Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних наук

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему *Метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу*

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 4 курсу, група КН-18-1* (Курс, група виконавця) _____ (Підпис) *О.О. Лабань* (Ініціали, прізвище)

Керівник: *к.ф.-м.н., доцент кафедри КН* (Науковий ступінь, посада) _____ (Підпис) *В.Ц. Міхалевський* (Ініціали, прізвище)

Нормоконтроль: *к.т.н., доцент кафедри КН* (Науковий ступінь, посада) _____ (Підпис) *Р.О. Багрій* (Ініціали, прізвище)

До захисту допускаю:
Зав. кафедри КН, д.т.н., професор _____ (Підпис) *О.В. Бармак* (Ініціали, прізвище)
13 червня 2022 р.

Хмельницький 2022

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет *інформаційних технологій*
Кафедра *комп'ютерних наук*
Освітній ступінь *бакалавр*
Галузь знань *12 – Інформаційні технології*
Спеціальність *122 – Комп'ютерні науки*

ЗАТВЕРДЖУЮ
Завідувач кафедри *комп'ютерних наук*

(підпис)
д.т.н., професор О.В. Бармак
«*25*» *березня* *2022* року

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

1. Тема кваліфікаційної роботи бакалавра: *«Метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу»*

2. Завдання видано студенту *Лабаню Олегу Олеговичу*
(прізвище, ім'я, по батькові)

3. Керівник роботи *доц. каф. КН Міхалевський Віталій Цезарійович*
(посада, прізвище, ім'я, по батькові)

4. Затверджено наказом університету від «*01*» *березня* *2022* р. № *18*

5. Зміст пояснювальної записки (перелік задач) та вихідні дані:
Мета роботи – розробка методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та відповідної програмної реалізації розробленого методу в вигляді системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу. Розроблена система виконує ряд функцій, серед яких: попередня обробка тексту, векторизація, побудова триграм для подальшого знаходження синонімів, редагування існуючого переліку синонімів, знаходження синонімів у користувацьких текстах тощо.

Виконавець: *студент 4 курсу, група КН-18-1* _____ *О.О. Лабань*
Курс, група виконавця — Підпис — Ініціали, прізвище

Керівник: *доцент кафедри КН* _____ *В.Ц. Міхалевський*
Науковий ступінь, посада — Підпис — Ініціали, прізвище

Анотація

Тема кваліфікаційної роботи бакалавра: *«Метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу»*

Виконавець кваліфікаційної роботи бакалавра: *студент групи КН-18-1 Лабань Олег Олегович*

Керівник кваліфікаційної роботи бакалавра: *к.ф.-м..н., доцент кафедри КН Міхалевський Віталій Цезарійович*

Кваліфікаційна робота бакалавра містить:

Пояснювальна записка				Кількість додатків
Сторінок	Рисунків	Таблиць	Джерел інформації	
66	31	12	42	3

Метою кваліфікаційної роботи бакалавра є розробка методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та відповідної програмної реалізації розробленого методу в вигляді системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу. Для розробки автоматизованої системи було використано мову програмування C#, а також систему керування базами даних MS SQL Server.

Розроблена система призначена для SEO оптимізації та створення різноманітної кількості пошукових запитів, а також для копірайтерів для покращення унікальності текстів шляхом використання синонімів.

Напрямками практичного використання розробленої системи є автоматизований пошук синонімів користувачами для вирішення проблем аналізу текстової інформації та розширення словникового запасу.

Ключові слова: аналіз текстової інформації, n-грами, триграми, ключові слова.

Виконавець: *студент 4 курсу, група КН-18-1* (Курс, група виконавця) _____ (Підпис) *О.О. Лабань* (Ініціали, прізвище)

Зміст

Перелік скорочень

Скорочення, термін, позначення	Пояснення
БД	База даних
ІС	Інформаційна система
ІТ	Інформаційні технології
КРБ	Кваліфікаційна робота бакалавра
КН	Комп’ютерні науки
ПП	Програмний продукт
СКБД	Система керування базами даних
ХНУ	Хмельницький національний університет.
FCL	Framework Class Library
LSA	Latent Semantic Analysis
MS	Microsoft
MFC	Microsoft Foundation Class
TF	Term Frequency
TF-IDF	Term Frequency – Inverse Document Frequency
SEO	Search Engine Optimization

Вступ

Один із типів сприйняття інформації людиною є сприйняття її через текст. Також текст є одним із каналів передачі інформації від людини до людини, від людини до машини. Важливим є правильна та чітка подача інформації в тексті, а якщо це стосується літературного тексту чи тексту, який висвітлює певні події в журналістиці, то він повинен відповідати певному культурному та літературному рівню.

Перенасичення інформаційного простору неякісним контентом знижує рівень зацікавленості читача в певних ресурсах, тому важливо, щоб текст, який публікується на загал, відповідав ряду критеріїв високоякісного тексту, а саме грамотності, унікальності, лаконічності тощо. Уміння створювати якісний текст потребує неабияких зусиль від автора, який його створює, адже сучасний читач має доступ до різних джерел інформації, тому зацікавленість автора в тому, що цей читач залишився з ним.

Одним із способів зробити текст унікальним та не нудним для читача – це використання синонімів для ключових термінів написаного тексту. Важливо здійснювати підбір синонімів відповідно до стилю тексту.

Інформаційні технології надають різноманітні засоби для автоматизації роботи з текстами – виділення семантичного ядра (ключових слів, словосполучень та термінів), структуризація тексту тощо. Тому є перспективною розробка методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу. Це надасть можливість проводити пошук синонімів слів у тексті з подальшою можливістю семантичного аналізу даного тексту.

Мета кваліфікаційної роботи бакалавра – розробка методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та відповідної програмної реалізації розробленого методу в вигляді системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.

Об’єкт дослідження – процес семантичного аналізу цифрового тексту.

Предмет дослідження – інформаційні технології, моделі, методи та засоби для автоматизованого пошуку синонімів у цифрових текстах.

У роботі проаналізовано сучасні методи та технології для автоматизованого пошуку синонімів, також розглянуто приклади вже існуючого програмного забезпечення для вирішення даної задачі, запропоновано власний підхід, що базується на побудові n-грам.

Розділ 1 Характеристика предметної області: аналіз моделей, методів та реалізацій

1.1 Аналіз інформаційних моделей

У еру інформаційних технологій людей усюди оточує текстова інформація. Люди проводять багато часу за читанням статей, книг, газет, журналів, як в друкованій так і в електронній версіях. Важливим на сьогодні є також і вміння писати тексти, аби вони були на високому рівні та не відштовхували читача.

Уміння журналістами висвітлювати певні події, уміння письменників писати привабливі для читача твори, а також уміння копірайтерів створити такий текст, після читання якого захочеться придбати товар – доволі нелегка справа. Тим паче, беручи до уваги високу ерудованість та вибагливість сучасного читача. Тому можна впевнено говорити про те, що якість текстів має бути на високу рівні.

Насичення інтернет-простору різноманітною текстовою інформацією відбувається щосекундно, проте не можна сказати, що вона є якісною. Ряди авторів такого контенту поповнюють щоденно, проте не можна сказати, що вони є компетентними в цій справі.

Можна виділити декілька критеріїв, за якими можна оцінити якість тексту [1]:

– унікальність – текст, який що публікується в інтернеті, або в іншому місці повинен бути унікальним. Важко створити 100% унікальний текст, проте відсоток унікальності повинен варіюватись 80-90%;

– грамотність та простота подачі – інтерес до написаного тексту суттєво знижується, якщо у ньому є помилки (граматичні, пунктуаційні тощо) та якщо він написаний складно для розуміння, до прикладу має велику кількість складнопідрядних речень, що важко пов’язуються між собою;

– інформаційне насичення – текст повинен повністю розкривати суть того, про що в ньому розповідається з максимальною кількість подробиць, для повного розуміння читачем того, що бажає донести автор;

– лаконічність – текст повинен розкривати заявлену тему в повному обсязі, не відхиляючись на інші теми;

– структурованість – структурований текст завжди краще читається та сприймається. Він повинен мати назву, вступ, абзаци, підзаголовки, якщо необхідно.

Тексти повинні відповідати високому рівню культури мови. Згідно визначення, культура мови – це рівень володіння нормами літературної мови (як писемної, так і усної), а також використання мовно-виражальних засобів на високому рівні зважаючи на те з якою метою ведеться мовлення та за яких обставин [2].

В свою чергу літературна мова – це форма національної мови, яка відповідає певним стандартам, нормам, високим рівнем грамотності та має функцію обслуговувати культурне життя народу та всі сфери його суспільної діяльності [3].

Також написаний текст повинен відповідати певному стилю мовлення, що дозволяє автору більш чітко виражати свої думки, що стосуються певної тематики. В українській мові існує шість стилів мовлення [4]:

– публіцистичний;
– науковий;
– офіційно-діловий;
– художній;
– розмовний;
– конфесійний.

Кожному із цих стилів мовлення відповідають певні мовні явища, усі ці стилі мають своє призначення і їх невідповідне використання може спотворювати думку автора [5].

Окрім цього тексти, що публікуються повинні бути оцінені за різними показниками якості тексту. Відбувається це за допомогою контент-аналізу текстів. Він в свою чергу поділяється на два типи аналізу:

– кількісний – направлений на виявлення частоти окремих символів, слів або тем у тексті;

– якісний – призначений для виявлення нетривіальних висловлювань, інтонацій повідомлення, що доноситься у тексті [6].

До прикладу, у SEO-оптимізації використовується такий термін, як «нудота тексту». Це важливий показник, який відноситься до якісних показників якості тексту і виражає частоту вживання одних і тих же слів у тексті [7]. Оцінка цього показника виражається в цифрах та не буває нульовою.

Показник нудоти має великий вплив на те, як текст може сприйматися читачем. І чим він більший, тим меншу якість має текст. Ключовим моментом, який допоможе знизити нудоту тексту – це зменшення кількість найбільш вживаних слів. Часто для цього використовують заміну слів на їх синоніми [8].

Синонімами називаються слова однієї частини мови, які мають близьке, або однакове лексичне значення, проте вони різні за звучанням та написанням [9].

Головна мета застосування синонімів – це уникнення одноманітності у тексті. Застосування синонімів урізноманітнює написаний текст, але водночас зберігає його сенс. Синоніми також бувають різних видів [10]:

– семантичні синоніми – це ті синоніми, які відрізняються за відтінками значення;

– стилістичні синоніми – використовуються у різних сферах і залежно від цього відрізняються емоційним забарвленням;

– семантико-стилістичні – відрізняються і відтінками значень, і емоційним забарвленням одночасно.

Для того, щоб знаходити найбільш повторювані слова у тексті виконується пошук ключових слів у тексті.

Ключове слово – це таке слово, яке найбільш точно передає сенс тексту, воно має істотний смисловий вплив. Ключових слів у тексті завжди декілька, саме за допомогою визначення сукупності таких слів можна визначити сенс тексту. [11]. Ключові слова формують зміст тексту та служать для створення структурно-семантичної єдності та цілісності тексту, а також здатні задавати стиль тексту. Також часто для визначення сенсу тексту знаходження лише ключових слів є недостатнім, тому проводиться пошук ключових словосполучень та термінів.

Ключове словосполучення – це сукупність слів кількістю два і більше, які визначають сенс тексту [12].

Ключовими термінами називають слова або словосполучення, що застосовуються для позначення деякого поняття [13].

Існує декілька методів пошуку ключовий слів та словосполучень. До прикладу, для пошуку ключових слів використовують:

– частотний аналіз;

– TF-IDF:

– RAKE;

– TextRank тощо.

Частотний аналіз тексту – найпростіший аналіз, який визначає частоту зустрічання слова в тексті відносно інших слів. Визначається за формулою [14]:

$$Freq_x = \frac{Q_x}{Q_{all}}, \quad (1)$$

де, $Freq_x$ – частота слова x; Q_x – кількість вживання слова x у тексті; Q_{all} – загальна кількість слів у тексті.

TF-IDF – це оцінка частоти слів, які виділяє слова більш важливі для контексту, а не ті слова, які часто з'являються в документі. Для визначення ключового слова у тексті недостатньо просто провести підрахунок повторів

кожного зі слів у тексті, адже є велика кількість певних загальних слів, які не визначають сенс цього текст [15]. Визначається за формулою:

$$TF - IDF = \frac{n_i}{\sum_k n_k} \cdot \log \frac{|D|}{|d_i \ni t_i|}, \quad (2)$$

де n_i – число входжень слова в текст;

$\sum_k n_k$ – кількість слів у тексті;

$|D|$ – кількість текстів у корпусі;

$|d_i \ni t_i|$ – кількість текстів, в яких зустрічається слово t_i.

Таким чином TF-IDF визначає, що значимість слова пропорційна кількості повторів цього слова у тексті, і обернено пропорційна кількості повторів слова у інших текстах корпусу.

TextRank – це алгоритм пошуку ключових слів тексту на основі графів. Розділивши текст на кілька складових одиниць, будується граф з'єднань вузлів, подібність між реченнями використовується як вага ребер, значення TextRank обчислюється шляхом ітерації циклу, а ключові слова з високим рейтингом попадають в результуючий набір [16].

Для застосування цих всіх методів пошуку ключових слів створюють корпуси текстів.

Корпус текстів – це підібрана та оброблена за певними правилами сукупність текстів, що використовуються для досліджень мови. Корпуси мають різні характеристики [17]:

– розмічені – можуть розмічуватись, наприклад, за тематикою текстів, їх емоційним забарвленням;

– електронний – обов’язково має бути в електронному вигляді для проведення різних маніпуляцій з ним програмним шляхом;

– репрезентативний – повинен представляти в повному обсязі об’єкт, який моделює;

– тощо.

Також корпуси бувають як одномовні, так і багатомовні, а також паралельні (паралельний переклад одного тексту на одну чи більше мов).

Серед відомих українських корпусів варто відзначити: корпус ГРАК [18], корпус порталу MOVA.info [19], корпус бібліотеки «Чтиво» [20], корпус Лейпцизького університету [21] тощо.

Генеральний регіонально анотований корпус української мови (ГРАК) – колекція текстів українською мовою, що відповідає характеристикам репрезентативності, структурованості. Дозволяє створювати свої вибірки текстів для проведення різних лінгвістичних досліджень з граматики, лексики тощо. Вміщує тексти різних жанрів з 1816 до 2020 року кількістю 90 000.

Корпус лінгвістичного порталу MOVA.info – безкоштовний корпус української мови, що також надає доступ до електронних словників. Також автори корпусу надають консультації з мовних питань. Надає можливість роботи не тільки з ключовими словами, а й з N-грамами.

Корпус бібліотеки «Чтиво» – неанотований та несистематизований корпус української мови. Тексти мають не дуже добру якість, містять багато помилок, так як не є перевіреними. Корпус не відповідає вимогам корпусної лінгвістики, тому може надавати спірні результати. Містить понад 6 гігабайт текстів українською мовою.

Корпус Лейпцизького університету – містить тексти з інтернету, загальною кількістю токенів близько 1 млрд., українською мовою доступно близько 500 тис. токенів. Доступ до корпусу відкритий та надається онлайн. Окрім пошуку за конкретним словом, доступний також пошук за словоформою.

Отже, мистецтво написання якісного текстового контенту є важливим завданням у будь-якій галузі. Якісно написаний текст важливий не тільки у художній літературі, а і для написання різних новинних статей, тексту, що призначений «продати» товар тощо. Повторення одних і тих же слів у тексті робить його нудним та не цікавим, що знижує зацікавленість читача в ресурсі, де розміщений цей текст. Для того, щоб текст був насичений, цікавий

використовують прийом заміни ключових слів на відповідні синоніми. Для цього існують спеціальні словники синонімів.

1.2 Огляд теоретичних підходів до розв'язку подібних задач

Для пошуку ключових слів та словосполучень у тексті існує ряд методів. Деякі методи для пошуку ключових слів було розглянуто у попередньому пункті. Для пошуку ключових словосполучень використовуються інші методи засновані на пошуку n-грам.

N-грама з семантичної точки зору – це послідовність слів, складів, букв. Найчастіше в якості n-грам описують послідовність саме слів [22].

N-грами бувають різних розмірів:

– біграми – послідовність з двох слів;

– триграми – послідовність з трьох слів;

– тетраграми – послідовність з чотирьох слів;

– N-грами – послідовність з більше, ніж чотирьох слів, в залежності від їх кількості N замінюється на відповідну цифру.

Наприклад, з речення «Кіт стрибає через тин» можна утворити наступні біграми:

– кіт стрибає;

– стрибає через;

– через тин.

Триграми матимуть наступний вигляд:

– кіт стрибає через;

– стрибає через тин.

Для обрахунку кількості n-грам у реченні застосовується формула:

$$N_{gramsK} = X - (N - 1) \quad (3)$$

де N_{gramsK} – кількість n-грам порядку N у тексті K з кількістю слів X

N-грами використовуються в задачах вивчення тексту та обробки природної мови. При розробці мовної моделі n-грами знайшли своє застосування для розробки уніграмних, біграмних і триграмних моделей. Google і Microsoft розробили веб-масштабні n-грам моделі, які можна застосувати для виправлення орфографії, узагальнення тексту тощо. Деякі з таких моделей можна знайти у відкритому доступі. Також вони використовуються для розробки функцій для контрольованих моделей машинного навчання SVM, MaxEnt, Naive Bayes тощо. [23].

Існує багато різних задач, в яких використовується підхід побудови n-грам. Наприклад, виявлення плагіату текстів, аналіз робочих чатів з метою виявлення конфліктних ситуацій, виявлення емоційного забарвлення тексту, використання n-грам для виявлення та виправлення помилок в текстах, виявлення образливого вмісту в тексті, аналіз n-грам в контекстній рекламі. Також n-грам знаходять своє застосування в задачах, в яких необхідно передбачити порядок слів.

Отже, n-грами широко застосовуються у різних наукових задачах, це стосується не тільки задач з обробки природньої мови, а й наприклад, пошуку генетичних послідовностей, в стисканні даних, обробці звуку тощо, так як n-грами описують не тільки послідовність слів, а й символів, складів, звуків. Використання n-грам такими великими компаніями як Google і Microsoft дало можливість розробити вільні для доступу проєкти виправлення помилок, розпізнавання мовлення, які містять трильйони слів у своїх корпусах.

1.3 Аналіз існуючих програмних рішень

Задача пошуку синонімів у цифрових текстах є доволі популярною на сьогодні і для її вирішення пропонується широкий вибір програмного забезпечення, що її вирішує. Особливо актуальна задача пошуку синонімів у текстах для SEO-галузі. SEO-спеціалісти використовують синонімайзери –

застосунки для перефразування тексту шляхом підбору відповідних синонімів. Далі розглянуто деякі з них.

Текстовід – ресур для роботи з текстами, який дозволяє користувачам робити замовлення на певні види робів, а також виконувати пошук синонімів самостійно [24]. Ресур підтримує більше 30 мов, як показує аналіз, словники доволі обмежені. На рисунку 1.1 зображено пошук синонімів за допомогою даного ресурсу.

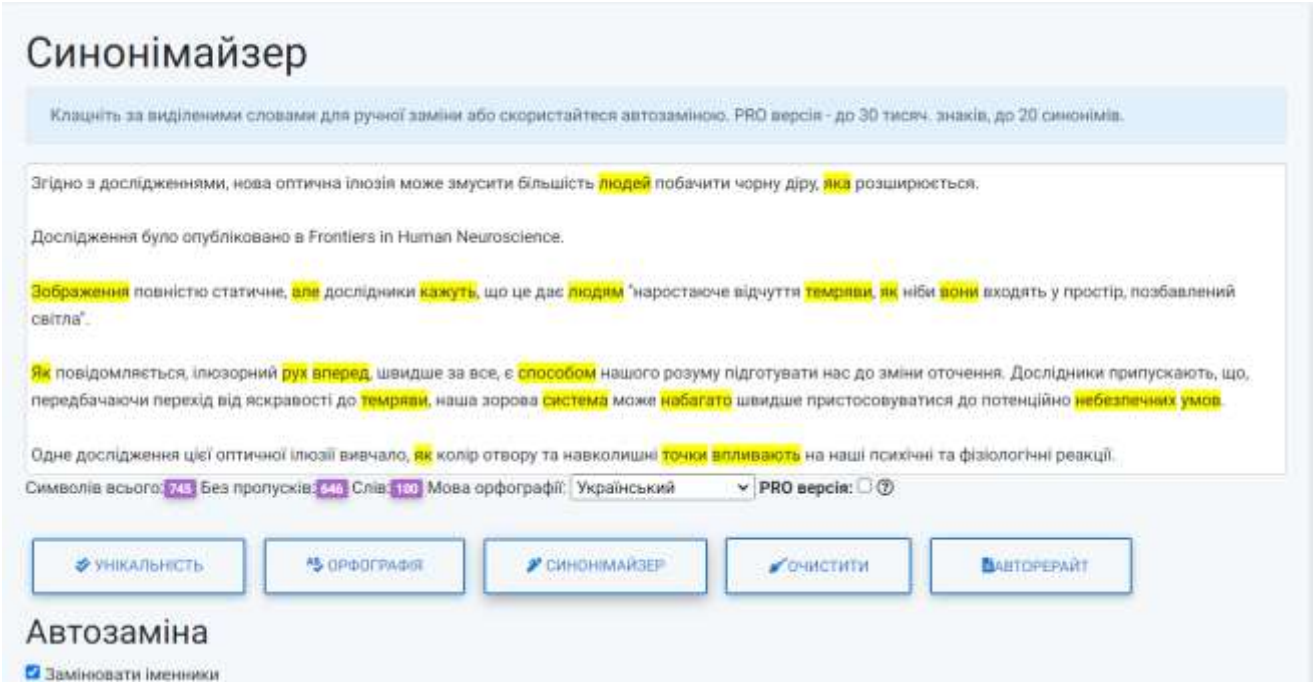


Рисунок 1.1 – Пошук синонімів за допомогою ресурсу Текстовід [24]

Як видно з рисунку, для української мови база даних цього ресурсу містить обмежену кількість синонімів, адже замін для більшості слів система не пропонує.

Smodin – ресурс, що створений для того, щоб зробити програмні застосунки доступними для кожної мови. Призначений для допомоги студентам, письменникам, викладачам для SEO-спеціалістам у роботі з текстами [25]. Застосунок реалізований у вигляді сайту, що надає доступ до перевірки тексту на плагіат, генерації цитат, перекладу на декілька мов та перефразування текстів. Функція перефразування, тобто підбору синонімів зображена на рисунку 1.2.

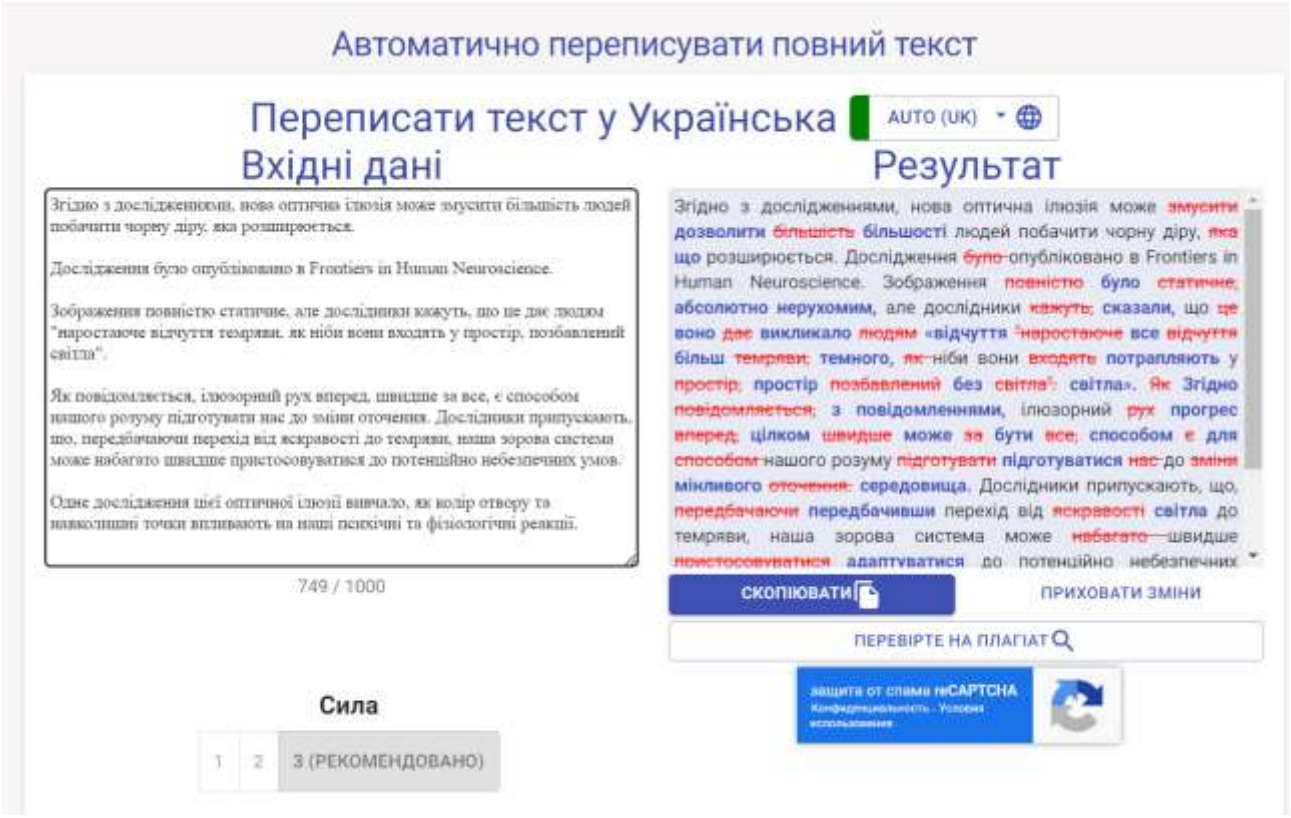


Рисунок 1.2 – Пошук синонімів за допомогою ресурсу Smodin [25]

Перевагами цього інструменту є те, що він може показувати заміни у тексті для того, щоб користувач наочно побачив яке слово на який синонім було замінене. Також він автоматично визначає мову тексту (підтримує близько 50 мов). Єдиним недоліком є те, що він має обмежену кількість спробу перефразування на добу.

Synonyma – синонімайзер, що підтримує декілька словників для однієї мови. Якщо користувачу не подобається підбір синонімів з одним словником, то він може обрати інший [26]. Результат підбору синонімів зображено на рисунку 1.3.

Користувач може самостійно проглянути варіанти, які пропонує для заміни система. Для цього достатньо клацнути на слово, що підсвічується сірим кольором в полі виведення результату. Як і більшість синонімайзерів ресурс має обмеження щодо кількості символів для вхідного тексту, проте це обмеження може бути зняти, якщо придбати підписку.

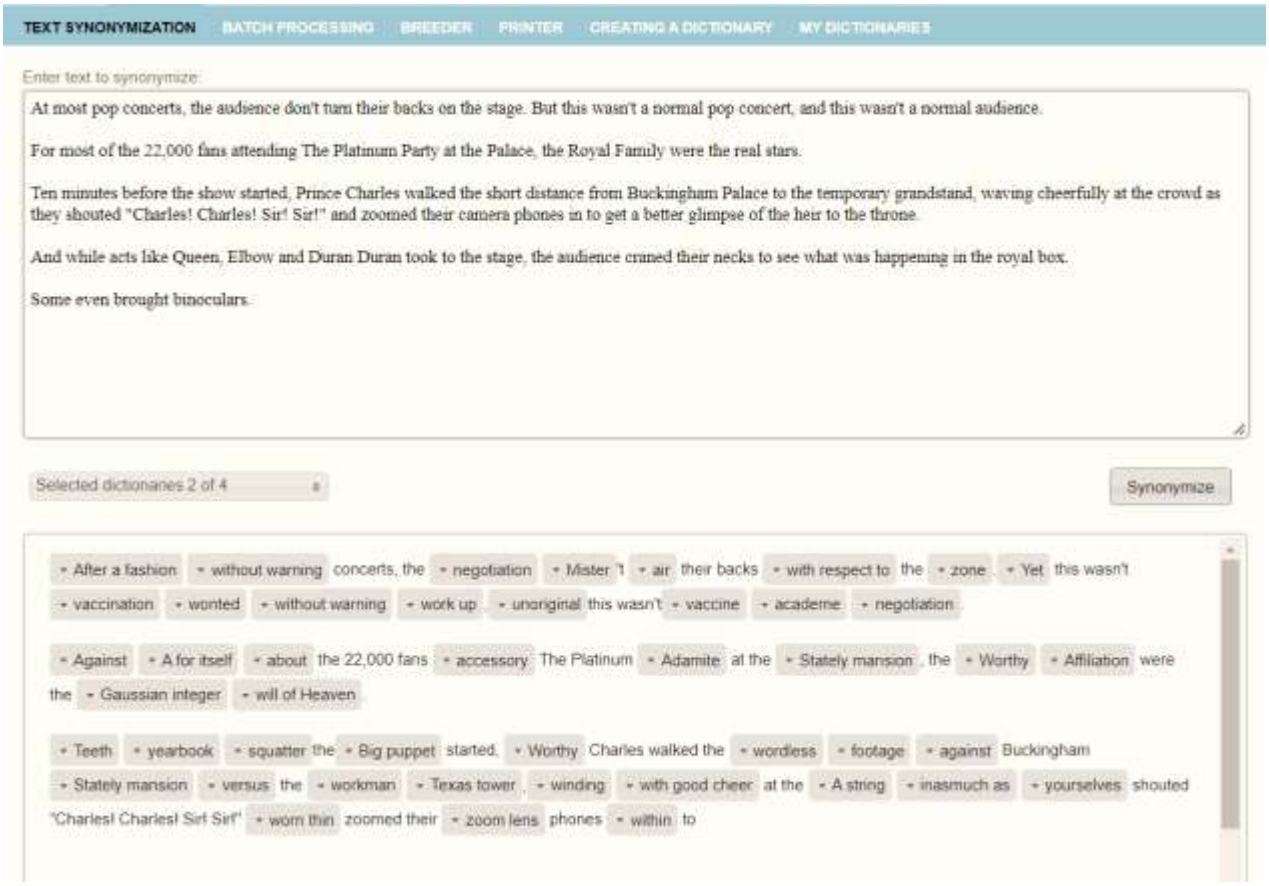


Рисунок 1.3 – Пошук синонімів за допомогою ресурсу Synonyma [26]

Також для української мови створено багато онлайн-словників, які не можуть шукати синоніми в тексті, але можу запропонувати користувачу синоніми до слова, яке вводиться.

Словник синонімів – великий словник синонімів української мови [27]. Користувач може переглядати усі слова, для яких є синоніми за алфавітом, або ж здійснювати пошук самостійно за потрібним словом. Приклад підбору синонімів до слова наведено на рисунку 1.4.

Окрім підбору синонімів також наводяться приклади вживання шуканого слова у різних реченнях (Рисунок 1.5).

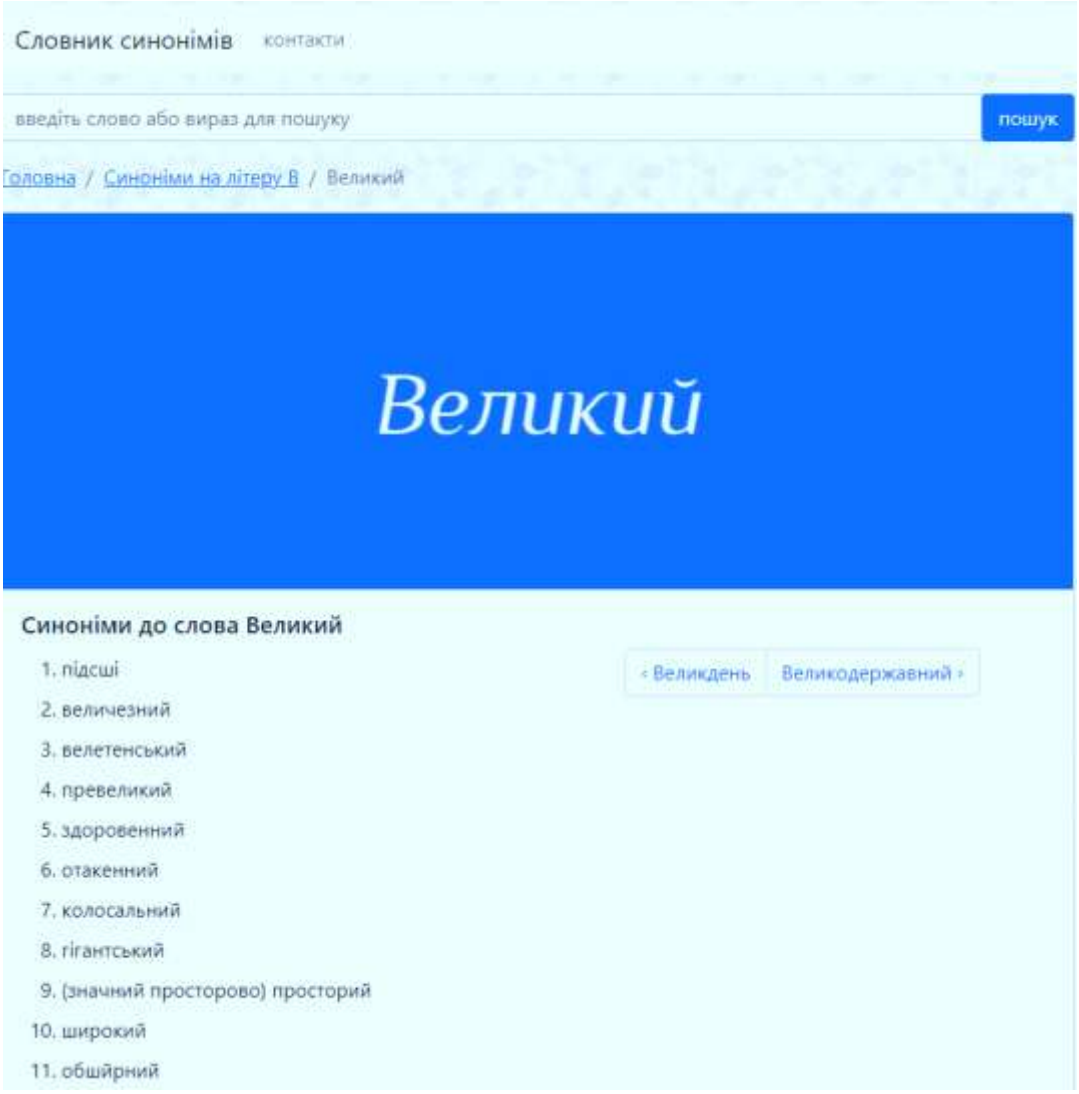


Рисунок 1.4 – Результат пошуку синонімів до слова у Словнику синонімів [27]

Рисунок 1.5 – Приклад вживання у різних реченнях шуканого слова у Словнику синонімів [27]

Синоніми.укр – сервіс містить синоніми української мови і за лічені секунди дозволяє підібрати синоніми до необхідних слів для того, щоб користувач міг урізноманітнити свій текст [28]. Результат пошуку синоніма до слова зображено на рисунку 1.6.

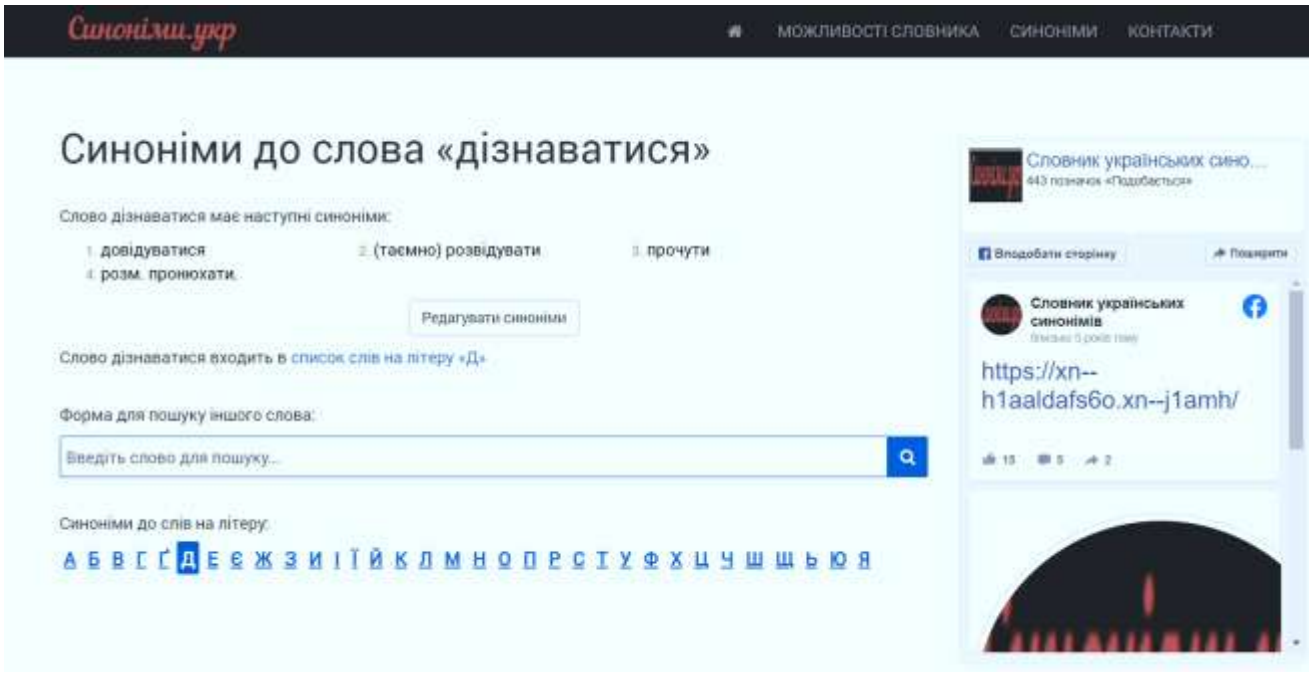


Рисунок 1.6 – Результат пошуку синонімів до слова у Синоніми.укр [28]

Дуже зручною в процесі пошуку синонімів є форма, яка пропонує варіанти синонімів по мірі введення користувачем літер необхідного слова (Рисунок 1.7).

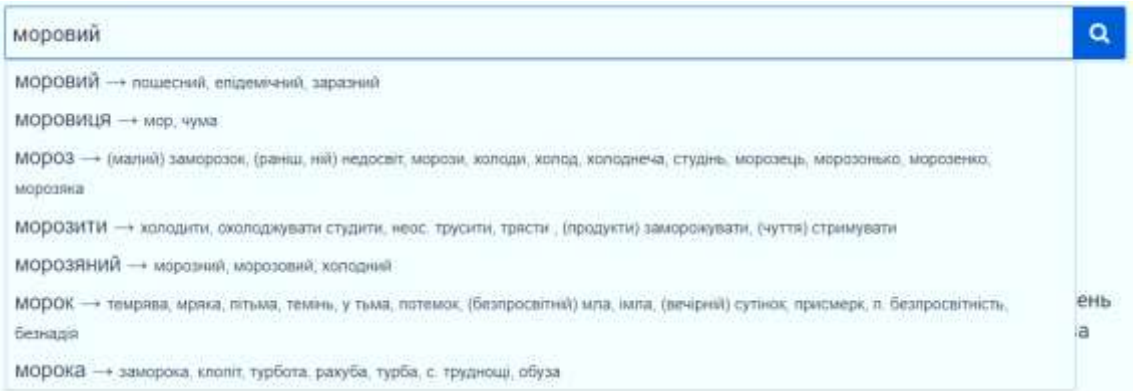


Рисунок 1.7 – Зручна форма пошуку синонімів слова у Синоніми.укр [28]

Отже, для пошуку синонімів у тексті існує багато інструментів, це словники, які допомагають знайти ряд синонімів для конкретного слова, і застосунки-синонімайзери, які підбирають синоніми автоматично і навіть можуть замінювати декілька слів на одне або навпаки. При цьому деякі з них дозволяють користувачу підібрати найбільш вдалий варіант синоніма, а деякі не мають цієї можливості. Можна сказати, що без втручання користувача шукати і підбирати синоніми у тексті майже неможливо, адже не завжди підібраний синонімайзером синонім підходить за стилем тексту. Також недоліком таких синонімайзерів є обмежена кількість синонімів у словнику.

1.4 Аналіз сучасних засобів створення програмного забезпечення

Різноманіття сучасних електронних пристроїв такого плану як персональні комп’ютери, мобільні телефони, планшети тощо вимагає від розробників проводити аналіз IT-ринку перед розробкою затребуваного програмного забезпечення. Це необхідно для того, щоб визначити потребу кінцевих користувачів у типі застосунку, його функціонального наповнення. Основні типи застосунків, які на теперішній час займають велику частину ринку це:

- мобільні застосунки.
- веб-застосунки;
- настільні застосунки;

Значення мобільних застосунків доволі очевидно. Кожен, хто користується смартфоном або планшетом, проводить значну частину свого повсякденного життя, взаємодіючи з цими пристроями. Для багатьох мобільні застосунки є необхідністю, що робить щоденні завдання надзвичайно зручними для користувачів. Часто відсутність необхідних застосунків фактично зупиняє значну частину щоденних справ [29].

Мобільні застосунки мають ряд переваг. Серед них [30]:

– покращена персоналізація – мобільні застосунки можуть дозволити користувачам налаштувати свої параметри на початку користування, на основі яких користувачі можуть отримувати необхідний вміст. Застосунки також можуть відстежувати взаємодію клієнтів, а потім використовувати цю інформацію для того, щоб пропонувати користувачам спеціальні рекомендації та оновлення;

– простота сповіщень – можна надсилати користувачу прості ненав'язливі сповіщення, які збільшують взаємодію користувача із застосунком;

– використання вбудованих функцій смартфона – мобільні застосунки мають перевагу, використовуючи функції смартфона, наприклад камера, GPS, телефонні дзвінки, акселерометр, тощо;

– робота в автономному режимі – хоча більшість застосунків мають необхідність Інтернет для виконання більшості завдань, проте мобільний застосунок може пропонувати основний вміст і функціональність користувачам в автономному режимі;

– гнучкість дизайну – користувач має можливість сам налаштувати зовнішній вигляд програми, також мобільні застосунки мають можливість підлаштовуватись під пору дня, а також рівень освітлення (використовуючи датчик освітлення).

З недоліків можна зазначити наступне: необхідність постійних оновлень, дороговартісна розробка, необхідність підтримки застарілих моделей смартфонів.

Розробка веб-застосунків відкрила цілий новий світ можливостей та ідей для людей, які ними користуються. Проте нерідко веб-застосунки підходять не для всіх, незважаючи на їх численні переваги. Серед переваг веб-застосунків варто відзначити [31].

– можливість спілкуватися з будь-ким, незалежно від технології, яку використовує адресат та адресант;

– велика аудиторія – Інтернетом користуються буквально мільйони людей з різною метою. Завдяки цьому можна знайти свою аудиторію і зацікавити своїм веб-застосунком;

– доступ 24/7 – доступ до веб-застосунку можна отримати з будь-якого пристрою і в будь-який час. Маючи свій профіль користувача можна паралельно працювати з веб-застосунком, маючи доступ до даних з різних пристроїв;

– простота розробки – існує безліч інструментів для розробки веб-застосунків, та вже готових шаблонів для створення, що робить процес створення веб-застосунку простим, швидким та відносно дешевим;

Головними недоліками є: постійне оновлення інформації, для того, щоб сайт залишався актуальним, велика кількість конкурентів, необхідність просування веб-застосунки, порівняно низький рівень безпеки.

Настільні застосунки є одними із перших застосунків, що з’явились та досі продовжують своє існування завдяки ряду переваг. До них можна віднести [32]:

– в більшості випадків настільні застосунки не покладаються на підключення до Інтернету;

– висока ефективність – настільні комп’ютери є потужнішими, ніж смартфони, та можуть надавати користувачу більше можливостей та виконувати задачі швидше;

– зручність використання – часто настільними застосунками зручніше користуватися, ніж веб-застосунками чи мобільними застосунками;

– немає великої залежності від Інтернет – всі необхідні дані для роботи з застосунком зберігаються на персональному комп’ютері користувача;

– гнучкість – набагато простіше настільні застосунки налаштовуються під потреби користувача для отримання більшої ефективності.

– можливість отримати доступ до застосунку та його даних за потреби через налаштування віддаленого мережевого доступу;

– безпечність – на відміну від веб-застосунків, настільні застосунки мають кращий рівень безпеки.

Недоліками є: необхідність встановлення настільного застосунку на ПК, дороговартісна розробка та потреба у покупці необхідного користувачу ПЗ для роботи з ним.

Отже, беручи до уваги усі переваги та недоліки вищеперерахованих типів застосунків найбільш доцільним буде розробка настільного застосунку, що дасть можливість користувачу отримати у зручний спосіб до широкого набору функцій.

Розробка настільних застосунків відбувається за допомогою інструментів програмування, які надаються програмними платформами. Найбільш відомі та популярні з них – це Java та .NET Framework. Далі розглянуто переваги та недоліки цих платформ.

Платформа Java – одна із найпопулярніших платформ для розробки програмного забезпечення уже понад 20 років. Java дозволяє розробляти безліч програм – від вбудованих пристроїв до смартфонів. Java була створена як платформа, яка може працювати на будь-яких пристроях безперебійно [33]. Розглянемо переваги даної платформи [33]:

– Java проста у використанні, а також у написанні, компіляції, налагодженні, ніж альтернативні мови програмування;

– код, що написаний на Java може працювати на будь-якій машині, яка не потребує встановлення спеціального програмного забезпечення, окрім Java Virtual Machine;

– застосування розподілених обчислень, що дозволяють запускати й виконувати завдання на кількох комп’ютерах у мережі, які працюють разом. Це дає можливість виконувати програму на різних комп'ютерах, збільшуючи її функціонал, проте розподіляючи ресурси для виконання на різні машини;

– багатопоточність дозволяє виконувати функцій програми в декілька потоків;

Недоліками платформи Java є [33]:

– споживання великих об’ємів пам’яті, а це означає, що їй потрібен значний об’єм пам'яті для швидкого виконання програм;

– швидкодія платформи Java є відносно меншою, ніж скомпільованих мов програмування;

– попередньо визначений зовнішній вигляд застосунків із графічним інтерфейсом, написаних на Java.

Можна зробити висновок про те, що платформа Java перевершує та працює краще, ніж інші і хоч має ряд недоліків, проте займає найвищий рейтинг в індексі TIOBE протягом останніх років.

.NET Framework – платформа для розробки програмних застосунків, розроблена компанією Microsoft для створення та виконання застосунків для операційної системи Windows. Фреймворк .NET містить інструменти розробника, мови програмування, бібліотеки для настільних застосунків, веб-застосунків, ігор. Платформа підтримує різні мови програмування, такі як Visual Basic, C#, C++, C тощо, для того щоб розробники могли вибрати мову для розробки необхідного застосунку [34].

.NET є однією з найкращих платформ для створення надійних, безпечних і гнучких веб- та настільних застосунків із-за своїх переваг:

– .NET надає багато засобів керування користувачами (UI). Маючи багатий набір вбудованих елементів керування інтерфейсом користувача, також підтримує елементи керування користувачами сторонніх компаній;

– використовує різні заходи безпеки. Має вбудовану автентифікацію Windows, яку можна використовувати для створення безпечних застосунків, а також має бібліотеки, що реалізовують криптографію для шифрування та дешифрування даних;

– легко інтегрується з іншими продуктами Microsoft, полегшує зв’язок із серверами обміну, та іншими програмами від Microsoft;

– дозволяє керувати пам’яттю під час розробки програмного забезпеченні;

– розширена робота бібліотеками надає можливість використовувати не тільки стандартні бібліотеки, а й сторонніх розробників.

Серед недоліків варто відзначити [35]:

– для професійної розробки необхідно придбати дороговартісну ліцензію;

– оновлення платформи може торкнутися функціоналу вже існуючого програмного забезпечення;

– застосунки створені на новішій версії .NET можуть не працювати на комп’ютерах, що використовують старішу версію.

Отже, виходячи з аналізу сучасних засобів створення програмного забезпечення, можна зробити висновок, що доцільним є обрати настільний тип застосунку та платформу .NET для комп’ютерів на базі операційної системи Windows.

1.5 Мета, задачі та вимоги до реалізації інформаційної системи

Метою кваліфікаційної роботи бакалавра є розробка методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та відповідної програмної реалізації розробленого методу. Для досягнення мети потрібно вирішити такі задачі:

1. Провести аналіз предметної області, у рамках якого оглянути існуючі методи для пошуку ключових слів та синонімів та існуючі програмі реалізації.
2. Розробити метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.
3. Розробити відповідну ІС автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу, зокрема структуру бази даних для даної предметної області.
4. Обрати засоби розробки для реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.
5. Створити відповідну програмну реалізацію згідно вищеописаних пунктів.
6. Провести тестування створеного за стосунку різними методами тестування та для зручності користування створити інструкцію користувача.

Розроблювана відповідно до методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу інформаційна система має виконувати функції роботи з навчальним корпусом текстів, з базою синонімічних груп та підбору синонімів до слів у тестовому тексті, зокрема наступні:

– Додавання навчального корпусу текстів для обробки.

– Здійснення базової обробка навчального корпусу текстів.

– Створення за дослідним текстом масиву слів.

– Створення за масивом слів дослідного тексту масиву оригінальних слів.

– Обрахунок оцінки TF для кожного слова з масиву оригінальних слів.

– Створення за масивом слів бази триграм.

– Обрахунок оцінки TF для кожної триграми з бази триграм.

– Формування переліку оригінальних слів з масиву оригінальних слів та значень їх оцінки TF.

– Пошук і відображення триграм, що містять обране слово на центральній позиції.

– Фіксування пар початкового і кінцевого слів триграм, що містять обране слово на центральній позиції.

– Пошук і відображення триграм, що містять ідентичні пари початкового і кінцевого слів, але інші слова на центральній позиції.

– Визначення множини інших слів з одержаних триграм як множини потенційних синонімів.

– Пошук і відображення триграм, що містять обране слово з множини потенційних синонімів на центральній позиції.

– Видалення користувачем обраних слів із множини потенційних синонімів.

– Додавання користувачем слів із масиву оригінальних слів до множини потенційних синонімів.

– Збереження результуючої множини слів як окремої множини синонімів – синонімічної групи.

– Додавання тестового тексту для обробки

– Створення за тестовим текстом масиву слів.

– Вибір користувачем робочого слова із масиву слів тестового тексту.

– Відображення множини слів синонімічної групи робочого слова.

– Вибір користувачем синоніма із множини слів синонімічної групи робочого слова.

– Відображення фрагментів тексту, що містять обраний синонім.

Розділ 2 Проєктування інформаційної системи

2.1 Метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу

Схематично метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу умовно можна розділити на три блока (рисунок 2.1). У першому блоці відбувається робота з навчальним корпусом текстів, та виконується ряд підзадач. Для формування достатньої дослідницької бази необхідно забезпечити додавання навчального корпусу текстів для обробки, який потім повинен буде пройти базову обробку навчального корпусу текстів. Тут відбувається переведення слів у нижній регістр, викидання спеціалізованих символів типу коми, крапки тощо. Також у межах першого блоку проходить створення за дослідним текстом масиву слів та перетворення його на масив оригінальних слів з обрахунком оцінки TF для кожного слова з масиву оригінальних слів. Коли масив оригінальних слів сформовано, відбувається створення за масивом слів бази триграм, для яких також відбувається обрахунок оцінки TF для кожної триграми.

Другий блок відповідає за роботу з базою синонімічних груп. В рамках блоку відбувається формування переліку оригінальних слів з масиву оригінальних слів та значень їх оцінки TF, після чого здійснюється пошук і відображення триграм, що містять обране слово на центральній позиції. За чим потім фіксуються пари початкового і кінцевого слів триграм, що містять обране слово на центральній позиції. Далі згідно пропонованого методу здійснюється пошук і відображення триграм, які містять ідентичні пари початкового і кінцевого слів, але інші слова на центральній позиції, за результатами якого визначається множини інших слів, як множини потенційних синонімів. У автоматизовано створений перелік користувач може самостійно видалити обране слово із множини потенційних синонімів, або за потреби додати слово з масиву оригінальних слів до множини потенційних синонімів. Останнім пунктом даного

блоку є збереження результуючої множини слів як окремої множини синонімів – синонімічної групи.

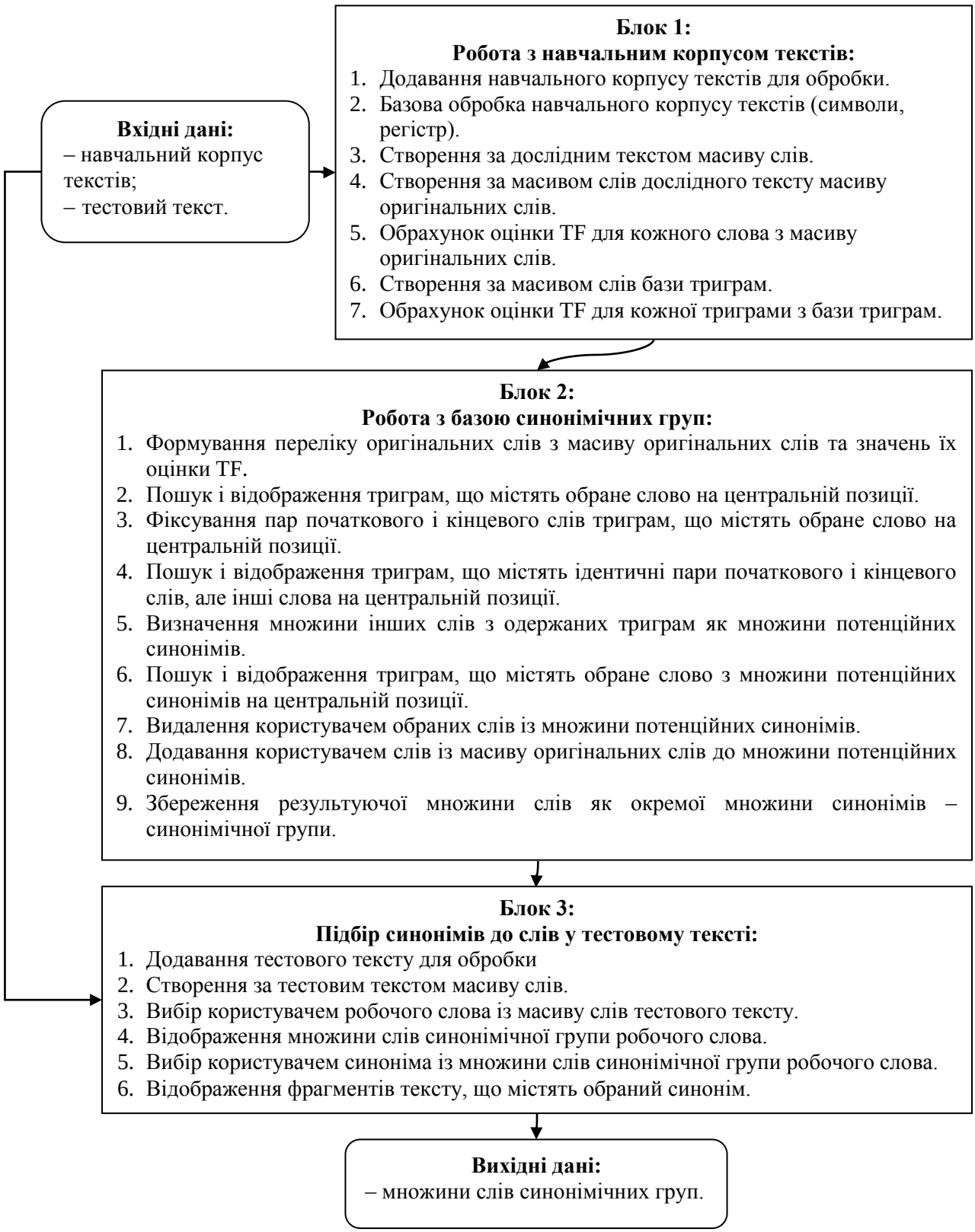


Рисунок 2.1 – Схема методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу

Третім блоком передбачено підбір синонімів до слів у тестовому тексті. У цьому блоці вирішується задача додавання тестового тексту для обробки, створення за тестовим текстом масиву слів. Після чого слідує можливість вибору користувачем робочого слова із масиву слів тестового тексту та відображення множини слів синонімічної групи робочого слова. Далі користувач може обрати синонім із множини слів синонімічної групи робочого слова та відобразити фрагменти тексту, що містять обраний синонім.

Робота з навчальним корпусом текстів представлена на рисунку 2.2

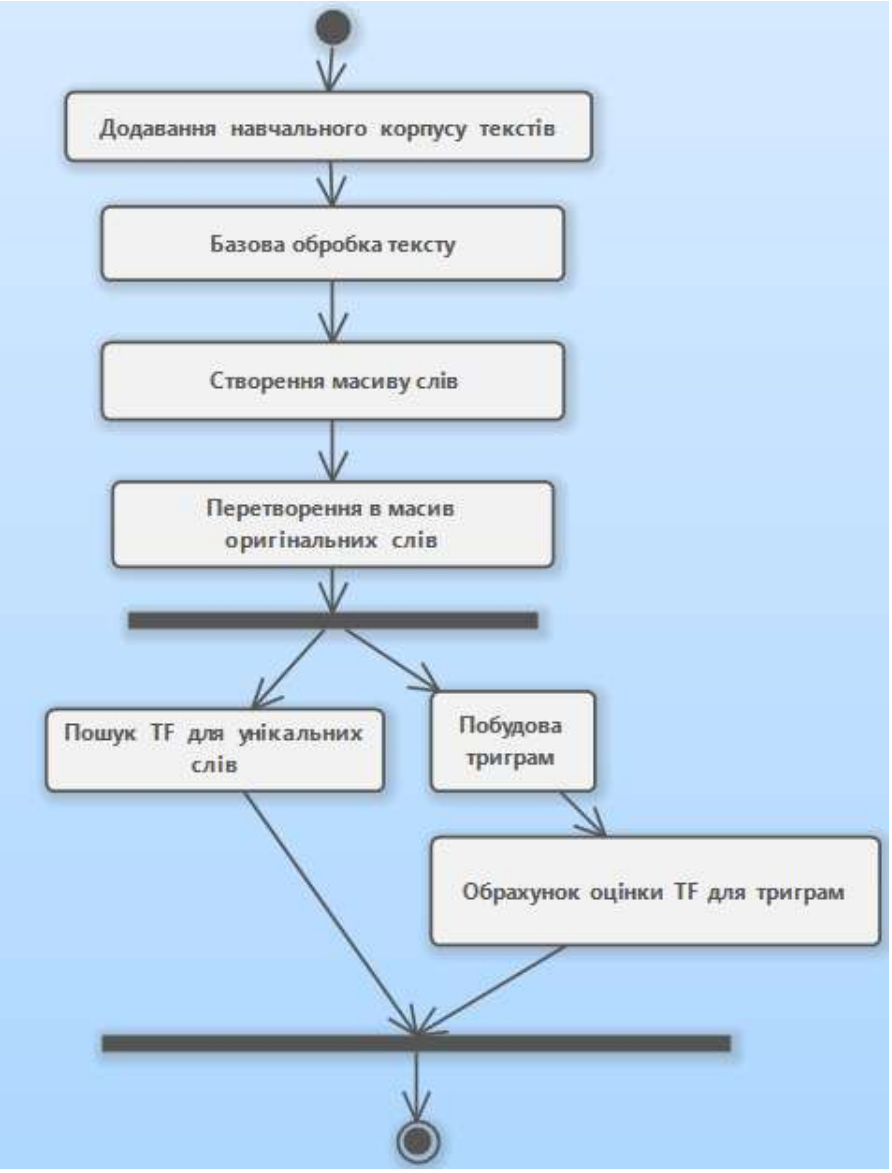


Рисунок 2.2 – Діаграма діяльності для блоку роботи з навчальним корпусом текстів

Вхідними даними пропонованого методу є навчальний корпус та тесовий текст. Як результат роботи методу – вихідними даними є отримання множини синонімічних груп.

2.2 Інформаційна структура системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу

2.2.1 Проектна архітектура системи та взаємозв'язок компонентів

Система автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу складається із трьох основних модулів та бази даних, що схематично зображені нижче (рисунок 2.3).

Архітектурна частина модуль роботи з навчальним корпусом текстів призначена для побудови триграм, які у подальшому будуть використовуватись модулем роботи з базою синонімічних груп. Також у межах модуля роботи з навчальним корпусом текстів є методи для базової обробки тексту та методи створення масиву унікальних слів з частотою їх зустрічання у тексті.

У рамках модулю роботи з базою синонімічних груп здійснюється пошук і відображення триграм, що містять обране слово на центральній позиції. Також для знаходження синонімів є метод пошуку і відображення триграм, що містять ідентичні пари початкового і кінцевого слів, але інші слова на центральній позиції. Якщо певне слово визначене системою як синонім, але насправді таким не є, існує метод видалення користувачем обраних слів із множини потенційних синонімів. Також можна вручну додати користувачу слова із масиву оригінальних слів до множини потенційних синонімів.

Третій модуль присвячений для роботи з тестовими текстами. У його рамках можна додати тестовий текст для обробки, на базі якого буде створено масив унікальних слів тестового тексту. Далі користувач може обрати слово для аналізу із масиву слів обраного текста, для якого можна виконати відображення множини слів синонімічної групи робочого слова.

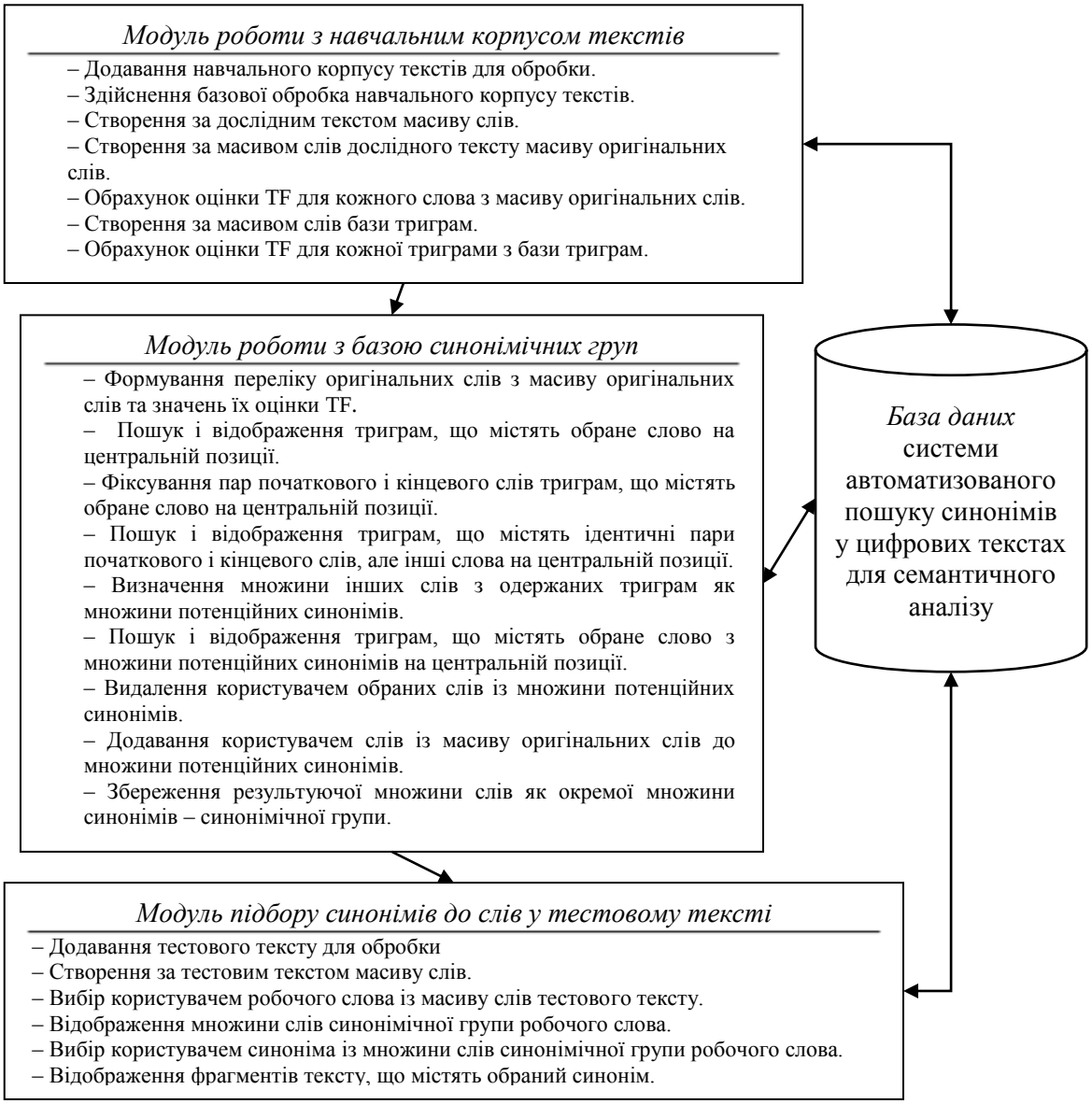


Рисунок 2.3 – Схема системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу

За обраним із множини слів синонімічної групи робочим словом користувач зможе виконати відображення фрагментів тексту, що містять обраний синонім. Для зв’язку перелічених модулів та збереження усієї потрібної інформації використовується база даних системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу, структура якої описана нижче.

2.2.2 Структура бази даних для автоматизованої системи

Для реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та забезпечення коректної роботи програмного застосунку на його основі, необхідно створити базу даних, що зберігатиме вхідні дані та результати обчислень. На рисунку 2.4 зображено даталогічну модель, відповідно до якої було реалізовано необхідні таблиці БД.

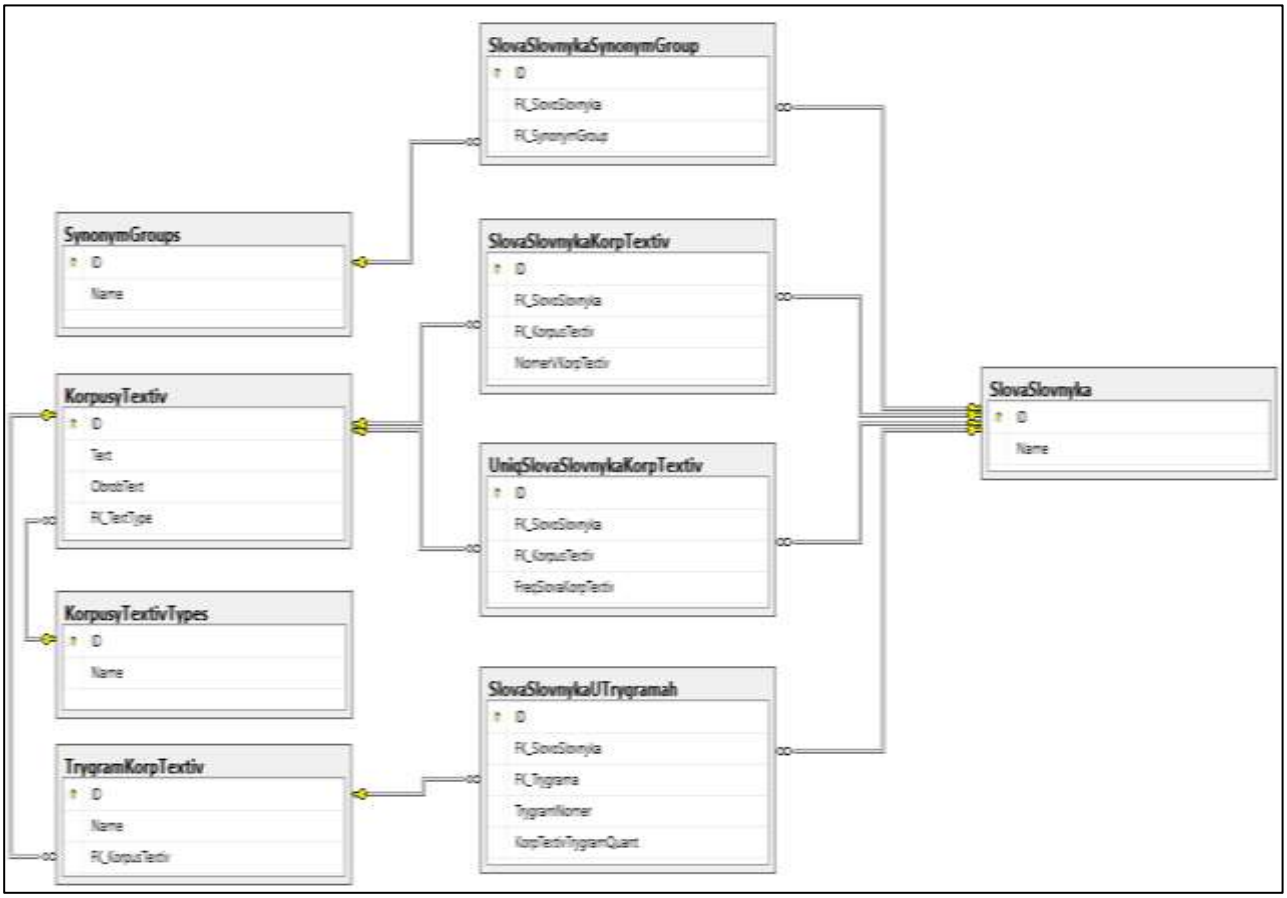


Рисунок 2.4 – Даталогічна модель бази даних методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу

Таблиця «KorpusyTextiv» (таблиця 2.1) призначена для збереження корпусів текстів.

Таблиця 2.1 – Атрибути таблиці «KorpusyTextiv»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	text	text	Контент цифрового тексту.
3.	ObrobText	text	Контент повідомлення в нормалізованому вигляді
4.	FK_TextType	int	Вторинний ключ, посилання на таблицю «TextTypes» для співставленням із відповідним типом тексту.

Таблиця «SlovaSlovnyka» (таблиця 2.2) призначена для збереження окремих слів словника.

Таблиця 2.2 – Атрибути таблиці «SlovaSlovnyka»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	Name	varchar(50)	Контент слова.

Таблиця «KorpusyTextivTypes» (таблиця 2.3) створена для збереження назв типів корпусів текстів.

Таблиця «SlovaSlovnykaKorpSliv» (таблиця 2.4) зберігає слова із словника у корпусі текстів. Зокрема, таблиця містить поля для визначення відповідного корпусу та порядкового номеру в корпусі тексту.

Таблиця 2.3 – Атрибути таблиці «KorpusyTextivTypes»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	Name	varchar(50)	Назва типу корпусу текстів.

Таблиця 2.4 – Атрибути таблиці «SlovaSlovnykaKorpTexstiv»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_SlovoSlovnyka	int	Вторинний ключ, посилання на таблицю «SlovaSlovnika» для співставленням із відповідним словом словника.
3.	FK_KorpusTextiv	int	Вторинний ключ, посилання на таблицю «KorpusyTextiv» для співставленням із відповідним корпусом текстів.
4.	NomerVKorpTextiv	int	Порядковий номер в корпусі текстів.

Таблиця «SlovaSlovnykaSynonymGroup» (таблиця 2.5) створена для збереження інформації щодо слів словника у синонімічних групах.

Таблиця «UniqSlovaSlovnykaKorpTextiv» (таблиця 2.6) створена для зберігання унікальних слів з словника у корпусі текстів.

Таблиця 2.5 – Атрибути таблиці «SlovaSlovnykaSynonymGroup»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_SlovoSlovnyka	int	Вторинний ключ, посилання на таблицю «SlovaSlovnika» для співставленням із відповідним словом словника.
3.	FK_SynonymGroup	int	Вторинний ключ, посилання на таблицю «SynonymGroups» для співставленням із відповідною синонімічною групою.

Таблиця 2.6 – Атрибути таблиці «UniqSlovaSlovnykaKorpTextiv»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці.
2.	FK_SlovoSlovnyka	int	Вторинний ключ, посилання на таблицю «SlovaSlovnika» для співставленням із відповідним словом словника.
3.	FK_KorpusTextiv	Int	Вторинний ключ, посилання на таблицю «KorpusyTextiv» для співставленням із відповідним корпусом текстів.
4.	FreqSlovaKorpTextiv	float	Числове значення частоти слова у корпусі текстів.

Таблиця «SynonymGroups» (таблиця 2.7) призначена для збереження даних щодо синонімічних груп.

Таблиця 2.7 – Атрибути таблиці «SynonymGroups»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці.
2.	Name	varchar(50)	Назва синонімічної групи.

Таблиця «SlovaSlovnykaUTrygramah» (таблиця 2.8) зберігає інформацію про слова словника у триграмах, зокрема номер в триграмі та кількість триграм у корпусі текстів.

Таблиця 2.8 – Атрибути таблиці «SlovaSlovnykaUTrygramah»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	FK_SlovoSlovnyka	int	Вторинний ключ, посилання на таблицю «SlovaSlovnika» для співставленням із відповідним словом словника.
3.	FK_Trygrama	int	Вторинний ключ, посилання на таблицю «TrygramKorpTextiv» для співставленням із відповідною триграмою.
4.	TrygramNomer	int	Номер в триграмі.
5.	KorpTextivTrygramQuant	int	Кількість триграм у корпусі текстів.

Таблиця «TrygramKorpTextiv» (таблиця 2.9) створена для збереження інформацію щодо триграм у корпусі текстів.

Таблиця 2.9 – Атрибути таблиці «TrygramKorpTextiv»

№ п/п	Назва атрибуту	Тип даних	Опис
1.	ID	int	Первинний ключ, числовий ідентифікатор для однозначного визначення запису таблиці
2.	Name	varchar(50)	Назва триграми
3.	FK_KorpusTextiv	int	Вторинний ключ, посилання на таблицю «KorpusyTextiv» для співставленням із відповідним корпусом текстів.

Таким чином, в результаті виконання розділу було створено даталогічну модель бази даних методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та відповідні таблиці із початковими вхідними даними.

2.3 Вибір засобів розробки інформаційної системи

Розробка програмного забезпечення є доволі складною задачею, тому потребує спеціальних засобів для його створення. Перед тим як розпочинати розробку застосунку необхідно визначитись на якій мові програмування буде створюватись застосунок, в якому середовищі програмування та яку систему керування базами даних обрати.

Вибір засобів розробки системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу буде розглянуто далі.

2.3.1 Вибір мови програмування

Вибір мови програмування є важливим етапом у підготовці до розробки інформаційної системи, адже різні мови програмування підходять для різних цілей, а також відрізняються складністю синтаксису тощо.

Так як у попередньому розділі для розробки застосунку обрано платформу .NET Framework, то доцільним буде обрати одну з мов, що підтримується цією платформою. Платформа підтримує наступні мови програмування: C++, C#, J#, Visual Basic [36].

Прийнято рішення обрати мову програмування C# так як ця мова програмування була створена спеціально для .NET [37]. Далі детальніше розглянуто мову програмування C# її переваги та недоліки. Серед основних переваг мови варто відзначити [38]:

– легка для вивчення та має багато функцій, які полегшують навчання;

– висока швидкодія під час процесу компіляції та запуску програм;

– об'єктно-орієнтована мова програмування, що спрощує розробку та обслуговування коду;

– є type-safe мовою, що покращує безпеку програми;

– підтримка мовної сумісності – це здатність коду взаємодіяти з кодом, написаним іншою мовою програмування, що допомагає повторно використовувати код;

– постійні оновлення та підтримка від Microsoft;

– велика кількість вбудованих бібліотек, які прискорюють розробку, та можливість використання сторонніх бібліотек.

Головним недоліком мови програмування є те, що вона повністю орієнтована на платформу .NET, що робить її не такою гнучкою в плані використання для інших платформ.

Можна зробити висновок, що мова програмування C# цілком підходить для розробки настільного додатку на платформі .NET адже має ряд переваг і мінімум недоліків. А також вона створена спеціально для роботи на платформі .NET.

2.3.2 Вибір редактора програмного коду

Для зручності створення програмного забезпечення розробники використовують спеціальний редактор програмного коду, який називається середовищем розробки. При чому такі середовища можуть бути орієнтовані на декілька мов програмування.

Для створення .NET-програм на мові програмування C# орієнтовані наступні середовища розробки: Microsoft Visual Studio, SharpDevelop, Borland Developer Studio тощо [36].

Доцільним є обрати середовище програмування Microsoft Visual Studio, так як його розробником, як і .NET є компанія Microsoft, що забезпечуватиме найкращий набір засобів для розроби інформаційної системи.

Microsoft Visual Studio є інтегрованим середовищем розробки, що вимагає його встановлення на ПК. Саме тому Microsoft значно покращила встановлення та обслуговування Visual Studio у версії 2016 року. Ця версія містить інсталятор Visual Studio, а також окрему програму, яка керує встановленням та оновленням усіх випусків Visual Studio [39].

За допомогою цього середовища розробки можна створювати різні типи застосунків, таких як ASP.NET та веб-розробка, Python, Azure, Xamarin, .NET, застосунки C++ тощо. Для цього достатньо лише встановити відповідні компоненти Visual Studio.

Visual Studio – це повноцінний інструмент для розробки ПЗ з великим набором функцій. Він підходить для створення великих проєктів. Проте має деякі недоліки. Visual Studio не є кросплатформенною та для отримання додаткових функцій розробки необхідно придбати ліцензію.

Отже, не беручи до уваги деякі недоліки, обране середовище розробки Microsoft Visual Studio є зручним для розробки системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу адже підтримує розробку настільних застосунків мовою C.

2.3.3 Вибір СКБД

Невід’ємною частиною великої інформаційної системи є база даних. У базі даних зберігаються необхідні для роботи дані. Для створення, підтримки та контролю доступу до бази даних використовується система керування базами даних. На сьогоднішній день їх існує декілька, наприклад, Microsoft Access, PostgreSQL, MySQL, Microsoft SQL Server, Oracle Database тощо [40]. Для створення та контролю доступу до бази системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу обрано Microsoft SQL Server.

Microsoft SQL Server – продукт компанії Microsoft, основною функцією якого є зберігання та робота даних за допомогою запитів інших програм, які можуть працювати на тому самому комп’ютері, або на комп’ютері в тій же мережі [41]. Microsoft SQL Server використовує Transact-SQL як мову запитів. Особливістю є те, що призначена дана СКБД для роботи з базами невеликого та середнього розмірів. Для роботи з великими базами даних варто обрати іншу СКБД [42].

Microsoft SQL Server у нових версіях використовує «хмарні» технології, а саме Windows Azure SQL Database Data Sync, тому це забезпечує безпеку даних та їх швидку синхронізацію з сервером. Для безпеки даних використовується надійне шифрування [42].

Великою перевагою Microsoft SQL Server є те, що це продукт Microsoft, тому платформа .NET може легко інтегрувати з ним, оскільки він також належить до тієї ж організації. А оскільки в попередніх розділах в якості інструментів для розробки було обрано платформу .NET та мову програмування C#, то Microsoft SQL Server в якості системи керування базами даних матиме місце в процесі розробки системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.

Розділ 3 Програмна реалізація інформаційної системи

3.1 Структура та функціональне призначення програмних складових системи

Для структуризації системи автоматизованого пошуку синонімів у цифрових текстах було створено діаграму класів, зображену нижче, на рисунку 3.1.

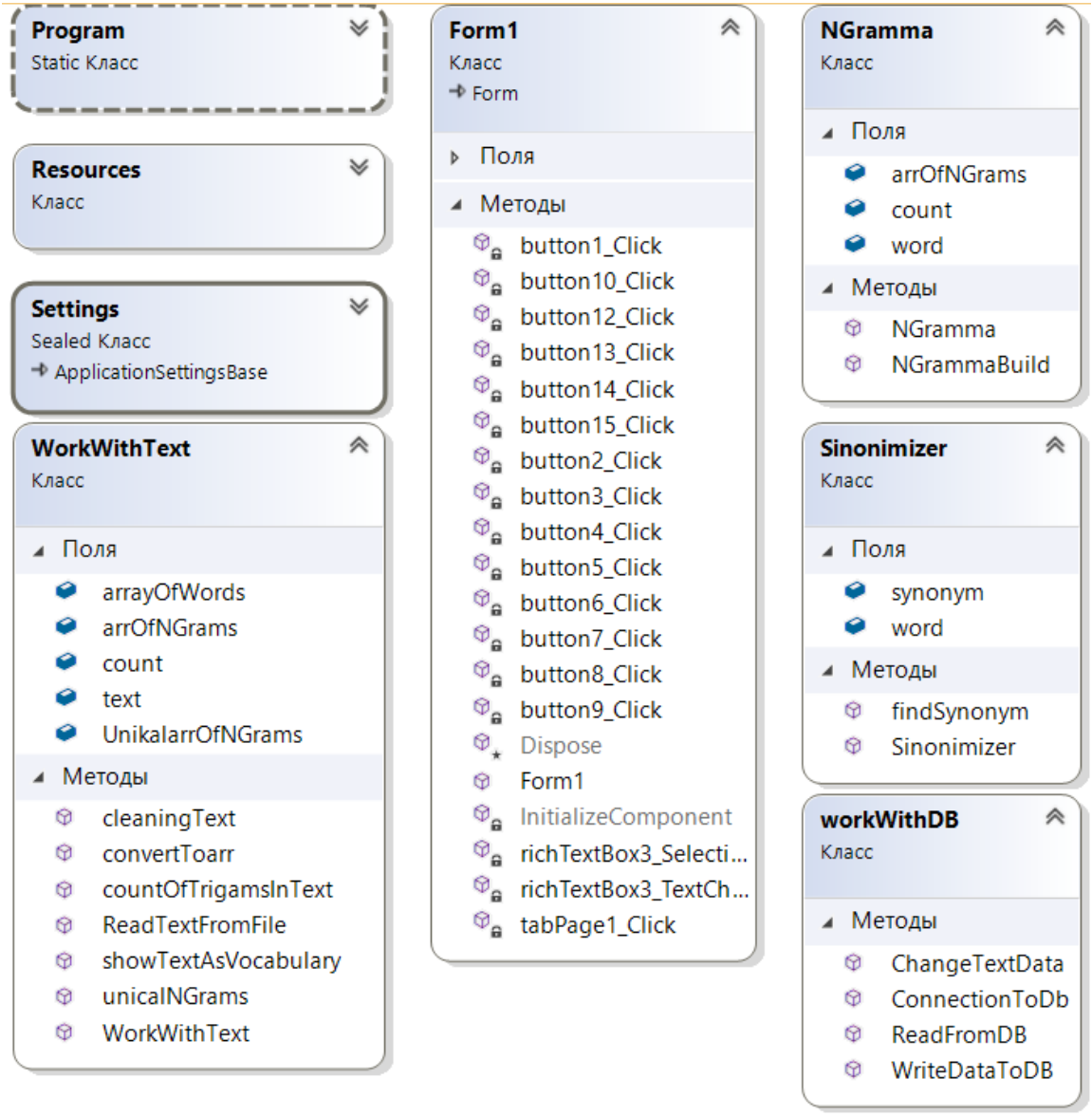


Рисунок 3.1 – Діаграма класів реалізації методу автоматизованого пошуку синонімів у цифрових текстах

Робота застосунку розпочинається із класу Program. На Form1 передбачені три вкладки для роботи із текстовою інформацією та є основний графічним інтерфейс, з яким працюватиме користувач. Реалізовано наступні вкладки:

– Робота з корпусом текстів.
– Робота з базою синонімічних груп.
– Пошук та аналіз синонімів.

Клас «WorkWithText» містить методи, що допомагають обробити текст для подальшої векторизації. До таких методів належить метод cleaningText(), що прибирає лишні символи, переводить все до нижнього регістру та будує масив слів з повторами. Метод convertToArr() призначений для конвертації перелічення слів у рядковий масив. Для завантаження тексту з файлу використовується метод readFromFile(), для відображення текста як словника без повторів написано метод showTextAsVocabulary(). Також є метод для побудови унікальних триграм – unicalNGrams(), та метод підрахунку частоти зустрічання нграм у тексті – countOfTrigramsInText().

Клас NGramma є матеріалом для формування триграм та є зручним представленням інформації для подальшої обробки. Містить конструктор та безпосередньо метод побудови триграм із можливими повторами – NgrammaBuild().

Клас WorkWithDB призначений для забезпечення взаємодії користувача та бази даних. Тут присутні методи для читання, запису та зміни даних у базі.

Клас Sinonimizer виконує пошук синонімів за методом аналізу триграм. Містить конструктор та відповідний метод пошуку синонімів – findSynonym(). Решта функціоналу реалізована безпосередньо обробниками подій відповідних компонентів. Більш детально реалізація розкрита у пункту нижче.

3.2 Особливості реалізації програмних складових системи

Під час реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу було створено ряд функцій для досягнення поставленої в 1.5 мети. Оскільки метод працює на основі підходу триграм, для коректної побудови триграм потрібно спершу провести очистку тексту від знаків пунктуації та від незначущих слів довжиною в одну букву. Для цього сформовано відповідний метод:

```
public IEnumerable<string> cleaningTextProcess(String text) {
var words = text.Split(new char[] { ' ', '.', ',', '!', '?', '/', '-', ')',
        '(', '[', ']', '—', '–', '»', '«', '#', '=', '”', '“', '@', '+','0', ':',
        '1','2','3','4','5','6','7','8','9' }, StringSplitOptions.RemoveEmptyEntries);
   return words;
}
```

Даний метод видалить не лише знаки пунктуації, але і цифри. Метод для конвертації перерахування у масив виглядає так:

```
    public String[] convertToarr(IEnumerable<string> s)
        {
            return s.Select(s1 => s1).ToArray();
        }
```

Наступним методом буде метод розбиття очищеного тексту на триграми, для подальшого виведення триграм користувачу.

```
public NGramma(String word, String[]arr, int pos)
        {
            this.word = word;
            arrOfNGrams = new String[3];
            if (pos+2 < arr.Length)
            {
                arrOfNGrams[0] = arr[pos];
                arrOfNGrams[1] = arr[pos + 1];
                arrOfNGrams[2] = arr[pos + 2];
            }
        }
```

Фрагмент програмного коду, який використовує вищеописані методи та здійснює виведення триграм на екран користувача наведено нижче:

```
arrOfWords = w.cleaningText(text);

                string[] s = w.convertToarr(arrOfWords);
                w.arrOfNGrams = new NGramma[s.Length];

                for (int k = 0; k < s.Length; k++)
                {
                        w.arrOfNGrams[k] = new NGramma(s[k],s,k);
                        dataGridView1.Rows.Add(w.arrOfNGrams[k].arrOfNGrams);
                }
```

Описаний вище програмний код дасть результати як на рисунку 3.2

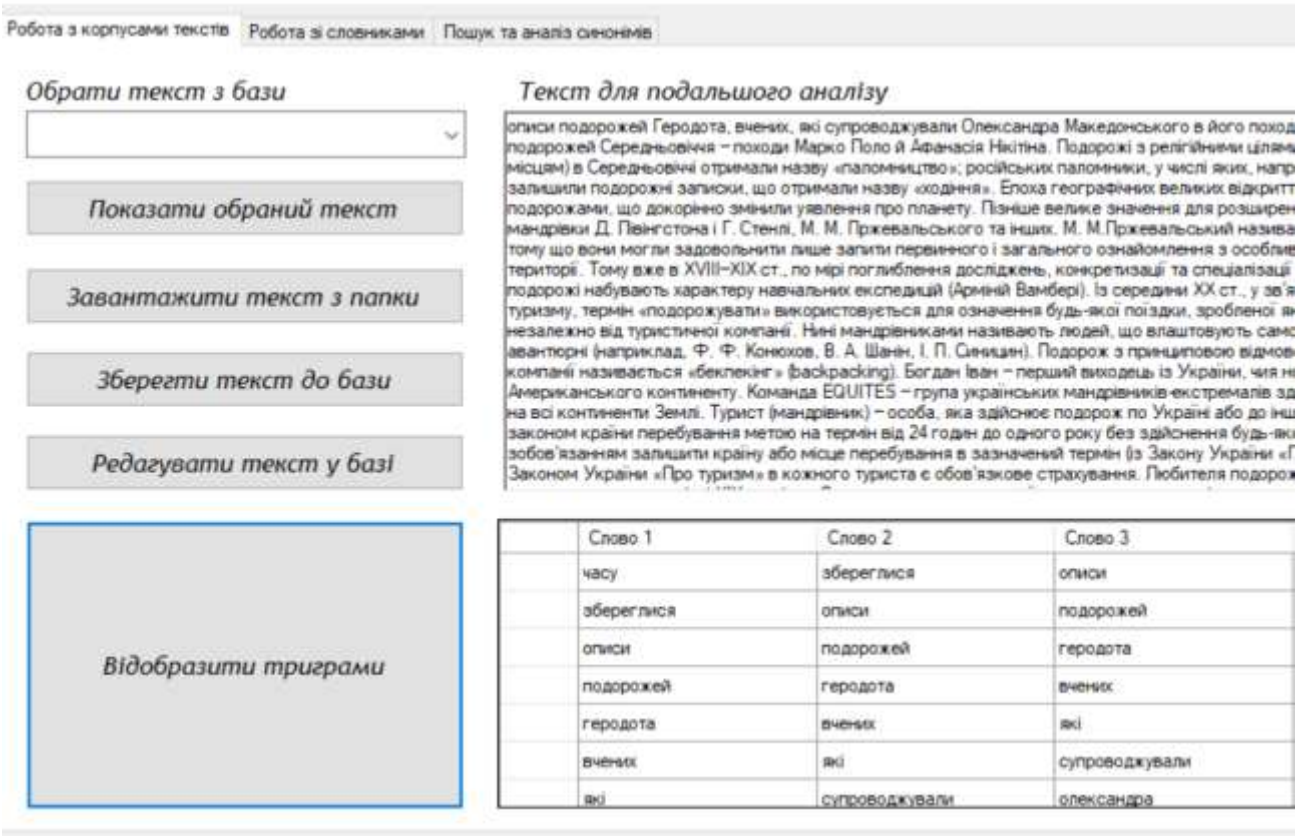


Рисунок 3.2 – Відображення триграм тексту

Оскільки важливо бачити не лише самі триграми, а й їх кількість зустрічань у тексті, наступним буде метод для підрахунку триграм у тексті.

```
public int countOfTrigamsInText(NGramma []nGrammaArr, NGramma nGramma) {
            int count=0;
            for (int i = 0; i < nGrammaArr.Length; i++)
            {
                bool isEqual = Enumerable.SequenceEqual(nGrammaArr[i].arrOfNGrams,
nGramma.arrOfNGrams);
```

```
                if (isEqual)
                { count++;
                }
            }
            return count;
        }
```

Після запуску програми користувач вже буде бачити не лише перелік триграм, а і їх кількість зустрічання у тексті (рисунок 3.3).

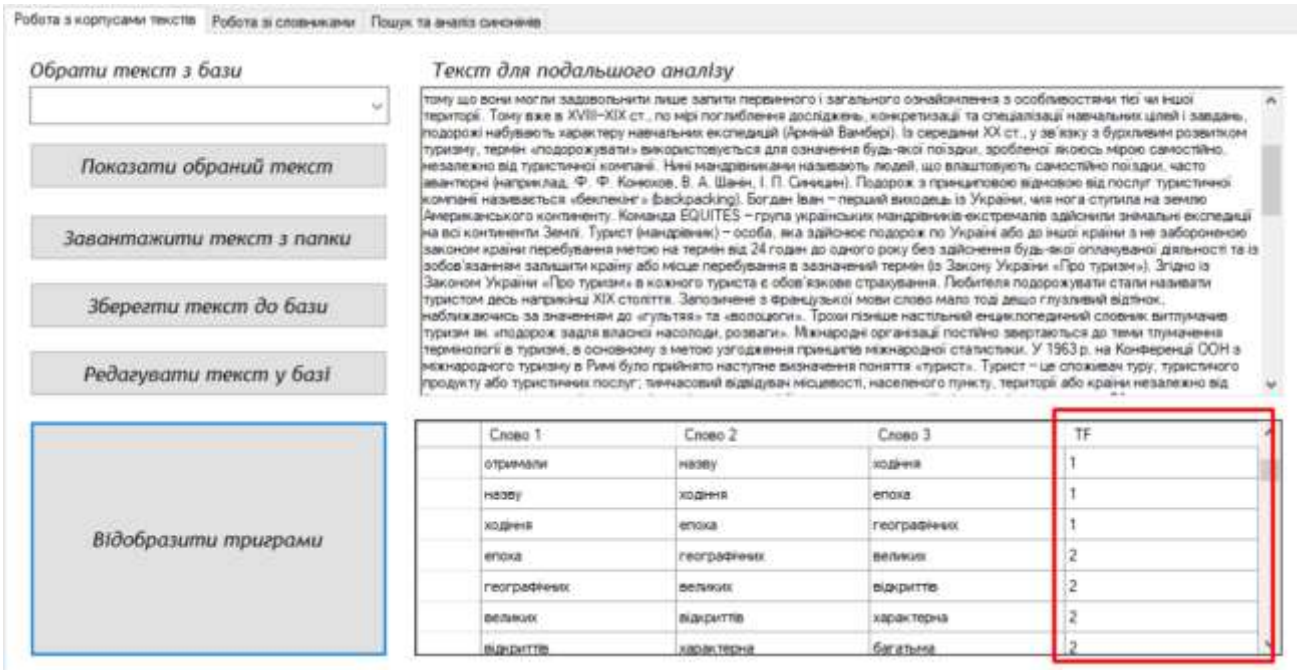


Рисунок 3.3 – Відображення триграм тексту з частотою зустрічання

Для пошуку триграм, які мають певне слово, яке цікавить користувача для аналізу використовується лямбда-вираз, а результат елементів, що задовольняють вказаним критеріям воводиться у відповідну таблицю (рисунок 3.4). Фрагмент програмного коду проілюстровано нижче:

```
  for (int i = 0; i < w.arrOfNGrams.Length-2; i++) {

if(w.arrOfNGrams[i].arrOfNGrams.Any(listBox2.SelectedItem.ToString().Contains))
                    {
                        dataGridView2.Rows.Add(w.arrOfNGrams[i].arrOfNGrams);
                    }
                }
```

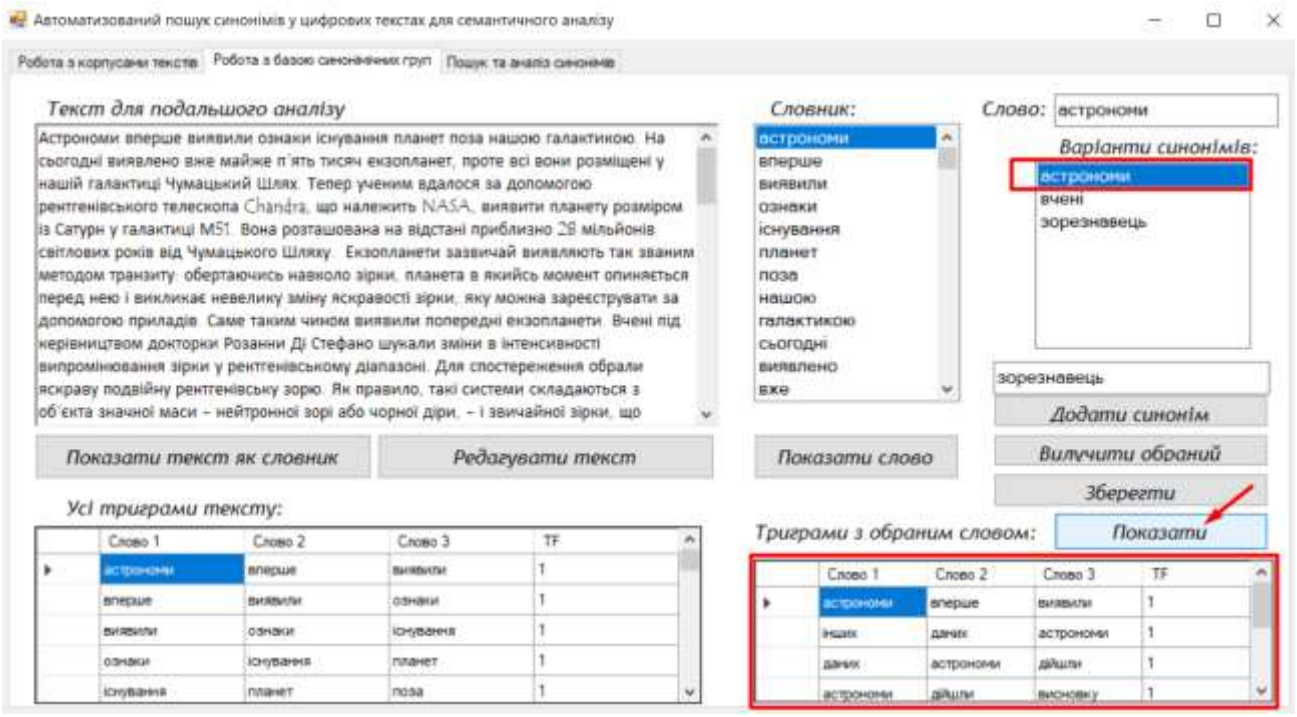


Рисунок 3.4 – Відображення триграм тексту з обраним словом

Таким чином, було проілюстровано деталі реалізації програмних складових інформаційної системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.

3.3 Тестування інформаційної системи

Функціональність створеного програмного забезпечення необхідно перевірити за допомогою тестування, а саме: тестування за допомогою тест-кейсів та функціональне тестування.

Перший тестовий випадок – перевірка завантаження навчального тексту вкладки «Робота з корпусами текстів».

Після запуску програми необхідно виконати кроки, вказані у таблиці 3.1. Після чого у програмі користувачу відобразиться результат у вигляді тексту у текстовому полі «Текст для подальшого аналізу» (рисунок 3.5).

Таблиця 3.1 Тест-кейс LA0001

Тест-кейс ID: LA0001	**Приорітет:** 1	**Створено:** 15.05.2022, Лабань О.О.
Назва: Перевірка завантаження навчального тексту вкладки «Робота з корпусами текстів» **Вхідні дані:** Додати текст із назвою «All» папки проєкту		
Кроки		**Очікуваний результат**
1. Запустити програму 2. На формі обрати вкладку «Робота з корпусами текстів» 3. Натиснути кнопку «Завантажити текст з папки» 4. Додати текст із назвою «All» з діалогового вікна 5. Перевірити наявність доданого тексту у текстовому полі.		Доданий текст успішно відображається у текстовому полі
Результат виконання тест-кейсу: перевірку пройдено успішно.		

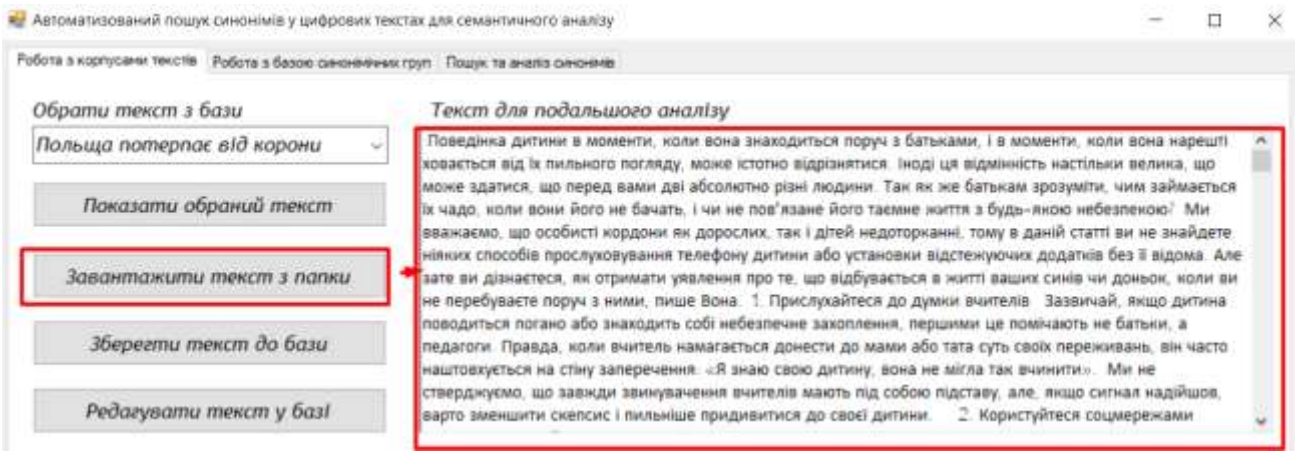


Рисунок 3.5 – Відображення завантаженого з папки тексту

Наступним тестовим випадком буде побудова триграм на базі завантаженого тексту. Кроки тест-кейсу проілюстровані у таблиці 3.2.

Таблиця 3.2 Тест-кейс LA0002

<table>
<tr><td colspan="2">Тест-кейс ID: LA0002</td><td>Приорітет: 1</td><td>Створено: 15.05.2022, Лабань О.О.</td></tr>
<tr><td colspan="4">Назва: Побудова триграм на базі завантаженого тексту з вкладки «Робота з корпусами текстів»
Вхідні дані: Додати текст із назвою «Фізики», натиснути кнопку «Відобразити триграми»</td></tr>
<tr><td colspan="3">Кроки</td><td>Очікуваний результат</td></tr>
<tr><td colspan="3">1. Запустити програму
2. На формі обрати вкладку «Робота з корпусами текстів»
3. Натиснути кнопку «Завантажити текст з папки»
4. Обрати із переліку текстів діалогового вікна текст «Фізики»
5. Натиснути на кнопку «Відобразити триграми»
6. Перевірити наявність триграм у таблиці з триграмами.</td><td>Доданий текст успішно відображається у текстовому полі

Триграми відобразились у таблиці триграм</td></tr>
<tr><td colspan="4">Результат виконання тест-кейсу: перевірку пройдено успішно.</td></tr>
</table>

При повторенні кроків, вказаних в таблиці 3.2 під час запуску програми некоректних моментів виявлено не було. Результат підтвердження коректності роботи проілюстровано на рисунку 3.6.

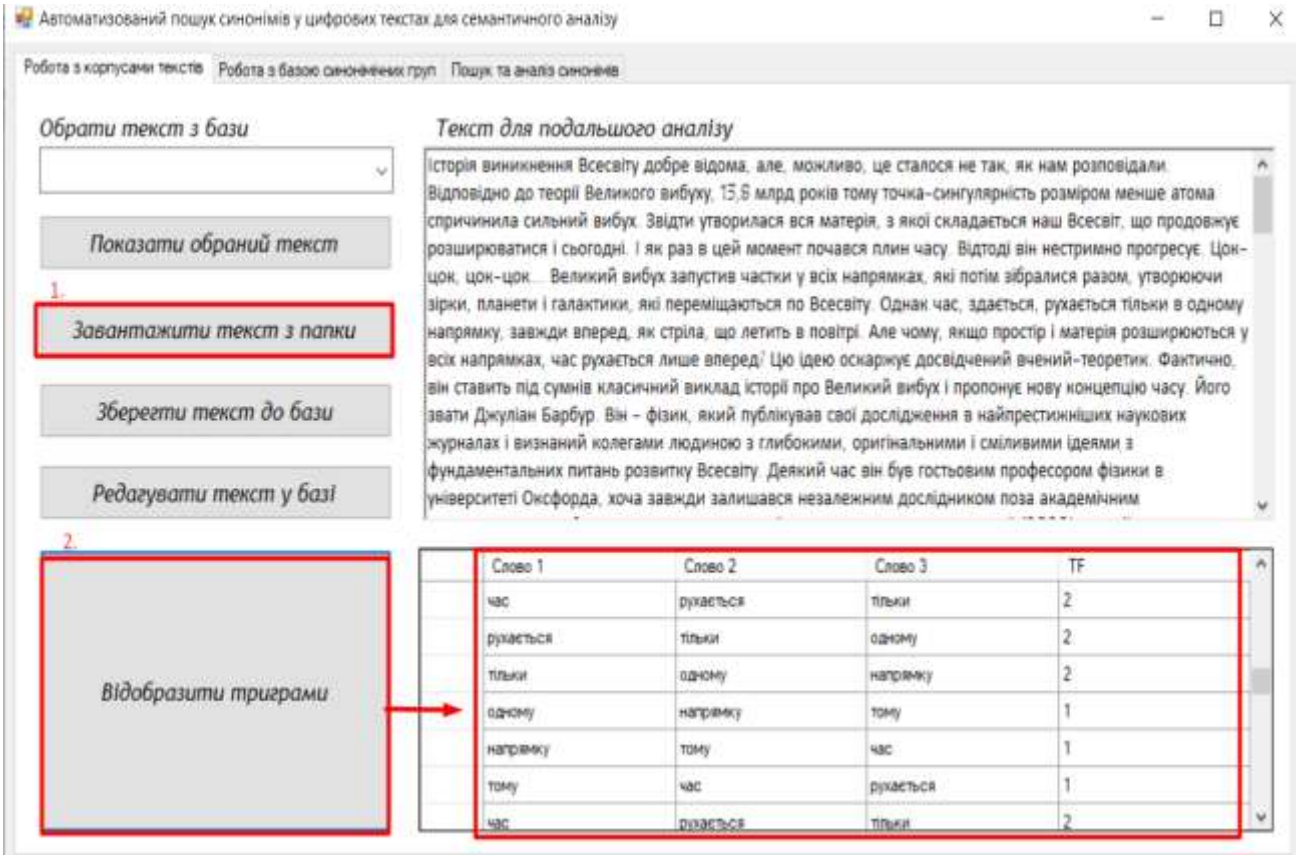


Рисунок 3.6 – Відображення триграм завантаженого тексту

Також для переконання у коректності створення триграм був створений відповідний юніт-текст, код якого наведено нижче:

```
    [TestClass()]
    public class NGrammaTests
    {
        [TestMethod()]
        public void NGrammaBuildTest()
        {
            NGramma m = new NGramma();
            String []result = m.NGrammaBuild("вчений", new string[] { "вчений", "геній",
"виконав", "план", "дій", "вказаний" }, 2);
            String[] planResult = new string[] { "виконав", "план", "дій" };
            Assert.IsTrue(result.SequenceEqual(planResult));
        }
    }
```

Тест виконано успішно, результати зображені на рисунку 3.7.

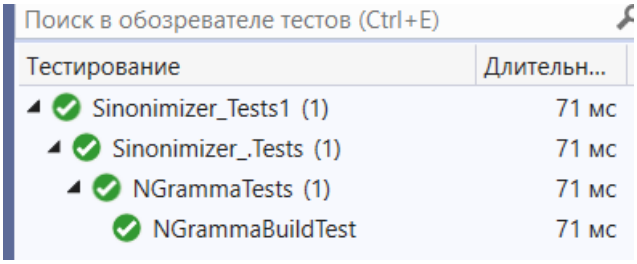


Рисунок 3.7 – Результат виконання юніт-тесту

Наступним тестовим випадком буде створення словника унікальних слів тексту. Послідовність кроків для виконання відображена у таблиці 3.3.

Таблиця 3.3 Тест-кейс LA0003

Тест-кейс ID: LA0003	**Приорітет:** 1	**Створено:** 15.05.2022, Лабань О.О.
Назва: Побудова словника унікальних слів на базі завантаженого тексту з вкладки «Робота з базою синонімічних груп» **Вхідні дані:** Додати текст із назвою «Сад», перейти на вкладку «Робота з базою синонімічних груп» та натиснути кнопку «Показати текст як словник»		
Кроки		**Очікуваний результат**
1. Запустити програму 2. На формі обрати вкладку «Робота з корпусами текстів» 3. Натиснути кнопку «Завантажити текст з папки» 4. Обрати із переліку текстів діалогового вікна текст «Сад» 5. Натиснути на кнопку «Відобразити триграми» 6. Перейти на вкладку «Робота з базою синонімічних груп». 7. Натиснути кнопку «Показати текст як словник»		Текст відобразився у формі вектора унікальних слів
Результат виконання тест-кейсу: перевірку пройдено успішно.		

При відтворенні послідовності кроків, вказаних в таблиці 3.3 під час запуску програми некоректних моментів виявлено не було. Результат підтвердження коректності роботи проілюстровано на рисунку 3.8.

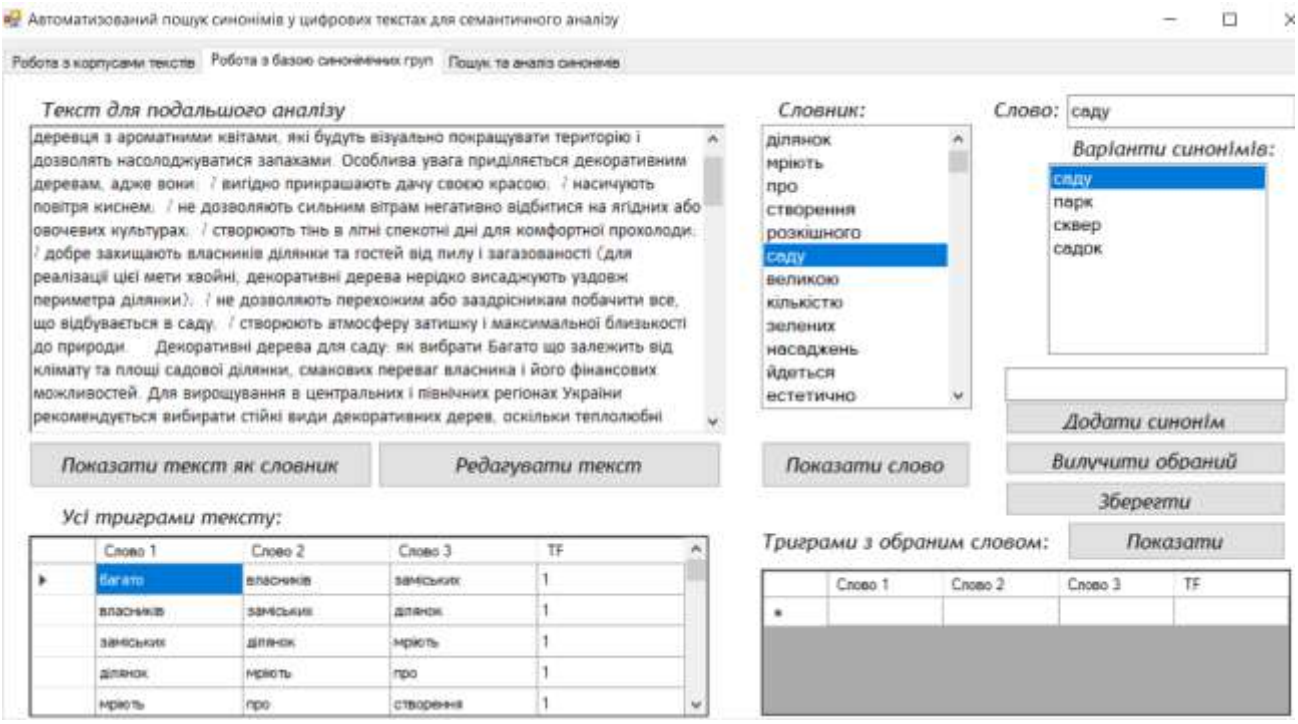


Рисунок 3.8 – Відображення вектору унікальних слів тексту

При тестуванні створеного програмного застосування некоректно працюючих функцій не виявлено. У результаті проведеного тестування можна зробити висновок, що застосунок реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу працює коректно згідно поставленої задачі.

3.4 Інструкція користувача

Для зручності використання реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу потрібно забезпечити інструкцію користувача. При запуску програми користувач побачить першу вкладку – «Робота з корпусами текстів» (рисунок 3.9).

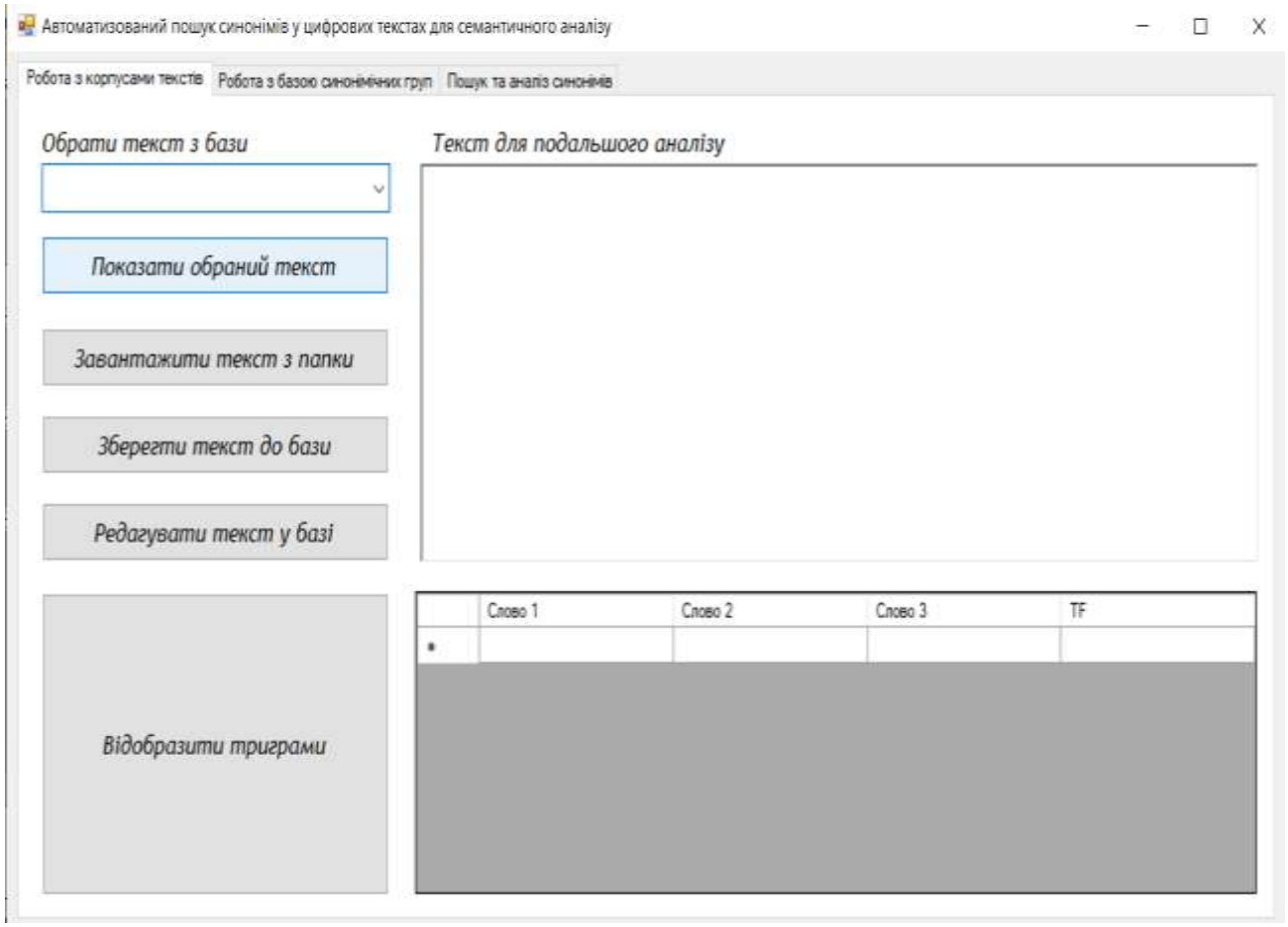


Рисунок 3.9 – Базовий екран програми

З поточної вкладки у користувача є можливість завантажити тексти з бази даних для аналізу (з випадаючого списку «Обрати текст з бази»), або завантажити текст з папки комп’ютера. Якщо користувач захоче відобразити текст з випадаючого переліку, потрібно обрати бажаний текст та натиснути кнопку «Показати обраний текст», після чого обраний текст відобразиться у полі «Текст для подальшого аналізу» (рисунок 3.10).

Якщо ж користувачу потрібно завантажити текст для аналізу з комп’ютера, потрібно натиснути кнопку «Завантажити текст з папки», після чого обрати шлях до потрібного тексту для аналізу та підтвердити вибір файлу, натиснувши кнопку «Відкрити» (рисунок 3.11).

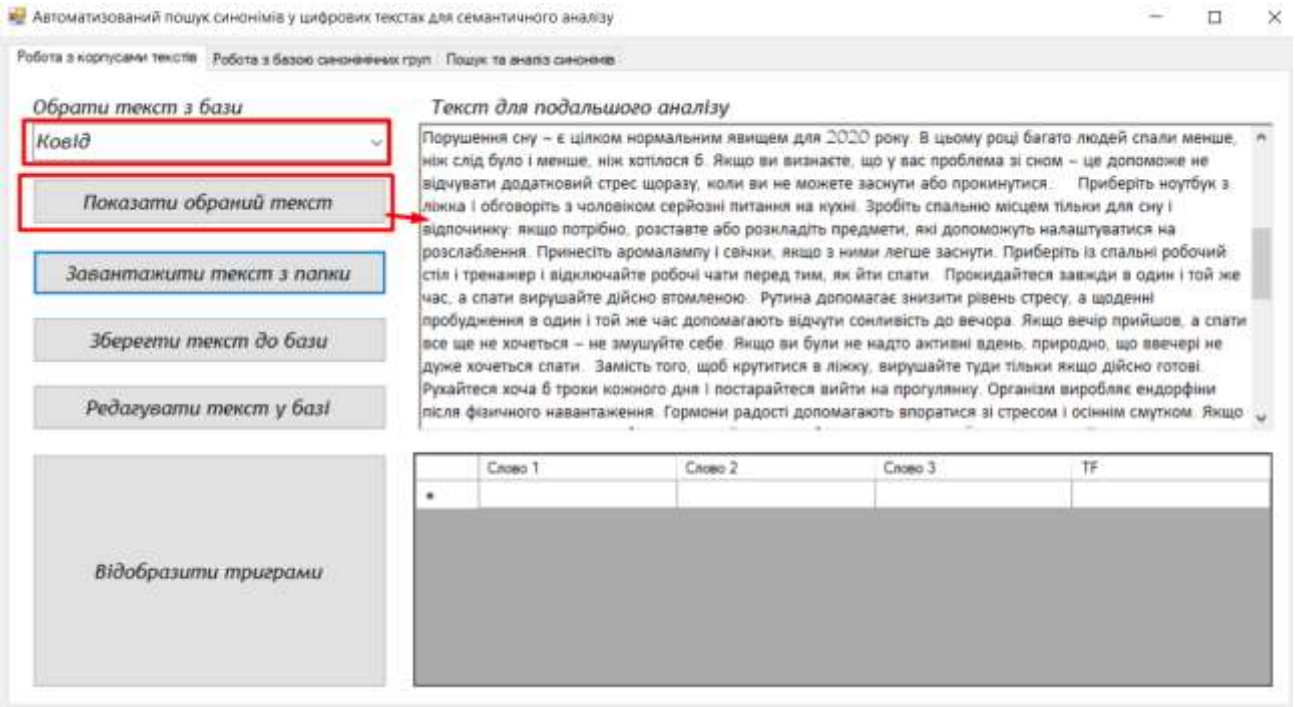


Рисунок 3.10 – Відображення тексту з бази даних

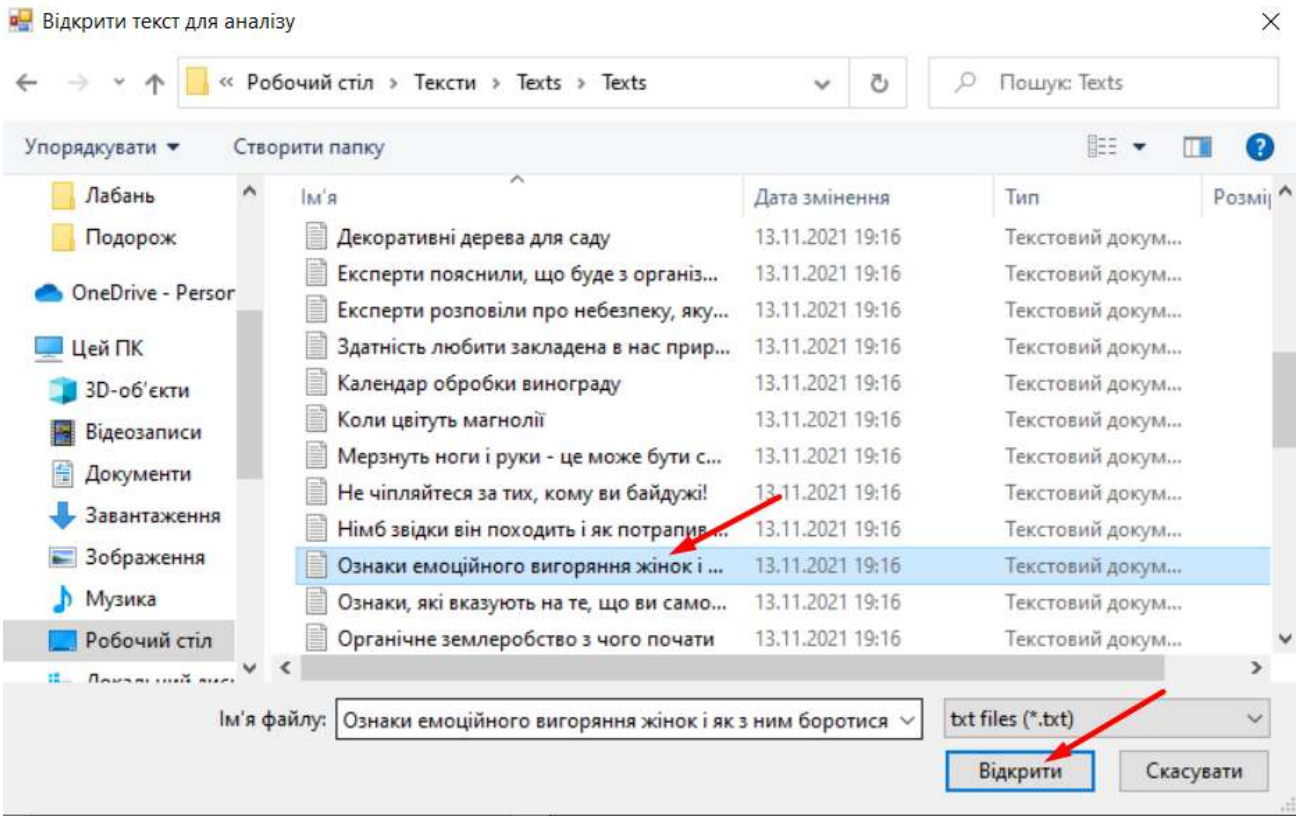


Рисунок 3.11 – Відкриття тексту з папки для аналізу

Обраний текст буде відображено у полі «Текст для подальшого аналізу». Також на першій вкладці у користувача є можливість зберегти завантажений

текст до бази даних , для цього потрібно натиснути кнопку «Зберегти текст до бази». Після чого користувач побачить повідомлення, що текст було збережено (рисунок 3.12).

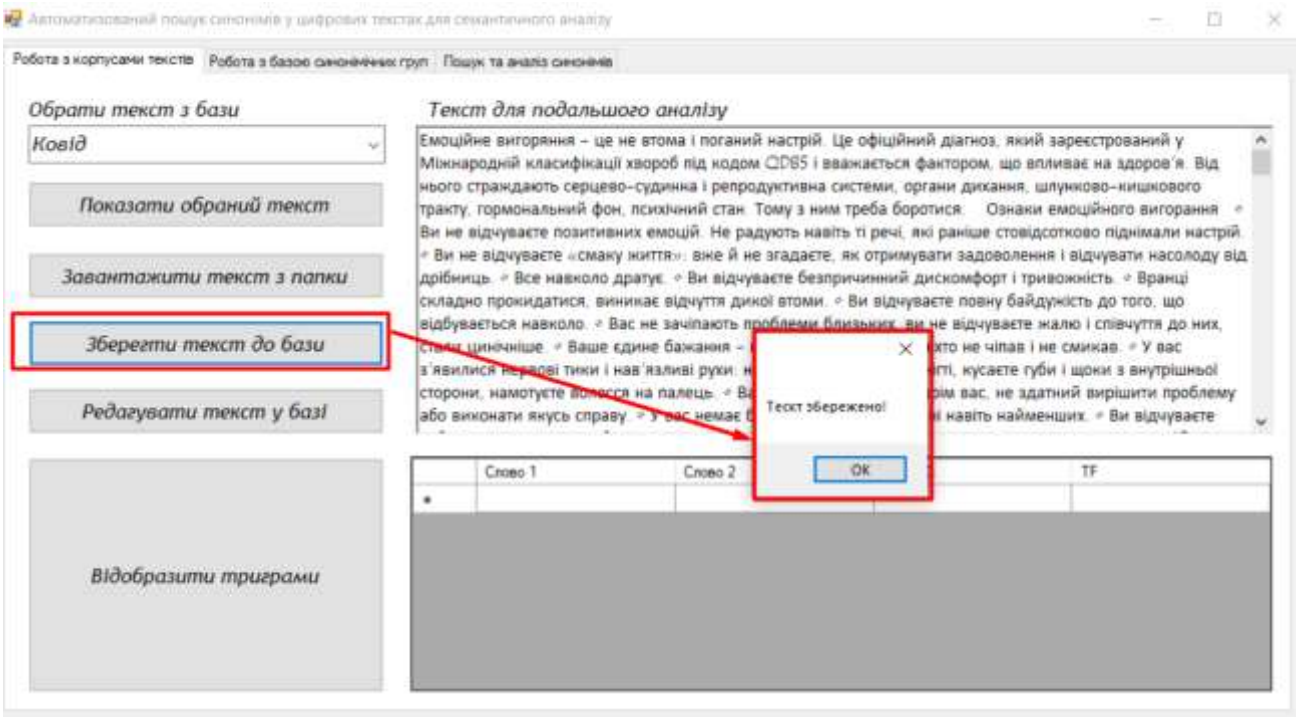


Рисунок 3.12 – Збереження завантаженого тексту до бази існуючих текстів

Аналогічно можна внести зміни у вже існуючий текст. Для внесення змін користувачем потрібно натиснути кнопку «Редагувати текст у базі». Також можна переглянути триграми обраного для аналізу тексту. Для цього потрібно натиснути на кнопку «Відобразити триграми». Результат буде як на рисунку 3.13. Окрім триграм у таблиці відображено частоту зустрічання для кожної триграми тексту.

На цьому функціональність першої вкладки розглянуто і можна перейти до другої вкладки програми – «Робота з базою синонімічних груп». При переході на другу вкладку завантажений текст зберігається та одразу відображаються його триграми (рисунок 3.14)

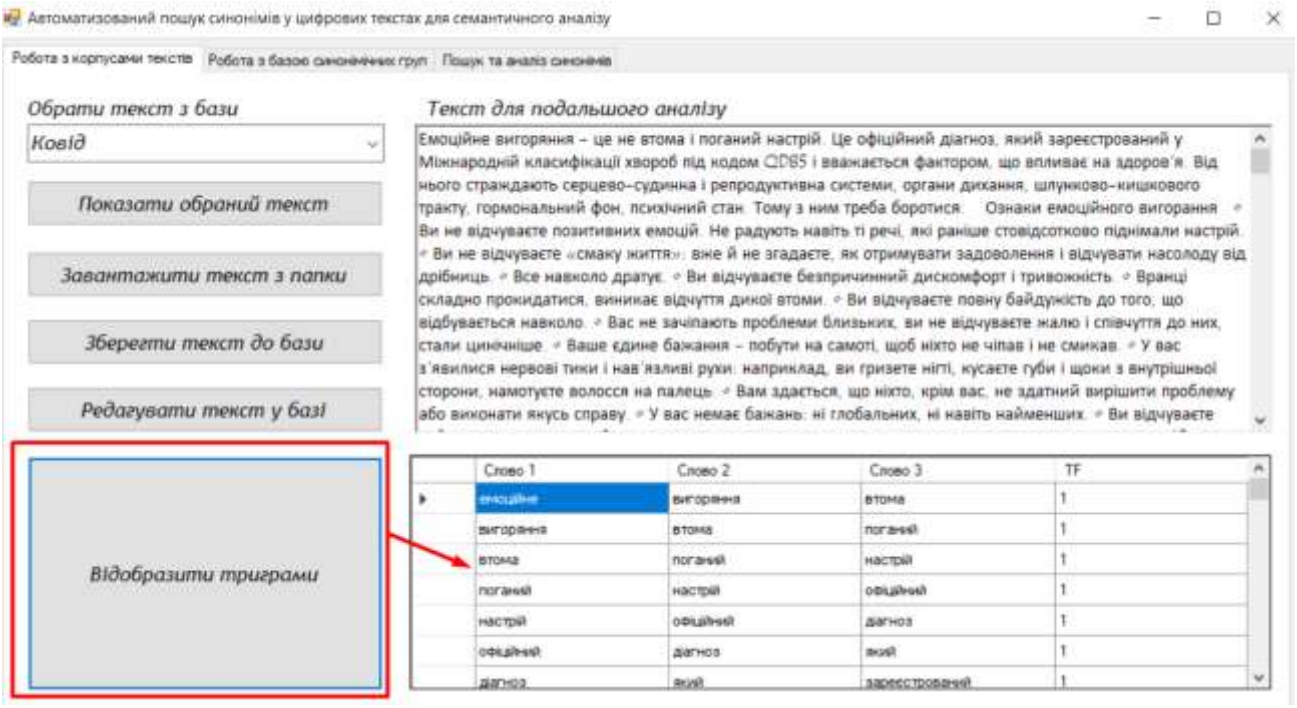


Рисунок 3.13 – Відображення триграм обраного тексту

Рисунок 3.13 – Вкладка «Робота з базою синонімічних груп»

Для даної вкладки доступно перетворення тексту у словник унікальних слів. Для цього користувачу потрібно натиснути кнопку «Показати текст як словник». Після чого текст відобразиться у полі «Словник» (рисунок 3.14).

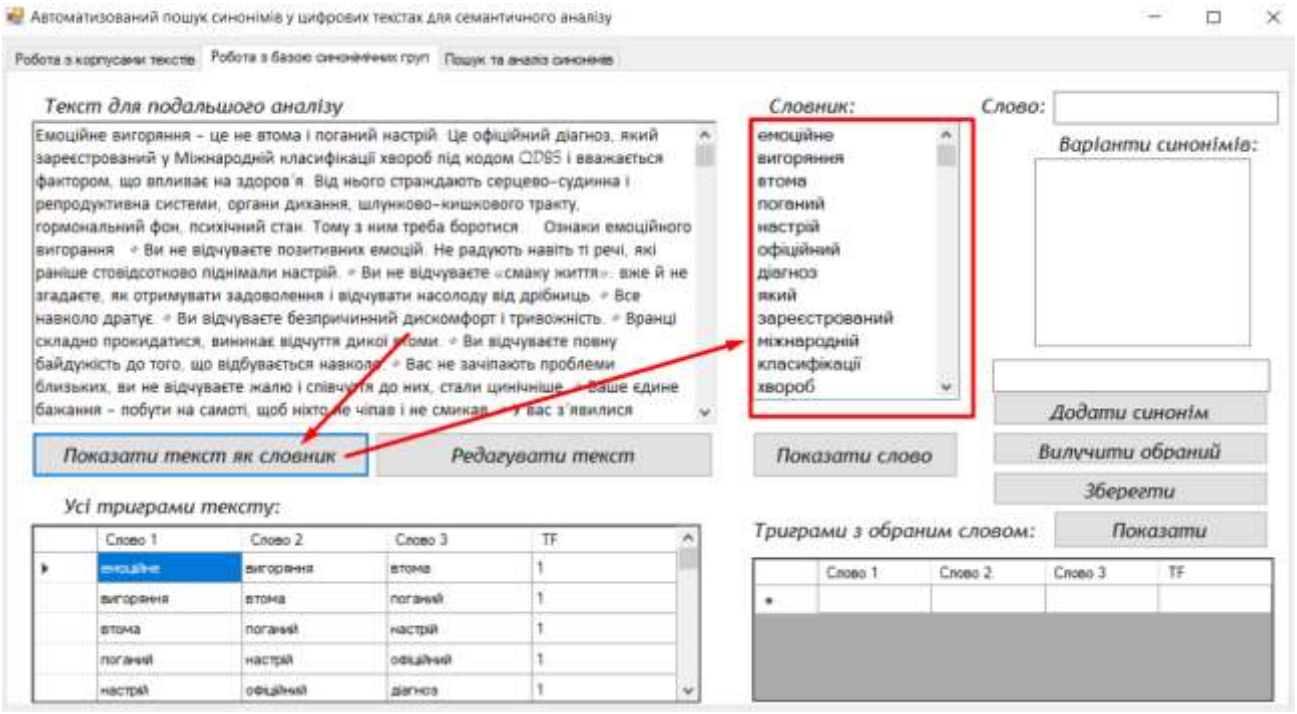


Рисунок 3.14 – Вкладка «Робота з базою синонімічних груп»

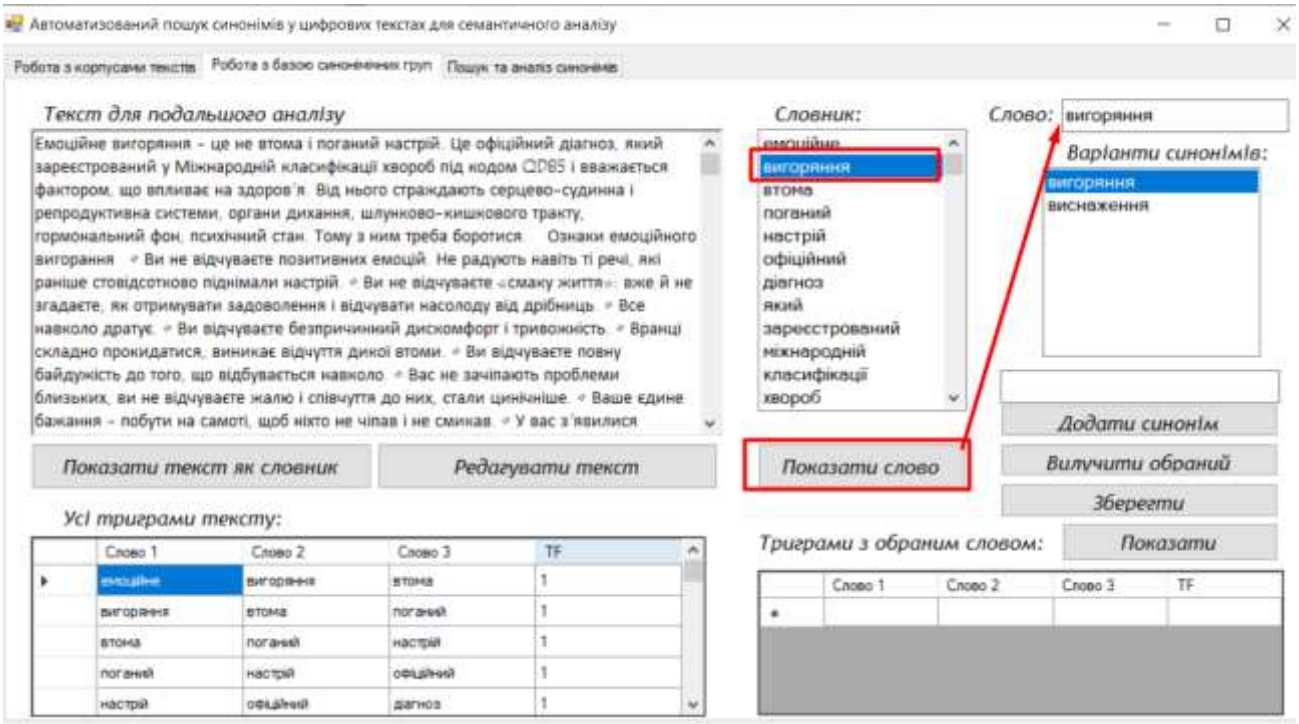


Рисунок 3.15 – Відображення синонімів

Також є можливість редагувати текст, для цього текст редагується у полі «Текст для подальшого аналізу», та натиснути кнопку «Редагувати текст». Текст буде відредаговано та перебудується таблиця триграм. Також є можливість працювати з окремими словами для відображення варіантів синонімів. Для цього

потрібно обрати слово зі словника яке потрібно проаналізувати та натиснути кнопку «Показати слово». При цьому обране слово буде відображено у текстовому полі «Слово», а також буде виведено автоматизований перелік синонімів (рисунок 3.15).

Окрім вищеописаного функціонування системи є можливість додати синонім «вручну». Для цього у текстове поле потрібно вписати синонім та натиснути кнопку «Додати синонім» (рисунок 3.16).

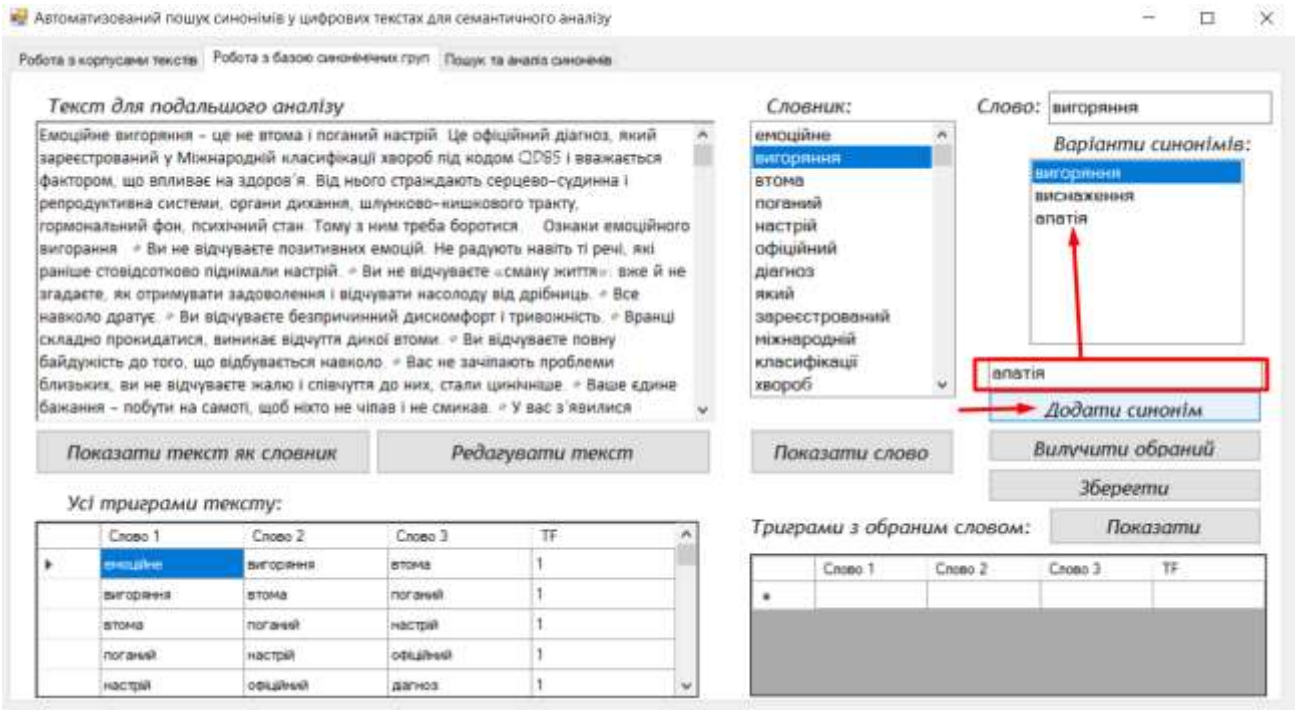


Рисунок 3.16 – Додавання синоніма

Як видно з рисунка 3.16 – слово додано до переліку синонімів. Також можна вилучити певне слово, обравши його у списку та натиснувш кнопку «Вилучити обраний». При натисненні кнопки «Зберегти» відбудеться збереження синонімічного ряду у базі даних. При натисненні на кнопку «Показати» будуть відображені усі триграми з обраним словом (рисунок 3.17).

Остання вкладка – «Пошук та аналіз синонімів». На цій вкладці користувач завантажує тестовий текст для аналізу, що можна зробити як «Вручну», так і використавши функцію завантаження тестового тексту з файлової системи.

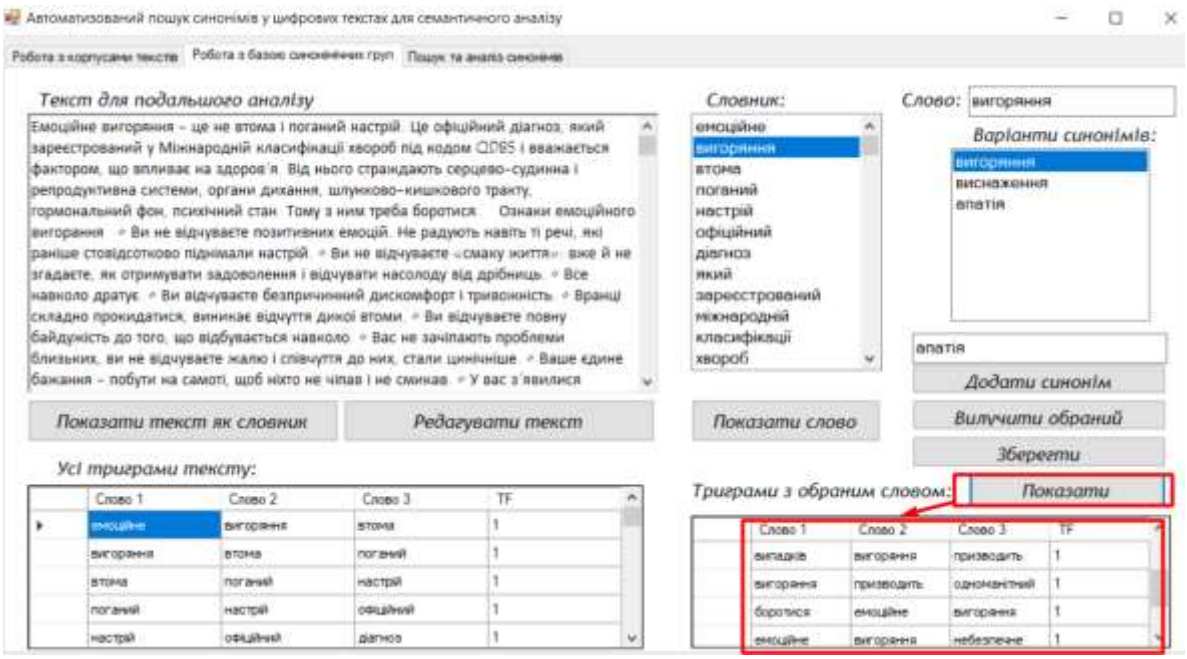


Рисунок 3.17 – Відображення триграм з обраним словом

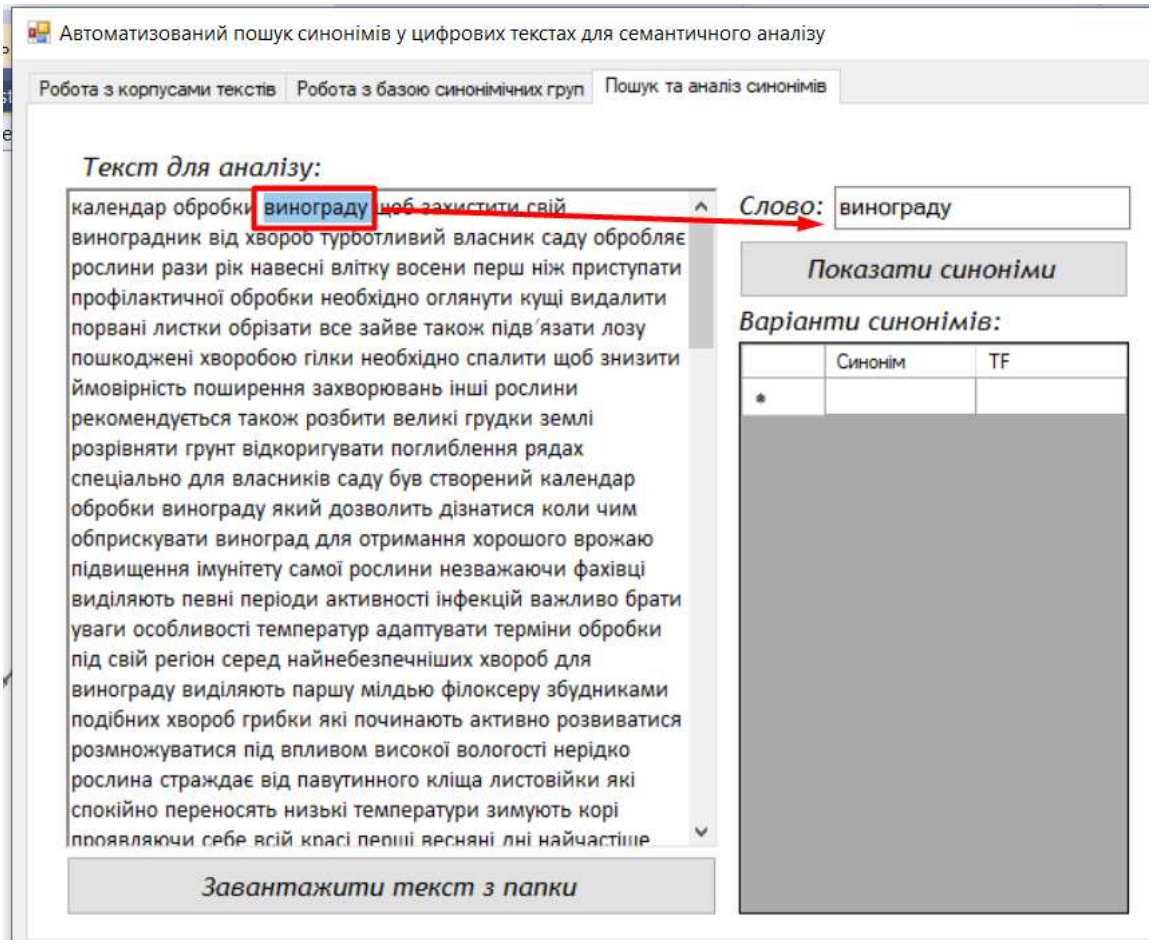


Рисунок 3.18 – Відображення обраного користувачем слова з тексту

Після завантаження тексту користувач мишкою виділяє слово, яке потрібно проаналізувати, обране слово автоматично буде відображено у відповідному текстовому полі (рисунок 3.18). Також можна проглянути усі триграми, для перегляду яких потрібно натиснути кнопку «Показати триграми» (рисунок 3.19).

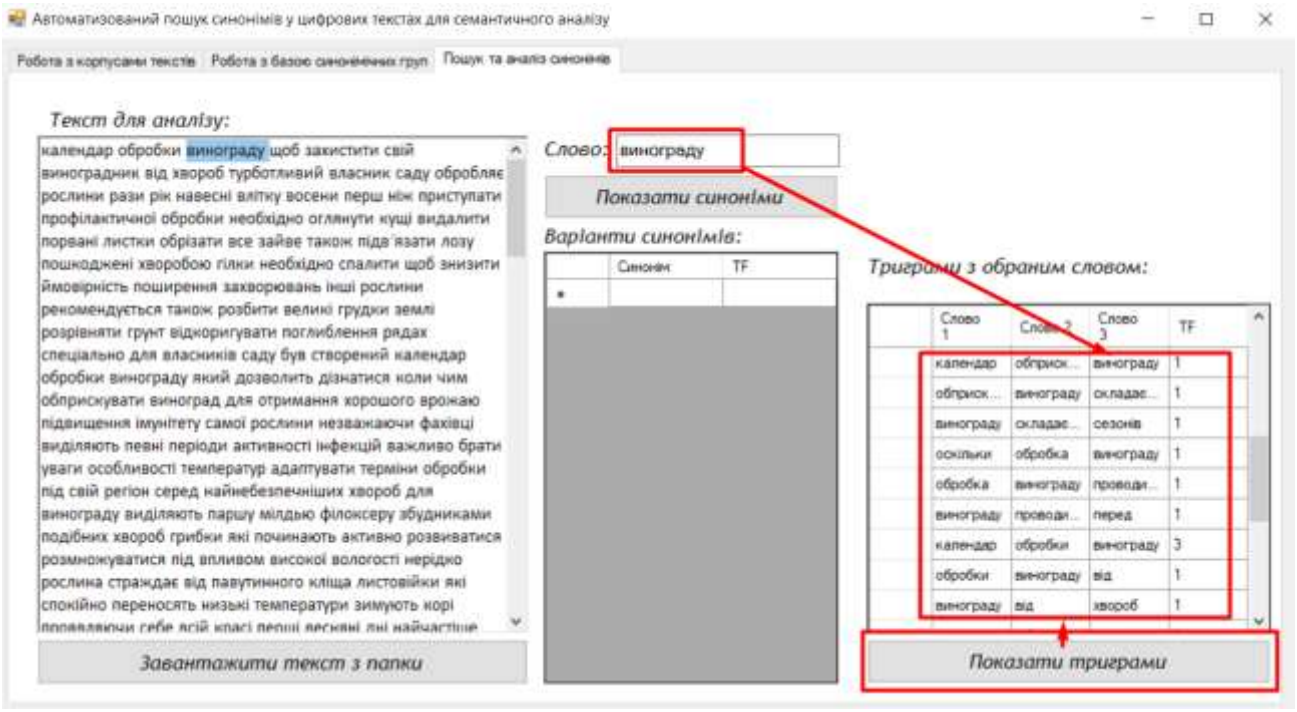


Рисунок 3.19 – Відображення триграм із обраним з тексту словом

Також є функціонал відображення переліку синонімів для обраного слова. Для цього потрібно натиснути на кнопку «Показати синоніми», після чого, перелік синонімів буде виведений у таблиці варіантів синонімів з частотою їх зустрічання (рисунок 3.20).

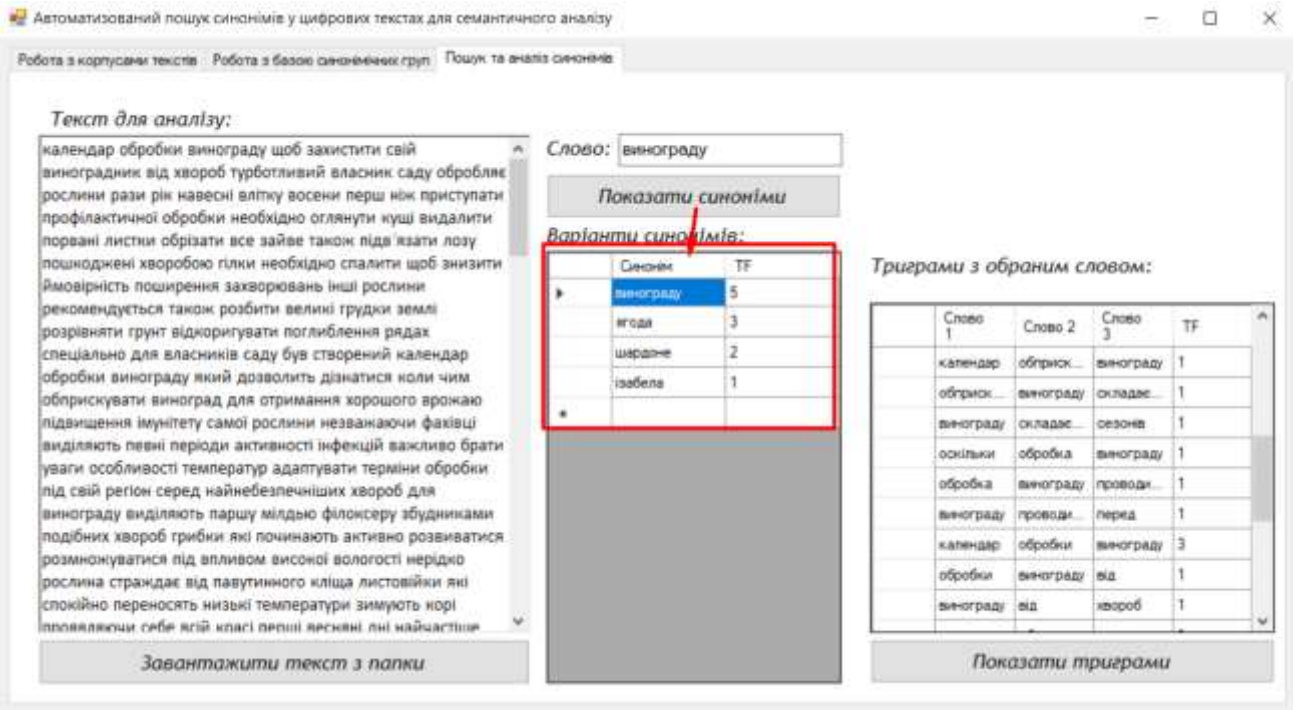


Рисунок 3.20 – Відображення синонімів обраного слова

Таким чином, створену програмну реалізацію методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.

3.4 Вимоги до розгортання інформаційної системи

Для забезпечення коректної роботи методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу рекомендовано нижченаведені технічні засоби.

Мінімальні вимоги до апаратних засобів:

– Процесор: AMD Atlon 640 DualCore.

– RAM: 4 Gb.

– Вільний дисковий простір: 64 Gb.

Рекомендовані вимоги до апаратних засобів:

– Процесор: Intel Core I5;

– RAM: 8 Gb;

– Вільний дисковий простір: 128 Gb.

Вимоги до програмних засобів:

– ОС: Windows 7/10.
– СКБД MS SQL Server (версія від 2012 року).
– .NET Framework 4.5.

Висновки

Мета кваліфікаційної роботи бакалавра була досягнута, а саме розроблено метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та розроблена відповідна програмна реалізація згідно запропонованого методу. Для досягнення мети були вирішені такі задачі:

1. Проведено аналіз предметної області, у рамках якого розглянуті існуючі методи для пошуку ключових слів та синонімів, а також існуючі програмі реалізації.
2. Розроблено метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.
3. Розроблено відповідну ІС автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу, зокрема структуру бази даних для даної предметної області.
4. Обрано відповідні засоби розробки для реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.
5. Створено відповідну програмну реалізацію згідно вищеописаних пунктів.
6. Проведено тестування створеного застосунку різними методами тестування, а для зручності користування створено інструкцію користувача.

Розроблена відповідно до методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу інформаційна система виконує функції роботи з навчальним корпусом текстів, з базою синонімічних груп та підбору синонімів до слів у тестовому тексті, зокрема наступні:

- Додавання навчального корпусу текстів для обробки.
- Здійснення базової обробка навчального корпусу текстів.
- Створення за дослідним текстом масиву слів.
- Створення за масивом слів дослідного тексту масиву оригінальних слів.
- Обрахунок оцінки TF для кожного слова з масиву оригінальних слів.

– Створення за масивом слів бази триграм.

– Обрахунок оцінки TF для кожної триграми з бази триграм.

– Формування переліку оригінальних слів з масиву оригінальних слів та значень їх оцінки TF.

– Пошук і відображення триграм, що містять обране слово на центральній позиції.

– Фіксування пар початкового і кінцевого слів триграм, що містять обране слово на центральній позиції.

– Пошук і відображення триграм, що містять ідентичні пари початкового і кінцевого слів, але інші слова на центральній позиції.

– Визначення множини інших слів з одержаних триграм як множини потенційних синонімів.

– Пошук і відображення триграм, що містять обране слово з множини потенційних синонімів на центральній позиції.

– Видалення користувачем обраних слів із множини потенційних синонімів.

– Додавання користувачем слів із масиву оригінальних слів до множини потенційних синонімів.

– Збереження результуючої множини слів як окремої множини синонімів – синонімічної групи.

– Додавання тестового тексту для обробки

– Створення за тестовим текстом масиву слів.

– Вибір користувачем робочого слова із масиву слів тестового тексту.

– Відображення множини слів синонімічної групи робочого слова.

– Вибір користувачем синоніма із множини слів синонімічної групи робочого слова.

– Відображення фрагментів тексту, що містять обраний синонім.

Перелік посилань

1. Textelit. Якісний текст: 5 основних критеріїв. URL: https://textelit.com.ua/top-5-kriteriev/#KO_1

2. Wikipedia. Культура мови. URL: https://uk.wikipedia.org/wiki/Культура_мови

3. Wikipedia. Літературна мова. URL: https://uk.wikipedia.org/wiki/Літературна_мова

4. Wikipedia. Стиль мовлення. URL: https://uk.wikipedia.org/wiki/Стиль_мовлення

5. Освіта.ua. Стилі літературної української мови як різновиди мови. URL: https://osvita.ua/vnz/reports/dilovodstvo/24233/

6. Wikipedia. Контент аналіз. URL: https://uk.wikipedia.org/wiki/Контент_аналіз

7. iGroup Ukraine. Нудота сторінки.URL: https://igroup.com.ua/seo-articles/nudota/

8. Savelink. Нудота тексту: що це і якою повинна бути? URL: http://savelink.org.ua/nudota-tekstu-shho-tse-i-yakoyu-povinna-buti/

9. Wikipedia. Синонім. URL: https://uk.wikipedia.org/wiki/Синонім

10. Тренажер з правопису української мови. Синоніми та антоніми. URL: https://webpen.com.ua/pages/vocabulary/synonyms.html

11. Wikipedia. Ключове слово. URL: https://uk.wikipedia.org/wiki/Ключове_слово

12. Wikipedia. Словосполучення. URL: https://uk.wikipedia.org/wiki/Словосполучення

13. Wikipedia. Термін. URL: https://uk.wikipedia.org/wiki/Термін

14. Wikipedia. Frequency analysis. URL: https://en.wikipedia.org/wiki/Frequency_analysis

15. Visitor Analytics. Частота термінів, зворотна частоты документів (TF-IDF). URL: https://www.visitor-analytics.io/ru/glossary/t/term-frequency-inverse-document-frequency-tf-idf/

16. Analytics India Magazine. Guide to NLP’s Textrank Algorithm. URL: https://analyticsindiamag.com/guide-to-nlps-textrank-algorithm/

17. Wikipedia. Корпус текстів. URL: https://uk.wikipedia.org/wiki/Корпус_текстів

18. ГРАК. Генеральний регіонально анотований корпус української мови (ГРАК). URL: http://uacorpus.org/Kyiv/ua

19. MOVA.info. Корпус української мови URL: http://www.mova.info/corpus.aspx

20. Електронна бібліотека Чтиво. Корпус української мови. URL: http://korpus.org.ua/

21. Deutscher Wortschatz. Ukrainisches Gemischt-Korpus. URL: https://corpora.uni-leipzig.de/de?corpusId=ukr_mixed_2014

22. Wikipedia. N-грама. URL: https://uk.wikipedia.org/wiki/N-грама

23. Kavita Ganesan, PhD. What are N-Grams? URL: https://kavita-ganesan.com/what-are-n-grams/#.YpzYk6jP1PY

24. Текстовід. Синонімайзер. URL: https://textovod.com/synonymizer

25. Smodin. Автоматично переписувати повний текст. URL: https://smodin.io/uk

26. Synonyma. Синонімізація тексту. URL: http://synonyma.ru/tools/synonymize/

27. Synonimy. Словник синонімів. URL: https://synonimy.info/

28. Синоніми.укр. Словник синонімів онлайн, підбір синонімів до слів. URL: https://xn--h1aaldafs6o.xn--j1amh/

29. Hokuapps. Benefits of Custom Mobile Application Development for Your Enterprises. URL: https://www.hokuapps.com/blogs/benefits-custom-mobile-application-development-enterprises/

30. Vwo. Mobile App Or Website? 10 Reasons Why Apps Are Better. URL: https://vwo.com/blog/10-reasons-mobile-apps-are-better

31. E-startupindia. Advantages & Disadvantages of website. URL: https://www.e-startupindia.com/learn/advantages-disadvantages-of-website/

32. VinitySoft. Advantages of Desktop Over Web-Based Applications. URL: https://www.vinitysoft.com/2015/05/advantages-of-desktop-over-web-based-applications/

33. Nimapinfotech. Advantages of java and disadvantages of java. URL: https://nimapinfotech.com/blog/advantages-and-disadvantages-of-java

34. ACTE. What is .Net FrameWork? Uses and its Benefits | Everything You Need to Know. URL: https://www.acte.in/what-is-net-framework-uses-and-its-benefits-article

35. ScoutAPM. .NET Core vs. .NET Framework: Side-by-Side Comparison. URL: https://scoutapm.com/blog/net-core-vs-net-framework

36. Wikipedia. .NET_Framework. URL: https://uk.wikipedia.org/wiki/.NET_Framework

37. Metanit. Мова C# і платформа .NET. URL: https://metanit.com/sharp/tutorial/1.1.php

38. CodeXoxo. What Is C# Language, Advantages & Features Of C# Language. URL: https://codexoxo.com/advantages-c-sharp-language/

39. The Register. Microsoft's do-it-all IDE Visual Studio 2022 came out late last year. How good is it really? URL: https://www.theregister.com/2022/01/25/visual_studio_2022/

40. Wikipedia. Database. URL: https://en.wikipedia.org/wiki/Database#Database_management_system

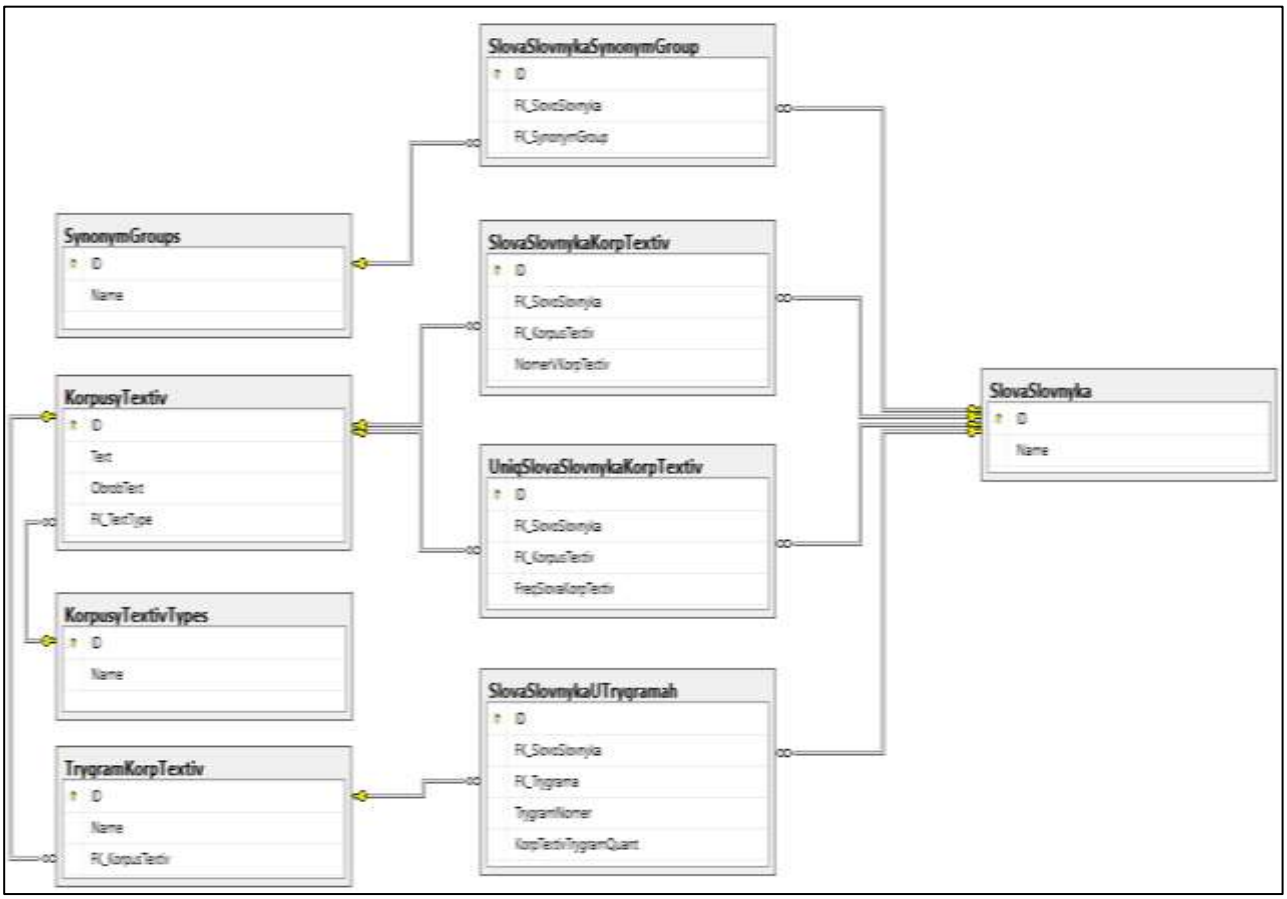
41. Wikipedia. Microsoft SQL Server. URL: https://en.wikipedia.org/wiki/Microsoft_SQL_Server

42. MyDiv. Microsoft SQL Server. URL: https://soft.mydiv.net/win/download-Microsoft-SQL-Server.html

ДОДАТКИ

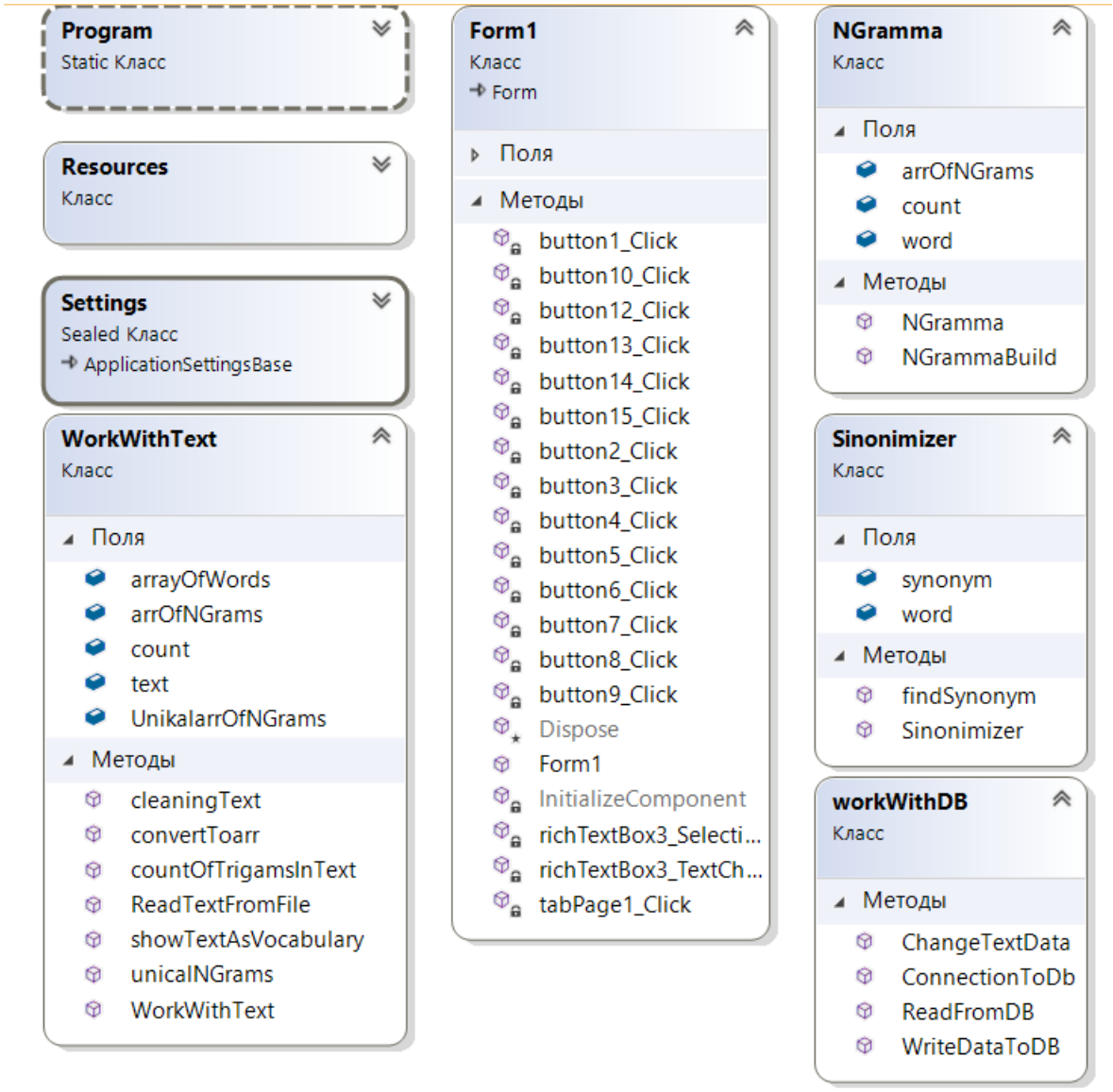
Додаток А

Структура бази даних системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу



Додаток Б

Розгорнута структура класів системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу



Додаток В

Презентаційний матеріал

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

МЕТОД АВТОМАТИЗОВАНОГО ПОШУКУ СИНОНІМІВ У ЦИФРОВИХ ТЕКСТАХ ДЛЯ СЕМАНТИЧНОГО АНАЛІЗУ

Виконав:
студент 4 курсу, групи КН-18-1
Лабань Олег Олегович

Керівник:
к.ф.-м.н., доцент кафедри КН
Міхалевський Віталій Цезарійович

Актуальність

Перенасичення інформаційного простору неякісним контентом знижує рівень зацікавленості читача в певних ресурсах, тому важливо, щоб текст, який публікується на загал, відповідав ряду критеріїв високоякісного тексту, а саме грамотності, унікальності, лаконічності тощо. Уміння створювати якісний текст потребує неабияких зусиль від автора, який його створює, адже сучасний читач має доступ до різних джерел інформації, тому зацікавленість автора в тому, що цей читач залишився з ним.

Одним із способів зробити текст унікальним та не нудним для читача – це використання синонімів для ключових термінів написаного тексту. Важливо здійснювати підбір синонімів відповідно до стилю тексту.

Інформаційні технології надають різноманітні засоби для автоматизації роботи з текстами – виділення семантичного ядра (ключових слів, словосполучень та термінів), структуризація тексту тощо. Тому є перспективною розробка методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу. Це надасть можливість проводити пошук синонімів слів у тексті з подальшою можливістю семантичного аналізу даного тексту.

Мета і задачі роботи

Метою кваліфікаційної роботи бакалавра є розробка методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та відповідної програмної реалізації розробленого методу.

Для досягнення мети потрібно вирішити такі задачі:

- Провести аналіз предметної області, у рамках якого оглянути існуючі методи для пошуку ключових слів та синонімів та існуючі програмі реалізації.
- Розробити метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.
- Розробити відповідну ІС автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу, зокрема структуру бази даних для даної предметної області.
- Обрати засоби розробки для реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.
- Створити відповідну програмну реалізацію згідно вищеописаних пунктів.
- Провести тестування створеного за стосунку різними методами тестування та для зручності користування створити інструкцію користувача.

Схема методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу

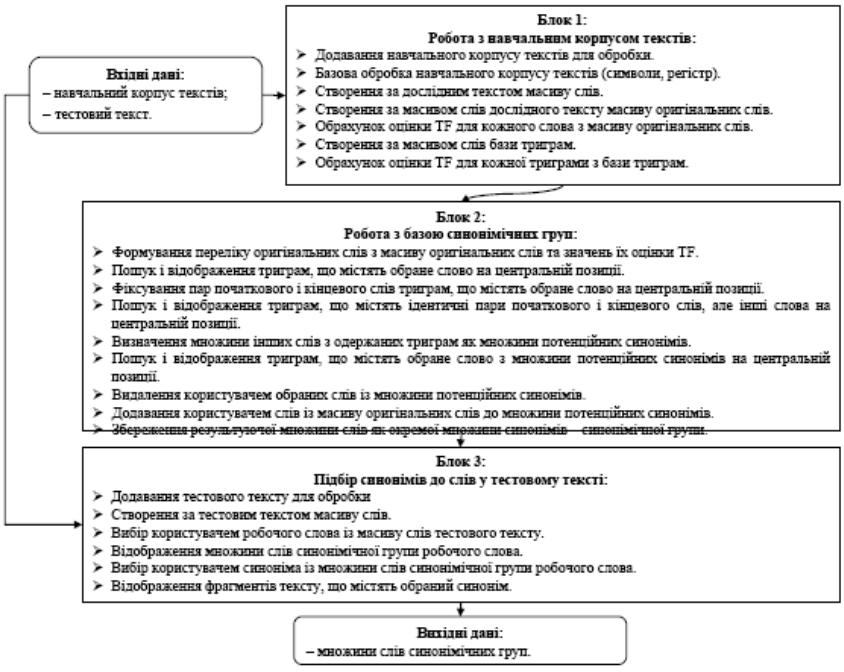
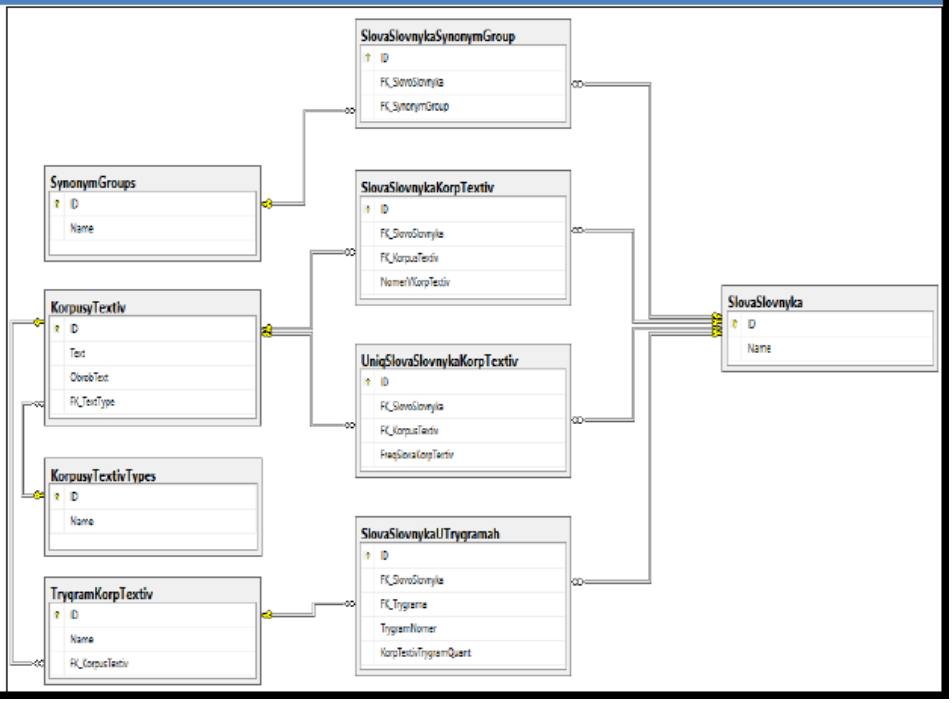
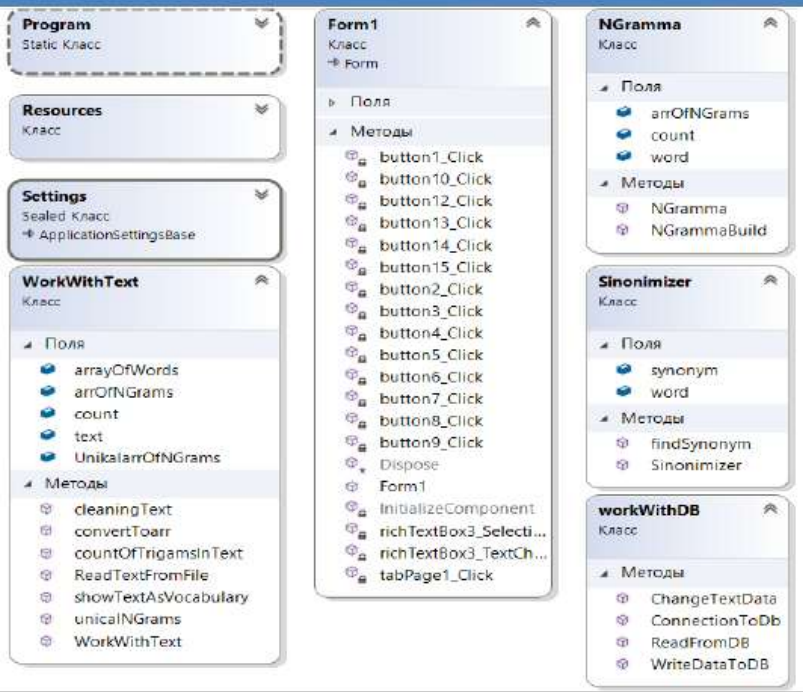


Схема системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу

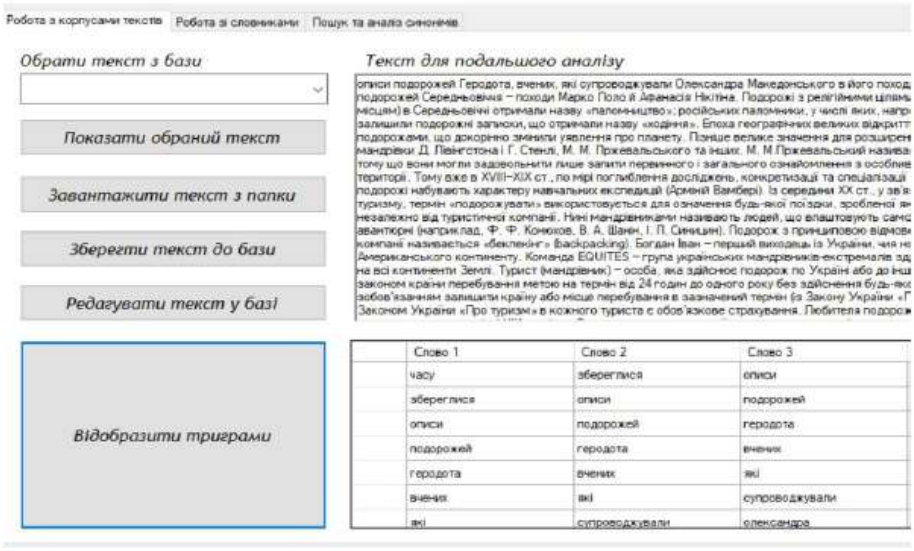
Схема бази даних



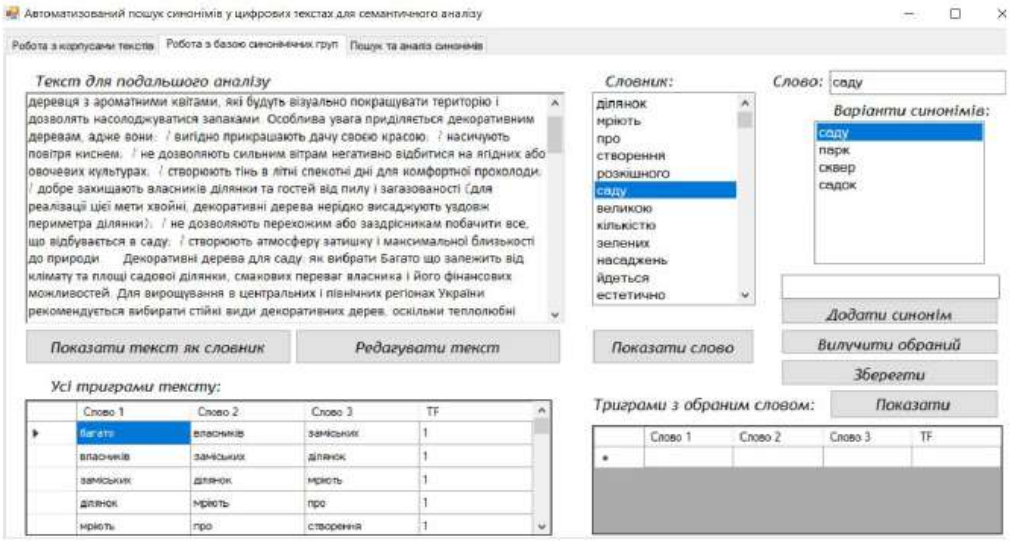
Діаграма класів програмної реалізації методу автоматизованого пошуку синонімів у цифрових текстах



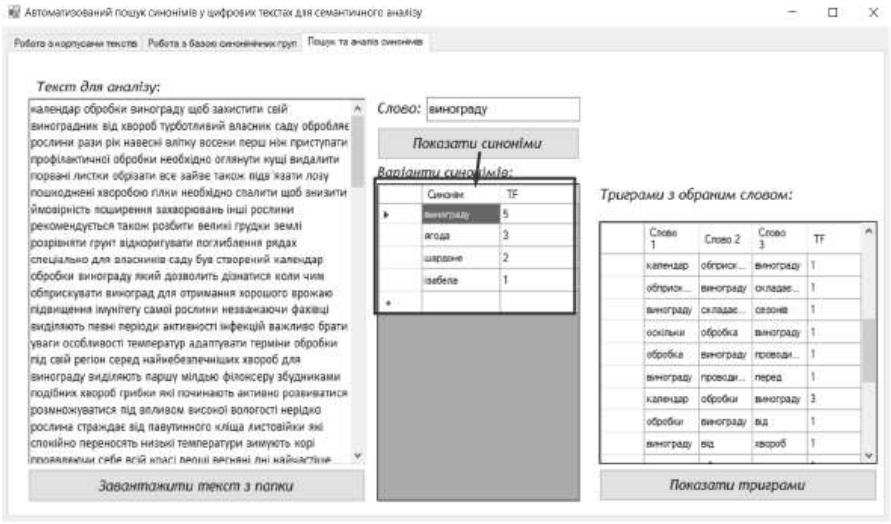
Відображення триграм тексту



Відображення триграм з обраним словом



Відображення синонімів обраного слова



Висновки

Мета кваліфікаційної роботи бакалавра була досягнута, а саме розроблено метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та розроблена відповідна програмна реалізація згідно запропонованого методу.

Для досягнення мети були вирішені такі задачі:

- Проведено аналіз предметної області, у рамках якого розглянуті існуючі методи для пошуку ключових слів та синонімів, а також існуючі програмі реалізації.
- Розроблено метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.
- Розроблено відповідну ІС автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу, зокрема структуру бази даних для даної предметної області.
- Обрано відповідні засоби розробки для реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.
- Створено відповідну програмну реалізацію згідно вищеописаних пунктів.
- Проведено тестування створеного застосунку різними методами тестування, а для зручності користування створено інструкцію користувача.

Висновки

Розроблена відповідно до методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу інформаційна система виконує функції роботи з навчальним корпусом текстів, з базою синонімічних груп та підбору синонімів до слів у тестовому тексті, зокрема наступні:

- ✓ Додавання навчального корпусу текстів для обробки.
- ✓ Здійснення базової обробка навчального корпусу текстів.
- ✓ Створення за дослідним текстом масиву слів.
- ✓ Створення за масивом слів дослідного тексту масиву оригінальних слів.
- ✓ Обрахунок оцінки TF для кожного слова з масиву оригінальних слів.
- ✓ Створення за масивом слів бази триграм.
- ✓ Обрахунок оцінки TF для кожної триграми з бази триграм.
- ✓ Формування переліку оригінальних слів з масиву оригінальних слів та значень їх оцінки TF.
- ✓ Пошук і відображення триграм, що містять обране слово на центральній позиції.
- ✓ Фіксування пар початкового і кінцевого слів триграм, що містять обране слово на центральній позиції.
- ✓ Пошук і відображення триграм, що містять ідентичні пари початкового і кінцевого слів, але інші слова на центральній позиції.
- ✓ Визначення множини інших слів з одержаних триграм як множини потенційних синонімів.
- ✓ Пошук і відображення триграм, що містять обране слово з множини потенційних синонімів на центральній позиції.
- ✓ Видалення користувачем обраних слів із множини потенційних синонімів.
- ✓ Додавання користувачем слів із масиву оригінальних слів до множини потенційних синонімів.
- ✓ Збереження результуючої множини слів як окремої множини синонімів – синонімічної групи.
- ✓ Додавання тестового тексту для обробки
- ✓ Створення за тестовим текстом масиву слів.
- ✓ Вибір користувачем робочого слова із масиву слів тестового тексту.
- ✓ Відображення множини слів синонімічної групи робочого слова.
- ✓ Вибір користувачем синоніма із множини слів синонімічної групи робочого слова.
- ✓ Відображення фрагментів тексту, що містять обраний синонім.

UNICHECK

Ім'я користувача:
Кафедра КН

ID перевірки:
1011548910

Дата перевірки:
12.06.2022 09:26:16 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
12.06.2022 09:55:10 EEST

ID користувача:
100005671

Назва документа: Лабань_ЗАПИСКА_short

Кількість сторінок: 65 Кількість слів: 9782 Кількість символів: 73584 Розмір файлу: 4.69 MB ID файлу: 1011420943

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

6.43%
Схожість

Найбільша схожість: 3.76% з джерелом з Бібліотеки (ID файлу: 1011420917)

1.7% Джерела з Інтернету	77	Сторінка 67
5.88% Джерела з Бібліотеки	135	Сторінка 67

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи	3
Підозріле форматування	22 сторінки

Sun Jun 12 08:36:44 EEST 2022, Петровський Сергій Степанович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 10.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 10%**

	Документ		Суммарное совпадение по Базе Данных	
ID: 105047 Название: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу Добавлено в БД: 2022-06-12 Авторы: О.О. Лабань Руководители: В.Ц. Міхалевський Консультанты: Опоненты:	Символы	Лексемы	Символы	Лексемы
	57762	832	7676 (13%)	95 (11%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы
105009	Название: ЗВІТ з професійної практики база практики SOA LABS Добавлено в БД: 2022-06-10 Авторы: Лабань О.О. Руководители: Скрипник Т.К. Консультанты: Опоненты:	5615 (10.0%)	93 (11.0%)

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНИХ НАУК
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

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

Назва: *Метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу*

Автор: *студент групи КН-18-1 Лабань Олег Олегович*

Спеціальність: *122 – Комп'ютерні науки*

Освітня програма: *освітньо-професійна*

Науковий керівник: *доцент кафедри КН Міхалевський Віталій Цезарійович*

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 О.О. Лабаня, є законними і не є плагіатом, оскільки:

1) за програмою Anti-Plagiarism виявлені 10% запозичень вказують на документ автора роботи та містять його ж Звіт з практики.

2) За програмою UNICHECK виявлені 6,43%, які є фрагментарними, не більше 3,76% на джерело – містять поширені конструкції, загальновідомі терміни та визначення.

3) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи.

Керівник роботи _____ *Віталій МІХАЛЕВСЬКИЙ*

Гарант ОП _____ *Олександр МАЗУРЕЦЬ*

Завідувач кафедри КН _____ *Олександр БАРМАК*

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МОН УКРАЇНИ

Кафедра комп'ютерних наук

ВІДГУК НАУКОВОГО КЕРІВНИКА

на кваліфікаційну роботу бакалавра

студента *гр. КН-18-1 Лабаня Олега Олеговича*

за темою *Метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу*

1. Актуальність теми

Мистецтво написання якісного текстового контенту є важливим завданням у будь-якій галузі. Якісно написаний текст важливий не тільки у художній літературі, а і для написання різних новинних статей, тексту, що призначений «продати» товар тощо. Повторення одних і тих же слів у тексті робить його нудним та не цікавим, що знижує зацікавленість читача в ресурсі, де розміщений цей текст. Для того, щоб текст був насичений, цікавий використовують прийом заміни ключових слів на відповідні синоніми. Для цього існують спеціальні словники синонімів. Відповідно, автоматизація процесів пошуку синонімів у цифрових текстах для семантичного аналізу є актуальною задачею комп'ютерних наук.

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

Поставлена у кваліфікаційній роботи бакалавра мета стосується розробки методів і технологій отримання, зберігання, обробки, передачі та використання інформації, інтелектуального аналізу даних і прийняття рішень, а саме розробки методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу. При цьому при вирішенні поставлених задач використовуються математичні моделі, методи та алгоритми розв'язання теоретичних і прикладних задач, що виникають при розробці інформаційних технологій. Тому результати виконання кваліфікаційної роботи бакалавра відповідають стандарту бакалавра спеціальності 122 – Комп'ютерні науки.

3. Професійні та особистісні якості бакалавра

При роботі над кваліфікаційною роботою бакалавра Лабань Олег Олегович проявив себе достатньо кваліфікованим фахівцем та дисциплінованим студентом, вчасно виконував поставлені етапи. Як в процесі написання пояснювальної записки, так і при розробці прикладного програмного забезпечення проявив достатні для одержання успішного результату компетентності.

4. Ступінь самостійності під час виконання кваліфікаційної роботи

Одержані в роботі результати є наслідком особистої діяльності студента, який самостійно виконував всі поставлені задачі.

5. Ступінь оволодіння методами дослідження

В роботі при розробці та прикладній реалізації методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу виявлено достатній ступінь оволодіння студентом необхідними інструментами та обладнанням, методами, методиками та технологіями предметної області комп'ютерних наук.

6. Повнота та якість розкриття теми роботи

Тема роботи в повній мірі обґрунтована й розкрита, проведено аналіз актуальності та відомих досліджень в межах обраної теми, поставлені завдання, які у роботі виконані, та розроблено програмне забезпечення для автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу.

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

Структура роботи та послідовність викладення логічні та відповідні поставленій меті. Викладення матеріалу грамотне та виявляє високий ступінь відповідності стилю.

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

Запропонований метод автоматизованого пошуку синонімів у цифрових текстах може мати практичне використання. Розроблена відповідна система призначена для SEO оптимізації та створення різноманітної кількості пошукових запитів, а також для копірайтерів для покращення унікальності текстів шляхом використання синонімів. Напрямками практичного використання розробленої системи є автоматизований пошук синонімів користувачами для вирішення проблем аналізу текстової інформації та розширення словникового запасу.

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

Враховуючи високий рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «задовільно».

Керівник _____ *к.ф.-м.н., доц. каф. КН Віталій МІХАЛЕВСЬКИЙ*

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МОН УКРАЇНИ

Кафедра комп'ютерних наук

РЕЦЕНЗІЯ

на кваліфікаційну роботу бакалавра

студента *гр. КН-18-1 Лабань Олег Олегович*

за темою: *Метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу*

1. Актуальність обраної теми

Одним із способів зробити текст унікальним та не нудним для читача – це використання синонімів для ключових термінів написаного тексту. Важливо здійснювати підбір синонімів відповідно до стилю тексту. Інформаційні технології надають різноманітні засоби для автоматизації роботи з текстами – виділення семантичного ядра (ключових слів, словосполучень та термінів), структуризація тексту тощо. Тому є перспективною розробка методу автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу, що є актуальною задачею комп'ютерних наук. Це надасть можливість проводити пошук синонімів слів у тексті з подальшою можливістю семантичного аналізу тексту.

2. Повнота розкриття мети та завдань дослідження

Кваліфікаційна робота бакалавра студента Лабаня О.О. виконана в повному обсязі, було проаналізовано предметну область, створено метод автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу та реалізовано програмний застосунок на базу вищезазначеного методу.

3. Зміст кожного розділу роботи

Перший розділ присвячений проведенню аналізу предметної області та визначенню основних параметрів для розв'язку поставленої задачі. Другий розділ присвячений проєктуванню функціональної структури інформаційної системи автоматизованого пошуку синонімів у цифрових текстах для семантичного аналізу. Третій розділ присвячений програмній реалізації спроєктованої функціональної структури інформаційної системи. Також сформовано основний висновок роботи.

4. Оцінка розробленої інформаційної системи, її практична цінність

Створений метод та програмний застосунок на його основі дозволяють користувачеві виконувати автоматизований пошук синонімів у цифрових текстах для семантичного аналізу.

5. Якість оформлення кваліфікаційної роботи бакалавра

Робота виконана на належному науково-методичному рівні та відповідає встановленим вимогам щодо оформлення такого роду праць.

6. Недоліки кваліфікаційної роботи бакалавра

Кваліфікаційна робота бакалавра містить кілька синонімічних варіантів назви створеного програмного продукту, варто було використати лише один; втім це не впливає на одержані результати.

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

Враховуючи рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка ***«добре»****.*

Рецензент Мартинюк В.В.