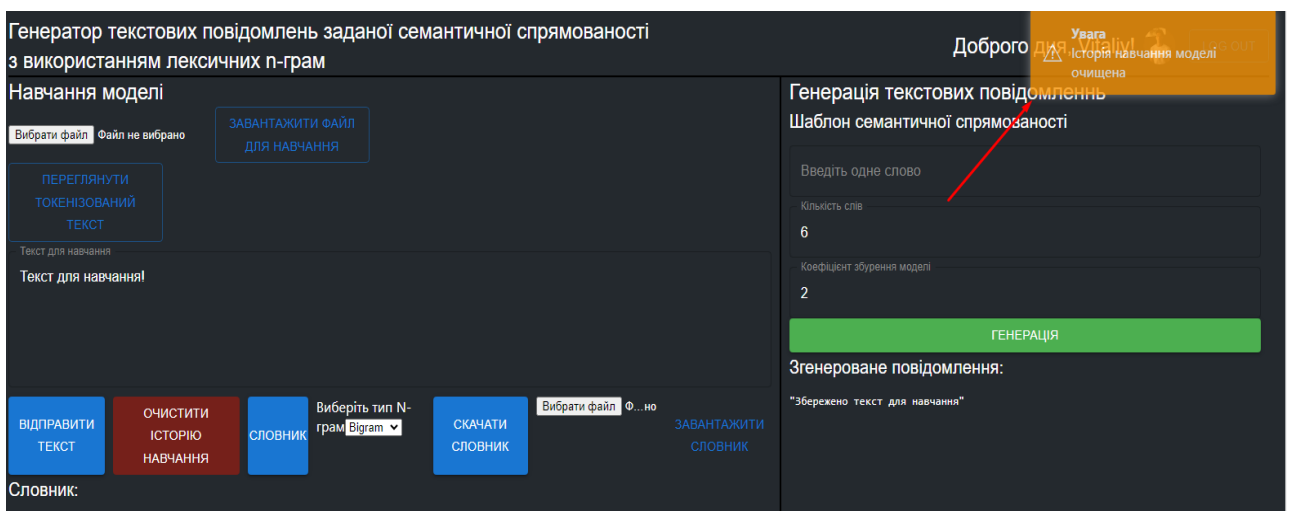


Рисунок 4.7 – Результат проходження тест-кейсу «Перевірка коректності очищення історії навчання»

Таблиця 4.3 – Тест-кейс коректності очищення історії навчання

Тест-Кейс ID: 0.0.3	Пріоритет: 1	Створено: 23.04.2023
Назва: Перевірка коректності очищення історії навчання		
Вхідні дані: Невеликий об'єм тексту		
Кроки		Очікуваний результат
<ol style="list-style-type: none"> 1. Натиснути на поле для вводу тексту «Текст для навчання» 2. Ведіть текст для навчання або вставте скопійований 3. Натиснути кнопку «Відправити текст» 4. Натиснути кнопку «Очистити історію навчання» 		Поява повідомлення із попередженням про успішне очищення історії навчання моделі
Результат виконання тест-кейсу: пройдено успішно		

У тест-кейсі з таблиці 4.4 проводиться перевірка коректності авторизації в системі за допомогою опції "Увійти через Google". Цей тест визначає, чи відбувається правильна авторизація користувача, який вибрав опубліковуй запис Google для входу в систему.

У вхідних даних передбачено наявний обліковий запис Google. Кроки тестування включають натискання на кнопку "Увійти через Google", вибір власного опублікованого запису та подальше натискання кнопки "Підтвердити".

Очікується, що після цих дій система повинна взаємодіяти із сервісом Google, отримати валідні дані для авторизації та відобразити головну сторінку системи із наявним функціоналом. Результат виконання тест-кейсу має бути відмічений як "пройдено успішно". Цей тест допомагає перевірити, чи коректно взаємодіє система з сервісом Google під час авторизації користувача та чи надає вірний доступ до функціоналу. Результат виконання тест кейсу зображено на рисунку 4.8.

Таблиця 4.4 –Тест-кейс перевірки коректності авторизації в системі

Тест-Кейс ID: 0.0.4	Пріоритет: 1	Створено: 23.11.2023
Назва: Перевірка коректності авторизації в системі		
Вхідні дані: Опубліковуй запис Google		
Кроки	Очікуваний результат	
<ol style="list-style-type: none"> 1. Натиснути кнопку «Увійти через Google» 2. Вибрати свій обліковий запис 3. Натиснути кнопку «Підтвердити» 	Поява головної сторінки системи із наявним функціоналом	
Результат виконання тест-кейсу: пройдено успішно		

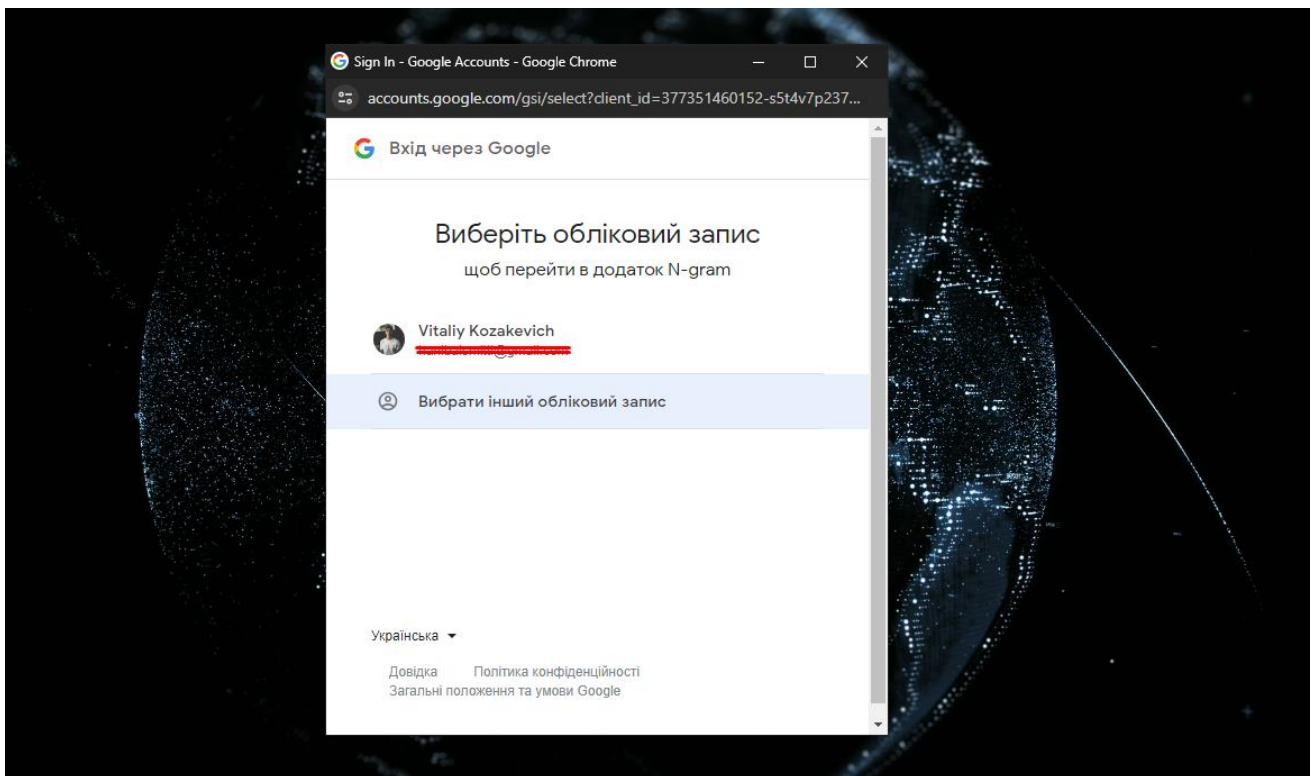


Рисунок 4.8 – Результат проходження тест-кейсу «Перевірка коректності авторизації в системі»

У загальному контексті тест-кейси, визначають і валідують ключові функціональності системи. Перевірки коректності завантаження тексту для навчання, очищення історії навчання та авторизації через Google надають

повноту та точність в роботі системи. Тестування обмежених об'ємів тексту, взаємодія з інтерфейсом та коректність авторизації відображають готовність системи до реального використання. Усі ці тест-кейси вказують на високий рівень надійності та функціональності системи, що дозволяє впевнено вважати їх пройденими успішно. Тест-кейси, які включають в себе перевірку завантаження тексту, очищення історії навчання та процес авторизації, становлять важливий комплекс контролю якості для системи. Їх успішне виконання свідчить про ефективність функціоналу, що пропонуються користувачам. Враховуючи великий пріоритет управління текстовим контентом та безпеки доступу, висновок можна зробити про готовність системи до продуктивного використання та задоволення потреб користувачів. Ці тест-кейси допомагають не лише перевірити основні функціональності, але й гарантують, що система демонструє стабільність та надійність в усіх аспектах її використання.

4.4 Дослідження ефективності генерації текстових повідомлень заданої семантичної спрямованості

Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам дає можливість генерувати текстові повідомлення заданої семантичної спрямованості, адже звичайна n-грамна модель не має такої можливості. В аналізі тексту застосовуються різні види n-грам для отримання контексту та визначення частот слів. Два основних варіанти – біграми та триграми – володіють своїми перевагами та недоліками. Розглянуто обидва підходи та розглянуто ситуації, в яких один вид n-грам може бути більш ефективним за інший.

Біграми, які представляють собою пари сусідніх слів у тексті, мають свої власні переваги та недоліки. Однією з основних переваг біграм є їхня компактність, що робить їх легшими для обробки та зберігання в порівнянні з великими n-грамами. Вони також можуть забезпечувати деякий рівень контексту, що корисно для завдань, де важливий локальний зв'язок між словами.

З іншого боку, біграми мають обмежену спроможність враховувати довший контекст та складні взаємозв'язки між більш ніж двома словами. Це обмеження може призвести до втрати інформації про глибокий зміст тексту, особливо в тематично багатих або технічно спеціалізованих областях. Також слід враховувати, що біграми можуть недостатньо виявляти взаємодії та контекст при аналізі більш складних текстових даних. Для проходження тесту, в додатку додана можливість перегляду словників з біграм. Для прикладу було взято твір «Кайдашева сім'я». Даний словник зображений на рисунку 4.9.

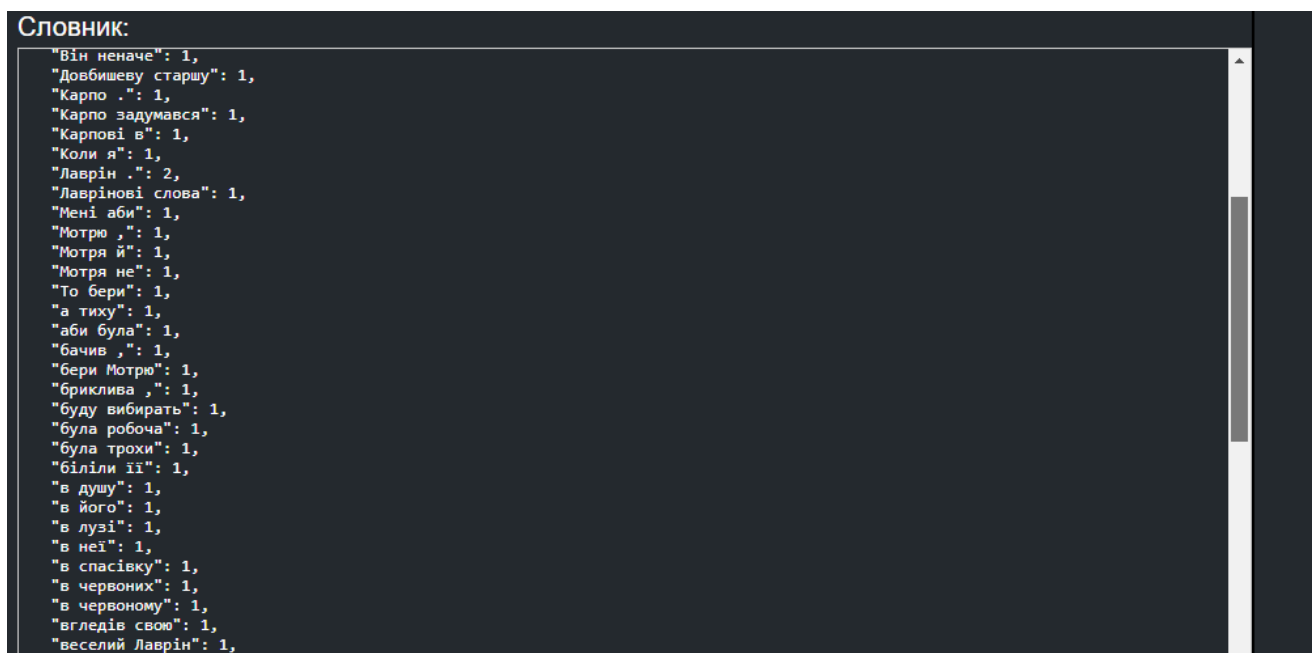


Рисунок 4.9 – Словник біграм

Триграми, що складаються з трьох послідовних слів у тексті, володіють своїми перевагами та недоліками. Основною перевагою триграм є їхня спроможність взяти до уваги більший контекст та виявляти складні взаємодії між словами, що робить їх ефективними для аналізу довших текстів та розкриття глибоких змістовних зв'язків. Триграми також можуть допомагати розрізняти контексти та виражати тонкі семантичні відтінки. Однак використання триграм може збільшити обсяг даних та вимагати більше ресурсів для обробки. Також слід враховувати, що у випадку коротших текстів чи обмежених тематикою документів триграми можуть бути менш ефективними та менш інформативними.

Тому вибір між використанням триграм залежить від конкретних вимог завдання та особливостей текстових даних. На рисунку 4.10 зображено словник триграм.

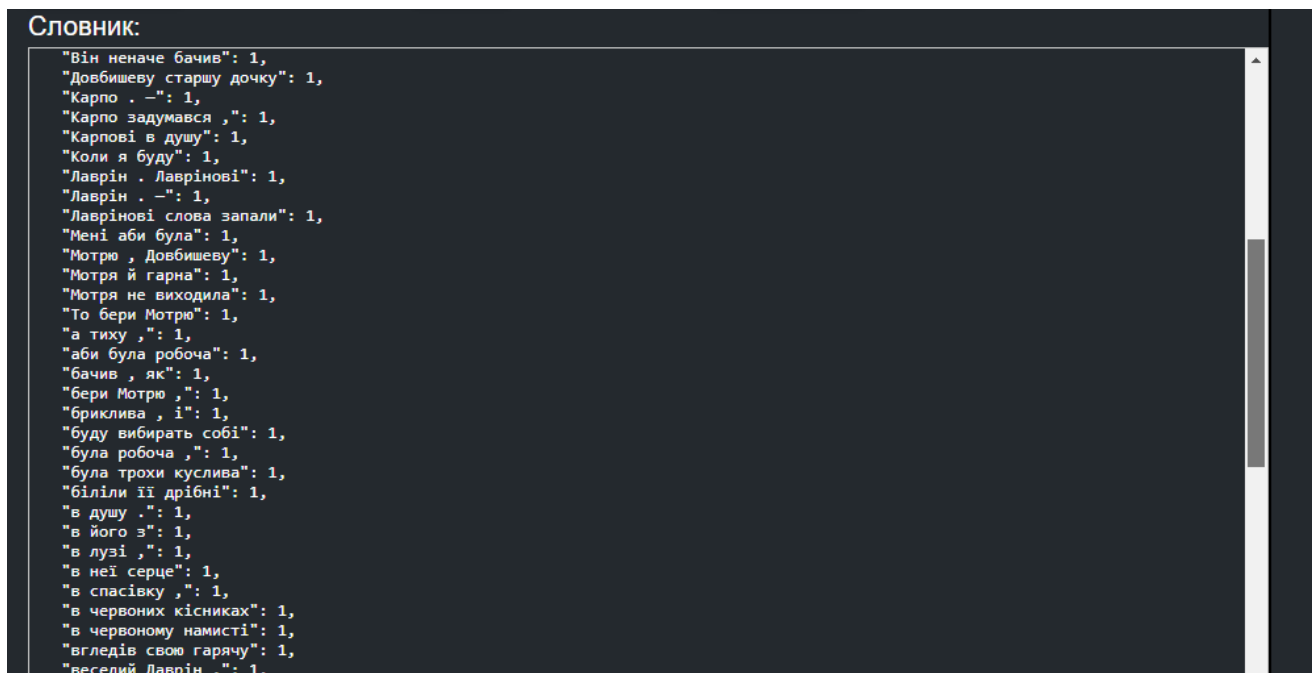


Рисунок 4.10 – Словник триграм

Один з важливих параметрів методу – це кількість слів у згенерованому повідомленні. Проведено тестування з різною кількістю слів, спрямоване на визначення оптимального значення для досягнення балансу між змістом та зрозумілістю. Використовуючи тестові дані та об'єктивні критерії оцінки, досліджено, як кількість слів впливає на якість згенерованих повідомлень. Для прикладу, для навчання моделі було взято уривок із твору «Кайдашева сім'я», а як вхідне слово, було вибрано слово «гарячу мрію». Для першого тесту було вказано 6 слів для генерації. В результаті було отримано повідомлення «вгледів свою гарячу мрію в червоних». Результат зображено на рисунку 4.11.

Наступним було проведемо тестування з обмеженням кількості слів у згенерованому повідомленні до 6, з метою визначення впливу цього обмеження на якість та структуру згенерованого тексту. Вхідним словом було вибрано знову слово «гарячу мрію». В результаті було отримано таке повідомлення «де він ніби вгледів свою гарячу мрію в червоних кісниках на голові». Результат зображено на рисунку 4.12.

Генерація текстових повідомлень

Шаблон семантичної спрямованості

Семантична спрямованість

гарячу мрію

Кількість слів

6

Коефіцієнт збурення моделі

2

ГЕНЕРАЦІЯ

Згенероване повідомлення:

"вгледів свою гарячу мрію в червоних"

Рисунок 4.11 – Згенероване повідомлення із 6 слів

Генерація текстових повідомлень

Шаблон семантичної спрямованості

Введіть одне слово

гарячу мрію

Кількість слів

12

Коефіцієнт збурення моделі

2

ГЕНЕРАЦІЯ

Згенероване повідомлення:

"де він ніби вгледів свою гарячу мрію в червоних кісниках на голові"

Рисунок 4.12 – Згенероване повідомлення із 12 слів

Після проведення тестування методу генерації текстових повідомлень із обмеженням у 6 та 12 словах можна дійти висновку, що метод показав відмінну ефективність при генерації текстів обмеженої довжини в 6 слова. Генеровані повідомлення виявилися лаконічними та зрозумілими. Незважаючи на обмежену кількість слів, метод зумів зберегти семантичний контекст та вираження семантичної спрямованості.

Для прикладу, в таблиці 4.5 показано результати роботи програми із різними коефіцієнтами збурення. Для цього використовувався один і той самий текст семантичної спрямованості, із використанням різного коефіцієнту збурення і різними типами n-грам.

Таблиця 4.5 – Результати роботи програми із різними коефіцієнтами збурення

Використаний тип n-грами	Коефіцієнт збурення	Кількість слів у цільовому текстовому повідомленні	Результат – згенероване текстове повідомлення семантичної спрямованості
Біграма	1	10	«Лаврінові слова запали Карпові в лузі , як мухи.»
Біграма	2	10	«Лаврінові слова запали Карпові в лузі , – сказав він»
Триграма	1	10	«Лаврінові слова запали Карпові в лузі , як мухи.»
Триграма	2	10	«Лаврінові слова запали Карпові в лузі , – сказав він»

Також метод продемонстрував адаптацію до обмеження в 12 слів, дотримуючись зрозумілості та виражаючи значущий зміст у відносно невеликому обсязі. Семантичний контекст і спрямованість повідомлень були в цілому збережені, хоча були відзначені деякі обмеження у глибині вираження.

Обмеження кількості слів виявилось ефективним стратегічним підходом до контролю довжини генерованих текстів. Якщо користувачеві важлива лаконічність та врахування семантичної насиченості, обмеження в 6 слова може бути оптимальним вибором. З іншого боку, обмеження в 12 слів дозволяє генерувати більш деталізовані повідомлення, зберігаючи при цьому загальний семантичний характер. Далі слід розглядати можливості налаштування методу для забезпечення оптимального балансу між довжиною тексту та збереженням семантики відповідно до конкретних потреб користувача.

У результаті проведення тестувань визначено важливість параметру – кількості слів у згенерованому тексті. Експерименти із різною кількістю слів були спрямовані на визначення оптимального значення, забезпечуючи баланс між змістом та зрозумілістю. Використовуючи тестові дані та об'єктивні критерії оцінки, вивчено вплив кількості слів на якість згенерованих повідомлень. Досліджено також три тест-кейси, які перевірили ключові функції системи генерації тексту. В цілому, метод демонструє високу ефективність при генерації текстів обмеженої довжини та забезпечує збереження семантичного контексту та спрямованості при обмеженій кількості слів.

Сучасні технології обробки природної мови, зокрема системи генерації тексту на базі штучного інтелекту, надають унікальні можливості для дослідження ефективності та семантичної спрямованості в області текстового вивчення. В рамках цього дослідження, акцент робиться на використанні методу генерації текстових повідомлень з використанням лексичних n-грам.

Під час експерименту проаналізовано відповіді чат-бота на ключові слова, з метою оцінки семантичної спрямованості генерації тексту даною моделлю. Це дозволить визначити оцінку якості згенерованого тексту. Для тестування було використано частину твору «Кайдашева сім'я», а словом для семантичної спрямованості було використано «Карпові слова». Дані були вказані в програмі, що вказано на рисунку 4.13 і були введені початкові параметри, після чого отримано результат у вигляді речення «Лаврінові слова запали Карпові в лузі , сказав він»

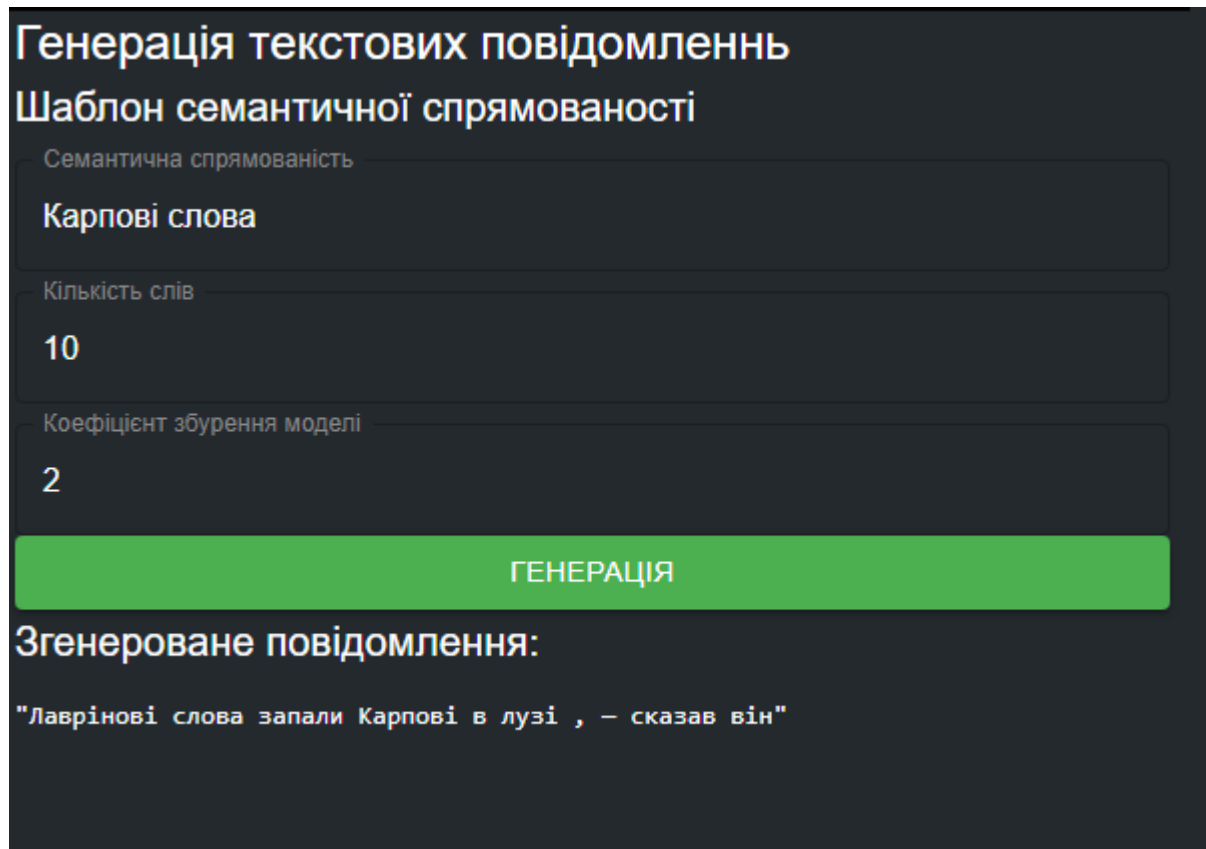


Рисунок 4.13 – Результат роботи моделі із вказаними параметрами

Результати експерименту можуть слугувати основою для подальших вдосконалень та використання систем штучного інтелекту в області обробки тексту. Для цього результати роботи інформаційної системи відправлено на оцінку чату GPT (рисунок 4.14).

Отримані результати свідчать про потенціал методу генерації текстових повідомлень на основі лексичних n -грам для заданої семантичної спрямованості. У даному дослідженні було проведено аналіз різних видів n -грам для визначення їхньої ефективності в генерації текстових повідомлень з заданою семантичною спрямованістю. Біграми та триграми виявилися ключовими елементами у визначенні контексту та взаємозв'язків між словами.

Біграми, представляючи пари сусідніх слів, відзначаються компактністю та можливістю враховувати локальний зв'язок між словами. Однак їхня обмежена спроможність виявляти довший контекст може призводити до втрати глибокого змісту тексту, особливо в тематично багатих областях.

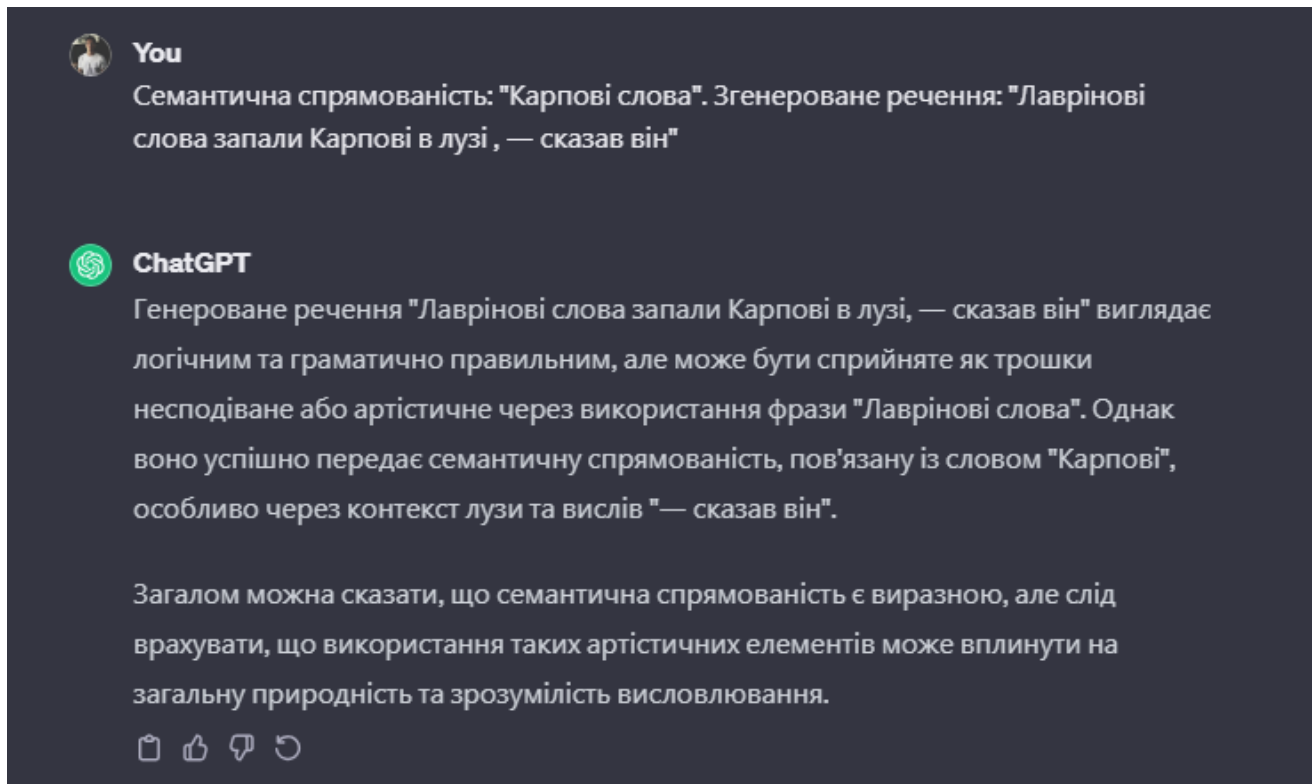


Рисунок 4.14 – Оцінка згенерованого речення чатом GPT

Триграми, складаючись з трьох послідовних слів, дозволяють враховувати більший контекст та виявляти складні взаємодії між словами. Вони особливо ефективні для аналізу довших текстів та вираження глибоких змістовних зв'язків. Однак використання триграм може вимагати більше ресурсів та бути менш ефективним у коротших текстах.

Досліджено вплив кількості слів у згенерованих повідомленнях, а також визначено оптимальні значення для досягнення балансу між змістом та зрозумілістю. Обмеження в 3 слова демонструє ефективність у створенні лаконічних і зрозумілих повідомлень, водночас зберігаючи семантичний контекст.

Обмеження в 10 слів показує адаптивність методу до більшої кількості слів, зберігаючи якість та семантичну насиченість. Результати вказують на можливість налаштування методу для забезпечення оптимального балансу між довжиною тексту та збереженням семантики відповідно до конкретних потреб користувача.

Проведене дослідження ефективності методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних програм, надало важливі висновки щодо його потенційного застосування та обмежень.

Також ефективність розробленого методу було перевірено за індексом читабельності тексту.

Індекс читабельності тексту Флеша (Flesch Reading Ease) – міра визначення складності сприйняття тексту читачем. Він може обчислюватися на основі декількох параметрів: довжини речень, слів, питомої кількості найбільш частотних (або рідкісних) слів [42]. Розраховується він за наступною формулою:

$$206.835 - 1.015\left(\frac{\text{кількість слів}}{\text{кількість речень}}\right) - 84.6\left(\frac{\text{кількість складів}}{\text{кількість слів}}\right)$$

Розрахований індекс може мати наступну градацію, що наведена у таблиці 4.6.

Таблиця 4.6 – Інтерпретація отриманого індексу читабельності тексту

Індекс Флеша	Рівень читабельності тексту
90-100	дуже високий
80-90	високий
70-80	вище за середній
60-70	середній
50-60	нижче за середній
30-50	низький
0-30	дуже низький

Для дослідження ефективності розробленого методу згідно індексу читабельності тексту Флеша, згенеровано 100 текстових повідомлень з загальною кількістю слів в повідомленні від 20 до 40 за допомогою біграм та 100

для триграм. Розподіл обрахованих індексів наведено у таблиці 4.7 та рисунку 4.15 для біграм та у таблиці 4.8 та рисунку 4.16 для триграм.

Таблиця 4.7 – Розподіл згенерованих текстів за допомогою біграм згідно рівня читабельності

Індекс Флеша	Кількість текстів
90-100	14
80-90	19
70-80	23
60-70	25
50-60	7
30-50	9
0-30	3

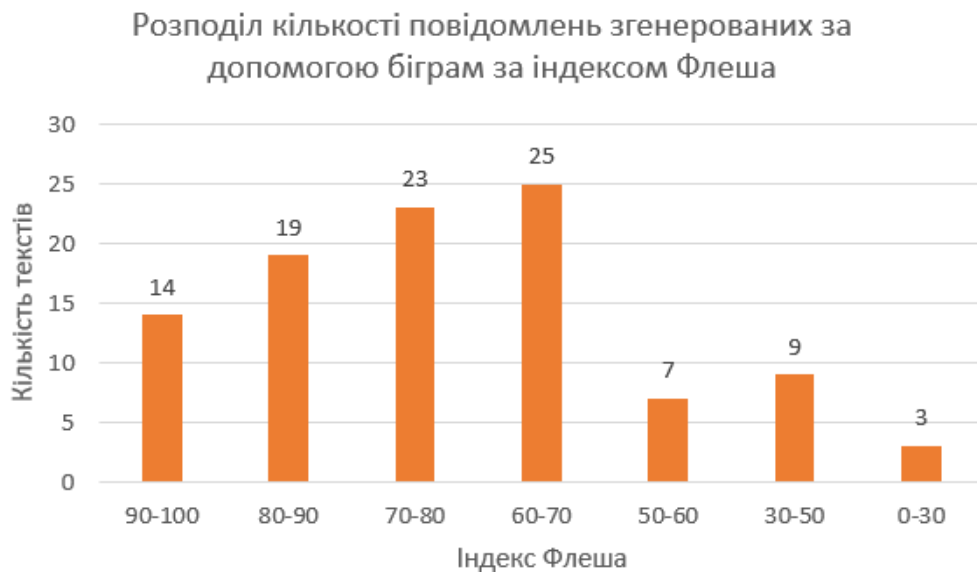


Рисунок 4.15 – Діаграма розподіл згенерованих текстів за допомогою біграм за індексом Флеша

З діаграма розподілу згенерованих текстів за допомогою біграм за індексом Флеша видно, що найбільша кількість згенерованих текстових повідомлень має середній та вище середнього рівень читабельності.

Таблиця 4.8 – Розподіл згенерованих текстів за допомогою триграм згідно рівня читабельності

Індекс Флеша	Кількість текстів
90-100	17
80-90	37
70-80	28
60-70	9
50-60	3
30-50	4
0-30	2

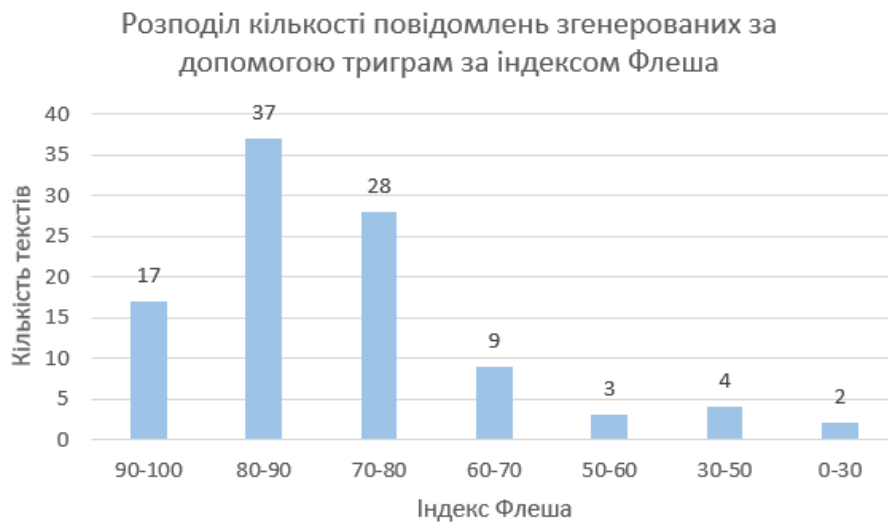


Рисунок 4.16 – Діаграма розподіл згенерованих текстів за допомогою триграм за індексом Флеша

З діаграма розподілу згенерованих текстів за допомогою триграм за індексом Флеша (рисунок 4.12) видно, що найбільша кількість згенерованих текстових повідомлень має вище середнього та високий рівень читабельності.

Отже, проаналізувавши отримані результати можна зробити висновок про те, що текстові повідомлення, що були згенеровані з використанням біграм мали гіршу читабельність, аніж згенеровані за допомогою триграм. Однак, переважна

кількість згенерованих текстових повідомлень мала рівень читабельності Флеша середній та вище, а саме для біграм – 81% та для триграм 91%.

Використання лексичних n-грам дозволяє системі враховувати контекст і створювати повідомлення, які мають логічний та семантичний зв'язок між словами та фразами. Також, відзначається здатністю враховувати семантичні відтінки та задану спрямованість, що робить генеровані повідомлення відповідними тематиці чи ключовим словам. Також важливо враховувати що, ефективність методу суттєво залежить від якості та репрезентативності використовуваних лексичних n-грам, враховуючи, що вони витягаються з доступних текстових даних. Генерація базується на існуючих n-грамах, що може обмежити творчий аспект та різноманіття виразу у згенерованих повідомленнях.

У майбутньому можна розглядати розширення методу та його використання в різних сферах для створення ефективного та змістовного текстового контенту з заданою семантичною спрямованістю.

Висновки до розділу 4

В цьому розділі було представлено докладний огляд програмної реалізації інформаційної системи, спрямованої на генерацію текстових повідомлень з заданою семантичною спрямованістю за допомогою лексичних n-грам. Застосування цієї технології виявилось ключовим для досягнення точності та природності згенерованого контенту.

Програмна реалізація використовує мову програмування Python та фреймворк Flask для реалізації бекенду, що забезпечує швидку та ефективну роботу системи. Важливим аспектом реалізації є використання бібліотеки NLTK для обробки тексту та лексичних n-грам, що дозволяє враховувати семантичні зв'язки у генерації текстового контенту.

Описано взаємодію між фронтендом та бекендом за допомогою API-запитів та важливий механізм CORS для забезпечення безпечної взаємодії між

різними частинами системи. Також висвітлено роль авторизації через Google та роботу APIHandler у визначенні різних маршрутів для обробки запитів.

Успішно здійснено обробку та аналіз текстових даних для генерації повідомлень, а також систематично представлено схему бекенд частини додатку. Особлива увага була приділена забезпеченню ефективної та надійної взаємодії між фронтендом та бекендом, що враховує вимоги до обробки текстового контенту та забезпечення його семантичної точності.

В цілому, програмна реалізація відображає комплексний підхід до генерації текстових повідомлень та демонструє успішну інтеграцію ключових технологій у цей процес. Розроблена система генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам виявляється ефективним інструментом для генерації контенту з заданою семантичною спрямованістю та відкриває нові можливості для подальших досліджень у сфері обробки природної мови.

Загальні висновки

Кваліфікаційна робота магістра розв'язує задачу автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.

Результатом роботи є метод, що призначений для генерації текстових повідомлень заданої семантичної спрямованості, який за навченою частотною n-грамною моделлю та заданою користувачем семантичною спрямованістю, генерує відповідне текстове повідомлення. Отримані результати та програмна реалізація системи створені з урахуванням важливих аспектів обробки природної мови та інтеграції цих аспектів у генерацію текстового контенту.

За вирішення задачі автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам було виконано завдання дослідження:

- проведено аналіз предметної області, зокрема визначено особливості застосування n-грам для задач генерації текстів семантичної спрямованості;
- виконано аналіз існуючих рішень щодо подібних задач;
- розроблено метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам;
- спроектовано структуру інформаційної системи генерації текстових повідомлень заданої семантичної спрямованості й структуру відповідної бази даних;
- створено прикладний застосунок, що використовує розроблений метод, та виконано його тестування;
- дослідити ефективність розробленого методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.

Проведене дослідження ефективності методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам дозволило зробити висновок про те, що текстові повідомлення, що були

згенеровані з використанням біграм мали гіршу читабельність, аніж згенеровані за допомогою триграм. Однак, переважна кількість згенерованих текстових повідомлень мала рівень читабельності Флеша середній та вище, а саме для біграм – 81% та для триграм 91%. Використання лексичних n-грам виявилось ключовим елементом для досягнення семантичної точності та природності в генерації текстових повідомлень. Цей метод дозволяє уникати загальних недоліків та забезпечує більш глибоке розуміння контексту при створенні висловлень.

Загалом, використання лексичних n-грам дозволяє системі враховувати контекст і створювати повідомлення, які мають логічний та семантичний зв'язок між словами та фразами. Також, відзначається здатністю враховувати семантичні відтінки та задану спрямованість, що робить генеровані повідомлення відповідними тематиці чи ключовим словам. Також важливо враховувати що, ефективність методу суттєво залежить від якості та репрезентативності використовуваних лексичних n-грам, враховуючи, що вони витягаються з доступних текстових даних. Генерація базується на існуючих n-грамах, що може обмежити творчий аспект та різноманіття виразу у згенерованих повідомленнях.

Інформаційна система, розроблена на основі методу автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, використовує мову програмування Python та фреймворк Flask для реалізації бекенду, що дозволяє ефективно взаємодіяти з фронтендом та обробляти API-запити. Важливим аспектом реалізації є взаємодія з бібліотекою NLTK для обробки тексту та аналізу лексичних структур.

Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам має ряд перспектив розвитку. Одним із напрямків розвитку є використання більш складних моделей генерації текстів, які можуть враховувати структуру тексту. Це дозволить створювати більш якісні і творчі тексти. Іншим напрямком розвитку є використання додаткових даних, таких як граматичні правила або тезауруси, для підвищення

точності прогнозування лексичних n-грам, що дозволить краще розуміти семантику оригінального тексту і створювати більш точні тексти.

Вирішені завдання відображають потенціал використання даного методу для генерації текстових повідомлень з визначеною семантичною спрямованістю. Результати дослідження відкривають перспективи для подальших досліджень та вдосконалення методів генерації контенту, що враховують складність семантичних відносин у природній мові.

Основні наукові й практичні результати кваліфікаційної роботи магістра доповідались у доповіді на тему «Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам» на XV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2023», також за темою кваліфікаційної роботи магістра автором виконано наукову публікацію [43].

Перелік посилань

1. Richard Jones, Ronald Martin, "Communication in the Real World: An Introduction to Communication Studies". URL: <https://open.lib.umn.edu/communication/>.
2. R.L. Trask "Language and Linguistics: The Key Concepts". URL: <https://ds.amu.edu.et/xmlui/bitstream/handle/123456789/13878/0415413583.Routledge.Language.and.Linguistics.The.Key.Concepts.Jul.2007.pdf?sequence=1&isAllowed=y>.
3. Frawley, W "Linguistic Semantics.". URL: https://books.google.com.ua/books?hl=uk&lr=&id=yZ77AQAAQBAJ&oi=fnd&pg=PP1&dq=Linguistic+Semantics&ots=yvslH8JFs7&sig=xRhyEhblsA3MKUyrVoKBaJDEDgw&redir_esc=y#v=onepage&q=Linguistic%20Semantics&f=false.
4. J. Allan "Natural Language Semantics.". URL: <https://seop.illc.uva.nl/entries/situations-semantics/>.
5. Geeraerts, D "Theories of Lexical Semantics.". URL: <https://books.google.com.ua/books?id=Lf5QEAAAQBAJ&lpg=PR7&ots=UuSKl3lyiE&dq=5.%09Geeraerts%2C%20D%20%20%22Theories%20of%20Lexical%20Semantics.%22%20&lr&hl=uk&pg=PR7#v=onepage&q=5.%09Geeraerts,%20D%20%20%22Theories%20of%20Lexical%20Semantics.%22&f=false>.
6. Bach "Semantics in Generative Grammar". URL: <https://www.jbe-platform.com/content/journals/10.1075/li.00033.cip>.
7. Koehn, P "Statistical Machine Translation.". URL: <https://books.google.com.ua/books?id=kKYgAwAAQBAJ&lpg=PR11&ots=k8wiWXf54q&dq=Koehn%2C%20P%20%20%22Statistical%20Machine%20Translation.%22%20&lr&hl=uk&pg=PR11#v=onepage&q=Koehn,%20P%20%20%22Statistical%20Machine%20Translation.%22&f=false>.
8. Jose B. Marino "N-gram-based Machine Translation". URL: <https://direct.mit.edu/coli/article/32/4/527/1929/N-gram-based-Machine-Translation>.

9. Atanu Dey "Senti-N-Gram: An n-gram lexicon for sentiment analysis"..
URL: <https://www.sciencedirect.com/science/article/abs/pii/S095741741830143X>
10. Maja Popovic "Character n-gram F-score for automatic M evaluation ".
URL: <https://aclanthology.org/W15-3049.pdf>.
11. Wei-Jen Li, "Fileprints: identifying file types by n-gram analysis". URL:
<https://ieeexplore.ieee.org/abstract/document/1495935>.
12. Arun Kumar Yadav, Samir Kumar Borgohain, "Sentence generation from a bag of words using N-gram model". URL:
<https://ieeexplore.ieee.org/abstract/document/7019414>.
13. Z Hu, Z Yang, X Liang "Toward controlled generation of text". URL:
<https://proceedings.mlr.press/v70/hu17e.html>.
14. K McKeown, "Text generation". URL:
<https://www.annualreviews.org/doi/pdf/10.1146/annurev.cs.02.060187.002153>.
15. Roy T. Fielding "Principled design of the modern Web architecture".
URL: <https://dl.acm.org/doi/abs/10.1145/514183.514185>
16. Ian Horrocks. "Semantic Web Architecture: Stack or Two Towers?".
URL: https://link.springer.com/chapter/10.1007/11552222_4.
17. Raymond Cheng. "Radiatus: a Shared-Nothing Server-Side Web Architecture". URL: <https://dl.acm.org/doi/abs/10.1145/2987550.2987571>.
18. Mike Conway "Classifying disease outbreak reports using n-grams and semantic features". URL:
<https://www.sciencedirect.com/science/article/abs/pii/S1386505609000537>.
19. Ashraf S. Hussein "Visualizing document similarity using n-grams and latent semantic analysis". URL:
<https://ieeexplore.ieee.org/abstract/document/7555994>.
20. Samira Abnar "Expanded N-Grams for Semantic Text Alignment". URL:
<https://ceur-ws.org/Vol-1180/CLEF2014wn-Pan-AbnarEt2014.pdf>.
21. Hai Hieu Vu "Sentence Similarity by Combining Explicit Semantic Analysis and Overlapping N-Grams". URL:
https://link.springer.com/chapter/10.1007/978-3-319-10816-2_25.

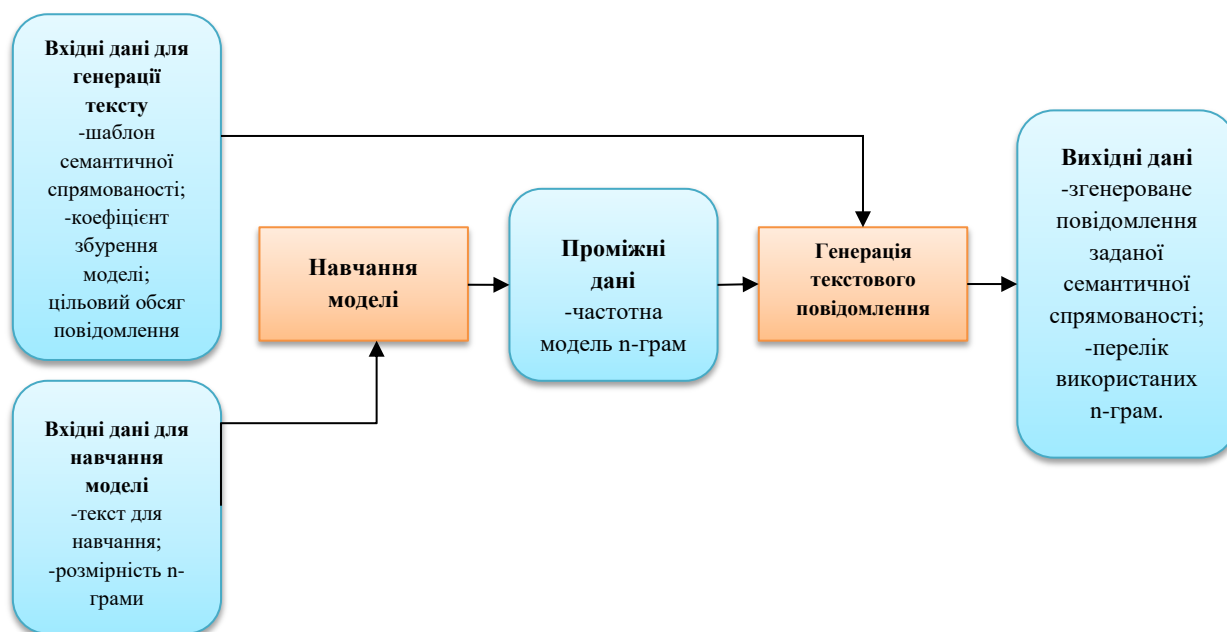
22. Gwénolé Lecorvé "Automatically finding semantically consistent n-grams to add new words". URL: <https://ieeexplore.ieee.org/abstract/document/5947398>.
23. Stewart Yang "N-gram Statistics in English and Chinese: Similarities and Differences". URL: <https://ieeexplore.ieee.org/abstract/document/4338381>.
24. Grigori Sidorov "Syntactic Dependency-Based N-grams as Classification Features". URL: https://link.springer.com/chapter/10.1007/978-3-642-37798-3_1.
25. Bofang Li "Neural Bag-of-Ngrams". URL: <https://ojs.aaai.org/index.php/AAAI/article/view/10954>.
26. M Grinberg "Flask web development: developing web applications with python". URL: <https://books.google.com.ua/books?id=cVIPDwAAQBAJ&lpg=PT25&ots=xPBUdq5l9W&dq=26.%09M%20Grinberg%20%22Flask%20web%20development%3A%20developing%20web%20applications%20with%20python%22&lr&hl=uk&pg=PT25#v=onepage&q&f=false>.
27. Dmitriy Beshpalov "Sentiment classification based on supervised latent n-gram analysis". URL: <https://dl.acm.org/doi/abs/10.1145/2063576.2063635>.
28. Radford, A "Improving Language Understanding by Generative Pretraining.". URL: <https://www.mikecaptain.com/resources/pdf/GPT-1.pdf>.
29. Shiva Raj Pokhrel "A Semantic Communication Approach to 6G Networks".. URL: <https://ieeexplore.ieee.org/abstract/document/9937052>
30. Hai Dang, Karim Benharrak "Beyond Text Generation: Supporting Writers with Continuous Automatic Text Summaries". URL: <https://dl.acm.org/doi/abs/10.1145/3526113.3545672>
31. Touseef Iqbal "The survey: Text generation models in deep learning". URL: <https://www.sciencedirect.com/science/article/pii/S1319157820303360>
32. "Word n-gram language model". URL: https://en.wikipedia.org/wiki/Word_n-gram_language_model
33. Manning, Raghavan ta Schütze "N-Gram Language Models" . URL: https://tarekamr.com/static/pdf/URL_Classification_MSc_Dissertation_CCBYSA.pdf

34. "Syntagmatic, Paradigmatic, and Automatic N-gram". URL: https://www.researchgate.net/profile/Danielle-Mcnamara/publication/285641506_Syntagmatic_paradigmatic_and_automatic_N-gram_approaches_to_assessing_essay_quality/links/5dc5b8baa6fdcc575034824f/Syntagmatic-paradigmatic-and-automatic-N-gram-approaches-to-assessing-essay-quality.pdf
35. Visual Studio Code. URL: <https://code.visualstudio.com/>
36. Python. URL: <https://www.python.org/>
37. Flask. URL: <https://flask.palletsprojects.com/en/3.0.x/>
38. J Perras "Flask Blueprints". URL: <https://books.google.com.ua/books?id=SfSoCwAAQBAJ&lpg=PP1&ots=Aamv-PxfzE&dq=38.%09J%20Perras%20%22Flask%20Blueprints%22%20&lr&hl=uk&pg=PP1#v=onepage&q&f=false> .
39. A O'rinboev "optimizing performance in a dental queue web app". URL: <http://econferences.ru/index.php/dptms/article/view/8371>.
40. React Patterns. URL: <https://reactpatterns.com/>
41. Redux. URL: <https://redux.js.org/>
42. What Is Flesch-Kincaid Readability? URL: <https://www.webfx.com/tools/read-able/flesch-kincaid/>
43. Козакевич В.А., Собко О.В., Тищенко О.О., Вознюк Л.О., Медведчук В.Ю. Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023». Хмельницький, 2023. с. 136-141. URL: <https://kn.khmnu.edu.ua/wp-content/uploads/sites/18/apkn-2023-corporpaper.pdf>

ДОДАТКИ

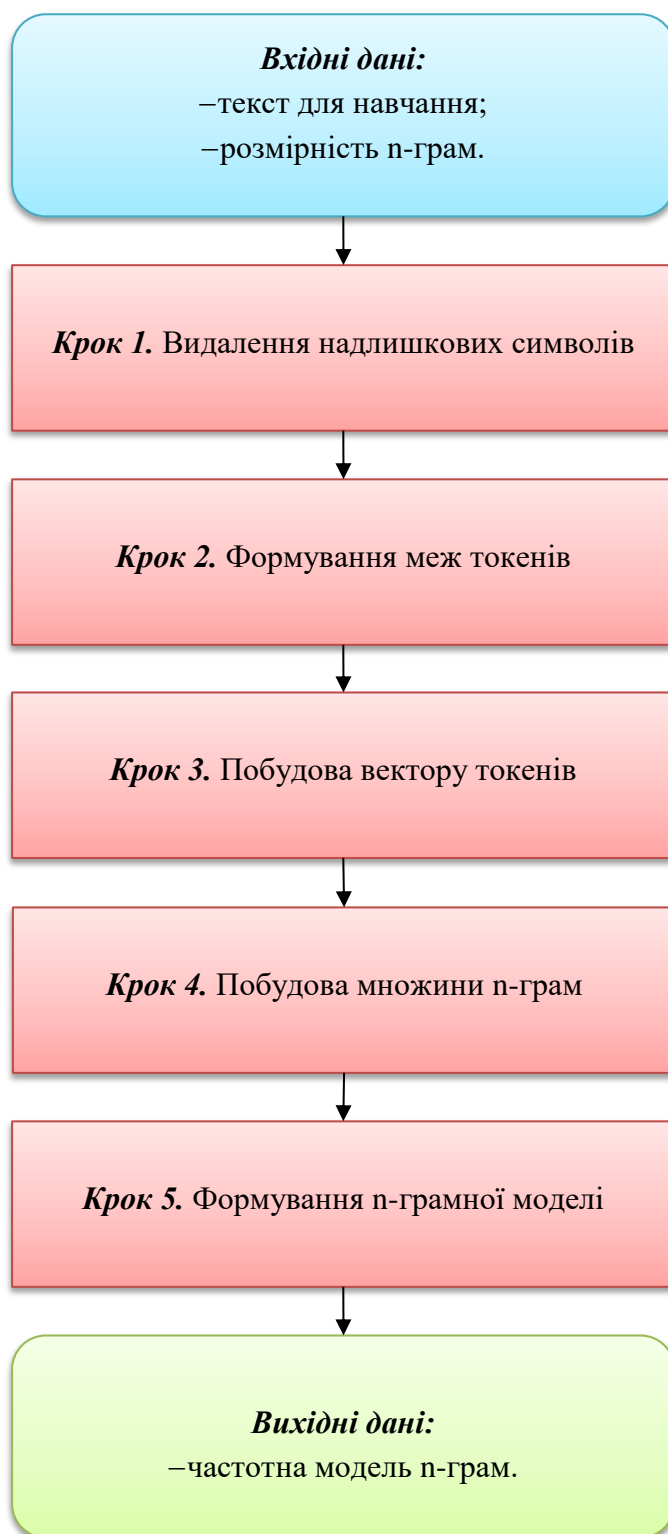
Додаток А

Етапи виконання методу генерації текстових повідомлень



Додаток Б

Схема навчання n-грамної моделі



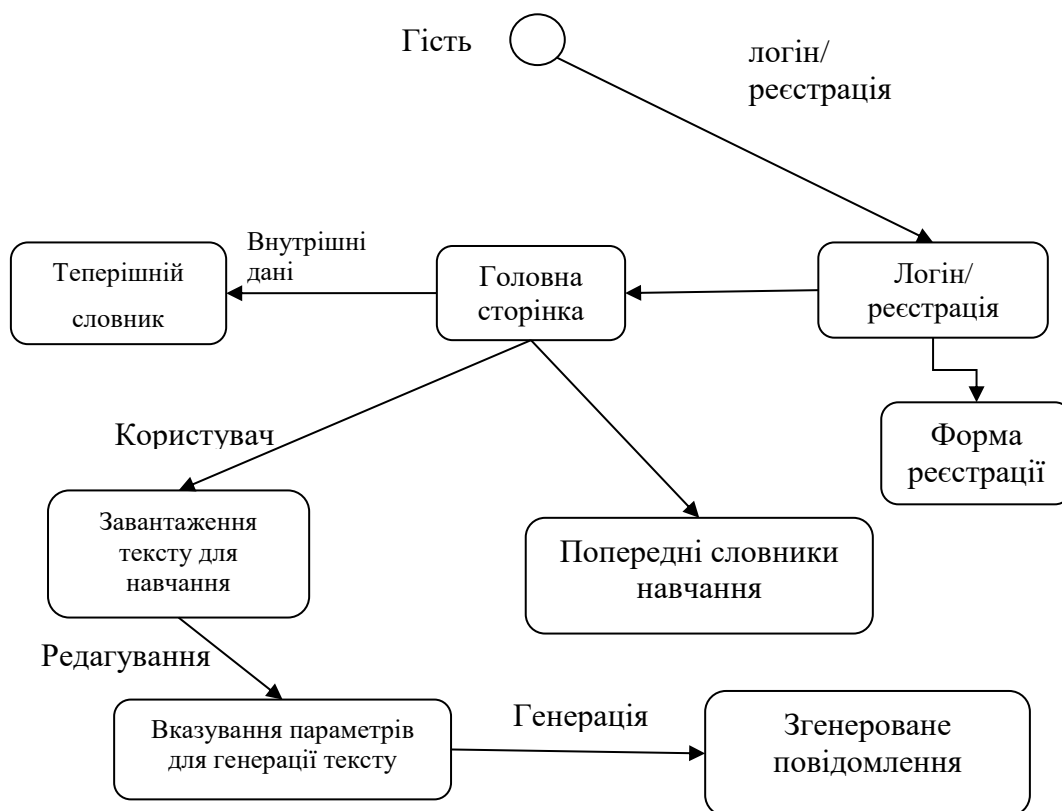
Додаток В

Схема генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам



Додаток Г

UML-діаграма варіантів використання сайту



Додаток Д

Світлини наукових публікацій, виконаних при роботі над кваліфікаційною роботою магістра

(ксерокопії титульної сторінки, сторінки змісту та всіх сторінок із публікацією)

Наукова публікація:

Козакевич В.А., Собко О.В., Тищенко О.О., Вознюк Л.О., Медведчук В.Ю. Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам. Збірник наукових праць за матеріалами XV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2023». Хмельницький, 2023. с. 136-141. URL: <https://kn.khmnu.edu.ua/wp-content/uploads/sites/18/apkn-2023-corpuspaper.pdf>

Міністерство освіти і науки України
Хмельницький національний університет



ЗБІРНИК НАУКОВИХ ПРАЦЬ
за матеріалами XV Всеукраїнської науково-практичної конференції
«Актуальні проблеми комп'ютерних наук АПКН-2023»

17-18 листопада 2023

Хмельницький 2023

Залуцька О.О., Молчанова М.О., Віт Р.В., Мазурець О.В. Конфігурування нейронної мережі для класифікації емоційної тональності текстової інформації за показниками семантичної зв'язності	102
Запорожець М.В., Молчанова М.О., Скрипник Т.К. Метод виявлення патологій мозку за зображеннями магнітно-резонансної терапії нейромережевими засобами	108
Карлечук Д.Т., Багрій Р.О., Скрипник Т.К., Тищенко О.О. Метод структурування тексту оголошень для об'єктів нерухомості засобами NLP	111
Карпович В.В., Дрозд А.І., Жуковський П.О., Мельник В.В. Методи вирішення проблем пропускну здатності дисків для застосунків з інтенсивним обсягом даних	116
Каушан С.О., Лисенко С.М. Дослідження інформаційних систем електронного рекрутингу персоналу	118
Качур А.В., Лисенко С.М. Виклики в розвитку технології віртуальної реальності: оптимізація архітектури VR.....	121
Качур О.І. Перспективні напрямки розвитку сучасного антивірусного захисту мереж та роль методів на основі генетичних алгоритмів.....	124
Кирилюк О.О., Онишко О.Г. Дослідження використання інструменту Elasticsearch для оптимізації вебдодатків, розроблених з використанням фреймворку Laravel.....	128
Кльоц Ю.П., Петляк Н.С., Чвалов А.А. Технології тестування безпеки вебресурсів	130
Коверник Д.С. Мобільний додаток для читання книг з Google Books: методології програмної інженерії та архітектурні рішення.....	133
Козакевич В.А., Собко О.В., Тищенко О.О., Вознюк Л.О., Медведчук В.Ю. Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам	136
Козельський О.В. Методи та засоби створення мультикомп'ютерних систем з подвійною автентифікацією потоків даних в корпоративних мережах	142

УДК 004.4

Козакевич В.А., Собко О.В., Тищенко О.О., Вознюк Л.О., Медведчук В.Ю.

*Хмельницький національний університет***МЕТОД АВТОМАТИЗОВАНОЇ ГЕНЕРАЦІЇ ТЕКСТОВИХ ПОВІДОМЛЕНЬ
ЗАДАНОЇ СЕМАНТИЧНОЇ СПРЯМОВАНОСТІ З ВИКОРИСТАННЯМ
ЛЕКСИЧНИХ N-ГРАМ**

Було розглянуто ключові приклади розробки інформаційних систем, які автоматично генерують текстові повідомлення з визначеною семантичною спрямованістю за допомогою лексичних n-грам і були отримані результати роботи цих систем для подальшого аналізу.

The key examples of the development of information systems that automatically generate text messages with a defined semantic orientation using lexical n-grams were considered, and the results of the work of these systems were obtained for further analysis.

Генерація текстових повідомлень – це автоматичне створення або надсилання невеликих обсягів текстової інформації окремим користувачам або групам користувачів з невеликими змінами або без них. Автоматизація такого маркетингу набула популярності серед багатьох брендів, оскільки вона дозволяє їм ефективно взаємодіяти зі своєю новою аудиторією в широкому масштабі. Крім того, автоматична генерація текстових повідомлень часто застосовується в системах сповіщення [1].

У наш час, коли сучасні технології швидко розвиваються і стають все більш інтуїтивно зрозумілими, продуктивними і зручними для користувача, важливо мати засоби ефективного спілкування з ним. Це стає невід'ємною складовою кожного проекту чи роботи. Для підтримки зв'язку з потенційними або постійними клієнтами, укладення контрактів з працівниками компанії та в щоденній робочій атмосфері все частіше використовують засоби для автоматизованої генерації невеликих текстових повідомлень.

Такі системи для автоматичної генерації невеликого обсягу текстових повідомлень застосовуються в різних сферах. Вони можуть бути використані як чат-боти, програми для автоматичного сповіщення користувачів, для модерації активності постійних учасників у групових чатах і соціальних мережах.

Однією з ключових сфер використання є чат-боти. У цьому випадку обмін текстовими повідомленнями між користувачем і сервісом відбувається негайно та нагадує звичайний діалог між двома особами. Однак, з одного боку, одним із учасників є реальна особа, а з іншого - система з автоматизованою генерацією невеликих текстових повідомлень. Також можливий обмін повідомленнями з

більшою кількістю зареєстрованих користувачів. Якщо в бесіді бере участь більше двох осіб одночасно, це вже називається чатом. Також важливою є можливість використання аудіо асистентів [2], які автоматично генерують відповіді у текстовому або аудіо форматі.

Один із прикладів використання цього підходу полягає у використанні типових моделей, який детально розглядається у роботі Лангкілде та Найтта. Задача генерації природної мови [3] є важливою складовою роботи системи автоматичної відповіді користувачу.

Моделі, які ґрунтуються на роботі з конкретними випадками, в даному контексті аналізують вхідні дані як набір порівнюваних ситуацій, які відбуваються у конкретному середовищі. Наприклад, задавши однакові запитання великій кількості людей, можна аналізувати їх відповіді, що в свою чергу утворює групу відповідей з однаковою семантикою та змістом. Grounded theory[4] ґрунтується на цьому підході у своїй роботі. Вони використовували модель HALogen [5], яка представлена на рисунку 1.

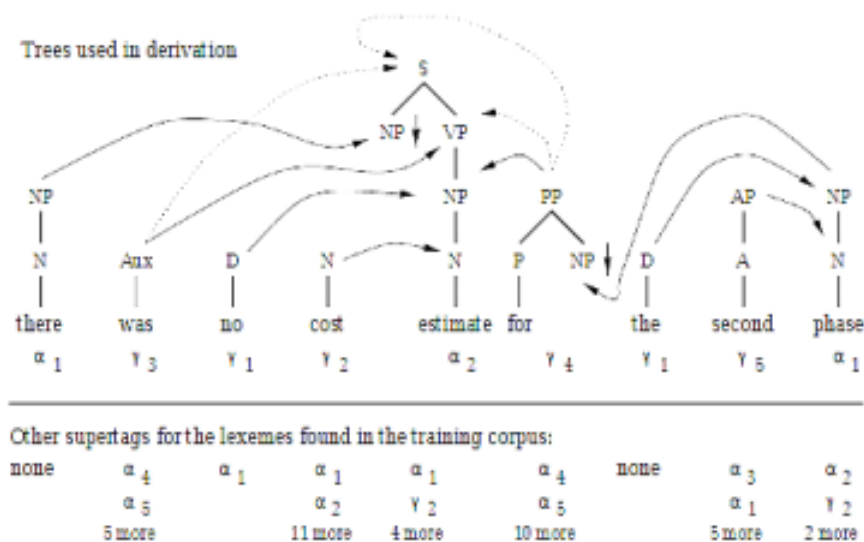


Рисунок 1 – HALogen [5]

Граматична модель у цій реалізації представлена у вигляді структури, що називається "випадковий ліс" (random forest). За допомогою стохастичного алгоритму оцінювання може бути вибраний найбільш підходящий варіант. У даному випадку цей підхід використовує базу у формі n -грам, тоді як сучасні підходи виконують великі експерименти з сучасними та інноваційними моделями для досягнення кращих результатів[6]. Існують різні види n -грам.

Уніграми (1-грами) – це найпростіші n-грами, які представляють окремі слова. Вони не містять контексту та розглядаються ізольовано. Наприклад, у реченні "Сьогодні гарний день", уніграмами будуть "Сьогодні", "гарний", "день".

Біграми (2-грами) – це n-грами, які складаються з двох слів. Вони дозволяють враховувати контекст, але тільки в обмеженому обсязі. Наприклад, для того ж речення, біграми будуть "Сьогодні гарний", "гарний день".

Триграми (3-грами) – це n-грами, які мають три слова в послідовності. Вони враховують більший контекст, ніж біграми. Наприклад, для того ж речення, триграми будуть "Сьогодні гарний день".

Чотириграми (4-грами) – це n-грами з чотирма словами в послідовності.

Що стосується використання n-грамів, вони використовуються для створення статистичних моделей мови, які визначають ймовірність входження певної послідовності слів у текст. Автоматичний переклад, використовуються для визначення найбільш ймовірних перекладів.

Розпізнавання мови в основному використовуються для визначення найімовірніших слів або фраз в аудіозаписах. Автокоректори та системи підказки. Використовують n-грами для визначення найбільш ймовірного продовження введеного тексту.

Важливо враховувати, що зі збільшенням значення n (кількість слів в n-грамі), зростає складність обробки та обсяг затребуваної статистики для навчання моделі. З іншого боку, більші значення n дозволяють враховувати більший контекст та покращувати якість результатів.

N-грами знаходять застосування у багатьох галузях, де важливо аналізувати текстову інформацію та розуміти її контекст. Використовуються для прогнозування ймовірностей появи певних слів чи фраз у тексті. Це важливо для завдань, таких як автоматичний переклад, розпізнавання мови, генерація тексту тощо. N-грами використовуються для визначення найбільш ймовірних перекладів для конкретних слів чи фраз. Використовують n-грами для визначення найбільш ймовірного продовження введеного тексту, що допомагає вказувати на можливі помилки або надавати рекомендації. N-грами використовуються для визначення найбільш ймовірних слів чи фраз в контексті генерування тексту. N-грами можуть бути використані для прогнозування подій чи трендів на основі аналізу текстових даних.

N-грами, хоч і є потужним інструментом для аналізу тексту та розуміння контексту, мають кілька недоліків. Для багатьох завдань, особливо в глибоких аналізах, n-грами обмежені в тому, що вони не можуть врахувати дуже великий контекст. Це особливо стає проблемою у випадках, коли значення слова залежить від довшого контексту. Велика кількість n-грам може бути рідкісними, тобто вони можуть взагалі не зустрічатися в тренувальних даних. Це ускладнює побудову надійних статистичних моделей. Для великих значень n (наприклад, 4-грами та вище), розмір словника та потрібна кількість обчислень для побудови моделі можуть бути вкрай великими. N-грами враховують тільки послідовність слів, але не

їхній фактичний порядок. Це може призводити до неправильних або незрозумілих висновків.

Інший приклад цього може бути підхід, що не вимагає складних обчислень для сортування та автоматичної генерації. Він використовує статичну інформацію для генерації відповіді у момент вибору та прийняття рішення. Цей метод часто використовується в системі PCRU, яка була вперше запропонована в 2007 році [7]. Робота цієї системи показана на рисунку 2. Вона може генерувати потенційне закінчення речення, яке має найбільшу ймовірність.

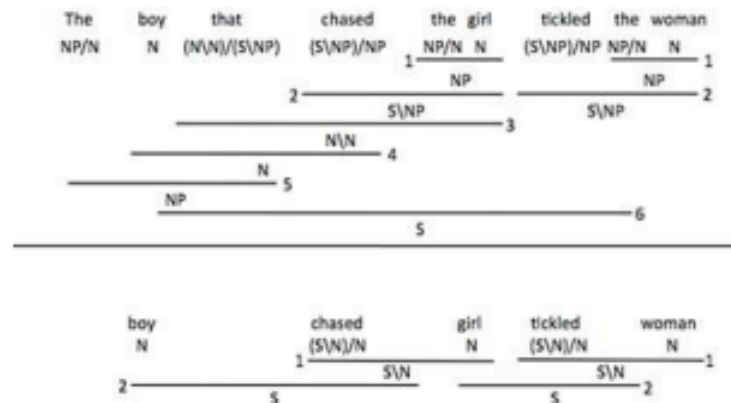


Рисунок 2 – Робота системи PCRU

Вагомим прикладом є фреймворк OpenCCG[8]. Цей інструмент є потужним засобом для аналізу великих повідомлень різних форматів та складності. Він використовує граматичний підхід [9] і базу початкових даних, побудовану на основі бібліотеки Penn Treebank (рисунок 3), яка широко використовується для статистичної оцінки мовних моделей.

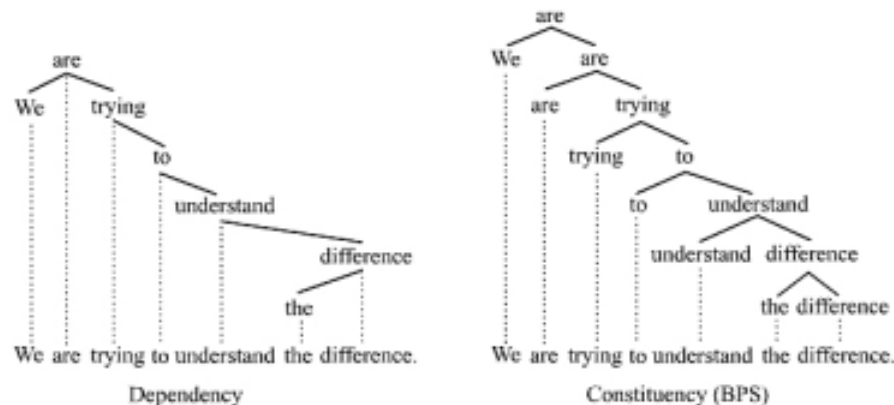


Рисунок 3 – Penn Treebank

Щоб розпочати роботи з цією системою автоматичної генерації текстових повідомлень потрібно підготувати корпус (див. рисунок 4), на якому ми будемо навчати нашу модель, а також визначити необхідну кількість слів у вихідному тексті. Генератор видасть "очищену" послідовність слів та всіх розділових знаків. Також доданий ще один генератор, який надає три токени посліпль. У цьому випадку токенами можуть бути слова або символи.

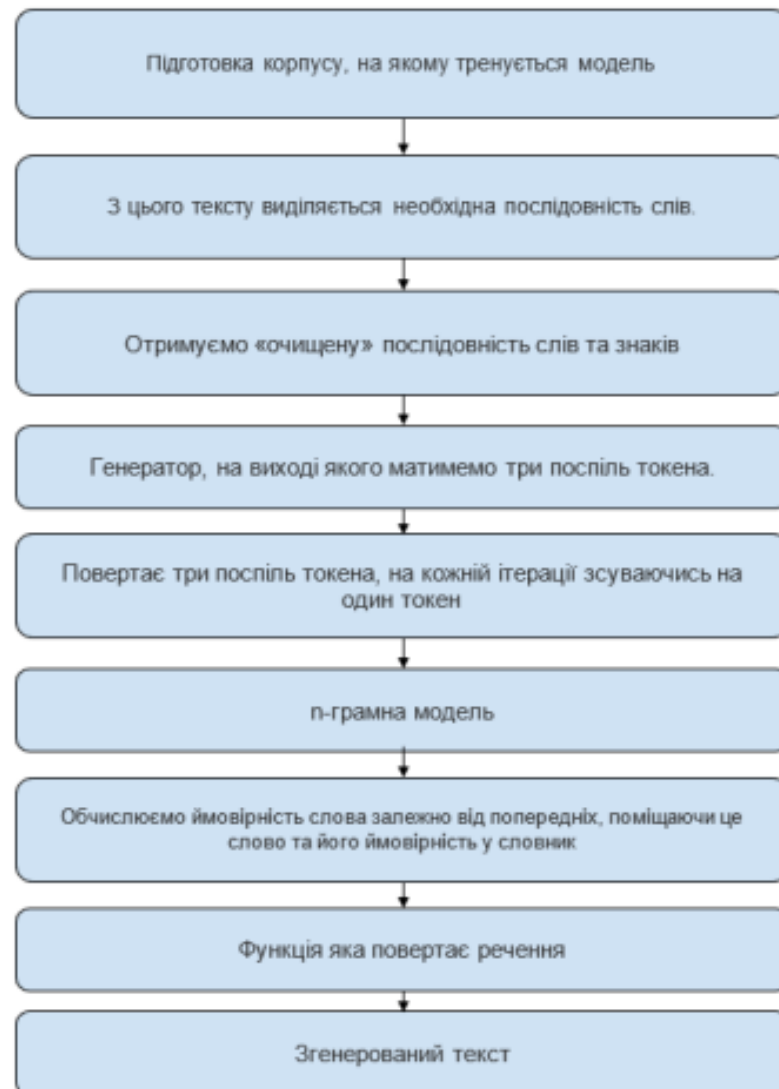


Рисунок 4 – Загальна схема роботи методу побудови текстів за допомогою n-грам

Наступним кроком є додавання функції, яка виводить кілька слів поспіль. Це полегшує вибір першого слова у фразі, що генерується. Узагалі, метод працює так: він повертає три токени поспіль, кожен наступний зрушується на один рівень після кожної ітерації.

Для початку ініціалізуємо генератори. Потім обчислюємо n-грами та визначаємо ймовірність кожного слова в залежності від попередніх. Далі, це слово та його ймовірність додаються у словник. Важливо зауважити, що цей метод може не бути найбільш оптимальним, оскільки він може вимагати значних ресурсів пам'яті. Проте для невеликих корпусів він є досить ефективним.

Цей метод ґрунтується на поступовому виборі найбільш ймовірних слів та розділових знаків до того моменту, коли виявляємо початок наступної фрази.

Основною метою є інтеграція всіх компонентів та етапів в одній системі машинного навчання. Це може призвести до створення оптимізованих та зручних систем, які не потребують попередньої обробки вхідних даних або редагування та форматування тексту.

Тому були ретельно проаналізовані ключові приклади розробки інформаційних систем, які використовують лексичні n-грами для автоматичної генерації текстових повідомлень з визначеною семантичною спрямованістю. Результати цієї роботи свідчать про значущий прогрес у сфері автоматизованої генерації контенту та його аналізу.

Перелік посилань

1. Text Message Automation. URL: <https://www.slicktext.com/text-message-automation.php>
2. Google Assistant. URL: <https://www.techrepublic.com/article/google-assistant-the-smart-persons-guide/>
3. Natural language generation. URL: <https://research.aimultiple.com/nlg/>
4. A design framework for novice researchers. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6318722/>.
5. Model for Generation. URL: https://www.researchgate.net/publication/2473335_Exploiting_a_Probabilistic_Hierarchical_Model_for_Generation;
6. OpenCCG. URL: <http://openccg.sourceforge.net/>;
7. Combinatory categorial grammar. URL: https://en.wikipedia.org/wiki/Combinatory_categorial_grammar;
8. Chart Generation grammar. URL: <https://www.inf.ed.ac.uk/teaching/courses/nlg/readings/KayACL96.pdf>;
9. Grammar-Based Approach to Microplanning. URL: <https://www.aclweb.org/anthology/J17-1001.pdf>

Додаток Г

Презентаційний матеріал

Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

Виконав: студент 2 курсу, група КНм-22-1,
Козакевич Віталій Анатолійович

Керівник роботи: викладач кафедри КН,
Собко Олена Віталіївна

Актуальність

На сьогоднішній день організації в багатьох галузях промисловості використовують автоматизацію текстових повідомлень. Щодня з'являються нові варіанти використання інструментів автоматизації. Деякі з найпопулярніших застосувань включають в себе електронну комерцію, автоматизацію і упорядкування візитів, підтримку потенційних клієнтів і чат-ботів. Одним із вагомих переваг в автоматизації те, що вона так само добре працює як з 10 людьми, так і зі 100 000. Важливо пам'ятати, що потрібно персоналізувати кожне надіслане текстове повідомлення. Це призводить до кращого залучення та рівня відповіді. Також використання такої служби з генерації текстових повідомлень, дозволяє впоратися з важкою роботою, пов'язаною з підтримкою взаємодії з аудиторією, що інколи відіграє основну роль в роботі з клієнтами.

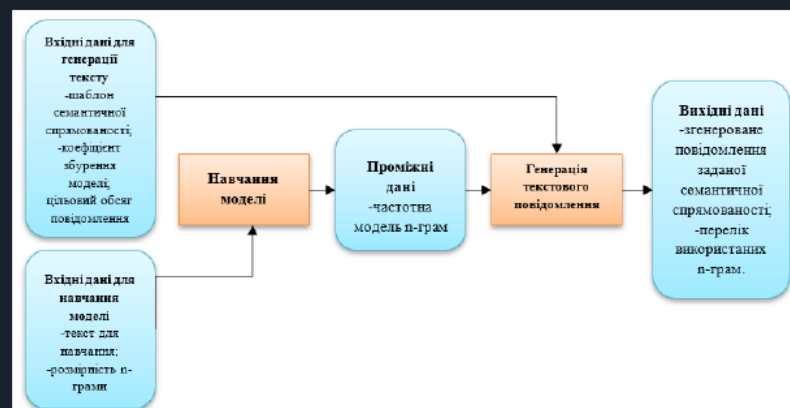


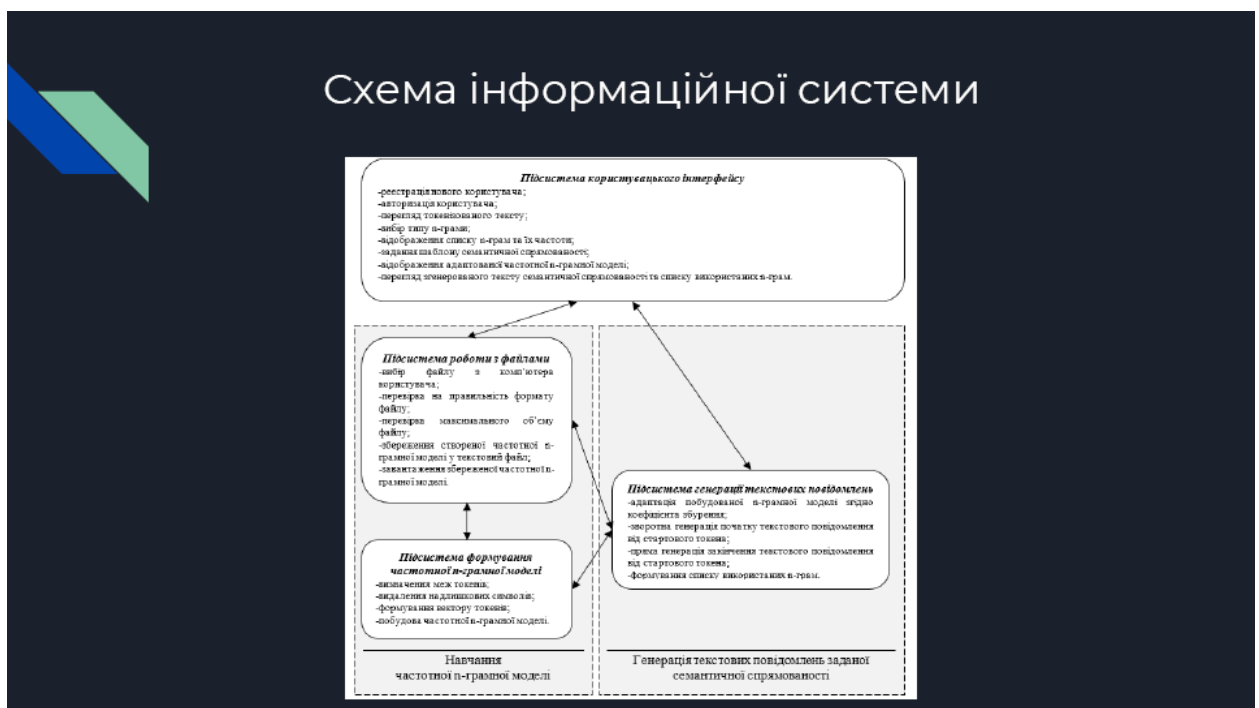
Наукова новизна

Результати виконання кваліфікаційної роботи магістра містять інновації та наукову новизну, зокрема було удосконалено метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, який дозволяє виконувати генерації текстових повідомлень заданого цільового обсягу.

Розроблений метод дозволяє генерувати текстові повідомлення як з використанням біграм, так і на основі триграм, із визначеним середнім рівнем читабельності тексту вище середнього. В ході дослідження ефективності було зроблено висновок, що текстові повідомлення, які були згенеровані з використанням біграм мали гіршу читабельність, ніж згенеровані за допомогою триграм. Однак, переважна кількість згенерованих текстових повідомлень мала рівень читабельності Флеша середній та вище, а саме для біграм – 81% та для триграм 91%.

Етапи виконання методу генерації текстових повідомлень





Результати роботи

Генерація текстових повідомлень

Шаблон семантичної спрямованості

Введіть одне слово

гарячу мрію

Кількість слів

12

Коефіцієнт збурення моделі

2

ГЕНЕРАЦІЯ

Згенероване повідомлення:

"де він ніби вгледів свою гарячу мрію в червоних кісниках на голові!"

Генерація текстових повідомлень

Шаблон семантичної спрямованості

Семантична спрямованість

Карпові слова

Кількість слів

10

Коефіцієнт збурення моделі

2

ГЕНЕРАЦІЯ

Згенероване повідомлення:

"Лаврінові слова запали Карпові в лузі, — сказав він"

Оцінка згенерованого текстового повідомлення за допомогою чату GPT



You

Семантична спрямованість: "Карпові слова". Згенероване речення: "Лаврінові слова запали Карпові в лузі, — сказав він"



ChatGPT

Генероване речення "Лаврінові слова запали Карпові в лузі, — сказав він" виглядає логічним та граматично правильним, але може бути сприйняте як трошки несподіване або артистичне через використання фрази "Лаврінові слова". Однак воно успішно передає семантичну спрямованість, пов'язану із словом "Карпові", особливо через контекст лузи та вислів "— сказав він".

Загалом можна сказати, що семантична спрямованість є виразною, але слід врахувати, що використання таких артистичних елементів може вплинути на загальну природність та зрозумілість висловлювання.



Діаграми розподілу згенерованих текстів за допомогою біграм та триграм за індексом Флеша



Висновок

Кваліфікаційна робота магістра розв'язує задачу автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.

Результатом роботи є метод, що призначений для генерації текстових повідомлень заданої семантичної спрямованості, який за навченою частотною n-грамною моделлю та заданою користувачем семантичною спрямованістю, генерує відповідне текстове повідомлення. Отримані результати та програмна реалізація системи створені з урахуванням важливих аспектів обробки природної мови та інтеграції цих аспектів у генерацію текстового контенту.

Поставлені у меті завдання було вирішено, вони відображають потенціал використання даного методу для генерації текстових повідомлень з визначеною семантичною спрямованістю. Результати дослідження відкривають перспективи для подальших досліджень та вдосконалення методів генерації контенту, що враховують складність семантичних відносин у природній мові.

Основні наукові й практичні результати роботи доповідались у доповіді на тему «Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам» на XV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2023», також за темою кваліфікаційної роботи магістра було виконано наукову публікацію.

Ім'я користувача:
Кафедра КН

Дата перевірки:
12.12.2023 09:10:49 EET

Дата звіту:
12.12.2023 09:26:22 EET

ID перевірки:
1015995976

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100005671

Назва документа: КНм-22-1 Козакевич

Кількість сторінок: 81 Кількість слів: 15185 Кількість символів: 117593 Розмір файлу: 1.92 MB ID файлу: 1015678868

10.2% Схожість

Найбільша схожість: 3.89% з Інтернет-джерелом (https://kn.khmnmu.edu.ua/wp-content/uploads/sites/18/apkn2022_corp).

9.48% Джерела з Інтернету

901

Сторінка 83

3.3% Джерела з Бібліотеки

99

Сторінка 88

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 1.0%

Словники перевірки: en_US, ru_RU, ua_UA. Помилки в документах: 9%

ID: 122623 Назва: КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА на тему Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам Додано в БД: 2023-12-12 Автора: В.А. Козакевич Керівники: О.В. Собко Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	98480	1440	2307 (2%)	36 (3%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНИХ НАУК

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ МАГІСТРА ДО ЗАХИСТУ ЗА РЕЗУЛЬТАТАМИ АНАЛІЗУ ЗВІТУ ПОДІБНОСТІ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

Автор: Козакевич Віталій Анатолійович

Спеціальність: 122 – Комп'ютерні науки

Освітня програма: освітньо-професійна

Науковий керівник: викл. кафедри КН, Собко О.В.

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

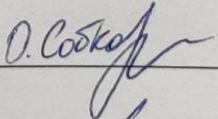
Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) за програмою Anti-Plagiarism виявлені 1% є фрагментарними – містять поширені конструкції, загальновідомі терміни, скорочення та визначення.
- 2) За програмою UNICHECK виявлені 10,2%, що є запозиченнями, які розміщені в розділах аналізу існуючих технологій та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи.

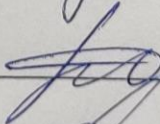
Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 1% і 10,2% відповідно, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи



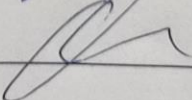
Олена СОБКО

Гарант ОП



Руслан БАГРІЙ

Завідувач кафедри КН



Олександр БАРМАК



ВІДГУК НАУКОВОГО КЕРІВНИКА

на кваліфікаційну роботу магістра

гр. КНм-22-1 Козакевича Віталія Анатолійовича за темою: Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

1. Актуальність теми

Актуальність теми «Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам» обумовлена потребою в ефективних методах обробки та аналізу текстової інформації. Розроблений метод має потенціал для застосування в різних сферах, включаючи маркетинг, комерцію, освіту.

2. Відповідність роботи предметній області спеціальності 122 Комп'ютерні науки та загальним вимогам до наукових робіт

Тема кваліфікаційної роботи "Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам" повністю відповідає предметній області спеціальності 122 Комп'ютерні науки, а також вимогам до кваліфікаційної роботи магістра, так як об'єктом дослідження є процес генерації текстових повідомлень заданої семантичної спрямованості, тобто має місце використання інформаційних технологій, моделей та методів, алгоритмів та засобів для генерації текстових повідомлень заданої семантичної спрямованості.

3. Професійні та особистісні якості магістранта

Козакевич Віталій Анатолійович під час роботи над кваліфікаційною роботою проявив себе як цілеспрямований та дисциплінований студент та продемонстрував достатній рівень знань та умінь за спеціальністю "Комп'ютерні науки".

4. Ступінь самостійності під час виконання кваліфікаційної роботи

Кваліфікаційна робота магістра виконана здобувачем самостійно, в роботі наведено посилання на використані джерела.

5. Наукова новизна та оригінальність запропонованих підходів

Отримані такі результати: вдосконалено підхід до генерації текстових повідомлень за допомогою n-грам, що дало можливість генерувати текстові повідомлення семантичної спрямованості, що досягається адаптацією частотної n-грамної моделі за допомогою коефіцієнта збурення; розроблено метод генерації текстових повідомлень заданої семантичної спрямованості, що дозволило підвищити семантичну спрямованість текстів

генерованих текстів з рівнем читабельності тексту вище середнього. Отримані результати оприлюднені на XV Всеукраїнській науково-практичній конференції «Актуальні проблеми комп'ютерних наук АПКН-2023» 17-18 листопада 2023 р., м. Хмельницький, в доповіді на тему «Метод автоматизованої генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам».

6. Ступінь оволодіння методами дослідження

Студент Козакевич В.А. виявив достатньо високий рівень володіння методами проведеного дослідження, що були використані у роботі.

7. Повнота та якість розкриття теми роботи

Мета дослідження досягнута, отримані результати розкриті в достатній мірі, надано їх детальне обґрунтування.

8. Логічність, послідовність, аргументованість, літературна грамотність викладу матеріалу

Матеріал викладено логічно, послідовно і обґрунтовано. Мова і стиль роботи відповідають вимогам до наукових робіт спеціальності 122 – Комп'ютерні науки, що забезпечує її зрозумілість і доступність.

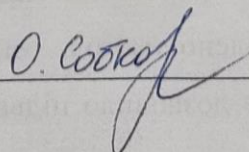
9. Можливість практичного застосування кваліфікаційної роботи, окремих її частин

Створена інформаційна система для генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам є прикладною програмною реалізацією розробленого метода генерації текстових повідомлень і використовувалась для проведення експериментальних досліджень з метою апробації запропонованого метода. Проведені дослідження ефективності розробленого в роботі метода генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам свідчать про те, що даний підхід можна використовувати для генерації текстових повідомлень для прикладних застосувань на кшталт чатботів та текстрайтингу, а розроблений метод має потенціал для застосування в різних сферах, включаючи маркетинг, комерцію й освіту.

Висновок про можливість допуску кваліфікаційної роботи до захисту, на яку оцінку заслуговує робота

Вважаю, що кваліфікаційна робота студента Козакевича Віталія Анатолійовича може бути рекомендована до захисту та заслуговує на оцінку "добре".

Науковий керівник



викладач кафедри КН Олена СОБКО



ВІДГУК ОПОНЕНТА

на кваліфікаційну роботу магістра

гр. КНМ-22-1 Козакевича Віталія Анатолійовича за темою: Метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам

1. Актуальність обраної теми

У сучасному світі, де інформація є одним з найважливіших ресурсів, зростає потреба в автоматизації процесів її створення та обробки. Це стосується і текстової інформації, яка використовується у різних сферах діяльності. Тому метод генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам є актуальним, оскільки дозволяє створювати тексти, які відповідають вимогам семантичної спрямованості щодо змісту.

2. Відповідність роботи предметній області спеціальності 122 Комп'ютерні науки та загальним вимогам до наукових робіт

Обрана тема генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам, в межах якої виконані поставлені задачі, повною мірою відповідає предметній області спеціальності 122 «Комп'ютерні науки» та вимогам до кваліфікаційної роботи магістра.

3. Повнота розкриття мети та завдань дослідження

В кваліфікаційній роботі магістра автор повною мірою розкриває мету дослідження, а також виконує завдання, що були поставлені в межах теми.

Наявність наукової новизни

Результати виконання кваліфікаційної роботи магістра містять інновації та наукову новизну, що відповідають спеціальності 122 «Комп'ютерні науки». В роботі надано результати, які мають наукове та інноваційне значення, та їх детальне обґрунтування. Результати дослідження оприлюдненні на науково-практичній конференції «АПКН-2023».

4. Зміст кожного розділу роботи

Робота містить чотири розділи. У першому розділі виконано дослідження предметної області генерації текстових повідомлень за семантичною спрямованістю. Другий

розділ присвячено розробці методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам. У третьому розділі опис розробленого веб-застосунку, що дозволяє генерувати текстові повідомлення заданої семантичної спрямованості. У четвертому розділі описано результати дослідження ефективності розробленого методу генерації текстових повідомлень заданої семантичної спрямованості з використанням лексичних n-грам.

5. Ступінь розкриття теми роботи

Тема кваліфікаційної роботи магістра достатньо розкрита, проведений аналіз відомих джерел, що стосуються предметної області генерації текстових повідомлень за семантичною спрямованістю, досягнуто поставленої у роботі цілі, а також проведено аналіз результатів застосування запропонованих методу і відповідних засобів.

6. Якість оформлення кваліфікаційної роботи

Оформлення кваліфікаційної роботи відповідає необхідним нормам та вимогам, які ставляться до оформлення кваліфікаційних робіт.

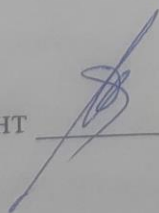
7. Недоліки кваліфікаційної роботи

Проаналізовано недостатню кількість джерел, які стосуються проблеми генерації текстових повідомлень саме за семантичною спрямованістю. Розроблене прикладне програмне забезпечення має недостатню кількість функціональних можливостей та має досить простий інтерфейс користувача. Було б доцільним більш детальне дослідження ефективності розробленого методу генерації текстових повідомлень заданої семантичної спрямованості.

8. Загальний висновок (допускається чи не допускається до захисту), якої оцінки заслуговує кваліфікаційна робота

Враховуючи достатньо високий рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка добре.

Опонент

 Шинкевич О.М. к.ф. НТБ