Тема : ***«Інформаційна технологія виділення багатослівних термінів з текстів природньою мовою»***

Шифр – ***«Programmer Dictionary»***

ЗМІСТ

ВСТУП

Словники предметних областей (тезауруси) широко використовуються при проектуванні програмних продуктів починаючи з етапу виявлення вимог і закінчуючи супроводом. На їх основі створюються бази даних, посадові інструкції та інші документи.

Оскільки складання словника предметної області «вручну» - надзвичайно трудомісткий процес, що вимагає участі кваліфікованих фахівців, то метою даної роботи є скорочення часу на побудову словника предметної області.

Основним етапом побудови словника є виділення термінів з текстів, що представляють певну предметну область. Найбільші проблеми цього етапу пов'язані з виділенням багатослівних термінів (БТ) [6]. У даній роботі пропонується метод автоматизованого виділення багатослівних термінів з текстів російською мовою [7,8].

Відомі рішення по автоматизації розробки словників предметної області, орієнтовані на англо-німецьку групу мов - система jalingo [2]. Система полегшує роботу експерта в предметній області, однак вимагає участі фахівця в лінгвістичному аналізі.

Існує ряд варіантів виділення термінів з тексту: виділення максимальних ланцюжків, що містять терміни [9], використання автоматичних запитів до пошукової машини [10], синтаксичний аналіз [11,12]. Аналіз відомих рішень показав, що розглянуті рішення дозволяють виділити загальновживані стійкі словосполучення, які задовольняють шаблонами або словосполучення, що використовуються в Інтернеті, однак проблема визначення термінів у вузькій конкретної предметної області залишається невирішеною.

Аналіз раніше виконаних досліджень показав, що проблема виділення БТ має ряд невирішених питань, пов'язаних з визначенням кількості слів,

складових БТ, формулюванням БТ, що містять однакові слова, великою витратою часу на процес виділення БТ [5].

На даний момент активно розвиваються інформаційні технології, а саме в галузі розробки програмного забезпечення. Майже для всіх сфер діяльності людини були розроблені та розроблюються програмні рішення, які полегшують роботу людини та роблять її менш трудомісткою. Розробка програмного продукту є складним процесом, який проходить у декілька етапів, кожен з них має специфічні складнощі, та потребує чималих затрат часу і роботи людини. Для вирішення цієї проблеми також були створені та створюються різноманітні продукти, які полегшують роботу команди розробки – це різноманітні програмні продукти, стратегії розробки, методи проектування систем, фреймворки і т.д.

Одним з таких рішень є розроблювана в рамках наукової роботи система виділення багатослівних термінів з текстів, яка пришвидшить процес створення словників предметної області на етапі аналізу вимог.

Завдання, які необхідно вирішити для досягнення мети наукової роботи:

1. Розробка математичної моделі для автоматизованого формування словника багатослівних термінів в тексті;
2. Розробка алгоритмів функціонування елементів системи та взаємодії між ними;
3. Програмна реалізація математичної моделі та алгоритмів, що розроблювалися.

У даній роботі представлені результати рішення вище перелічених завдань та наведено висновки щодо повноти досягнення поставленої цілі.

1 СЦЕНАРІЇЇ ВАРІАНТІВ ВИКОРИСТАННЯ

1.1 Визначення проблеми

Саме розробка програмного забезпечення є одною із провідних галузей людської діяльності, оскільки програмні продукти, які призначені для полегшення роботи людини вводяться в експлуатацію практично у всіх сферах життя людини: навчання, медицина, торгівля, бізнес, будівництво, всі види промисловостей і т.д.

Важливо відзначити, що і розробка програмних рішень є дуже трудомістким і складним процесом, що проходить в декілька етапів. Таких як отримання замовлення на розробку, аналіз вимог, планування розробки, проектування архітектури, написання коду, тестування, відладка програми, введення в експлуатацію. Всі вони виконуються послідовно, і до наступного можна перейти лише коли поточний буде пройдено. Наслідком прискорення виконання одного етапу розробки стане пришвидшення виконання проекту загалом. Це дасть змогу команді виконати проект в коротші строки або використати час, що «звільнився» на вирішення важливих задач, наприклад розширення функціоналу програми або більш детального тестування продукту.

Розроблювана в якості наукової роботи система буде використовуватись розробником та експертом предметної області саме на етапі аналізу вимог. Предметна область будь-якого підприємства або організації в достатній мірі описана текстами документів, які використовуються в діяльності організаційних структур. І оскільки словник предметної області (тезаурус) формується на основі текстів документів, він дає уявлення розробнику про предметну область та про діяльність організації.

Окрім використання СПО (словник предметної області) в розробці програмних продуктів існує ще багато завдань [8], пов'язаних з діяльністю будь-якої організації, для вирішення яких існування такого словника було б

дуже бажано. Це завдання, пов'язані зі створенням і розвитком інформаційних систем, підготовкою кадрів, створенням нової документації, чітким розподілом обов'язків між співробітниками.

1.2 Сценарії варіантів використання

Для того щоб зрозуміти, як повинен працювати кожен прецедент у програмі опишемо повний сценарій для кожного варіанту використання [4].

Опис сценарію варіанту використання **«Створити словник»**

Передумови: Користувач зайшов в систему.

Актори: Користувач.

Ініціатором початку сценарію є Користувач.

Гарантія успіху: збереження нового словника в системі.

Основний успішний сценарій:

1. Користувач обирає задачу «Пошук файлу». Система виводить форму для вибору текстового файлу.

2. Користувач обирає задачу відкриття файлу. Система відображає вміст файлу.

3. Користувач обирає задачу «Аналіз». Система аналізує текст та формує словник, Система відображає словник.

Альтернативні сценарії:

3.1 Користувач обрав невірний формат файлу: система виводить повідомлення «Некоректний формат файлу!».

Опис сценарію варіанту використання **«Переглянути словник»**

Передумови: Користувач зайшов в систему, в системі збережений хоча б один словник.

Актори: Користувач.

Ініціатором початку сценарію є Користувач.

Гарантія успіху: перегляд словника термінів.

Основний успішний сценарій:

1. Користувач обирає задачу «Відкрити словник». Система відкриває вікно вибору файлів.

2. Користувач обирає словник. Система відображає словник.

Альтернативні сценарії:

3.1 Обраний невірний формат файлу, система виводить повідомлення «Невірний формат файлу»

Опис сценарію варіанту використання «Зберегти словник»

Передумови: Користувач зайшов в систему, виконано прецедент «Переглянути словник» або «Створити словник».

Актори: Користувач.

Ініціатором початку сценарію є Користувач.

Гарантія успіху: перегляд словника термінів.

Основний успішний сценарій:

1. Користувач обирає меню «Зберегти». Система відкриває вікно збереження файлів.

2. Користувач обирає місце, куди хоче зберегти файл.

3. Користувач підтверджує. Система зберігає словник.

Опис сценарію варіанту використання «**Додавання терміну**»

Передумови: Користувач зайшов в систему, виконано прецедент «Створення словника» або «Перегляд словника».

Актори: Користувач.

Ініціатором початку сценарію є Користувач.

Гарантія успіху: термін додано в словник.

Основний успішний сценарій:

1. Користувач обирає пункт меню «Створити».
2. Система виводить форму для додавання терміну.
3. Користувач заповнює поля, зберігає термін. Система підтверджує. Система відображає словник з новим терміном.

Альтернативні сценарії:

3.1 Не заповнені поля форми: система виводить повідомлення «Заповніть поля!».

Опис сценарію варіанту використання **«Видалення терміну»**

Передумови: Користувач зайшов в систему, виконано прецедент «Переглянути словник» або «Створити словник».

Актори: Користувач.

Ініціатором початку сценарію є Користувач.

Гарантія успіху: видалення терміну зі словника.

Основний успішний сценарій:

1. Користувач обирає в словнику термін, який хоче видалити.

2. Користувач обирає пункт «Видалити». Система виводить форму підтвердження.

3. Користувач підтверджує. Система видаляє термін.

Опис сценарію варіанту використання **«Редагування терміну»**

Передумови: Користувач зайшов в систему, виконано прецедент «Переглянути словник» або «Створити словник».

Актори: Користувач.

Ініціатором початку сценарію є Користувач.

Гарантія успіху: видалення терміну зі словника.

Основний успішний сценарій:

1. Користувач обирає в словнику термін, який хоче редагувати.

2. Користувач обирає задачу «Редагувати». Система виводить форму редагування терміну.

3. Користувач вводить дані, зберігає. Система підтверджує. Система зберігає термін.

Альтернативні сценарії:

2.1 Не заповнені поля форми: система виводить повідомлення «Заповніть поля!».

1.1 Модель варіантів використання

Надалі будуть представлені моделі варіантів використання, записані в вигляді формул.

Пункт «Створити словник».

Опис пункту має вигляд (1.1):

$$Create = (N, [Actor, tp1, tt1, System, tp2, tf1], [Actor, tp1, tt2, System, tp2, File], [Actor, tp1, tt3, System, tp2, Dictionary]) \quad (1.1)$$

Розшифруємо позначення, наведені у формулі (1.1):

- N – номер прецеденту
- $tp1$ – «обирає задачу»
- $tp2$ – «виводить»
- $tt1$ – «пошук файлу»
- $tt2$ – «відкрити файл»
- $tt3$ – «аналіз»
- $tf1$ – форма для вибору файлу
- $Actor$ – користувач, який взаємодіє з системою (повинен бути визначеним у введені до прецеденту);
- $System$ – система, виступає в якості відповідача на дії користувача;
- $Dictionary$ – словник, який є списком термінів виділених з текстового файлу.

Пункт «Переглянути словник».

Опис пункту має вигляд (1.2):

$$Create = (N, [Actor, tp1, tt1, System, tp2, tf1], [Actor, tp1, tt2, System, tp2, Dictionary]) \quad (1.2)$$

Розшифруємо позначення, наведені у формулі (1.2):

- N – номер прецеденту
- $tp1$ – «обирає задачу»
- $tp2$ – «виводить»
- $tt1$ – «переглянути словник»
- $tt2$ – «відкрити файл словника»
- $tf1$ – форма для вибору файлу словника
- $Actor$ – користувач, який взаємодіє з системою (повинен бути визначеним у введені до прецеденту);
- $System$ – система, виступає в якості відповідача на дії користувача;
- $Dictionary$ – словник, який є списком термінів виділених з текстового файлу.

Пункт «Зберегти словник».

Опис пункту має вигляд (1.2):

$$Create = (N, [Actor, tp1, tt1, System, tp2, tf1], [Actor, tp1, tt1, System, tp3, Dictionary]) \quad (1.2)$$

Розшифруємо позначення, наведені у формулі (1.2):

- N – номер прецеденту
- $tp1$ – «обирає задачу»
- $tp2$ – «виводить»
- $tp3$ – «зберігає»

- $tt1$ – «зберегти словник»
- $tf1$ – форма для збереження файлу словника
- $Actor$ – користувач, який взаємодіє з системою (повинен бути визначеним у введені до прецеденту);

- $System$ – система, виступає в якості відповідача на дії користувача;
- $Dictionary$ – словник, який є списком термінів виділених з текстового файлу.

Пункт «Додати термін».

Опис пункту має вигляд (1.1):

$$Create = (N, [Actor, tp1, tt1, System, tp2, tf1], [Actor, tu1, Term],$$
$$[Actor, tp1, tt2, System, tp3, Dictionary, System, tp2, Dictionary]) \quad (1.1)$$

Розшифруємо позначення, наведені у формулі (1.1):

- N – номер прецеденту
- $tp1$ – «обирає задачу»
- $tp2$ – «виводить»
- $tp3$ – «зберігає»
- $tt1$ – «додати термін»
- $tt2$ – «зберегти»
- $tx1$ – введений користувачем текст
- $tf1$ – форма для додавання терміну
- $Actor$ – користувач, який взаємодіє з системою (повинен бути визначеним у введені до прецеденту);
- $System$ – система, виступає в якості відповідача на дії користувача;
- $Term$ – термін (поняття) в словнику, який позначає предмет чи явище;
- $Dictionary$ – словник, який є списком термінів виділених з текстового файлу (складається з термінів).

Пункт «Видалити термін».

Опис пункту має вигляд (1.1):

$$Create = (N, [Actor, tp1, Term, Actor, tp2, tt1, System, tp3, tf1], [Actor, tp2, tt2, System, tp4, Term]) \quad (1.1)$$

Розшифруємо позначення, наведені у формулі (1.1):

- N – номер прецеденту
- $tp1$ – «обирає»
- $tp2$ – «обирає задачу»
- $tp3$ – «виводить»
- $tp4$ – «видаляє»
- $tt1$ – «видалити»
- $tt2$ – «підтвердити»
- $tf1$ – форма підтвердження видалення терміну
- $Actor$ – користувач, який взаємодіє з системою (повинен бути визначеним у введені до прецеденту);
- $System$ – система, виступає в якості відповідача на дії користувача;
- $Term$ – термін (поняття) в словнику, який позначає предмет чи явище.

Пункт «Редагувати термін».

Опис пункту має вигляд (1.1):

$$Create = (N, [Actor, tp1, Term, Actor, tp2, tt1, System, tp3, tf1], [Actor, tx1, Term]$$
$$[Actor, tp2, tt2, System, tp4, Dictionary, System, tp3, Dictionary]) \quad (1.1)$$

Розшифруємо позначення, наведені у формулі (1.1):

- N – номер прецеденту
- $tp1$ – «обирає»
- $tp2$ – «обирає задачу»

- $tp3$ – «виводить»
- $tp4$ – «зберігає»
- $tt1$ – «редагувати»
- $tt2$ – «підтвердити»
- $tx1$ – введений користувачем текст
- $tf1$ – форма редагування терміну
- $Actor$ – користувач, який взаємодіє з системою (повинен бути визначеним у введені до прецеденту);
- $System$ – система, виступає в якості відповідача на дії користувача;
- $Term$ – термін (поняття) в словнику, який позначає предмет чи явище;
- $Dictionary$ – словник, який є списком термінів виділених з текстового файлу (складається з термінів).

2 РОЗРОБКА АЛГОРИТМУ ПОБУДОВИ СЛОВНИКА ТА МАТЕМАТИЧНОЇ МОДЕЛІ БАГАТОСЛІВНОГО ТЕРМІНУ

Алгоритми програмної системи розроблювалися на основі дослідження [8], задачами якого були:

- визначення характеристик БТ;
- формулювання методу виділення БТ;
- створення математичної моделі БТ.

2.1 Визначення характеристик багатослівних термінів

Для автоматизації процесу виділення БТ треба було виявити ряд характеристик БТ, що впливають на технологію цього процесу. Обумовлені характеристики використовують поняття «опорне слово» - іменник, що входить в БТ. Для роботи з БТ потрібні були такі характеристики:

- можлива кількість слів, що входять в термін;
- розташування «опорних слів» в БТ;
- можливу кількість «опорних слів» в БТ;
- визначення слів і знаків пунктуації, які обмежують БТ.

З огляду на те, що словник предметної області створюється для подальшого проектування і супроводу програмного продукту, для дослідження [8] були обрані текстові документи з ряду областей техніки і прикладних наук (інформаційні управляючі системи, прикладні аспекти математики і кібернетики, екологія, процеси управління і ін.) Російською, українською та хорватською мовами. Для кожної предметної області було виділено по 200 термінів.

На Рисунку 2.1 показаний середній розподіл ймовірностей входження в багатослівний термін певної кількості слів. Розкид значень, який визначається конкретною предметною областю не перевищив 1 - 2%.

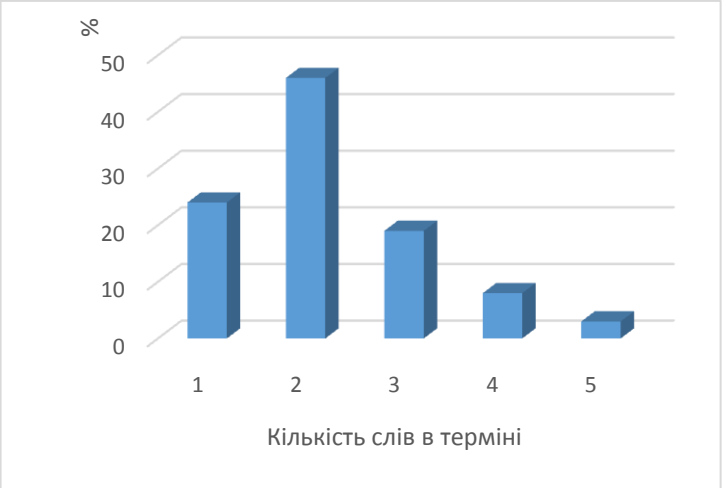

Рисунок 2.1 – Ймовірності появи терміну, що містить одне і більше слів

На Рисунку 2.2 наведені результати аналізу розташування опорного слова в багатослівному терміні. В якості опорного слова вибиралися іменники, наприклад, для терміну «інформаційна система» опорним словом є «система». Якщо в терміні виявилося більше одного іменника як, наприклад, в терміні «реляційні бази даних», то кожне з них було віднесено до відповідної категорії «бази» - посередині, «даних» - справа.

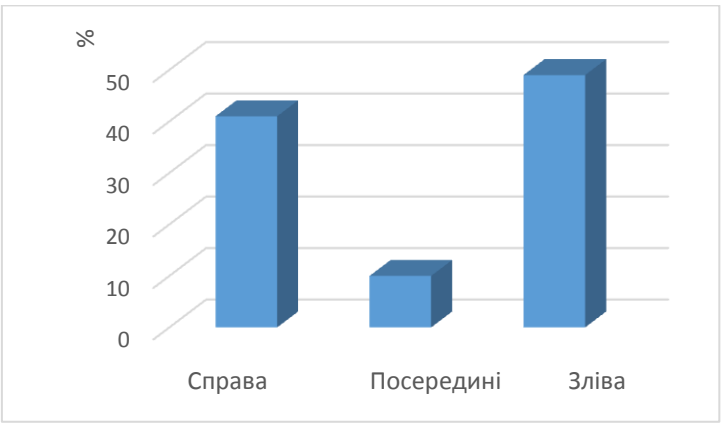

Рисунок 2.2 – Ймовірності розташування опорного слова в багатослівному терміні

На Рисунку 2.3 показана ймовірність появи декількох опорних слів (іменників) в БТ. Ймовірність появи декількох іменників висока, тому спосіб

виділення терміне по іменнику і узгодженим з ним прикметника приводить до великих погрішностей, а більш глибокий аналіз зв'язку слів вимагає складного синтаксичного аналізу. Це підтверджує необхідність шукати більш ефективний спосіб виділення БТ.

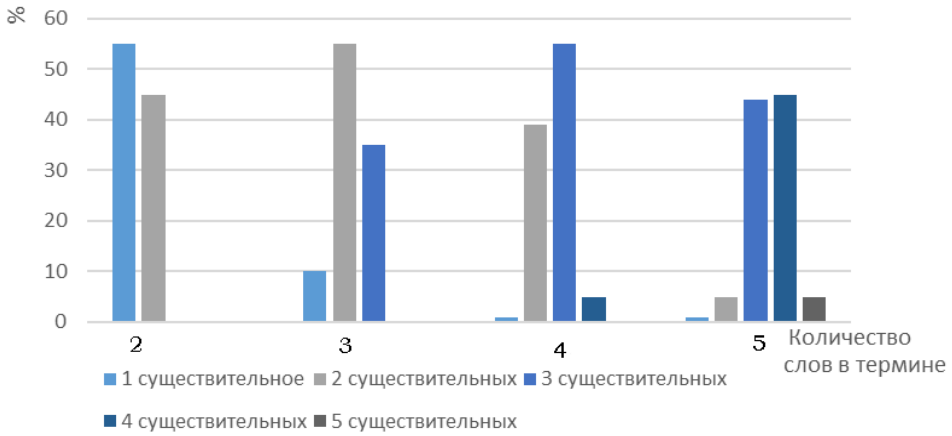

Рисунок 2.3 – Імовірність появи іменників в БТ

У Таблиці 2.1 наведені результати визначення можливих меж БТ.

Таблиця 2.1 – Можливі межі входження БТ в текст

№	**Обмеження зліва**	**Входження в БТ**	**Обмеження справа**
1	Пробіл	Входить	Пробіл
2	, пробіл	Входить*	,
3	– пробіл	Входить	Пробіл –
4	: пробіл	Не входить	:
5	; пробіл	Не входить	;
6	. пробіл	Не входить	.
7	? пробіл	Не входить	?
8	! пробіл	Не входить	!
9	– пробіл	Не входить	Пробіл –

Випадок, коли кома входить в БТ, виявився єдиним з 1000 проаналізованих БТ («особи, які приймають рішення»).

Відповідно до результатів дослідження зроблено такі висновки:

- багатослівний термін може бути представлений не більше ніж п'ятьма словами,

- розташування опорного слова в багатослівному терміні може бути будь-яким.

2.2 Метод виділення багатослівних термінів

Метод передбачає ряд етапів:

- морфологічний розбір аналізованого тексту, виділення іменників;

- визначення можливих БТ на основі опорних слів;

- підрахунок кількості входжень БТ в текст.

Для розробки методу виділення багатослівних термінів була представлена математична модель БТ [8]. Запропонована модель розглядає однослівні терміни як окремий випадок багатослівного терміну. В результаті обробки тексту повинен бути отриманий список термінів. Назвемо цей список словником, оскільки в подальшому при додаванні в цей список тлумачень термінів, він стає словником ПО. На етапі виділення термінів словник представимо у вигляді множини записів

$$D = \{r_i\} i = 1, n$$

Кожен запис має вигляд:

$$r = < tm, lsn, nf, q >$$

де

tm – множина варіантів представлення терміна;

lsn – список опорних слів (іменників), що входять в термін, в нормалізованому вигляді;

nf – нормалізоване подання терміна;

q – кількість входжень терміна в документ.

Представлення одного терміну множиною варіантів в процесі аналізу тексту дозволяє в кінці аналізу визначити правильне представлення багатослівного терміну, що містить кілька опорних слів. Кожен елемент множини складається з однакової послідовності слів. Розрізняються елементи відмінками а також числом (однина або множина) відповідних слів. Таблиця 2.2 ілюструє використання множини варіантів представлення одного терміна.

Таблиця 2.2 – Варіанти представлення багатослівного терміну

№	**Термін**		
	Російська мова	**Українська мова**	**Хорватська мова**
1	реляционными базами данных	реляційними базами даних	relacijskim bazama podataka.
2	реляционную базу данных	реляційну базу даних	relacijsku bazu podataka
3	реляционная база данных	реляційна база даних	Relacijska baza podataka

Нормалізована форма подання nf є єдиною для всіх варіантів представлення терміна. Наприклад, для терміна, наведеного в таблиці 2, вона буде представлена послідовністю з трьох слів «реляційний» «база» «дане». Введення нормализованої форми дозволяє порівнювати терміни за допомогою дуже простої процедури порівняння рядків, що суттєво скорочує час обробки тексту.

Список опорних слів терміна lsn при завершенні формування словника D дозволить вибрати найбільш підходящий варіант подання терміна tm.

Відповідно до діаграми на рис. 2.1 в багатослівний термін може входити до 5 слів. Відповідно до діаграми на рис. 2.2 опорне слово в багатослівному

терміні може займати будь-яку позицію. Тому запропоновано сформувати всі можливі групи слів щодо опорного слова. З цілю скорочення кількості можливих груп відповідно до табл. 2.1 визначено безліч видів лівих і правих меж БТ.

$$B = \{":",";",".","?","!", pron\}$$

Запропоновано формувати можливі терміни як послідовності з 5, 4, 3, 2 і одного слова, що містять, як мінімум, одне опорне слово. Попередньо припустимо, що в послідовність увійде тільки одне опорне слово.

Представимо фрагмент тексту S в виді послідовності елементів:

$$e_1,...,e_l,..,e_m$$

Елементом може бути окреме слово або знак пунктуації. Кожне слово представлено послідовністю букв W (безпосередньо з тексту), множиною атрибутів A і нормалізованою формою представлення nf (результат роботи аналізатора):

$$e = < W, A, nf >$$

Визначимо атрибути, які будуть необхідні для визначення меж БТ. Нехай А1 представляє частину мови, А2 – число, А3 – рід, А4 – особу, А5 – відмінок.

Розділові знаки представляються тільки своїм написанням $e = < W, \varnothing, * >$.

Нехай деякий елемент є опорним словом $e_0 = < W, A, q >$,

де $A1 = noun$ (іменник);

q – кількість появи терміну в тексті S .

Сформулюємо правила складання послідовностей слів:

– послідовність формується з елементів, розташованих поруч один з одним;

– опорне слово обов'язково входить в послідовність;

– число елементів в послідовності не повинно бути більше 5 і менше 1 (розділові знаки, що входять до послідовність, не враховуються);

– послідовність може бути обмежена зліва чи справа від опорного слова, якщо деяким елементом речення e_i при умові, що $e_j \in B$.

Нехай в деякому тексті є послідовність елементів:

$$e_{-5}e_{-4}e_{-3}e_{-2}e_{-1}e_0e_1e_2e_3e_4e_5,$$

де e_0 – опорне слово.

Тоді можливими послідовностями слів (без врахування обмежень) будуть:

$$\begin{bmatrix} e_{-4}e_{-3}e_{-2}e_{-1}e_0 \\ e_{-3}e_{-2}e_{-1}e_0e_1 \\ e_{-2}e_{-1}e_0e_1e_2 \\ e_{-1}e_0e_1e_2e\ e_3 \\ e_0e_1e_2e\ e_3e_4 \\ e_{-3}e_{-2}e_{-1}e_0 \\ ...,e_{-1}e_0 \\ e_0e_1e_2e\ e_3 \\ ...,e_0e_1 \end{bmatrix}$$

Пропонується формула для визначення кількість можливих комбінацій:

$$K = \sum_{i=0}^{i\le 5-2}(5-i) = 14$$

Врахуємо можливі межі для комбінацій. Нехай деякий елемент $e_j \in B$. Тоді всі комбінації, в які входять елементи з індексами $i \le j$, виключаються з подальшого аналізу. Формула для визначення кількості можливих комбінацій при обмеженні зліва має вигляд:

$$K_l = 14 - \sum_{i=1}^{5-j}(5-j+i-1)$$

Кількість можливих послідовностей слів при наявності в послідовності декількох опорних слів залежить від числа опорних слів, але не може перевищувати 14 в розрахунку на одне опорне слово.

2.3 Представлення алгоритмів

В даному пункті розібрані дві основні функції програми – аналіз XML-файлу, та формування можливих комбінацій слів для терміну. Алгоритми цих функцій будуть представлені за допомогою блок-схем.

Блок-схема – поширений тип схем (графічних моделей), що описують алгоритми або процеси, в яких окремі кроки зображуються у вигляді блоків різної форми, з'єднаних між собою лініями, що вказують напрямок послідовності

На рисунку 2.4 представлена схема алгоритму підбору комбінацій терміну (повний опис алгоритму надано в пункті 2.2).

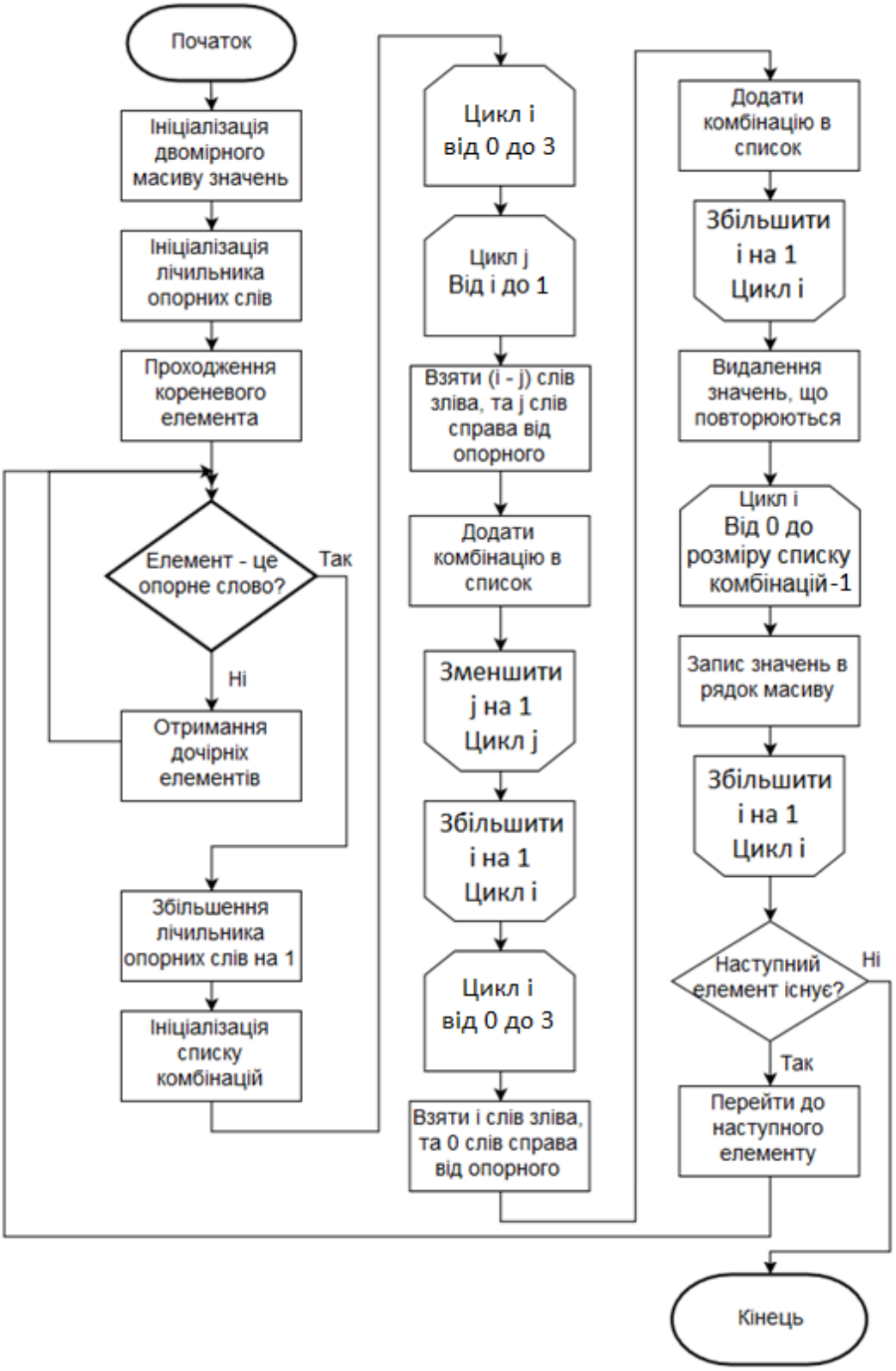

Рисунок 2.4 – Алгоритм підбору комбінацій терміну

2.4 Апробація

Для реалізації запропонованих рішень був розроблений продукт ProgrammerThesaurus. Функціональна схема створення словника наведена на рисунку 2.5.

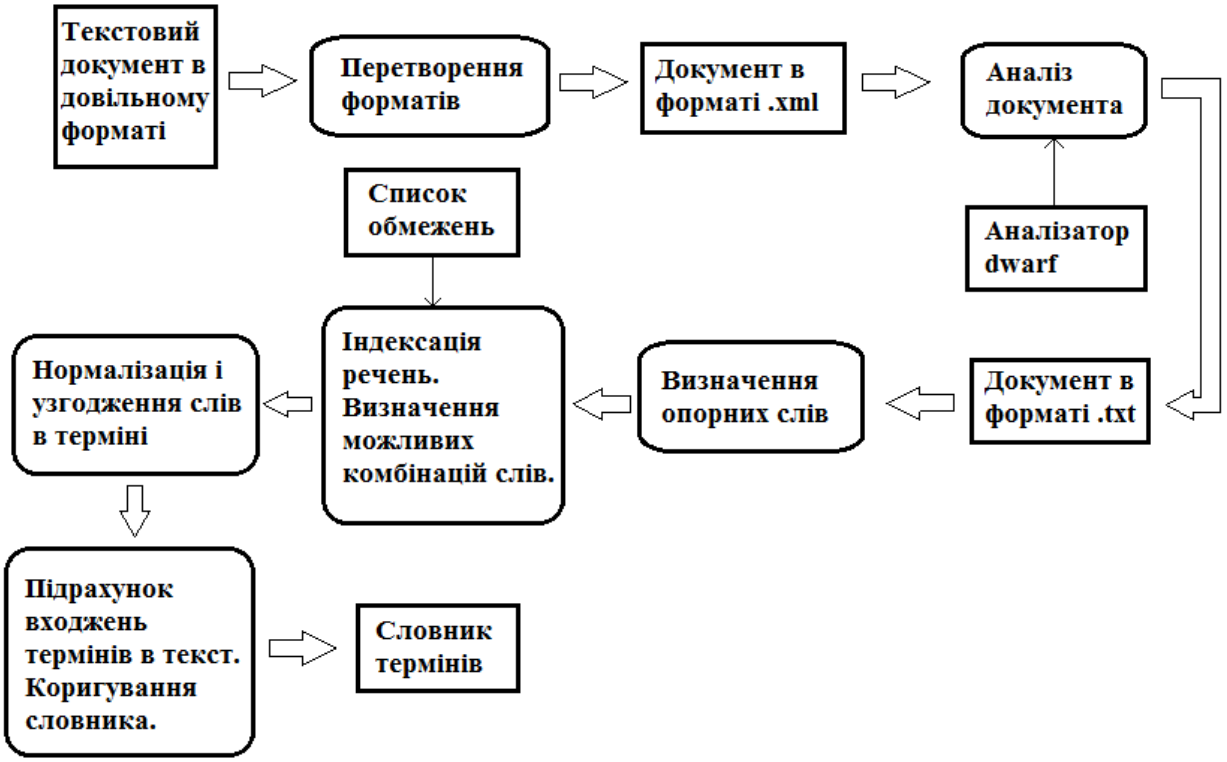

Рисунок 2.5 – Основні етапи створення словника

На рисунку 2.6 представлене вікно, що дозволяє експерту (користувачу) редагувати термін.

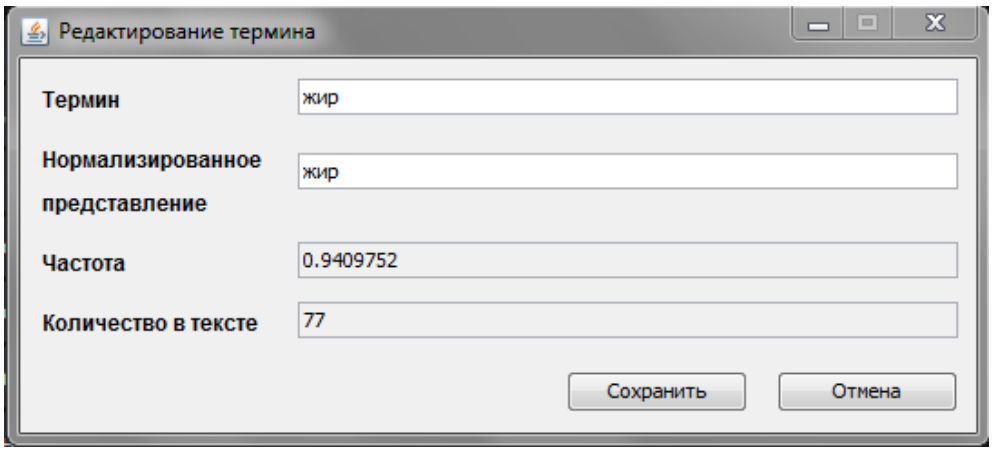

Рисунок 2.6 – Вікно редагування терміну

На рисунку 2.7 представлене вікно списку термінів (словник) з обраного файлу.

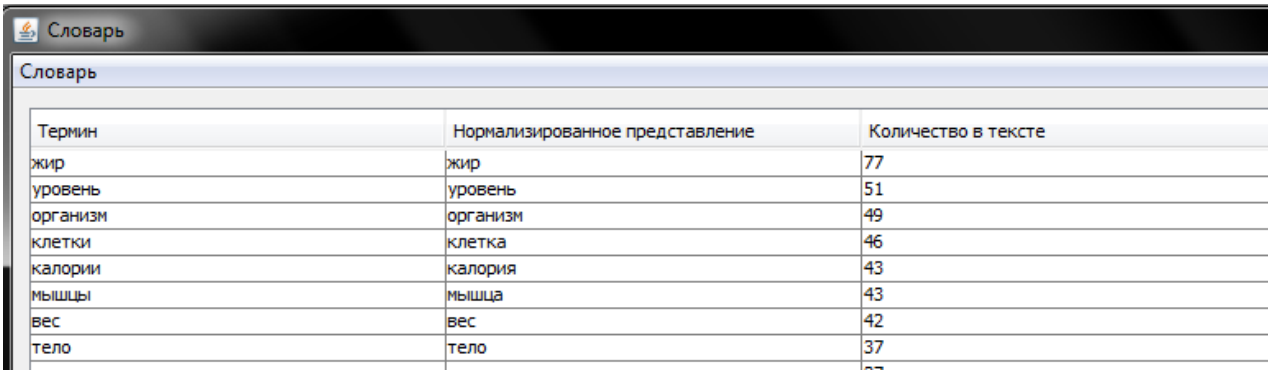

Рисунок 2.7 – Вікно словника

Експериментальні дані, отримані за результатами аналізу ряду документів, наведені в табл. 2.3. Час представлено в хвилинах.

Таблиця 2.3 – Оцінка часу формування словника

Спосіб складання словника	t_r	t_w	t_g	t
«Вручну»	3	20	15	38
За допомогою програми	10^{-3}	10^{-3}	10^{-3}	$3*10^{-3}$

Час складання словника (t) для сторінки тексту документу D_1 в режимі ручної роботи - близько 28 хвилин, а в режимі автоматизованої - близько 0.18 секунд. З урахуванням часу на коригування експертом отриманих результатів – 6 хвилин. З результатів аналізу ефективності видно, що з використанням програми задача виконується в 6,3 рази разів швидше, аніж вручну.

ВИСНОВКИ

У ході роботи успішно виконані наступні задачі:

1. Розроблено математичну модель для автоматизованого формування словника багатослівних термінів в тексті;
2. Розроблено алгоритми функціонування елементів системи та взаємодії між ними;
3. Програмно реалізовано математичні моделі та алгоритми, що розроблювалися.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ларман, К. Использование UML 2.0. и шаблонов проектирования. / К. Ларман. – К: Наука, 2010. – 349 с.
2. JaLingo [Electronic resource]. –Available at: \www/URL: http://jalingo.sourceforge.net/
3. Фаулер, М. UML. Основы. / М. Фаулер. – Символ-Плюс, 2005. – 192 с.
4. Любченко, В.В. Конспект лекций по дисциплине «Архитектура и проектирование программного обеспечения» для студентов направления 6.050103 – Программная инженерия. / В. В. Любченко. – Одесса: ОНПУ, 2012. – 87 с.
5. Кунгурцев, А. Б. Формирование словаря предметной области / А. Б. Кунгурцев, И. В. Барыкина // Искусственный интеллект. – 2006. – № 1. –144–151 с.
6. Побудова словника предметної області на основі автоматизованого аналізу текстів українською мовою / О. Кунгурцев, С. Ковальчук, Я. Поточняк, М. Широкоступ // Технічні науки та технології. – Чернігів, 2016. – № 3 (5). – 164–174 с.
7. Кунгурцев А. Б., Поточняк Я. В., Силяев Д. А. Метод автоматизированного построения толкового словаря предметной области / А.Б. Кунгурцев, Я.В. Поточняк, Д.Ф. Силяев // Технологический аудит и резервы производства — № 2/2(22), 2015. – 58–63 с.
8. Kungurtsev O. Development of information technology of term extraction from documents in natural language / O. Kungurtsev, S. Zinovatnaya, Ia. Potochniak, M. Kutasevych // Eastern-European Journal of Enterprise Technologies. Vol 6, No 2 (96) (2018). pp. 44-51. DOI: https://doi.org/10.15587/1729-4061.2018.147978 (SCOPUS)
9. Bourigault, D. Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases [Text] / D. Bourigault // Proc. of COLING-92. –Nantes, France, 1992. – P. 977–981.
10. Baroni, M. Bootstrapping Corpora and Terms from the Web [Text] / M. Baroni, S. Bernardini // Proceedings of LREC. – Lisbon: ELDA, 2004. – P. 1313–1316.

11. Программный пакет синтаксический анализ. Проект АОТ [Электронный ресурс]. – Режим доступа: \www/URL: http://www.aot.ru/docs/synan.html
12. Шелов, С. Д. Терминоведение: семь вопросов и семь ответов по семантике термина [Текст] / С. Д. Шелов // НТИ. Сер. 2. Информационные процессы и системы. – 2001. – № 2. – С. 1–11.
13. Кунгурцев А.Б., Кутасевич М.А, Поточняк Я. В. / Алгоритм автоматического выделения ключевых слов из документов на естественном языке // Материали VII Международной научно-практической конфиренции – 2018 – ИУСТ-Одесса – С. 155-158.

АНОТАЦІЯ

Метою розробки технології є скорочення часу на побудову словника предметної області за рахунок автоматизації процесу виділення багатослівних термінів з текстових документів.

Створений алгоритм виділення багатослівних термінів на основі визначення іменників в якості опорних слів.

Розроблений механізм підрахунку входжень термінів в документ, що базується на нормалізованому представлені термінів.

Тестування ПЗ показало якісне виділення термінів та значне скорочення часу аналізу документів порівняно з ручною роботою.

Ключові слова: словник предметної області, багатослівний термін, опорне слово, частота, текстовий документ, нормалізоване представлення.